DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-014

Robust extensions to guided image filtering

Oleg V. Michailovich; University of Waterloo; Waterloo, Ontario/Canada

Abstract

Image denoising is commonly regarded as a problem of fundamental importance in imaging sciences. The last few decades have witnessed the advent of a wide spectrum of denoising algorithms, capable of dealing with noises and images of various types and statistical natures. It is usually the case, however, that the effectiveness of a given denoising procedure and the complexity of its numerical implementation increase pro rata, which is often the reason why more advanced solutions are avoided in situations when data images have relatively large sizes and/or acquired at high frame rates. As a result, substantial efforts have been recently extended to develop efficient means of image denoising, the computational complexity of which would be comparable to that of standard linear filtering. One of such solutions is Guided Image Filtering (GIF) - a recently proposed denoising technique, which combines outstanding performance characteristics with real-time implementability. Unfortunately, the standard implementation of GIF is known to perform poorly in situations when noise statistics deviate from that of additive Gaussian noise. To overcome this deficiently, in this note, we propose a number of modifications to the filter, which allow it to achieve stable and accurate results in the case of impulse and Poisson noises.

Introduction

Despite recent advances in the design and manufacturing of imaging devices and related methodologies, the problem of image denoising still remains a problem of fundamental importance in imaging sciences and associated disciplines. The most archetypical question asked in this respect has always been: "Is there an *optimal* filter capable of rejecting the noise, while retaining the content of the original image as intact as possible?" Unfortunately, an ultimate answer to this question does not seem to exist, just as there may hardly ever be a unified agreed upon what the "optimal" means [1]. This being said, however, specific applications can still put forward certain technical requirements, with respect to which some filtering algorithms may be deemed to be more appropriate than others. In particular, when one is facing the necessity to processes images of extremely large sizes (as it would be the case, for instance, in digital pathology [2]), it makes practical sense to favour computationally efficient solutions. From this perspective, a less naïve question to ask would be: "What is the best performer among filters that have, say, linear complexity?"

Assuming the "best" to refer to the edge-preservation properties of a filter, the above question has been addressed by many authors [3, 4]. Recently, the arsenal of computationally efficient methods of image denoising was supplemented by an additional tool, known as *Guided Image Filtering* (GIF) [5]. Since the introduction of the original concept, GIF has undergone several modifications, which have enabled its application to a broad spectrum of important practical problems [6, 7]. Despite its apparent success, however, one particular aspect of GIF still seems to be unaddressed. Specifically, from the viewpoint of Bayesian estimation theory, GIF can be considered to be a maximum-a-posteriori (MAP) estimator derived under the assumption on noise to be additive, white, and Gaussian (AWG). Supporting this theoretical argument are numerous empirical observations, which reveal suboptimal performance of the filter in the cases when the noise happens to deviate from its expected (i.e., AWG) behaviour.

To address the above problem, we propose to modify the original formulation of GIF to improve its performance in two practically important cases, *viz.* when the images of interest are contaminated by: a) impulsive and b) Poisson (aka shot) noises. To this end, we take advantage of the Bayesian formalism, which leads to two optimization problems, neither of which admits a closed form solution (as opposed to the case of the original GIF). However, for each of the two problems, we propose a particularly fast recursive solution, which requires only a few applications of the standard GIF, thereby resulting in only a mild increase in the overall computational complexity of image filtering. To ensure the reproducibility of our results, we detail the structures of both filters and test their performances in terms of both perceptual signal-to-noise ratio (PSNR) and structural similarity (SSIM) index [8].

Guided image filtering

In its original version, GIF operates on a (noisy) data image g under the guidance of a reference image I. Without loss of generality, we can consider both g and I to be real-valued 2-D arrays of size $N \times M$. Let k be a 2-tuple index pointing to some arbitrary (yet fixed) elements of g. Also, let $\mathscr{N}_k$ be a local neighbourhood of k. Then, GIF is based on representing the values of the original (noise-free) image f in terms of the values of I as given by [5]

$$f_i = a_k I_i + b_k, \quad \forall i \in \mathscr{N}_k, \tag{1}$$

where a_k and b_k are filter parameters which are yet to be defined. In words, (1) suggests that each $\mathscr{N}_k$ neighbourhood of the original image f is an affine function of the corresponding neighbourhood of the guiding image I.

The model parameters a_k and b_k in (1) can be estimated using the data image g. Specifically, to facilitate the exposition, let $K := |\mathscr{N}_k|$ denote the number of pixels in $\mathscr{N}_k$, and let $\bar{g}^k$ and $\bar{I}^k$ be K-dimensional (columns) vectors obtained by column-stacking the values of $\{g_i\}_{i\in\mathscr{N}_k}$ and $\{I_i\}_{i\in\mathscr{N}_k}$, respectively. Then, the optimal a_k and b_k can be found by solving the quadratic minimization problem of the form

$$\min_{\bar{c}^k} \left\{ \|A^k \bar{c}^k - \bar{g}^k\|_2^2 + \varepsilon\, \bar{c}^{kT} B \bar{c}^k \right\}, \tag{2}$$

where $\bar{c}^k := [a_k, b_k]^T$, $A^k := \left[\bar{I}^k, \mathbf{1}_K\right] \in \mathbb{R}^{K\times 2}$ (with $\mathbf{1}_K$ being a K dimensional vector of ones), $B := \mathrm{diag}\{[1,0]\} \in \mathbb{R}^{2\times 2}$, and $\varepsilon > 0$ is a user-defined parameter that controls the level of smoothing

DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-014

applied by the filter. The solution to (2) is straightforward to compute in a closed form. In particular, the optimal values of the GIF parameters are given by

$$a_k^* = \frac{\mathrm{cov}(\bar{g}^k, \bar{I}^k)}{\mathrm{var}(\bar{I}^k) + \varepsilon'}, \quad b_k^* = \mathrm{mean}(\bar{g}^k) - \mathrm{mean}(\bar{I}^k)\, a_k^*, \tag{3}$$

with $\varepsilon' = K^{-1}\varepsilon$ and "mean" standing for the mean value of a vector (e.g., $\mathrm{mean}(\bar{I}^k) := (1/K)\sum_{i\in\mathscr{N}_k} \bar{I}_i^k$), while "cov" and "var" are used to denote the sample covariance and variance of corresponding vectors, respectively. Specifically,

$$\mathrm{cov}(\bar{g}^k, \bar{I}^k) = \frac{1}{K}\sum_{i\in\mathscr{N}_k} \bar{g}_i^k \bar{I}_i^k - \mathrm{mean}(\bar{I}^k)\,\mathrm{mean}(\bar{g}^k), \tag{4}$$

whereas $\mathrm{var}(\bar{I}^k) = \mathrm{cov}(\bar{I}^k, \bar{I}^k)$.

In this work, we are particularly interested in running GIF in the *self-guided* regime, i.e., when I is replaced by the data image g. In this case, the cost of computing a_k^* and b_k^* is dominated by the cost of computing $\mathrm{mean}(\bar{g}^k)$ and $\mathrm{var}(\bar{g}^k)$, which amounts to two convolutions. To see that, we assume $\mathscr{N}_k$ to be a rectangular neighbourhood of size $L_N \times L_M$ and choose h to be a convolution kernel defined by $h = (L_N L_M)^{-1}\mathbf{1}_{L_N}\mathbf{1}_{L_M}^T$. Then, the optimal values of a_k and b_k corresponding to *all* k in the domain of g can be arranged into two $N \times M$ matrices a^* and b^*, which in turn can be computed using Algorithms 1 specified below.

Algorithm 1 Self-Guided Image filtering (SGIF)

procedure SGIF(g, ε')
$g_m \leftarrow g * h$
$g_v \leftarrow (g \cdot g) * h - g_m \cdot g_m$
$a^* \leftarrow g_v / (g_v + \varepsilon')$
$b^* \leftarrow g_m \cdot (a^* - 1)$
return a^*, b^*

Above, the symbols $\cdot$ and $/$ denote point-wise multiplication and division, while $*$ stands for the operation of 2-D convolution. Note that, due to the separability of h, the 2-D convolution can be reduced to a series of 1-D convolutions along the row and column directions using kernels $h_N = (L_N)^{-1}\mathbf{1}_{L_N}$ and $h_M = (L_M)^{-1}\mathbf{1}_{L_M}$, respectively. Moreover, substantially better filtering results are usually obtained when the "box" kernels are replaced with "bell-shaped" ones. Thus, for instance, the results reported in this paper have been obtained using $h_N = h_M = (1/16)[1, 4, 6, 4, 1]^T$ [9].

It was argued in [5], that each pixel i is included in a total of K different neighbourhoods $\mathscr{N}_k$, and thus the value of f_i in (1) may not be the same when computed for different a_k and b_k. A possible solution to the above difficulty (as proposed in [5]) is to compute the filter output $\tilde{f}$ according to

$$\tilde{f} = (a^* * h) \cdot g + (b^* * h). \tag{5}$$

Unfortunately, while useful in general, the above approach tends to over smooth the images filtered in the self-guided regime. For this reason, in our experiments, the estimation has been performed according to

$$\tilde{f} = a^* \cdot g + b^*, \tag{6}$$

which corresponds to the case when a_k^* and b_k^* computed for various neighbourhoods $\mathscr{N}_k$ are used *only* for correcting the value of g_i at their respective centres.

Proposed method

The minimization problem (2) admits an alternative interpretation from the viewpoint of statistical estimation theory. Thus, the first term of the objective function could be interpreted as a (negative) log-likelihood, while the second term reflects some *a priori* knowledge on the statistical nature of an optimal c. Moreover, the quadratic form of the log-likelihood term suggests that the noise is assumed to obey an AWG model. Unfortunately, using such model does not guarantee satisfactory performance of the filter in the case of other types of noise which require a different formulation. In what follows, we consider two practically important scenarios, in which one deals with impulsive and shot noises.

Impulse noises

In the case of impulse noise, a more appropriate formulation of GIF could be obtained through replacing the ℓ_2-norm in (2) with an ℓ_1-norm, which leads to computation of the filter coefficients by solving

$$\min_{\bar{c}^k} \left\{ \|A^k \bar{c}^k - \bar{g}^k\|_1 + \varepsilon\, \bar{c}^{kT} B\, \bar{c}^k \right\}. \tag{7}$$

Although (7) is still a convex optimization problem, its solution is not available in a closed form. Moreover, even though this problem could be solved by a great variety of efficient optimization tools, doing so in a "k-by-k" manner would greatly impair the computational efficiency of the filter, which is, after all, the key attribute that we are interested to retain. Accordingly, in what follows, we detail an optimization routine that allows computing an optimal $\tilde{f}$ in a few iterations, with the cost of one iteration being virtually identical to that of a single application of SGIF.

The main idea of the proposed approach is rooted in using a particular method of convex optimization known as alternating directions method of multipliers (ADMM) [10]. In particular, we start with replacing the original problem (7) with an equivalent constrained minimization problem of the form

$$\min_{\bar{c}^k, \bar{z}^k} \left\{ \|\bar{z}^k\|_1 + \varepsilon\, \bar{c}^{kT} B\, \bar{c}^k \right\}, \quad \text{s.t. } \bar{z}^k = A^k \bar{c}^k - \bar{g}^k, \tag{8}$$

where $\bar{z}^k \in \mathbb{R}^K$ is an auxiliary optimization variable. The new problem can, in turn, be cast in its *augmented* Lagrangian form and solved iteratively using the following set of update equations

$$\bar{c}_{t+1}^k = \arg\min_{\bar{c}^k} \left\{ \varepsilon\, \bar{c}^{kT} B\, \bar{c}^k + \frac{\delta}{2}\|A^k \bar{c}^k - \bar{g}^k - \bar{z}_t^k + \bar{y}_t^k\|_2^2 \right\}, \tag{9}$$

$$\bar{z}_{t+1}^k = \arg\min_{\bar{z}^k} \left\{ \|\bar{z}^k\|_1 + \frac{\delta}{2}\|A^k \bar{c}_{t+1}^k - \bar{g}^k - \bar{z}^k + \bar{y}_t^k\|_2^2 \right\}, \tag{10}$$

$$\bar{y}_{t+1}^k = \bar{y}_t^k + A^k \bar{c}_{t+1}^k - \bar{g}^k - \bar{z}_{t+1}^k, \tag{11}$$

where subscript t is an iteration counter, $\bar{y}_t^k \in \mathbb{R}^K$ is a vector of (scaled) Lagrange multipliers, and $\delta > 0$ is a user-defined penalty parameter.

The minimization problems in (9) and (10) admit closed-form solutions. Thus, the computation of $\bar{c}_{t+1}^k$ amounts to solving a 2-by-2 system of equations of the form

$$\left(\delta A^{kT} A^k + 2\varepsilon B\right) \bar{c}_{t+1}^k = \delta A^{kT}\left(\bar{z}_t^k + \bar{g}^k - \bar{y}_t^k\right), \tag{12}$$

©2016 Society for Imaging Science and Technology
DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-014

while computing $\bar{z}_{t+1}^k$ boils down to standard *soft thresholding*, namely

$$\bar{z}_{t+1}^k = \text{soft}\{\bar{r}^k, 1/\delta\} := \max(\bar{r}^k - 1/\delta, 0) + \min(\bar{r}^k + 1/\delta, 0), \tag{13}$$

where we denote $\bar{r}^k := A^k \bar{c}_{t+1}^k - \bar{g}^k + \bar{y}_t^k$ for brevity. Although very efficient from computational perspective, the above update equations need to be computed for each image neighbourhood (as indexed by k) and each iteration t. Moreover, the algorithm also requires us to store and manipulate two $N \times M$ arrays of K-dimensional vectors $\{\bar{z}_t^k\}$ (auxiliary optimization variables) and $\{\bar{y}_t^k\}$ (scaled Lagrange multipliers). Needless to say, these additional storage requirements further diminish the overall efficiency of the procedure.

To restore the computational efficiency of GIF in the case of impulsive noises, we first note that the computation in (9) effectively amounts to *guided* filtering of $\bar{g}^k + \bar{z}_t^k - \bar{y}_t^k$, while using $\bar{g}^k$ as a *guiding* image patch and properly rescaling the regularization constant (i.e., using $2\varepsilon/\delta$ instead of ε). Although very efficient in principle, the main drawback of the above procedure stems from the necessity of storing *different* vectors $\bar{z}_t^k$ and $\bar{y}_t^k$ for each local neighbourhood $\mathcal{N}_k$. However, the shift-invariant nature of both GIF and soft-thresholding in (10) suggests that $\bar{z}_t^k$ and $\bar{y}_t^k$ can be defined in a manner similar to $\bar{g}^k$. In particular, by letting z_t and y_t be two $N \times M$ *images* of the auxiliary optimization variables and the scaled Lagrange multipliers, respectively, the elements of $\bar{z}_t^k$ and $\bar{y}_t^k$ can be shown to be given by $\{z_{t,i}\}_{i \in \mathcal{N}_k}$ and $\{y_{t,i}\}_{i \in \mathcal{N}_k}$. As a result, the update in (9) can be performed by means of the standard GIF algorithm of [5] with g and $g + z_t - y_t$ regarded as the guiding and data images, correspondingly.

Algorithm 2 Laplacian SGIF (LSGIF)

procedure LSGIF(g, ε', δ)
$z \leftarrow 0$
$y \leftarrow 0$
repeat until convergence
$(a, b) \leftarrow \text{SGIF}(g + z - y, 2\varepsilon'/\delta)$
$\tilde{f} \leftarrow a \cdot g + b$
$z \leftarrow \text{soft}\{\tilde{f} - g + y, 1/\delta\}$
$y \leftarrow y + \tilde{f} - g - z$
end repeat
return $\tilde{f}$

To further improve the computational efficiency of the proposed solution, the filtering of $g + z_t - y_t$ can be carried our in the "self-guided" regime, i.e., by exploiting $g + z_t - y_t$ for its own guidance. This substitution corresponds to, so-called, inexact update [11] (a formal convergence analysis of which is omitted here for the reason of space). It was observed through numerical experiments, however, that the above "trick" does not impair the convergence of the algorithm, while being capable of substantially improving its computational efficiency.

The conceptual outline of the proposed algorithm, which will be referred below to as *Laplacian self-guided image filtering* (LSGIF) (owing to the Laplacian nature of impulsive noises), is detailed in Algorithm 2 above. One can see that the computational complexity of each iteration of the algorithm is dominated by that of a standard SGIF procedure. In practice, we have observed that it rarely takes more than ten iterations for the algorithm to reach convergence (which, in turn, allows processing large scale data at a fraction of a second even using an average CPU). It also deserves noting that the convergence properties of the proposed method could be further improved using standard means for accelerating ADMM-based solvers, such as those based on Nesterov's scheme and/or adaptive restart [12].

Poisson noises

The image formation model of many key imaging modalities rely on the notion of event counts. The latter, for example, quantifies the number of gamma photons which pass though a single slit of the collimator of a gamma camera in positron emission tomography (PET) and single photon emission computer tomography (SPECT) [13]. In addition, statistical models of the same type are routinely used in optics to account for the process of "counting" the number of optical photons registered by the sensor of a (CCD) camera, which makes them ubiquitous throughout confocal microscopy, astronomical imaging, and turbulent imaging. In all the above case, data images are normally assumed to be contaminated by Poisson noise [14].

In the case of Poisson noises, the Gaussian log-likelihood term $\|A^k \bar{c}^k - \bar{g}^k\|_2^2$ needs to be replaced by its Poissonian counterpart, which transforms (2) into a minimization problem of the form

$$\min_{\bar{c}^k} \left\{ \mathbf{1}_K^T A^k \bar{c}^k - \bar{g}^{kT} \log(A^k \bar{c}^k) + \varepsilon \bar{c}^{kT} B \bar{c}^k \right\}, \tag{14}$$

where, as before, $\mathbf{1}_K$ denotes a K-length (column) vector of ones. Note that the model assumes the true image to be strictly positive, and thus the above minimization problem should be accompanied by a constraint requiring $A^k \bar{c}^k \succ 0$ (with element-wise inequalities implied). We will see, however, that the proposed solution will be able to automatically enforce this constraint.

To derive an efficient method for solving (14), we start with replacing the original problem by an equivalent problem of the form

$$\min_{\bar{c}^k, \bar{z}^k} \left\{ \mathbf{1}_K^T \bar{z}^k - \bar{g}^{kT} \log(\bar{z}^k) + \varepsilon \bar{c}^{kT} B \bar{c}^k \right\}, \quad \text{s.t. } \bar{z}^k = A^k \bar{c}^k, \tag{15}$$

with $\bar{z}^k \in \mathbb{R}^K$ being an auxiliary optimization variable, as before. Using ADMM allows solving the problem in (15) through a sequence of iterations given by

$$\bar{c}_{t+1}^k = \arg\min_{\bar{c}^k} \left\{ \varepsilon\, \bar{c}^{kT} B\, \bar{c}^k + \frac{\delta}{2} \|A^k\, \bar{c}^k - \bar{z}_t^k + \bar{y}_t^k\|_2^2 \right\}, \tag{16}$$

$$\bar{z}_{t+1}^k = \arg\min_{\bar{z}^k} \left\{ \mathbf{1}_K^T \bar{z}^k - \bar{g}^{kT} \log(\bar{z}^k) + \frac{\delta}{2} \|A^k\, \bar{c}_{t+1}^k - \bar{z}^k + \bar{y}_t^k\|_2^2 \right\}, \tag{17}$$

$$\bar{y}_{t+1}^k = \bar{y}_t^k + A^k\, \bar{c}_{t+1}^k - \bar{z}_{t+1}^k, \tag{18}$$

where, as before, $\bar{y}_t^k \in \mathbb{R}^K$ is a (column) vector of scaled Lagrange multipliers at iteration t.

Similarly to the case of impulsive noises discussed in the previous section, the subproblems (16) and (17) admit simple closed-form solutions. Specifically, the computation of $\bar{c}_{t+1}^k$ boils down to solving a 2-by-2 system of equations, virtually identical to the

DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-014

one in (12). At the same time, the solution of (17) is considerably simplified by the fact that this optimization problem is *separable* in the coordinates of its argument, which suggests that updating $\bar{z}^k$ can be performed by independently updating its K entries. In particular, application of the first-order optimality condition to (17) results in the following update equation.

$$\bar{z}_k^{t+1} = \frac{\bar{r}^k + \sqrt{(\bar{r}^k)^2 + 4\delta\bar{g}^k}}{2\delta}, \text{ with } \bar{r}^k := \delta(A^k\bar{c}_t^k + \bar{y}_t^k) - 1. \quad (19)$$

Here the operation of squaring as well as taking the square root are assumed to be performed element-wise, which makes this update somewhat similar to soft-thresholding in (13). To make the notations more compact, we will refer to the above update equation as positive rectifier (denoted by rect_+), owing to its associated functional characteristics. Specifically,

$$\bar{z}_k^{t+1} = \text{rect}_+(\bar{r}^k, \bar{g}^k, \delta). \quad (20)$$

It is important to note that, as the name suggests, the output of rect_+ is always guaranteed to be nonnegative-valued, which explicitly enforces the resulting estimate to be nonnegative-valued as well.

Algorithm 3 Poissonian SGIF (PSGIF)

procedure PSGIF(g, ε', δ)
$z \leftarrow 0$
$y \leftarrow 0$
repeat until convergence
$(a, b) \leftarrow \text{SGIF}(z - y, 2\varepsilon'/\delta)$
$\tilde{f} \leftarrow a \cdot g + b$
$z \leftarrow \text{rect}_+\{\delta(\tilde{f} + y) - 1, g, \delta\}$
$y \leftarrow y + \tilde{f} - z$
end repeat
return $\tilde{f}$

Finally, we can improve the computational efficiency of the filter using the same approach as in the case of LSGIF. Specifically, instead of two sets $\{\bar{z}_t^k\}_{k=1}^{NM}$ and $\{\bar{y}_t^k\}_{k=1}^{NM}$ of K-dimensional vectors of the auxiliary optimization variable and their related (scaled) Lagrange multipliers, we define two $N \times M$ images z_t and y_t, such that the values of $\bar{z}_t^k$ (resp. $\bar{y}_t^k$) are given by the values of $\{\bar{z}_{t,i}\}_{i\in\mathbb{Z}_k}$ (resp. $\{\bar{y}_{t,i}\}_{i\in\mathbb{Z}_k}$). Also, instead of filtering $z_t - y_t$ under the guidance of g (as suggested by (16)), we carry out the filtering in the self-guided regime. As was already argued earlier, this "trick" is equivalent to using an inexact update, which can be shown to produce an absolutely summable sequence of errors, which in turn guarantees the convergence of the filtering procedure [11].

A conceptual outline of the proposed method – which will be referred below to as *Poissonian self-guided image filtering* (PSGIF) – is given by Algorithm 3 above. Similarly to the case of LSGIF, each iteration of PSGIF is dominated by a standard SGIF procedure, with a total number of iterations rarely exceeding a tensome. Moreover, although already very efficient computationally, the convergence of PSGIF can be further improved through employing available schemes for accelerating ADMM-based optimization routines [12]. This being said, however, we stress the fact that, due to the high computational efficiency of SGIF, a typical execution of either LSGIF or PSGIF takes only a fraction of a second, when applied to high-dimensional images (e.g., $\min(N, M) \geq 1024$) and processed on an average CPU.

Results

In this section, we provide empirical evidence supporting the viability and usefulness of the proposed filtering solutions. In particular, we demonstrate the performance of LSGIF and PSGIF in the cases of image contamination by impulsive and shot noises, respectively. For the sake of comparative analysis, the proposed methods are contrasted with the standard GIF approach of [5] as well as two additional filters derived using a Bayesian formulation in conjunction with the powerful ideas of *sparse representations* [15]. In this case, the original (unknown) image f is assumed to be representable as $f = Wc$, with W being a *synthesis* operator corresponding to either a predefined basis or a frame in the associated signal space, and c being a sparse set of representation coefficients. The arsenal of currently available choices of W is impressively broad [16]. In this work, we choose W to correspond to the framelet analysis of [17], which we choose to its excellent orientational selectivity and sparsifying properties.

Assuming the framelet coefficients to be *i.i.d.* Laplacian random variables and the measured image g to be contaminated by additive white Laplacian noise (in congruence with the impulsive nature of the noise), the maximum-a-posteriori (MAP) estimation framework prescribes finding an optimal c^* as the global minimizer of the following optimization problem

$$c^* = \arg\min_c \{\|Wc - g\|_1 + \lambda\|c\|_1\}, \quad (21)$$

where $\lambda > 0$, conventionally regarded as a regularization parameter, depends on the parameters of the probability distribution of the framelet coefficients as well as that of the noise. In practice, however, a suitable value of λ is found by trials and errors, depending on the balance between noise suppression and preservation of image details required by a specific application at hand. In the case of this study, λ was set to be equal to 0.7, which resulted in the best performance of the filter in terms of PSNR and SSIM. For the convenience of referencing, the above method will be referred below as Sparse Bayesian with Laplacian noises (SB-L).

In the case of Poisson (aka shot) noise and the same sparse (Laplacian) priors on c, the MAP estimation leads to an optimization problem of the form

$$\min_c \{\langle \mathbf{1}_{N\times M}, Wc\rangle - \langle g, \log(Wc)\rangle + \lambda\|c\|_1\}, \text{ s.t. } Wc \succ 0, \quad (22)$$

where $\langle\cdot,\cdot\rangle$ denotes the standard (Euclidean) inner product, while $\mathbf{1}_{N\times M}$ is an $N \times M$ array of ones. As before, the optimal value of λ was chosen to be 0.1, which yielded the best possible performance of the method in terms of PSNR and SSIM. In what follows, the method is referred to as Sparse Bayesian with Poisson noises (SB-P).

The framelet analysis required by both SB-L and SB-P was based on a 5-level multi-resolution (with an associated redundancy factor of approximately 2.65), implemented by means of fast decomposition/reconstruction routines based on FIR filtering in conjunction with up/down sampling. The minimizations in (21) and (22) were carried out using basic ADMM-based optimization routines, without any attempts of additional acceleration

DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-014

(just like in the case of LSGIF and PSGIF) [18]. All the methods under comparison (except for SGIF) used the same termination criterion. Namely, the iterations were terminated when a relative change in the value of associated cost function dropped below 1%.

The proposed and reference methods have been compared in terms of PSNR, SSIM, as well as their respective execution times. (All the codes have been written and implemented in Matlab® (version R2015a) on a standard 2.6 GHz Intel Core i7 CPU). The test image used in our experimental study and shown in Fig. 1 represented a typical fragment of a digital pathology scan produced by a confocal microscope.

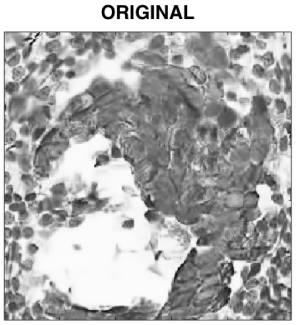

Figure 1. *Original test image.*

Table 1 summarizes the performance metrics of various filtering methods under comparison, obtained by averaging the results of 200 independent trials. In this case, the data image has been contaminated by salt & pepper noise (with the noise density equal to 0.1), giving rise to an (average) PSNR of 14.92 dB and SSIM of 0.39. One can see that the proposed LSGIF method is capable of achieving a higher estimation accuracy in less than 40 ms (as compared to roughly 3 secs required by SB-L). Note that it took LSGIF only about 10 iterations to converge till termination, which explains an approximately 10 fold increase in the execution time as compared to the case of SGIF. The parameters ε and δ of the method were set to be 4 and 5, respectively. It should be noted that, according to the theory, the convergence of ADMM is guaranteed for any $\delta > 0$, although the *rate* of convergence can differ for different δ. Even though there exist a number of numerical recipes allowing one to set the value of δ automatically, in this study, it was set empirically to produce the fastest possible convergence. Similarly, the value of $\varepsilon = 4$ was chosen so as to maximize the resulting values of PSNR and SSIM (just as in the case of SGIF, SB-L, and SB-P).

Table 1: Salt & pepper noises

	Data	GIF	SB-L	LSGIF
PSNR (dB)	14.92	22.08	25.18	**27.27**
SSIM	0.39	0.66	0.88	**0.92**
Time (ms)	-	3.7	3009	**38.4**

It deserves noting that the main goal of this study has been to improve the performance of GIF in the case of non-Gaussian noises (rather than formulating a novel filtering method, capable of "beating" some advanced image filters, such as SB-L). So, the authors have been genuinely surprised to see that, in addition to retaining the computational efficiency of SGIF, the proposed method has been able to actually outperform SB-L (which is based on much deeper considerations of harmonic analysis and statistical estimation theory). This conclusion is further supported by Fig. 2, which shows the results of various filters under comparison. One can see that SGIF retains a considerable portion of the noise, rending the resulting reconstruction inadequate. SB-L, on the other hand, produces a close-to-perfect result, albeit of a lower contrast (probably, due to the effect of residual noises). At the same time, the estimate computed by the proposed LSGIF method is virtually indistinguishable from the original image.

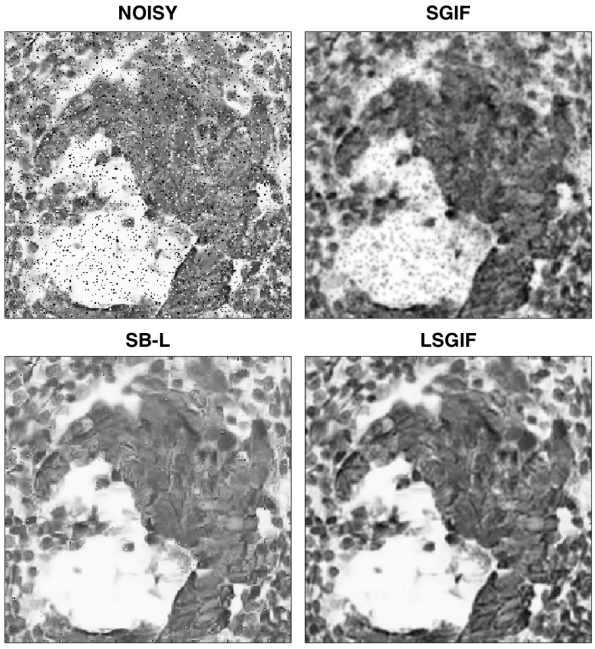

Figure 2. *Recovered images in the case of salt & pepper noises*

The performance of PSGIF in the case of Poisson (aka shot) noise contamination is summarized in Table 2 and Fig. 3. Once again, one can see that PSGIF not only retains the computational efficiency of SGIF (incurring only a ten-fold increase in the execution time, i.e., proportionally to the number of iterations), but also outperforms the other reference methods in terms of PSNR and SSIM. As in the case of LSGIF, the parameters ε and δ of the algorithm were adjusted empirically (so as to maximize its performance both in terms of accuracy and the rate of convergence) and set to 4 and 0.02, respectively. Note, however, that similar optimization has been done for all the methods under comparison.

Table 2: Poissonian noises

	Data	GIF	SB-P	PSGIF
PSNR (dB)	13.83	14.93	24.84	**25.49**
SSIM	0.63	0.67	0.81	**0.87**
Time (ms)	-	3.7	1522	**39.5**

Fig. 3 provides additional evidence in support of the viability of the proposed solution. In particular, while SGIF is incapable of rejecting the shot noise to any significant degree, a much better

DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-014

reconstruction is achieved by SB-P. However, the latter is clearly outperformed by PSGIF in terms of improved contrast and preservation of image details.

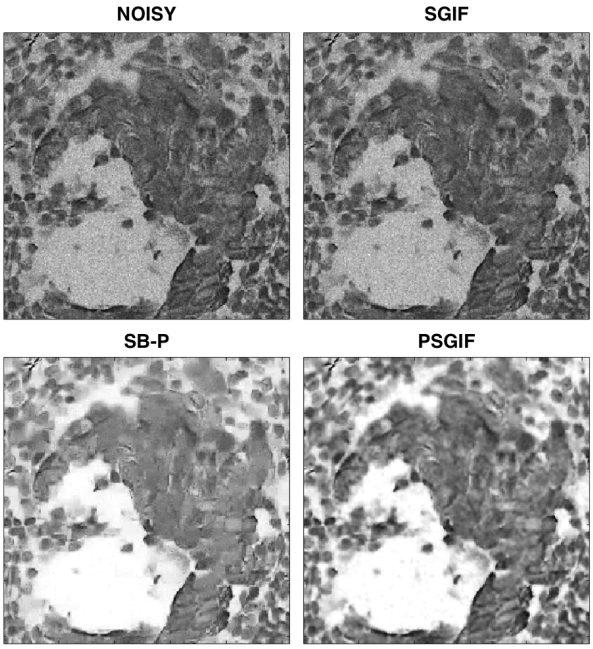

Figure 3. *Recovered images in the case of Poissonian noises*

Summary

In this paper, we proposed a number of modifications to guided image filtering which allow it produce useful results in the case of some common non-Gaussian noises. In particular, the proposed extensions, dubbed LSGIF and PSGIF, have been endowed with the ability to reject impulsive and shot noises, respectively. Note that both the derivation and experimental validation of the filters have been carried out in the *self-guided* regime (i.e., when the noisy image itself is used as guidance). However, the proposed framework is rather general, and is straightforward to extend to the cases when a guiding image is provided along with the noisy data image. Also, the main idea of this work has been to retain the relatively high computational efficiency of GIF, while enabling the latter to tackle broader classes of measurement noises. Although the proposed solutions are iterative in nature, their iterations are dominated by a single application of SGIF, which suggests that the overall complexity of the proposed algorithm scales linearly with the number of iterations, which rarely exceeds a few tens. Although unexpected, it was also observed that, while retaining the computational efficiency of GIF, the proposed solutions is capable of outperforming some advanced Bayesian filters (e.g., the one based on sparse approximation analysis) in terms of both PSNR and SSIM.

Acknowledgments

The author expresses his gratitude for the support provided by the Canadian Institute of Health Research (CIHR) as well as by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] J. M. Mendel, Lessons in Estimation Theory for Signal Processing, Communications, and Control, Pearson Education, 1995.

[2] Y. Sucaet and W. Waelput, Digital Pathology, SpringerBriefs in Computer Science, 2014.

[3] S. Paris and F. Durand, A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach, Proc. European Conf. Computer Vision (2006).

[4] A. Adams, N. Gelfand, J. Dolson, and M. Levoy, Gaussian KD-Trees for Fast High-Dimensional Filtering, Proc. ACM Siggraph, pp. 21:1-21:12 (2009).

[5] K. He, J. Sun, and X. Tang, Guided Image Filtering, IEEE Trans. Pattern Anal. Mach. Intell., 35, 6 (2013).

[6] L. De-Maeztu, S. Mattoccia, A. Villanueva, and R. Cabeza, Linear Stereo Matching, Proc. Int. Computer Vision Conf., pp. 1708-1715 (2011).

[7] Y. Ding, J, Xiao, and J. Xu, Importance Filtering for Image Retargeting, Proc. IEEE Conf. Comp. Vision and Pattern Recogn., pp. 89-96 (2011).

[8] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Proc., 13, 4, pp. 600-612 (2004).

[9] A. G. Dempster and N. P. Murphy, The Binomial Window: Heuristics and Metrics, Signal Processing, 80, 12, pp. 2641-2645 (2000).

[10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, Foundations and Trends in Machine Learning, 3, 1, pp. 1-122 (2010).

[11] E. G. Gol'shtein and N. V. Tret'yakov, Modified Lagrangians in Convex Programming and Their Generalizations, Point-to-Set Maps and Mathematical Programming, pp. 8697 (1979).

[12] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk, Fast Alternating Direction Optimization Methods, SIAM J. Imaging Sciences, 7, 3, pp. 1588-1623 (2014).

[13] M. N. Wernick and J. N. Aarsvold, Emission Tomography: The Fundamentals of PET and SPECT, Academic Press, 2004.

[14] F. A. Haight, Handbook of the Poisson Distribution, John Wiley & Sons, 1967.

[15] M. Elad, Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing, Springer, 2010.

[16] S. Mallet, A Wavelet Tour of Signal Processing: The Sparse Way, Academic Press, 2008.

[17] I. Daubechies, B. Han, A. Ron, and Z. Shen, Framelets: MRA-Based Constructions of Wavelet Frames, Appl. Comput. Harmon. Anal., 14, pp. 1-46 (2003).

[18] J.-F. Cai, S. Osher, and Z. Shen, Split Bregman Methods and Frame Based Image Restoration, Multiscale Modeling & Simulation, 8, 2, pp. 337-369 (2010).

Author Biography

Oleg Michailovich holds a double MASc degree in electrical and biomedical engineering from Saratov State University (1994) and Technion - IIT (2000), respectively, as well as a PhD in biomedical engineering from the Technion - IIT (2003). Currently, he is an associate professor at the Department of Electrical and Computer Engineering at the University of Waterloo (Waterloo, ON). His research focuses on inverse problems in signal and image processing as well as related computational methods. He is a member of IEEE and presently serves as an Associate Editor of IEEE Transactions on Image Processing.