Deblurring Poissonian Images via Multi-constraint Optimization

Stanislav Harizanov

Abstract This paper deals with the restoration of images corrupted by a non-invertible or ill-conditioned linear transform and Poisson noise. The paper is experimental and can be seen as a continuation of "as reported by Harizanov et al. (Epigraphical Projection for Solving Least Squares Anscombe Transformed Constrained Optimization Problems 2013)". The constraint set in the minimization problem, considered there, was too large and the results tend to oversmooth the initial image. Here, we consider various techniques for restricting this set in order to improve the image quality of the result, and numerically investigate them. They are based on image domain decomposition and give rise to multi-constraint optimization problems.

1 Introduction

Industrial computed tomography (CT) scanning uses irradiation (usually with x-rays) to produce three-dimensional representations of the scanned object both externally and internally. The latter is derived from a large series of two-dimensional radiographic images taken around a single axis of rotation. To create each of the planar images, a heterogeneous beam of X-rays is produced and projected toward the object. A certain amount of X-ray is absorbed by the object, while the rest is captured behind by a detector (either photographic film or a digital detector). The local magnitudes of the detected X-ray amount determine the corresponding gray-scale pixel values of the radiographic image. In such processes, where images are obtained by counting particles, Poisson noise occurs. Being interested in improving the quality of the 3D CT reconstruction, we are motivated to investigate 2D Poisson denoising techniques, and to apply them to each radiographic image.

S. Harizanov (✉)
Institute of Information and Communication Technologies,
Bulgarian Academy of Sciences, Sofia, Bulgaria
e-mail: sharizanov@parallel.bas.bg

S. Margenov et al. (eds.), *Innovative Approaches and Solutions in Advanced Intelligent Systems*, Studies in Computational Intelligence 648,
DOI 10.1007/978-3-319-32207-0_13

In mathematical terms, one wants to solve the ill-posed inverse problem of recovering the original 2D image $\bar{u} \in [0, v]^{M \times N}$ from observations

$$f = \mathcal{P}(H\bar{u}),$$

where v is the gray-scale intensity, $\mathcal{P}$ denotes an independent Poisson noise corruption process, and $H \in [0, +\infty)^{n \times n}$ is a blur operator, corresponding to a convolution with a Gaussian kernel. Here $n = MN$, because it is beneficiary to column-wise reshape the image into a long 1D vector. Note that blurring appears naturally in practice (e.g., when the industrial CT scan is not well calibrated) and needs to be incorporated in the problem.

Poisson denoising is a hot and active research field, so it is impossible to list all the related publications. We mention only few of them [1–10] in chronological order, as illustration. All the approaches are based on minimizing a regularization term $\Psi(u)$, where a data fidelity term $F(u,f)$ is either incorporated in the cost function as penalization

$$\underset{u}{\operatorname{argmin}}\ \Psi(u) + \lambda F(u,f), \quad \lambda \geq 0, \tag{1}$$

or considered as constraint

$$\underset{u}{\operatorname{argmin}}\ \Psi(u) \quad \text{subject to} \quad F(u,f) \leq \tau, \quad \tau \geq 0. \tag{2}$$

The problems (1) and (2) are closely related and, under some mild assumptions on Ψ and F, there is a one-to-one correspondence $\lambda \leftrightarrow \tau$, such that their solutions coincide (see [11]). In general, (1) is easier to solve, but the optimal parameter λ cannot be well approximated, while (2) is both mathematically and computationally more complex, but the optimal parameter τ is statistically estimated.

There are two main-stream directions for the choice of the data fidelity F. In the first one, the mean/variance dependence of the Poisson distribution can be reduced by using variance-stabilizing transformations (VST), such as the *Anscombe transform* [12]

$$T : [0, +\infty)^n \to (0, +\infty)^n : v = (v_i)_{1 \leq i \leq n} \mapsto 2\left(\sqrt{v_i + \frac{3}{8}}\right)_{1 \leq i \leq n}.$$

It transforms Poisson noise to approximately Gaussian noise with zero-mean and unit variance (if the variance of the Poisson noise is large enough), for which Least Squares estimates are maximum-likelihood ones. The second approach is closely related to a direct Maximum A Posteriori (MAP) estimate, where the neg-log-likelihood of the Poisson noise, i.e., the I-divergence (generalized Kullback-Leibler divergence)

$$u \mapsto D(f, Hu) := \begin{cases} \langle \mathbf{1}_n, f \log \frac{f}{Hu} - f + Hu \rangle & \text{if } Hu > 0, \\ +\infty & \text{otherwise,} \end{cases}$$

is used. Here $\langle \cdot, \cdot \rangle$ denotes the standard Euclidean inner product and $\mathbf{1}_n$ denotes the vector consisting of n entries equal to 1.

This paper is a continuation of our previous work [8], thus we deal with the Total Variation (TV) [13] constraint optimization problems

$$\underset{u \in [0,+\infty)^n}{\operatorname{argmin}} ||\nabla u||_{2,1} \quad \text{subject to} \quad ||T(Hu) - T(f)||_2^2 \leq \tau_A, \tag{3}$$

$$\underset{u \in [0,+\infty)^n}{\operatorname{argmin}} ||\nabla u||_{2,1} \quad \text{subject to} \quad D(f, Hu) \leq \tau_I, \tag{4}$$

where $\nabla \in \mathbb{R}^{2n \times n}$ is the discrete gradient operator (forward differences and Neumann boundary conditions are used), and $|| \cdot ||_{2,1}$ denotes the $\ell_{2,1}$ norm.

To both problems, we apply the **p**rimal- **d**ual **h**ybrid **g**radient algorithm [14, 15] with an extrapolation (**m**odification) of the dual variable (PDHGMp). At each iteration step, we compute n epigraphical projections [16] w.r.t. a 1D convex function related to T [8], respectively solve an I-divergence constrained least squares problem [6, 9]. The algorithms' description can be found in [8, Sect. 4]. We keep the notation Algorithms 1–2 from it.

The paper is organized as follows: In Sect. 2, we experiment with different choices of τ_A, τ_I and measure their effect on the output image quality. In Sect. 3, we propose various domain decompositions in order to improve that quality. Those decompositions give rise to multi-constraint optimization problems, for which Algorithms 1–2 are still applicable. Numerical experiments are conducted and the results are discussed. Conclusions are drawn in Sect. 4.

2 Single-Constraint Optimization with Optimal τ

In this paper, we test the same initial images $\bar{u}$ 'cameraman' (256×256), its central part (130×130), and 'brain' (184×140) as in [8] (Fig. 1), and we work with the same polluted images f. This allows us to compare numerical results. Again, we denote them by $\mathrm{B1}_\nu$, $\mathrm{B1part}_\nu$, and $\mathrm{B2}_\nu$, where ν stands for the gray-scale intensity. We recall that the peak signal to noise ratio (PSNR) and the mean absolute error (MAE) are computed via

$$\mathrm{PSNR} = 10 \log_{10} \frac{|\max \bar{u} - \min \bar{u}|^2}{\frac{1}{n} ||u - \bar{u}||_2^2}, \quad \mathrm{MAE} = \frac{1}{n\nu} ||\bar{u} - u||_1.$$

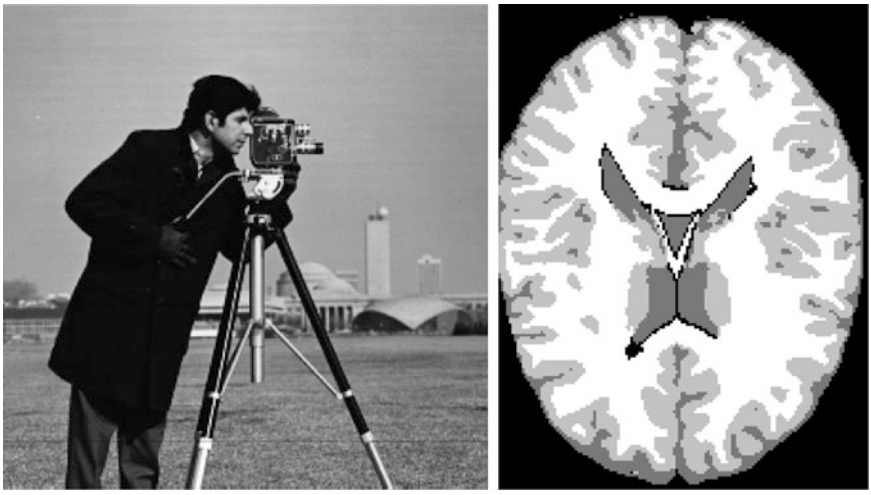

Fig. 1 Original images 'cameraman' (*left*) and phantom of a brain image (*right*)

The statistically motivated choice for the constraint parameter $\tau_A = n$ in (3), resp. $\tau_I = n/2$ in (4), places the true image $\bar{u}$ with high probability very close to the boundary of the constraint sets (see [8, Table 1] for experimental verification).

$$C_A = \{u : ||T(Hu) - T(f)||_2^2 \leq \tau_A\}, \quad C_I = \{u : D(f, Hu) \leq \tau_I\}, \tag{5}$$

thus guaranteeing those sets are non-empty and minimizers u_A, resp. u_I, exist. Moreover, since the TV functional is a semi-norm and every semi-norm is positively-homogeneous, $u_A \in \partial C_A$, resp. $u_I \in \partial C_I$, are unique, unless the constraint sets (5) contain constant images (the global minimizers of the TV functional), i.e., whenever

$$\min_{c \in \mathbb{R}} ||T(f) - c||_2^2 > \tau_A, \quad \text{resp.} \quad \min_{c \geq 0} D(f, c) > \tau_I.$$

We used that H reproduces constants ($H\mathbf{1}_n = \mathbf{1}_n$), which is true for convolution-based blur operators. Since $E(f_i) = (H\bar{u})_i$ and $T'(\bar{u}_i) \geq T'(v) = 1/\sqrt{v + 3/8}$ $\forall i = 1, \ldots, n$,

$$\min_{c \in \mathbb{R}} ||T(f) - c||_2^2 \geq \frac{1}{v + 3/8} \min_{c \in \mathbb{R}} ||H\bar{u} - c||_2^2.$$

Therefore, problems (3) and (4) admit unique solution if

$$\min_{c\in\mathbb{R}} ||H\bar{u} - c||_2^2 > v\tau_A, \quad \min_{c\geq 0} D(H\bar{u}, c) > \tau_I, \quad v \gg 0. \tag{6}$$

In particular, when $\tau_A = n$, $\tau_I = n/2$, (6) holds true for all *nontrivial* (e.g., edge-containing, not close-to-constant) initial images $\bar{u}$ and moderate blur operators H (such that $H\bar{u}$ remains nontrivial). To conclude, for $\tau_A = n$, $\tau_I = n/2$, and under some natural assumptions on $\bar{u}$, H, and v, problems (3) and (4) are well-posed, with $\bar{u}$ placed around the boundary of their constraint sets (5), thus being an admissible candidate for the unique solution.

Being an admissible candidate is not enough for $\bar{u}$ to be "close" to the actual solution! If the constraint set is too large, then the two images might still differ a lot. As Fig. 2 illustrates, this is indeed the case for our problem (3). The minimizer is significantly "oversmoothened". Hence, in order to let u_A better approximate $\bar{u}$, we need to restrict C_A.

The easiest way to do so is to simply decrease τ_A. We tried it on B1part_{3000} and the results can be viewed on Fig. 3. Since the minimizer of (3) tends to "stretch out" when τ_A decreases, we worked with box constraints $u \in [0, 3000]^n$ in C_A. Due to the $\lambda \leftrightarrow \tau$ relations between (1) and (2), $\tau_A(\lambda_A)$ is monotonically decreasing, thus invertible. We numerically solved the penalized version of (3) for various λ_A, which, as discussed in the introduction, is computationally much more efficient, and cheaply derived the solution u_A of (3) for the corresponding τ_A. We observe that the optimal values for τ_A w.r.t. both PSNR and MAE are smaller than n, which confirms the above conclusion that C_A is too large. Maximal PSNR is obtained for $\tau_A = 0.791n$, while minimal MAE is obtained for $\tau_A = 0.894n$. On the other hand, $\lambda_A = 10^4$ gives rise to $\tau_A = 0.5742n$, while $\lambda_A = 10^5$ gives rise to $\tau_A = 0.5452n$, thus we expect C_A to be empty for $\tau_A = 0.5$.

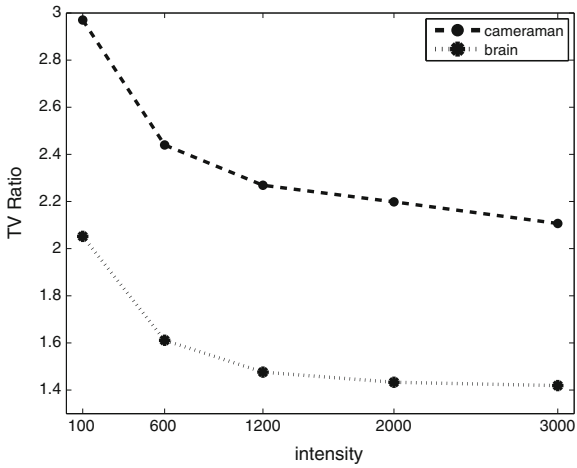

Fig. 2 The ratio $TV(\bar{u})/TV(u_A)$ as a function of v for the both test images

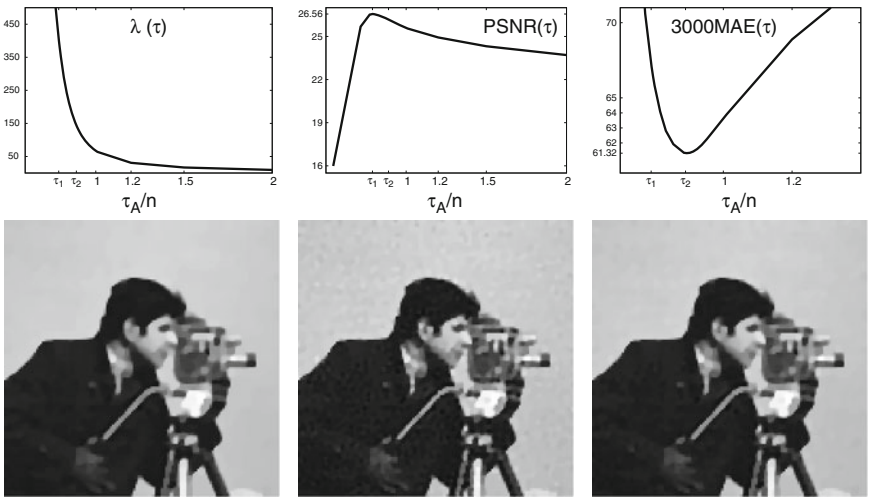

Fig. 3 *Left* λ as a function of τ_A and u_A for $\tau_A = n$. *Center* PSNR (u_A) as a function of τ_A and its minimizer u_A $(\tau_A = \tau_1 n)$. *Right* vMAE (u_A) as a function of τ_A and its minimizer u_A $(\tau_A = \tau_2 n)$

From the plots of PSNR (u_A) and 3000MAE (u_A) $(v = 3000)$ we see that the former is much more stable to the change of τ_A than the latter, e.g., when $\tau_A \in [0.72, 1]$, PSNR $(u_A) \in [25.5932, 26.5614]$, while v MAE $(u_A) \in [61.3239, 90.2654]$. Moreover, the optimal minimizer w.r.t. PSNR possesses quite large MAE (namely ≈67) and artifacts in the smooth regions of $\bar{u}$, while the optimal minimizer w.r.t. MAE has quite satisfactory PSNR (namely 26.16) and good visual properties (see Table 1). This is due to the different norms involved in the two functions, namely the 2-norm and the 1-norm. The initial image $\bar{u}$ was firstly blurred by a Gaussian kernel with $\sigma = 1.3$, thus smoothed around its edges (the TV semi-norm of the blurred image is 1.3146e+06 which is more than 2 times smaller than the TV semi-norm 2.9390e+06 of B1part_{3000}). In particular, the most problematic image part is the camera where we have very "thin" details. Since, we are also looking for the smoothest solution in C_A, the minimizers of (3) fail to recover the jump discontinuities of $\bar{u}$ in these regions and $||u_A - \bar{u}||_\infty$ is quite large there. The smaller the τ_A the larger the TV semi-norm of the minimizer and the better the capturing of the singularities. In the same time, the Poisson noise in the smooth regions becomes more and more problematic, since we further deviate from the statistically optimal value $\tau_A = n$ for its complete removal. PSNR is based on $\frac{1}{n}||\bar{u} - u_A||_2^2$ and the impact of $||u_A - \bar{u}||_\infty$ along the camera edges is very strong.

Table 1 Comparison among the optimal τ_A's from Fig. 2

	$\tau_A = 0.791n$	$\tau_A = 0.894n$	$\tau_A = n$
PSNR	26.5614	26.1581	25.5934
vMAE	67.0804	61.3239	63.6238

Hence the denoising failure in the smooth regions is undermined as long as the size of the artifacts there is not comparable to the one of the jump discontinuities at the edges. On the other hand, $3000MAE(u_A) = \frac{1}{n}||\bar{u} - u_A||_1$, which is a "sparser" norm and it is small when many entries of $u_A - \bar{u}$ are close to zero. Hence, it captures better the presence of denoising artifacts.

In conclusion, for the constraint optimization problem (3) it seems that, in order to obtain good visual results, it is better to minimize the MAE of u_A than to maximize its PSNR.

3 Multi-constraint Optimization

Even though decreasing τ_A may improve the image qualities of the solution of (3), it doesn't seem like a good strategy. We deviate from its statistical estimation, thus we need to guess the right value of τ_A, which is computationally expensive. Moreover, we lose the nice properties of $\bar{u}$ being close to the boundary of C_A, thus the guaranteed well-posedness of (3) as well as the admissibility of $\bar{u}$ to be the minimizer. In this section we follow a different approach for restricting C_A that is based on image domain decomposition and the use of independent constraints for the regions. In such a setup, Eq. (3) is reformulated into

$$\underset{u\in[0,+\infty)^n}{\operatorname{argmin}} ||u||_{2,1} \quad \text{subject to} \quad ||T(Hu) - T(f)||_{A_i} \leq \tau_i, \quad i = 1, \ldots, K. \tag{7}$$

where $\{A_i\}_{i=1}^K$ is a tessellation of the image domain ($int(A_i) \cap int(A_j) = \emptyset$), and $||\cdot||_{A_i}$ is a short notation for the squared 2-norm, restricted to the region A_i. Analogously, the multi-constraint version of (4) is

$$\underset{u\in[0,+\infty)^n}{\operatorname{argmin}} ||\nabla u||_{2,1} \quad \text{subject to} \quad D_{A_i}(f, Hu) \leq \frac{1}{2}\tau_i, \quad i = 1, \ldots, K. \tag{8}$$

Both Algorithms 1–2 can be straightforwardly modified to such multi-constraint setting, since no correlation among pixel data appear in $||\cdot||_2^2$ and $D(\cdot,\cdot)$, allowing for direct and complete component-wise splitting. Moreover, following the notation in [9], $\lambda_i^{(k+1)}$ is the solution of $D_{A_i}(f, g(p_1^{(k)} + Hu^{(k+1)}, \lambda/\sigma)) = \tau_i/2$ and for $\lambda_i := \lim_{k\to\infty}\lambda_i^{(k)}$ we have that the the minimizer of (8) also minimizes

$$\underset{u\in[0,+\infty)_n}{\operatorname{argmin}} ||\nabla u||_{2,1} + \sum_{i=1}^{K} \lambda_i D_{A_i}(f, Hu).$$

3.1 Block Subdivision

The first thing we try is a simple block subdivision of the spatial domain as illustrated on Fig. 4. More precisely, given a level l, we split the image into 4^l blocks $A_{i,j}^{(l)}$, $i,j = 0,\ldots,2^l-1$ of "equal" size (if the number of pixels in height or width is not divisible by 2^l, we of course need to round off and the blocks cannot be absolutely identical) and take $\tau_{i,j}^{(l)} = \text{card}(A_{i,j}^{(l)}) \approx n/4^l$. Due to triangle inequality, it is straightforward that the corresponding constraint set $C_A^{(l)}$ in (7) is a proper subset of C_A. Moreover, if we denote $C_A^{(0)} := C_A$, we have $C_A^{(l)} \subset C_A^{(l-1)}, \forall l \in \mathbb{N}$.

The results are summarized in Table 2. We observe that, as expected, increasing the level l we restrict the global constraint sets and increase the TV semi-norm of the solutions of (7) and (8). Up to a certain moment (in the particular case for B1part_{3000} this is $l = 3$) both PSNR and MAE values of the outputs improve. Then the MAE value has a significant jump and the visual quality of the output image

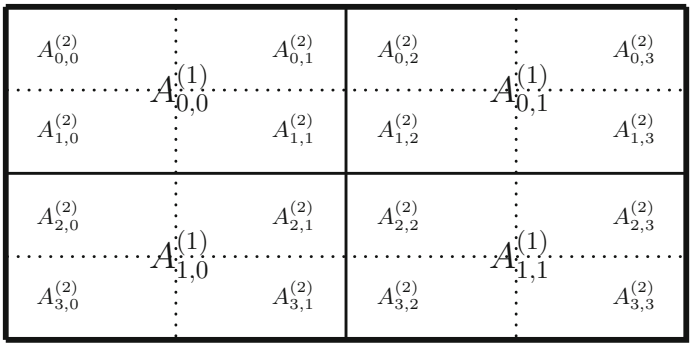

Fig. 4 Domain partition for block subdivision of levels 1 (*solid lines*) and 2 (*dotted lines*)

Table 2 Results of Algorithms 1–2 on B1part_{3000} for different l. When $l > 0$ we initialize with the output for $l-1$, and set $(\sigma,\rho) = (0.4, 0.3)$ in Algorithm 1

Level	#iter	TV semi-norm	PSNR	MAE·v
0	20000	1.7070e+6	25.5934	63.6238
		1.7073e+6	25.5949	63.6054
1	20000	1.7194e+6	25.6957	62.8892
		1.7197e+6	25.6975	62.8620
2	20000	1.7418e+6	25.8372	62.4555
		1.7423e+6	25.8385	62.4421
3	50000	1.7930e+6	25.9966	62.1405
	20000	1.7934e+6	25.9975	62.1315
4	50000	1.9490e+6	26.0686	64.6669
	20000	1.9649e+6	26.0298	65.1237

Fig. 5 Block subdivision. From *left* to *right*: Algorithm 1 outputs for $l = 3$ and $l = 4$

drops down (see Fig. 5). The reason is that, we are still using the statistically estimated bounds $\tau_{i,j}^{(l)}$ for each of the blocks. However, when l is large the size of the blocks $A_{i,j}^{(l)}$ is small, the law of large numbers that backs up the theoretical arguments in [12, 17] fails, and we cannot guarantee that the value of $\tau_{i,j}^{(l)}$ is adequate or even that the constraint problem (7) is block-wise well-posed.

Indeed, already at the third subdivision level we have trivial regions (constant images are admissible for some of the blocks). The uniqueness of the global minimizer is not affected, because those trivial blocks interfere with the others and, since we minimize the overall TV semi-norm, the constant intensity value on them is uniquely determined by its nontrivial neighbors. On the other hand, existence of the global minimizer becomes problematic at the fourth subdivision level. We have smaller blocks (their size is $\approx 8 \times 8$) and on some of them the original image $\mathrm{B1part}_{3000}$ is close-to-constant. When such a region $A_{i,j}^{(4)}$ is of small intensity (i.e., close to black) the Poisson noise is insignificant there and the oscillations of the neighboring pixel values in $f|_{A_{i,j}^{(4)}}$ are negligible. On top of that, the Anscombe transform is not reliable in the sense that $T(f|_{A_{i,j}^{(4)}})$ is not guaranteed to be normally distributed, so the choice $\tau_{i,j}^{(4)} = \mathrm{card}(A_{i,j}^{(4)})$ has no theoretical justification. To summarize,

$$||T(H\bar{u}) - T(f)||_{A_{i,j}^{(4)}} \sim 0 = \mathrm{card}(A_{i,j}^{(4)}) = \tau_{i,j}^{(4)},$$

and $\bar{u}|_{A_{i,j}^{(4)}}$ is well inside the interior of $C_{A_{i,j}^{(4)}}$. In the single-constraint case we have already verified that $\bar{u}$ is with high probability close to ∂C_A, thus the above relation implies for another region $A_{i,j}^{(4)}$, $||T(H\bar{u}) - T(f)||_{A_{i,j}^{(4)}} \gg \tau_{i,j}^{(4)}$, so $C_{A_{i,j}^{(4)}}$ could be empty.

Hence, Algorithm 1 may not converge on $A_{i,j}^{(l)}$. The following example of one such empty-constraint block at level 5 illustrates the problem

$$\bar{u}|_{A_{4,19}^{(5)}} = \begin{pmatrix} 2250.0 & 2262.3 & 2262.3 & 2176.2 \\ 2274.6 & 2262.3 & 2299.2 & 2286.9 \\ 2262.3 & 2311.5 & 2311.5 & 2299.2 \\ 2299.2 & 2323.8 & 2299.2 & 2348.4 \end{pmatrix}$$

$$f|_{A_{4,19}^{(5)}} = \begin{pmatrix} 2348 & 2243 & 2360 & 2183 \\ 2244 & 2295 & 2205 & 2234 \\ 2190 & 2364 & 2213 & 2393 \\ 2313 & 2269 & 2326 & 2264 \end{pmatrix}.$$

$\bar{u}|_{A_{4,19}^{(5)}}$ is almost constant and of high intensity, while the Poisson noise contributes significantly and visibly alters the entries of f.

As a result, the output image of Algorithm 1 for $l = 4$ possesses certain artifacts in some high, close-to-constant intense regions. As discussed in Sect. 2, those artifacts are captured by the MAE value of the output, which immediately jumps up, but not by its PSNR value, which still improves (see Table 2).

The block subdivision might be different. For example, it may be data-dependent and based on the output image at zeroth level $u^{(0)}$. We tested a 2-block subdivision, where A_0 is a 130×75 block that deviates most from the statistical expectation (i.e., it maximizes the quantity $\left| ||T(Hu^{(0)} - T(f)||_A - 130 \cdot 75 \right|$), and A_1 is its complement, as well as a 4-block subdivision, where A_0 is a 75×75 block that maximizes the analogous expression, A_1 and A_2 are the "horizontal" and "vertical" complements of A_0, and A_3 is the complement of their union $A_0 \cup A_1 \cup A_2$. For different original images and intensity levels the benefit of such data-dependency is different, but the quality of the output is comparable to that of the "standard" block subdivision presented above, possibly on a higher level. The close similarity between the solutions of (3) and (4), numerically observed in [8], holds true also for their multi-constraint versions (7) and (8). Only for $l = 4$ the outputs of Algorithms 1–2 differ visibly, but this is due to the ill-posedness of the optimization problems and the slow (or even lack of) convergence of the algorithms. This phenomenon appears for all test images and all constraint choices, we considered, therefore from now on we deal only with (7) and Algorithm 1.

3.2 *Intensity Tessellation*

The statistical choice $\tau_A = n$ in (3) is based not only on the law of large numbers, but also on the Central Limit Theorem. Therefore, it theoretically holds for independent and *identically distributed* Poisson random variables, which in our setting

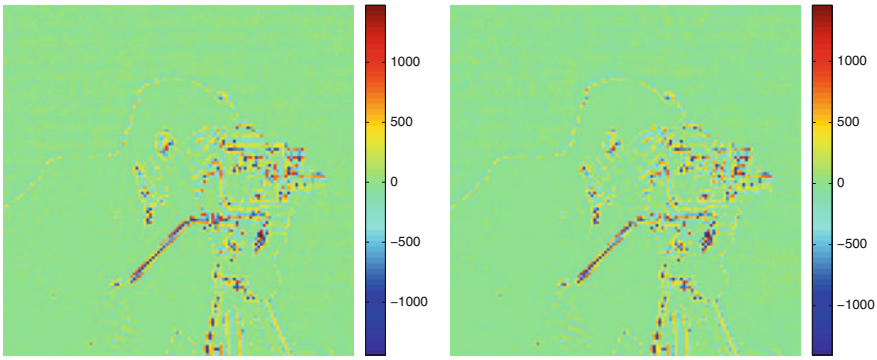

Fig. 6 Noise redistribution in block subdivision. The difference images between the output of Algorithm 1 at level 0 $u_A^{(0)}$ (*left*), respectively at level 3 $u_A^{(3)}$ (*right*) and the original image $\mathrm{B1part}_{3000}$

is equivalent to a trivial blurry image $H\bar{u} = const$. This is by far not true for the examples we consider, and even though in [8] $||T(H\bar{u}) - T(f)||_2^2/n \approx 1$ is numerically verified, the solution u_A of (3) is oversmoothened due to a "redistribution" of the noise among the pixels. Indeed, as illustrated on Fig. 6, around the edges (jump discontinuities) of the original image, where most of the TV semi-norm of $\bar{u}$ is concentrated, the big positive displacements $H\bar{u}_{i,j} - f_{i,j} \gg 0$ of the high-intensity pixels (i,j) related to the Poisson distribution are wrongly accumulated in the minimizer by the neighboring low-intensity pixels (i',j'), making $Hu_{i,j} \approx f_{i,j}$ and $Hu_{i',j'} \approx f_{i',j'} + (H\bar{u}_{i,j} - f_{i,j})$, while still $u \in C_A^{(l)}$. In other words, if a high-intensity edge pixel value of $\bar{u}$ is decreased by the noise, it is not properly denoised in $u_A^{(l)}$ but rather its neighboring low-intensity edge pixel increases its intensity with a reciprocal amount. Thus f is no longer a realization of independent Poisson random variables over $Hu_A^{(l)}$, meaning that $u_A^{(l)}$ does not approximate well $\bar{u}$ along the edges. On the other hand, away from the edges $u_A^{(l)}$ is a quite good approximation of $\bar{u}$. We back this up with a simple experiment. We replace all pixel intensities of $u_A^{(3)}$ from Table 2, that are more than d away from the corresponding values of the original image $\bar{u}$ with the true intensities and recompute the PSNR and the MAE values of such a "hybrid" image. The difference image $u_A^{(3)} - \bar{u}$ is the right one from Fig. 6. For $d = v/6 = 500$, only 397 pixels (≈2.35 % of all the pixels) are modified, while PSNR and νMAE improve to 30.1569 and 45.1862, respectively. For $d = v/15 = 200$, we modify 1334 (≈7.9 %) pixels and derive PSNR = 36.5034 and νMAE = 27.1241. Finally, $d = v/30 = 100$ gives rise to 2211 (≈13 %) modified pixels with PSNR = 40.1747 and νMAE = 19.7688. Finally if we take the opposite hybrid image for d = 500 (i.e., we use the $\bar{u}$ data for all the pixels but those 397 mentioned above), we use ≈97.65 % of the original pixels intensities, but PSNR = 28.0989 (worse than the counterpart test!) while νMAE = 16.9543 keeps improving.

Therefore, separating the image domain in regions A_i of similar intensity values sounds reasonable. This is what we do in this subsection. We again use subdivision, namely at level l we generate 2^l regions $\{A_i^{(l)}\}$ that decompose the intensity interval of $\bar{u}$ into intervals of equal length. In particular, for the B1part$_{3000}$ image

$$A_i^{(l)} := \{j | \bar{u}_j \in (i2^{-l}3000, (i+1)2^{-l}3000)\}, \quad \forall i = 0, \ldots, 2^l - 1. \tag{9}$$

In practice, the original image is not known a priori, so we use the output $u_A^{(0)}$ of the single-constraint Algorithm 1 in (9). The results are summarized in Table 3. Some comments are in order. Due to the subdivision technique, we again have the constraint set inclusion $C_A^{(l)} \subset C_A^{(l-1)}, \forall l \in \mathbb{N}$. Thus, $||\nabla u_A^{(l)}||_{2,1}$ is monotonically increasing with respect to l. We observe that PSNR $(u_A^{(l)})$ is also monotonically increasing with $l \geq 1$, while MAE $(u_A^{(l)})$ is not monotone at $l = 5$. In Sect. 3.1 we tessellated the image domain into 4^l regions, while here we used only 2^l. Therefore, it is reasonable to compare the quality of the l-level output images from Table 2 with the quality of the $2l$-level output images from Table 3. We see that, apart from the PSNR value for the image at level 3 in Table 2, respectively 6 in Table 3, both the PSNR and MAE values improve with intensity tessellation. Especially the MAE value which goes below its optimal value 61.3239 for the single-constraint optimization problem (3) (see Fig. 3). As before, there are indications that high l ($l = 7$ and $l = 8$) may lead to empty constraint sets $C_A^{(l)}$, thus the problem may be ill-posed and the algorithm may not converge. However, no visual artifacts appear (see Fig. 7) and MAE $(u_A^{(l)})$ continues to decrease.

The algorithm depends on the initial choice of image u in (9), and different u give rise to different outputs. We have always used $u_A^{(0)}$ in the experiments above, but we have also tested some of the block-subdivision outputs for higher levels, as well as some of the intensity-tessellation outputs for lower levels. The results are more or less comparable, with $u_A^{(0)}$ seeming to be the best option in general. Last but not least, tuning the regularization parameters σ and ρ was the key for the efficient performance

Table 3 Results of Algorithm 1 on B1part$_{3000}$ for different levels of intensity tessellation. For all levels we set $(\sigma, \rho) = (0.4, 0.3)$, and $\tau_i^{(l)} = \text{card}(A_i^{(l)})$

Level	#iter	TV semi-norm	PSNR	MAE·v
1	20000	1.7080e+6	25.5810	63.6927
2	20000	1.7228e+6	25.7251	62.7592
3	20000	1.7439e+6	25.8739	61.4654
4	50000	1.7535e+6	25.8895	61.2834
5	50000	1.7695e+6	25.9421	61.5316
6	50000	1.7775e+6	25.9923	61.1921
7	50000	1.8046e+6	26.0401	61.0400
8	50000	1.8491e+6	26.1230	60.7815

Fig. 7 The output images $u_A^{(4)}$ (*left*), $u_A^{(7)}$ (*center*), $u_A^{(8)}$ (*right*) from Table 2

of the single-constraint algorithm, while in both Sects. 3.1 and 3.2 the algorithms' convergence rate seems to be slow and independent of that choice, thus we always use $\sigma = 0.4$, $\rho = 0.3$. The same parameters also work for the other images $B1$ and $B2$ on all the considered intensity levels $v = 100, 600, 1200, 2000, 3000$.

The main drawback of the intensity tessellation algorithm is that we have no control on the size of the tessellated regions. It may happen that even at low levels, some of the regions consist of only few points (some of them might be even empty, but this is not a problem). Thus, the law of large numbers may be violated and $\tau_i^{(l)} = \text{card}(A_i^{(l)})$ may be a bad choice that leads to an ill-posed optimization problem or to a minimizer that is quite different from the initial image. A possible remedy is to adaptively split the intensity interval in order to guarantee $\text{card}(A_i^{(l)}) \approx n/2^l$, $\forall i = 0, \ldots, 2^l - 1$. This is left for future work.

3.3 2-Step Combined Tessellation

So far we saw that both block subdivision and intensity tessellation improve the image quality of the output of Algorithm 1. On the other hand, the former technique violates the Central Limit Theorem but allows for the application of the law of large numbers, while the latter one violates the law of large numbers but allows for the application of the Central Limit Theorem. Thus for both of them, the choice $\tau_i^{(l)} = \text{card}(A_i^{(l)})$ for constraint parameters is not theoretically justified and may lead to diverging algorithms or meaningless results. In this subsection, we try to combine the two approaches in a beneficial way. We use 1-level block subdivision together with 3-region-intensity tessellation of the image domain. The idea is the following: the block subdivision performs very well away from the edges, while along them it leads to noise redistribution (Fig. 6). If we assume that the redistribution always involves 2 neighboring pixels, one of law intensity and one of high intensity, and it simply exchanges their noisy values as discussed in Sect. 3.2, then we need to find all such pixel pairs, separate them into 2 regions—one $\bar{A}_0$ of low and one $\bar{A}_1$ of high intensity,

Fig. 8 Edge detection via difference images. *Left* $u_A^{(1)} - f$. *Center* $\bar{u} - f$. *Right* $u_A^{(1)} - u_A^{(0)}$. The images $u_A^{(i)}$, $i = 0, 1$ are taken from Table 2

and introduce a second constrained $||T(Hu) - T(f)||_{\bar{A}_i} \leq \text{card}(\bar{A}_i), \;\; i = 0, 1, 2$. (For the sake of symmetry, we take $\bar{A}_2$ to be the complement of $\bar{A}_0 \cup \bar{A}_1$.)

The operator H "smoothens" the edges of $\bar{u}$, but as long as the blur is not very strong (i.e., corresponds to a Gaussian kernel with close-to-one standard deviation) the edges "survive" it. The Poisson noise more or less preserves what is left from them. The block subdivision also smoothens the edges, but not as much as the blur operator. Thus, both the difference images $\bar{u} - f$ and $u_A^{(l)} - f$ contain enough edge information (see Fig. 8). However, the smoothing effect of the blur operator dominates and we cannot say from the second image where noise redistribution appear. Hence, we prefer to work with the difference image $\partial u := u_A^{(1)} - u_A^{(0)}$. Indeed, the subdivision of the image domain into 4 regions alleviates the noise redistribution effect from $u_A^{(0)}$ and part of the edge information is again visualized (the right image in Fig. 8). More importantly, all the pixels with significantly different $u_A^{(1)}$ and $u_A^{(0)}$ values indeed belong to the edges of $\bar{u}$, making edge-detection a plausible application of such multi-constraint optimization.

We compute $M := \max_i \partial u_i > 0$, $m := \min_i \partial u_i < 0$, fix a number $c > 0$, and set $\bar{A}_0 := \{i | \partial u_i > M/c\}$, $\bar{A}_1 := \{i | \partial u_i < m/c\}$. Then we solve

$$\underset{u \in [0, +\infty)^n}{\text{argmin}} \; ||\nabla u||_{2,1} \quad s.t. \quad \begin{array}{ll} ||T(Hu) - T(f)||_{A_{i,j}^{(1)}} \leq \text{card}\left(A_{i,j}^{(1)}\right), & i, j = 0, 1; \\ ||T(Hu) - T(f)||_{\bar{A}_k} \leq \text{card}(\bar{A}_k), & k = 0, 1, 2. \end{array}$$

We apply a straightforward modification of Algorithm 1, which decouples the two tessellations $\{A_{i,j}^{(1)}\}$ and $\{\bar{A}_k\}$. Results are summarized in Table 4.

We observe that the higher the intensity (thus the sharper the edges) the bigger the improvement in the quality of the result with respect to the corresponding images $u_A^{(1)}$ and $u_A^{(0)}$. For very low intensity ν the combined technique may even worsen the MAE of the output (see $B2_{100}$). The PSNR always improves.

Table 4 Results of the 2-steps combined tessellation for different images and intensity levels. For all examples we set $(\sigma, \rho) = (0.4, 0.3)$, and $c = 9$

Image	#iter	TV semi-norm	PSNR	MAE·v	TV($u^{(1)}$)	PSNR($u^{(1)}$)/PSNR($u^{(0)}$)	MAE($u^{(1)}$)/MAE($u^{(0)}$)
B1part_{3000}	100000	1.7573e+6	26.0000	60.9231	1.7194e+6	25.6957/25.5934	62.8892/63.6238
B1_{100}	20000	1.2602e+5	24.4036	3.0603	1.2312e+5	24.3061/24.2844	3.0659/3.0761
B1_{600}	20000	9.0592e+5	25.6788	15.2529	8.9419e+5	25.5992/25.5738	15.3439/15.4200
B1_{1200}	20000	1.9653e+6	26.2329	28.6136	1.9339e+6	26.1012/26.0742	28.8040/29.0003
B1_{2000}	20000	3.3570e+6	26.4457	46.1726	3.3308e+6	26.3738/26.3476	46.3732/46.6200
B1_{3000}	20000	5.2662e+6	26.7679	66.7055	5.2092e+6	26.6491/26.6338	67.1725/67.5230
B2_{100}	20000	9.9370e+4	19.9442	6.1998	9.3734e+4	19.8992/19.8949	5.9867/5.9804
B2_{600}	20000	7.5496e+5	21.9439	23.7810	7.4231e+5	21.8235/21.8230	23.8747/23.8499
B2_{1200}	20000	1.5802e+6	22.4779	42.5783	1.5561e+6	22.3595/22.3457	43.0012/43.0531
B2_{2000}	20000	2.7236e+6	22.9163	64.9695	2.6747e+6	22.7625/22.7535	66.1393/66.1365
B2_{3000}	20000	4.1120e+6	23.1652	93.2223	4.0658e+6	23.0049/22.9832	94.8958/94.9391

4 Conclusions

The constraint sets (5) for the optimization problems (3) and (4) are too large, thus their minimizers tend to oversmooth the image. We experimented with various restriction techniques on C_A, C_I.

Simply decreasing τ_A, τ_I does improve the image quality of the output at the beginning, but we deviate from their statistical estimations and need to guess their optimal values, which is computationally very expensive. Moreover, those optimal values depend on the quality measures we consider, differ significantly from minimizing MAE to maximizing PSNR, and does not necessary lead to an output with good visual properties.

Another option, suggested in [8] as a future work direction, is to consider multi-constraint optimization. We investigated such approach, within the framework (7) and (8). We considered spatial, intensity, and mixed domain decompositions of the image, and summarized the numerical results in Tables 2, 3 and 4.

In all the setups, we observed that the image quality of the output improved up to a certain level. After that, the optimization problems became ill-posed, the algorithms' convergence was unclear, and artifacts appeared. This effect was caught by the MAE output values, but not by the PSNR ones, which still increased. Multiple constraints slowed down Algorithm 1, and its convergence rate was poor, independent on the choice of the accelerators σ, ρ. Therefore, parallel implementation of Algorithm 1 is practically important and is an object of ongoing work. While the problems (7) and (8) were well-posed, the close similarity between their solutions u_A, u_I, numerically observed in [8], hold true. Hence, their difference image $u_A - u_I$ can be used as a criterion for well-posedness.

Acknowledgments This research is supported by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Program.

References

1. Dupé, F.X., Fadili, J., Starck, J.L.: A proximal iteration for deconvolving Poisson noisy images using sparse representations. IEEE Trans. Image Process. **18**(2), 310–321 (2009)
2. Zanella, R., Boccacci, P., Zanni, L., Bertero, M.: Efficient gradient projection methods for edge-preserving removal of Poisson noise. Inverse Prob. **25**(4), 045010 (2009)
3. Figueiredo, M.A.T., Bioucas-Dias, J.M.: Restoration of Poissonian images using alternating direction optimization. IEEE Trans. Image Process. **19**(12), 3133–3145 (2010)
4. Setzer, S., Steidl, G., Teuber, T.: Deblurring Poissonian images by split Bregman techniques. J. Vis. Commun. Image Represent. **21**(3), 193–199 (2010)
5. Mäkitalo, M., Foi, A.: Optimal inversion of the Anscombe transformation in low-count Poisson image denoising. IEEE Trans. Image Process. **20**(1), 99–109 (2011)
6. Carlavan, M., Blanc-Féraud, L.: Sparse Poisson noisy image deblurring. IEEE Trans. Image Process. **21**(4), 1834–1846 (2012)

7. Jezierska, A., Chouzenoux, E., Pesquet, J.-C., Talbot, H.: A primal-dual proximal splitting approach for restoring data corrupted with Poisson-Gaussian noise. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2012), pp. 1085–1088, Kyoto, Japan (2012)
8. Harizanov, S., Pesquet, J.C., Steidl, G.: Epigraphical projection for solving least squares Anscombe transformed constrained optimization problems. In: Kuijper et al., A. (eds.) Scale-Space and Variational Methods in Computer Vision (SSVM 2013), LNCS 7893, pp. 125–136, Springer, Berlin (2013)
9. Teuber, T., Steidl, G., Chan, R.H.: Minimization and parameter estimation for seminorm regularization models with *I*-divergence constraints. Inverse Prob. **29**, 1–28 (2013)
10. Li, J., Shen, Z., Yin, R., Zhang, X.: A reweighted ℓ_2 method for image restoration with Poisson and mixed Poisson-Gaussian noise. Inverse Prob. Imaging **9**(3), 875–894 (2015)
11. Ciak, R., Shafei, B., Steidl, G.: Homogeneous penalizers and constraints in convex image restoration. J. Math. Imaging Vis. **47**(3), 210–230 (2013)
12. Anscombe, F.J.: The transformation of Poisson, binomial and negative-binomial data. Biometrika **35**, 246–254 (1948)
13. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**, 259–268 (1992)
14. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
15. Pock, T., Chambolle, A., Cremers, D., Bischof, H.: A convex relaxation approach for computing minimal partitions. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 810–817 (2009)
16. Chierchia, G., Pustelnik, N., Pesquet, J.-C., Pesquet-Popescu, B.: Epigraphical projection and proximal tools for solving constrained convex optimization problems. SIViP **9**(8), 1737–1749 (2015)
17. Bardsley, J.M., Goldes, J.: Regularization parameter selection methods for ill-posed Poisson maximum likelihood estimation. Inverse Prob. **25**(9), 095005 (2009)