

# Blind Inpainting Using $\ell_0$ and Total Variation Regularization

Manya V. Afonso and Joao Miguel Raposo Sanches

**Abstract**—In this paper, we address the problem of image reconstruction with missing pixels or corrupted with impulse noise, when the locations of the corrupted pixels are not known. A logarithmic transformation is applied to convert the multiplication between the image and binary mask into an additive problem. The image and mask terms are then estimated iteratively with total variation regularization applied on the image, and  $\ell_0$  regularization on the mask term which imposes sparseness on the support set of the missing pixels. The resulting alternating minimization scheme simultaneously estimates the image and mask, in the same iterative process. The logarithmic transformation also allows the method to be extended to the Rayleigh multiplicative and Poisson observation models. The method can also be extended to impulse noise removal by relaxing the regularizer from the  $\ell_0$  norm to the  $\ell_1$  norm. Experimental results show that the proposed method can deal with a larger fraction of missing pixels than two phase methods, which first estimate the mask and then reconstruct the image.

**Index Terms**—Blind inpainting, image reconstruction, total variation, impulse noise, iterative methods.

## I. INTRODUCTION

**F**AULTY imaging sensors or bit errors during transmission can cause some pixels in an image to be lost or corrupted by impulse noise [12], [42]. In the case of missing pixel values, the corrupted pixels are assumed to have a value equal to zero, and the problem of estimating the complete image is called the inpainting problem [9], [48], [61].

We represent the image to be estimated with  $n$  pixels as a vector, say in lexicographic ordering,  $\mathbf{x} \in \mathbb{R}^n$ . Let  $m < n$  be the number of observed pixels or pixels free from impulse noise. The process of observing a partial set of  $m$  pixels out of  $n$  can be represented as an element-wise multiplication of the image with a binary mask in which all but  $m$  pixels are zero. In our representation, this observation process is represented as a multiplication of the vector  $\mathbf{x}$  with a size  $n \times n$  identity matrix  $\mathbf{A}$  with the respective diagonal elements corresponding to the  $(n - m)$  missing pixels set to zero.

Manuscript received May 5, 2014; revised October 6, 2014, January 11, 2015, and March 16, 2015; accepted March 22, 2015. Date of publication March 26, 2015; date of current version April 14, 2015. This work was supported in part by the Fundação para a Ciência e Tecnologia (FCT), Portuguese Ministry of Science and Higher Education through the Post-Doctoral Fellowship under Contract SFRH/BPD/79011/2011 and in part by FCT Project under Grant UID/EEA/50009/2013. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Alessandro Foi.

The authors are with the Instituto de Sistemas e Robótica, Instituto Superior Técnico, Lisboa 1049-001, Portugal (e-mail: mafonso@isr.ist.utl.pt; jmrs@isr.ist.utl.pt).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIP.2015.2417505

For the additive Gaussian noise model, the mapping from  $\mathbf{x}$  to the partially observed image  $\mathbf{y}$  is given by,

$$\mathbf{y} = \mathbf{A}(\mathbf{x} + \boldsymbol{\eta}_G), \quad (1)$$

where  $\boldsymbol{\eta}_G$  is an additive Gaussian noise term corrupting the observed pixel values. Note that the mask  $\mathbf{A}$  is applied to the sum of the image and the additive noise term. For other observation models, the mask  $\mathbf{A}$  is applied to the result of the algebraic operation relating the image and the noise term corresponding to the physics of the modality. For example, with multiplicative speckle noise which is the model in Ultrasound (US) and Synthetic Aperture Radar (SAR) imaging, equation (1) changes to,

$$\mathbf{y} = \mathbf{A}(\mathbf{x} \cdot \boldsymbol{\eta}_S), \quad (2)$$

where the speckle noise term  $\boldsymbol{\eta}_S$  is Rayleigh or Gamma distributed, and the multiplication is element-wise.

When the image is corrupted by impulse noise, those pixels of  $\mathbf{y}$  corresponding to the zeros on the diagonal of the observation matrix  $\mathbf{A}$  have a value that corresponds to the noise field  $\boldsymbol{\eta}_I$ ,

$$\mathbf{y} = \mathbf{A}(\mathbf{x} + \boldsymbol{\eta}_G) + (\mathbf{I}_n - \mathbf{A})\boldsymbol{\eta}_I. \quad (3)$$

Assuming that the pixel values of the image are in the interval  $[0, 255]$ , for the salt and pepper type of impulse noise the noisy pixels or elements of  $\boldsymbol{\eta}_I$  can have a value of either zero or 255. For the more difficult case of random valued impulse noise, the noisy pixels are uniformly distributed in the interval  $[0, 255]$ .

When the index set of the observed pixels or the observation mask  $\mathbf{A}$  is known, the inpainting problem can be solved by one of several existing methods for image reconstruction from a sparse set of observations [1], [7], [8], [21], [34], [41], [46]. Many of these methods were developed in the context of compressed sensing [14], [27] reconstruction, although it must be noted that in compressed sensing the observation operator needs to satisfy conditions to lead to incoherent observations [15]. For the removal of impulse noise, a common approach is to estimate the support set of the noisy pixels using an outlier detection method, to obtain an estimate of  $\mathbf{A}$  and then apply the reconstruction method.

Our goal in this paper is to estimate the image  $\mathbf{x}$  from the partial observations  $\mathbf{y}$  without knowing the observation mask  $\mathbf{A}$  beforehand. Such a situation occurs, for example, when it is not known which imaging sensors from the array are faulty, and therefore which pixels have values that are reliable and which ones are outliers.

### A. Related Work

The problems of reconstructing an image with missing data and removing impulse noise basically involve detecting outliers. Early approaches for estimating the missing values used median filtering, which discards outliers [42]. The Adaptive Median Filter (AMF) [42] and Adaptive Center Weighted Median Filter [19] were developed to detect the positions of noisy pixels with, respectively, salt-and-pepper and random valued impulse noise. Since median filters replace the value of the pixel at the center of the window with the statistical median of the neighborhood, they rely on the accuracy of the neighboring pixels and inherently cannot deal with a large percentage of outliers. In [25] an improved outlier detector based on thresholding, a measure called the Rank-Ordered Logarithmic Difference with Edge Preserving Regularization (ROLD-EPR) was proposed, for random valued impulse noise.

Two phase methods for estimating the image involve a mask estimation step, in which the observation mask  $\mathbf{A}$  is estimated using outlier detection, and a reconstruction step in which a standard convex optimization procedure is used with the estimate of  $\mathbf{A}$ . For example, some methods such as [13] and [17] use an absolute difference or  $\ell_1$ -data fidelity term  $\|\mathbf{Ax} - \mathbf{y}\|_1$ , with an edge preserving regularizer on the image. In [13], the total variation (TV) [18], [57] regularizer is used, leading to an  $\ell_1$ -TV optimization problem. In [68], a combination of  $\ell_1$ -norm and  $\ell_0$ -norm regularizers was used for simultaneous impulse noise removal and dictionary learning, after the mask has been estimated through outlier detection. The  $\ell_1$  term imposes sparsity on the difference terms corresponding to the noisy pixels, and the  $\ell_0$  regularizer tries to minimize the support set of the image on the learned dictionary.

Some methods do not use a separate mask detection stage, but estimate the mask or impulse noise field during the iterative process. In [66], a low rank matrix recovery method is presented where division of the image into patches corresponding to subspaces, and an alternating least squares method are used for reconstruction. A frame based method for image deblurring and decomposition into cartoon and texture components with impulse noise is presented in [24]. In this method, the impulse noise is considered additive and  $\ell_1$  regularization is used on both the image and noise components. An additive model was also used for the standard inpainting problem (*i.e.*, when the observation mask is known), in [38] with a non-local TV regularizer [37] on the image and  $\ell_1$  data fidelity term.

Another single phase iterative method called blind inpainting by Adaptive Outlier Pursuit (AOP) [69] simultaneously estimates the image and support set of the observed pixels, with wavelet regularization on the image and  $\ell_0$ -norm regularization on the mask.

A method for mixed impulse and Gaussian noise removal was proposed in [54], with an  $\ell_2$  data fidelity term corresponding to the observed pixels, an  $\ell_1$  term corresponding to the noisy pixels, and TV regularization on the image. The problem is reformulated as an iteratively reweighted quadratic problem by combining the  $\ell_2$  and  $\ell_1$  terms into a weighted  $\ell_2$  term, and the resulting  $\ell_2$ -TV problem is solved using the iteratively reweighted norm method [55].

Several of the above methods are based on alternating minimization, where two or more variables are iteratively estimated through a Gauss-Seidel method [49]. A related approach is the augmented Lagrangian (AL)/alternating direction method of multipliers (ADMM) framework [30] which has been used extensively in recent work on image reconstruction/restoration because of its mathematical elegance and computational speed [41], [70].

Finally, for the case of noise other than additive and Gaussian, there exist methods for image reconstruction from a partial set of pixels, such as [35], [36], and [44] for Poisson noise, and [3], [59] for Rayleigh speckle noise. There are also the classical interpolation methods that have been used in ultrasound imaging [62]. However, all these methods require the sampling matrix to be known. A method for image inpainting for the Poisson noise model was proposed in [6]. This approach filled in missing data through the minimization of the image gradient and an approximate solution of the mean curvature flow equation. In [39], a denoising method for Poisson noise was proposed, which divides the image into patches, which are modeled using sparse representation, and the dictionary is learnt during the denoising process. The same authors have also extended this method for inpainting under the Poisson noise model [40].

### B. Contributions

In this paper, we propose a method to estimate the image  $\mathbf{x}$  without knowing a priori the observation mask  $\mathbf{A}$ , *i.e.*, we simultaneously estimate the image and the mask. We formulate the masking operation as a summation after logarithmic compression, and apply a TV regularizer on the term corresponding to the logarithm of the image, and an  $\ell_0$ -norm regularizer on the term corresponding to the mask. The TV regularizer encourages the estimate of  $\mathbf{x}$  to be piecewise smooth, while the  $\ell_0$ -norm regularizer encourages the mask term to be sparse. The problem is solved iteratively using a Gauss-Seidel alternating minimization scheme. Experimental results show that our proposed method can deal with as many as 95% of the pixels missing, which is higher than reported in literature.

We extend the method to non-Gaussian noise models, namely multiplicative Rayleigh distributed speckle noise, and Poisson noise, by taking into account the data fidelity terms corresponding to their respective statistical models. Results of experiments with medical images for these models are presented.

A simple relaxation of the  $\ell_0$  regularizer to the  $\ell_1$ -norm continues to encourage the support set of the corrupted pixels to be sparse and works well for the removal of impulse noise, although not as well for the inpainting problem.

### C. Organization of the Paper

We formulate the estimation problem for blind inpainting for the additive and Gaussian noise model and present our proposed method in Section II. The  $\ell_1$  relaxation for impulse noise removal is presented in Sub-section II-B. We extend this method to Rayleigh speckle and Poisson noise

in Section III. In Section IV, we present experimental results on inpainting and impulse noise removal and comparison with existing methods within the range of fraction of missing pixels as reported in existing works. Sub-section IV-A presents results for blind inpainting with the additive and Gaussian noise model, Sub-section IV-B presents the results for impulse noise removal, and finally the results for blind inpainting with the Poisson and Rayleigh multiplicative noise models are presented in Sub-section IV-A. We also include a brief experimental study on the maximum extent to which the proposed method can recover missing pixels and the observation mask, in Sub-section IV-D. Section V concludes the paper with some pointers to future work.

## II. PROPOSED METHOD FOR BLIND INPAINTING

In the formulation of standard inpainting with TV regularization, with a known observation mask  $\mathbf{A}$ , the problem of estimating  $\mathbf{x}$  as reported in [1] and [21], is

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2 + \frac{\lambda}{2} TV(\mathbf{x}), \quad (4)$$

where  $TV(\cdot)$  is the isotropic total variation function, and  $\lambda > 0$  is the regularization parameter.

In our problem, we need to estimate both the image  $\mathbf{x}$  and the mask  $\mathbf{A}$ , therefore using a regularizer  $\phi(\cdot)$  on  $\mathbf{A}$ , leading to,

$$(\hat{\mathbf{x}}, \hat{\mathbf{A}}) = \arg \min_{\mathbf{x}, \mathbf{A}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2 + \frac{\lambda_1}{2} TV(\mathbf{x}) + \frac{\lambda_2}{2} \phi(\mathbf{A}), \quad (5)$$

where we now have two regularization parameters,  $\lambda_1, \lambda_2 > 0$ , for each of the two regularizer terms. The problem (5) is difficult to solve because it is not separable for our variables  $(\mathbf{x}, \mathbf{A})$ .

In [69], the problem is addressed by splitting the mask into the set of observed and noisy pixels. The observed image is defined as the sum of  $\mathbf{x}$  and the impulse noise term  $\mathbf{y} = \mathbf{x} + \mathbf{v}$ , where an element  $v_i$  of the vector  $\mathbf{v}$  is zero if the pixel is observed or non-zero if it is a noisy observation. Thus, a diagonal element  $a_i$  of the mask  $\mathbf{A}$  is 1 if a pixel is observed, *i.e.*, the corresponding element of the noise term  $v_i = 0$  or 0 if it is a noisy pixel, *i.e.*, the corresponding element of the noise term is non-zero. The problem solved is thus,

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{v}} \frac{1}{2} \|\mathbf{A}(\mathbf{x} - \mathbf{y})\|^2 + \frac{\lambda_1}{2} J(\mathbf{x}) + \frac{\lambda_2}{2} \|\mathbf{v}\|_0, \\ = \min_{\mathbf{x}, \mathbf{A}} \frac{1}{2} \|\mathbf{A}(\mathbf{x} - \mathbf{y})\|^2 + \frac{\lambda_1}{2} J(\mathbf{x}) + \frac{\lambda_2}{2} \|\text{diag}(\mathbf{I} - \mathbf{A})\|_0, \end{aligned}$$

where  $J(\mathbf{x})$  is a wavelet based regularizer term on  $\mathbf{x}$ , for example, the  $\ell_1$ -norm of wavelet coefficients in the analysis prior case [32]. Thus at each iteration,  $\mathbf{x}$  and  $\mathbf{A}$  are estimated by alternating minimization, gathering the terms containing each variable while keeping the other fixed.

### A. Blind Inpainting

We approach the problem of estimating  $\mathbf{x}$  and  $\mathbf{A}$  in a different manner. We use a logarithmic transform on both to convert the masking problem into an additive and separable one.

Since  $\mathbf{A}$  is a diagonal matrix  $\mathbf{A} = \text{diag}(\mathbf{a})$  and the masking operation is element-wise multiplication, we will optimize over the vector of diagonal elements  $\mathbf{a} \in \{0, 1\}^n$ . When a pixel with index  $i$  is observed, the corresponding mask element  $a_i = 1$ , and when pixel  $i$  is lost,  $a_i = 0$ . Thus, a pixel  $k$  in vector  $\mathbf{y}$  is defined as the scalar product,

$$y_i = x_i \times a_i. \quad (6)$$

We do not know apriori if a given pixel  $y_k$  corresponds to an observed one ( $a_k = 1$ ) or not ( $a_k = 0$ ).

Rather than have  $a_i = 0$  when the pixel is not observed, we can define  $a_i$  to be a small value in the order of  $10^{-K}$  or smaller,  $K$  being a positive integer greater than or equal to 3. Defining  $v_i = \log(a_i)$ , we have,

$$v_i = \begin{cases} 0, & \text{if } i \text{ is observed} \\ -K, & \text{otherwise} \end{cases} \quad (7)$$

If the maximum possible pixel value is 255, the value of  $K$  must satisfy  $K > \log 255$ . Assuming that  $\mathbf{y}$  and  $\mathbf{x}$  are always positive, applying a logarithmic transformation on (6) converts it into an additive model,

$$\log y_i = \log(x_i \times a_i), \quad (8)$$

$$\log y_i = \log x_i + \log a_i, \quad (9)$$

$$g_i = u_i + v_i, \quad (10)$$

where  $u_i = \log x_i$ , and  $g_i = \log(y_i + \delta)$ . A small positive bias term  $\delta > 0$  is added to  $\mathbf{y}$  to guarantee positivity. The base of the logarithm is 10, but we could use the natural logarithm or any other base for that matter, since this would only lead to an additive constant term.

Our problem is now estimating the vectors  $\mathbf{u}$  and  $\mathbf{v}$ , given the log transformed observation  $\mathbf{g}$ . We assume that our image  $\mathbf{x}$ , and therefore its logarithmic transformation  $\mathbf{u}$  are piece-wise smooth. In [69], the  $\ell_0$  norm was used as the regularizer directly on the binary mask. Our formulation differs in that we apply the  $\ell_0$ -norm regularization to the log transformed mask. The negative elements of  $\mathbf{v}$  therefore correspond to the non-observed pixels and those elements of  $\mathbf{v}$  which are equal to 0 correspond to the observed. Since the  $\ell_0$ -norm indicates the number of non-zero elements irrespective of their sign, minimizing  $\|\mathbf{v}\|_0$  minimizes the number of non-observed pixels. We therefore apply a TV regularizer on the log transformed image  $\mathbf{u}$ , and the  $\ell_0$ -norm regularizer on the log-transformed mask  $\mathbf{v}$ . Previously, TV regularization on log transformed images has been used in [45], [50], and [53]. Synthesis models which provide enhanced sparse representations in transform domains such as Block-matching and 3D filtering (BM3D) [20], [22] and K-means Singular Value Decomposition (KSVD) [4], [56] have also been used for image denoising and restoration, but will not be considered in this paper.

Our estimation problem is therefore,

$$(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = \arg \min_{\mathbf{u}, \mathbf{v}} \frac{1}{2} \|\mathbf{g} - \mathbf{u} - \mathbf{v}\|_2^2 + \frac{\lambda_1}{2} TV(\mathbf{u}) + \frac{\lambda_2}{2} \|\mathbf{v}\|_0, \quad (11)$$

where  $\lambda_1, \lambda_2 > 0$  are the respective regularization parameters.

**Algorithm 1** Blind Inpainting

1. Input  $\mathbf{y}$
2. Initialize parameters  $\delta > 0$ ,  $K > 0$ ,  $\lambda_1, \lambda_2$ , initial estimates  $\mathbf{u}^{(0)}, \mathbf{v}^{(0)}$ .
3. Compute  $\mathbf{g} = \log(\mathbf{y} + \delta)$ .
4. Set  $t = 0$
5. **repeat**
6.     Compute  $\mathbf{u}^{(t+1)}$  using (12).
7.     Compute  $\mathbf{v}^{(t+1)} = H_{\lambda_2}(\mathbf{g} - \mathbf{u}^{(t)})$ .
8.     Update values of  $\lambda_1, \lambda_2$ .
9.      $t \leftarrow t + 1$
10. **until** stopping criterion is satisfied.
11. Set estimates  $\hat{\mathbf{x}} = 10^{\hat{\mathbf{u}}}$ ,  $\hat{\mathbf{a}} = 10^{\hat{\mathbf{v}}}$ .

Formulations of the form (11) with different regularizers have appeared in the context of image decomposition [63] and deblurring with a sum of regularizers [10]. Alternating minimization schemes involving a sum of the  $\ell_0$ -norm and a convex term were also used in [11] and [51] for sparse image recovery.

Since (11) is a separable problem, we can apply a simple iterative alternating method as in [10] rather than resort to a variable splitting [23] and Alternating Direction Method of Multipliers (ADMM) [30] approach.

We apply an iterative alternating minimization to solve (11), by isolating the terms in each variable keeping the other fixed, leading to a Gauss-Seidel scheme. Solving for  $\mathbf{u}$  at iteration  $t$ ,

$$\hat{\mathbf{u}}^{(t)} = \arg \min_{\mathbf{u}} \frac{1}{2} \|\mathbf{g} - \mathbf{u} - \mathbf{v}^{(t)}\|_2^2 + \frac{\lambda_1}{2} TV(\mathbf{u}). \quad (12)$$

This is a TV regularized denoising problem, the solution of which can be computed efficiently using an algorithm such as Chambolle's algorithm [16].

Similarly, for  $\mathbf{v}$  at iteration  $t$  we have,

$$\hat{\mathbf{v}}^{(t)} = \arg \min_{\mathbf{v}} \frac{1}{2} \|\mathbf{g} - \mathbf{u}^{(t)} - \mathbf{v}\|_2^2 + \frac{\lambda_2}{2} \|\mathbf{v}\|_0. \quad (13)$$

This problem although non-convex, has a solution given by the hard threshold [28],

$$\hat{\mathbf{v}}^{(t)} = H_{\sqrt{2\lambda_2}}(\mathbf{g} - \mathbf{u}^{(t)}), \quad (14)$$

where  $H_{\lambda_2}(\cdot)$  is the hard threshold operator and is defined element-wise as,

$$v_i^{(t)} = \begin{cases} 0, & \text{if } (g_i - u_i^{(t)}) \leq \sqrt{2\lambda_2}, \\ (g_i - u_i^{(t)}), & \text{otherwise.} \end{cases} \quad (15)$$

This iterative process is run until the stopping criterion is satisfied. In practice, continuation schemes are used on the regularization parameters  $\lambda_1, \lambda_2$ , in which they are multiplied by a factor greater than one, until they reach a certain maximum value. Starting with smaller values and increasing them slowly can be interpreted as warm starting the problem at later iterations with the previous solutions [65]. The estimates of the image and mask are computed by inverting the logarithmic transformation,  $\hat{\mathbf{x}} = 10^{\hat{\mathbf{u}}}$  and  $\hat{\mathbf{a}} = 10^{\hat{\mathbf{v}}}$ .

The conditions for convergence [30], [36] do not require (12) to be solved exactly, as long as the error sequence decreases and the parameter  $\mu$  is positive. The proposed method is summarized in Algorithm 1.

**B. Impulse Noise Removal**

In the case of impulse noise, the corrupted pixels are no longer always equal to zero, but are never negative. From (3), it is obvious that if a pixel with index  $i$  is not corrupted, we have  $y_i = x_i$  and in the log domain,  $g_i = u_i + 0$ . If pixel  $i$  is noisy, we have  $y_i = \eta_{I,i}$  and  $g_i = \log \eta_{I,i}$ . In [24], the impulse noise field was treated as an additive component with a negative offset component to cancel out the pixel value, thereby absorbing the mask support in the noise value. We apply this logic to our log transformed variables, so as to absorb the observation mask in the variable  $\mathbf{v}$ . We therefore use the  $\ell_1$ -norm as was done in [24]. By using the  $\ell_1$  norm, we are no longer using sparsity on the cardinality of the support set of the missing pixels, but taking into account the fact that after logarithmic transformation, corrupted pixels have a value different from  $-K$ . The process of masking which is a binary logical operation is transformed into an additive operation with an offset term. In our case, we use this manipulation after logarithmic transformation.

We make a simple change to the framework described above for blind inpainting. We use the logarithmic transformation as before because the non-negativity assumption holds, but we replace the  $\ell_0$  term in (11) with an  $\ell_1$  term,

$$(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = \arg \min_{\mathbf{u}, \mathbf{v}} \frac{1}{2} \|\mathbf{g} - \mathbf{u} - \mathbf{v}\|_2^2 + \frac{\lambda_1}{2} TV(\mathbf{u}) + \frac{\lambda_2}{2} \|\mathbf{v}\|_1. \quad (16)$$

In this case, the variable  $\mathbf{v}$  corresponds to (the log transform of) the impulse noise field. For the non-corrupted pixels, this still corresponds to summation with zero in the log domain. Therefore, we now minimize the  $\ell_1$ -norm of the noise field. As in the previous section, we try to impose the condition of sparsity on the support set of the corrupted pixels.

Consequently, the solution for  $\mathbf{v}$  at each iteration will be different. We now have at iteration  $t$ ,

$$\hat{\mathbf{v}}^{(t)} = \arg \min_{\mathbf{v}} \frac{1}{2} \|\mathbf{g} - \mathbf{u}^{(t)} - \mathbf{v}\|_2^2 + \frac{\lambda_2}{2} \|\mathbf{v}\|_1. \quad (17)$$

This problem is convex and has a solution given by the well known shrinkage or soft threshold operator  $S_{\lambda_2}(\cdot)$  [26], [28],

$$\hat{\mathbf{v}}^{(t)} = S_{\lambda_2}(\mathbf{g} - \mathbf{u}^{(t)}). \quad (18)$$

**III. BLIND INPAINTING WITH NON-GAUSSIAN NOISE MODELS**

We extend the method described in the Section II to observation models other than additive and Gaussian. An intuitive way would be to apply a transformation such as a logarithmic transform for multiplicative speckle noise, and the variance stabilizing Anscombe transformation [5], [64] for Poisson noise, to convert the observation model into an additive one. For example, a more accurate method for computing the inverse Anscombe transform was proposed in [47]. Then, we could apply the method described in Algorithm 1. However this approach does not take into account the statistical model of the noise, and the appropriate data fidelity terms.

We therefore propose an extension of the proposed blind inpainting method to Rayleigh distributed multiplicative

speckle noise, and Poisson noise by using the appropriate data fidelity term. For multiplicative noise, multiplying a pixel whose value is 0 will always lead to the corresponding observed pixel being equal to 0 as well. In the case of Poisson counting processes, the count is always 0 when the parameter, in our case the pixel value, is equal to 0. Therefore, we interchange the order of the noisy observation and masking so that our observation  $\mathbf{y}$  is the result of observing the masked image  $\mathbf{Ax}$  under the noise model. For speckle noise, the observation model changes from (2) to,

$$\mathbf{y} = (\mathbf{Ax}) \cdot \eta_S, \quad (19)$$

and for Poisson noise the model is,

$$\mathbf{y} = \text{Poisson}(\mathbf{Ax}). \quad (20)$$

We can thus apply the likelihood function of the respective statistical model between  $\mathbf{y}$  the product  $\mathbf{Ax}$ . Recall that an element of  $\mathbf{Ax}$  is defined as  $\mathbf{Ax}_i = a_i x_i$ , where  $a_i$  is the  $i^{\text{th}}$  diagonal element of the matrix  $\mathbf{A}$ . For the Rayleigh multiplicative noise, the likelihood function is,

$$p(\mathbf{y}|\mathbf{Ax}) = \prod_{i=1}^n \frac{y_i}{a_i x_i} \exp\left(-\frac{y_i^2}{2(a_i x_i)}\right). \quad (21)$$

It is straightforward to show that (see [60] for more details) the associated data fidelity term between  $\mathbf{y}$  and  $\mathbf{Ax}$  is,

$$J_r(\mathbf{y}, \mathbf{Ax}) = \sum_{i=1}^n \left( \frac{y_i^2}{2(a_i x_i)} + \log(a_i x_i) \right). \quad (22)$$

Similarly for the Poisson model, the likelihood function is,

$$p(\mathbf{y}|\mathbf{Ax}) = \prod_{i=1}^n \frac{e^{-a_i x_i} (a_i x_i)^{y_i}}{y_i!}, \quad (23)$$

and the corresponding data fidelity term is,

$$J_p(\mathbf{y}, \mathbf{Ax}) = \sum_{i=1}^n (a_i x_i - y_i \log(a_i x_i)). \quad (24)$$

In both (22) and (24), we see that there appears a term with the product  $(a_i x_i)$  and a term involving its logarithm. We can therefore work with the log transformed variables  $\mathbf{u} = \log \mathbf{x}$ , and  $\mathbf{v} = \log \mathbf{a}$ . Thus (22) changes to

$$J_r(\mathbf{y}, \mathbf{u}, \mathbf{v}) = \sum_{i=1}^n \left( \frac{y_i^2}{2} e^{-(u_i+v_i)} + u_i + v_i \right), \quad (25)$$

and (24) changes to

$$J_p(\mathbf{y}, \mathbf{u}, \mathbf{v}) = \sum_{i=1}^n \left( e^{(u_i+v_i)} - y_i (u_i + v_i) \right). \quad (26)$$

Logarithmic transformations to deal with non-additive and Gaussian noise have been previously used in [39] and [58] for Poisson denoising and in [60] for Rayleigh despeckling.

We now formulate our optimization problem with TV regularization on  $\mathbf{u}$  and  $\ell_0$  regularization on  $\mathbf{v}$ . The data fidelity term  $J(\cdot)$  is changed accordingly. The problem (11)

for the additive Gaussian noise model changes to the more general problem,

$$(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = \arg \min_{\mathbf{u}, \mathbf{v}} J(\mathbf{y}, \mathbf{u}, \mathbf{v}) + \frac{\lambda_1}{2} TV(\mathbf{u}) + \frac{\lambda_2}{2} \|\mathbf{v}\|_0. \quad (27)$$

Since (25) and (26) both involve the sum of a linear term and an exponential term, they are non-separable for  $\mathbf{u}$  and  $\mathbf{v}$ . Therefore we need to use variable splitting [23] to be able to use the AL/ADMM method to solve (27). We therefore introduce two auxiliary variables  $\mathbf{z}$  and  $\mathbf{w}$  to act as the arguments of the TV and  $\ell_0$  regularizer terms respectively, leading to the constrained problem,

$$\begin{aligned} \min_{\mathbf{u}, \mathbf{v}, \mathbf{z}, \mathbf{w}} \quad & J(\mathbf{y}, \mathbf{u}, \mathbf{v}) + \frac{\lambda_1}{2} TV(\mathbf{z}) + \frac{\lambda_2}{2} \|\mathbf{w}\|_0 \\ \text{subject to} \quad & \mathbf{u} = \mathbf{z}, \mathbf{v} = \mathbf{w}. \end{aligned} \quad (28)$$

Using the augmented Lagrangian [43], [52], this problem can be shown to be equivalent to the minimization problem,

$$\begin{aligned} \min_{\mathbf{u}, \mathbf{v}, \mathbf{z}, \mathbf{w}} \quad & J(\mathbf{y}, \mathbf{u}, \mathbf{v}) + \frac{\lambda_1}{2} TV(\mathbf{z}) + \frac{\lambda_2}{2} \|\mathbf{w}\|_0 \\ & + \frac{\mu_1}{2} \|\mathbf{u} - \mathbf{z} - \mathbf{d}_z\|_2^2 + \frac{\mu_2}{2} \|\mathbf{v} - \mathbf{w} - \mathbf{d}_w\|_2^2, \end{aligned} \quad (29)$$

where  $\mu_1, \mu_2 \geq 0$  are the penalty parameters, and  $\mathbf{d}_z, \mathbf{d}_w$  are the so-called Bregman update vectors [41].

This problem is split into four problems at each iteration by gathering all the terms in each variable, and solving for each by keeping the others fixed. Thus, the AL algorithm iterates between minimizing the objective function in (29) with respect to  $\mathbf{f}$  and  $\mathbf{u}$ , leading to a Gauss-Seidel process (for more details, see [1], [2], [33] and the references therein) which at iteration  $t$  is summarized as,

$$\mathbf{u}^{(t+1)} = \arg \min_{\mathbf{u}} J(\mathbf{y}, \mathbf{u}, \mathbf{v}^{(t)}) + \frac{\mu_1}{2} \|\mathbf{u} - \mathbf{z}^{(t)} - \mathbf{d}_z^{(t)}\|_2^2 \quad (30)$$

$$\mathbf{v}^{(t+1)} = \arg \min_{\mathbf{v}} J(\mathbf{y}, \mathbf{u}^{(t)}, \mathbf{v}) + \frac{\mu_2}{2} \|\mathbf{v} - \mathbf{w}^{(t)} - \mathbf{d}_w^{(t)}\|_2^2 \quad (31)$$

$$\mathbf{z}^{(t+1)} = \arg \min_{\mathbf{z}} \frac{\mu_1}{2} \|\mathbf{u}^{(t)} - \mathbf{z} - \mathbf{d}_z^{(t)}\|_2^2 + \frac{\lambda_1}{2} TV(\mathbf{z}) \quad (32)$$

$$\mathbf{w}^{(t+1)} = \arg \min_{\mathbf{w}} \frac{\mu_2}{2} \|\mathbf{v}^{(t)} - \mathbf{w} - \mathbf{d}_w^{(t)}\|_2^2 + \frac{\lambda_2}{2} \|\mathbf{w}\|_0 \quad (33)$$

$$\begin{aligned} \mathbf{d}_z^{(t+1)} &= \mathbf{d}_z^{(t)} + \mathbf{z}^{(t+1)} - \mathbf{u}^{(t+1)}, \\ \mathbf{d}_w^{(t+1)} &= \mathbf{d}_w^{(t)} + \mathbf{w}^{(t+1)} - \mathbf{v}^{(t+1)}. \end{aligned}$$

As in the case of Gaussian noise, the  $\ell_2$ -TV denoising problem from (32) is solved using a few iterations of Chambolle's algorithm and the  $\ell_2$ - $\ell_0$  regularized denoising problem from (33) is solved using the hard threshold. The problems involving  $J(\cdot)$ , (30) and (31) can be solved approximately using a few iterations of Newton's method [49], after plugging in either (25) or (26).

The proposed method for blind inpainting with for Rayleigh multiplicative or Poisson noise is summarized in Algorithm 2.

#### IV. EXPERIMENTAL RESULTS

In this section we compare our proposed method with existing ones. In the synthetic experiments with the lena,

**Algorithm 2** Blind Inpainting - Non-Gaussian Noise

1. Input  $\mathbf{y}$
2. Initialize parameters  $\delta > 0$ ,  $K > 0$ ,  $\lambda_1, \lambda_2$ ,  $\mu_1, \mu_2$ , initial estimates  $\mathbf{u}^{(0)}, \mathbf{v}^{(0)}, \mathbf{z}^{(0)}, \mathbf{w}^{(0)}, \mathbf{d}_z^{(0)}, \mathbf{d}_w^{(0)}$ .
3. Set  $t = 0$
4. **repeat**
5.   Compute  $\mathbf{u}^{(t+1)}$  using Newton's method to solve (30).
6.   Compute  $\mathbf{v}^{(t+1)}$  using Newton's method to solve (31).
7.   Compute  $\mathbf{z}^{(t+1)}$  using Chambolle's method to solve (32).
8.   Compute  $\mathbf{w}^{(t+1)} \leftarrow H_{\lambda_2/\mu_2}(\mathbf{v}^{(t)} - \mathbf{d}_w^{(t)})$ .
9.    $\mathbf{d}_z^{(t+1)} \leftarrow \mathbf{d}_z^{(t)} + \mathbf{z}^{(t+1)} - \mathbf{u}^{(t+1)}$ .
10.    $\mathbf{d}_w^{(t+1)} \leftarrow \mathbf{d}_w^{(t)} + \mathbf{w}^{(t+1)} - \mathbf{v}^{(t+1)}$ .
11.   Update values of  $\lambda_1, \lambda_2$ .
12.    $t \leftarrow t + 1$
13. **until** stopping criterion is satisfied.
14. Set estimates  $\hat{\mathbf{x}} = e^{\mathbf{u}}$ ,  $\hat{\mathbf{a}} = e^{\mathbf{v}}$ .

cameraman, and peppers images, we present measures of the reconstruction error, and the number of incorrectly estimated mask pixels. The results presented in Tables I, II, III, and IV were averaged over 100 experiments. All experiments were performed on MATLAB on an Ubuntu Linux based server with 64 GB of RAM.

### A. Blind Inpainting

To test our proposed method, we generate a random binary mask with a fraction of its elements equal to zero and multiply it element-wise to our image corrupted with additive Gaussian noise. The criteria used to evaluate the accuracy of estimation are the Improvement in Signal to Noise Ratio (ISNR), which is defined as,

$$\text{ISNR} = 10 \log_{10} \left( \frac{\|\mathbf{y} - \mathbf{x}\|^2}{\|\mathbf{y} - \hat{\mathbf{x}}\|^2} \right), \quad (34)$$

the structural similarity index measure (SSIM) [67], and the fraction of incorrectly estimated mask pixels. The latter is computed using the binary exclusive or (XOR) operation with the original mask, which produces a logical value equal to 1 at the mask pixels estimated incorrectly, and zero otherwise. Hence, the sum over all the pixels of the logical XOR operation is a measure of the errors in the estimate of the mask.

The iterative process was run until the relative difference between successive iterates  $\|\hat{\mathbf{x}}^{(t+1)} - \hat{\mathbf{x}}^{(t)}\| / \|\hat{\mathbf{x}}^{(t+1)}\|$  fell below a threshold of  $10^{-3}$ . In the results for blind inpainting, the values of the regularization parameters used were  $\lambda_1 = 0.008$  and  $\lambda_2 = 0.1$ , which were found to work well. The parameters of other algorithms used in our comparison were the ones suggested by the respective authors, with hand-tuning for the best ISNRs when different images and image sizes and different experimental setups were used. Note that some methods may be computationally very slow or inaccurate beyond the range of missing pixels which were reported in their respective papers.

Figure 1 shows the worst case (*i.e.*, the setup with the highest fraction of missing pixels) results obtained with the proposed method for the  $512 \times 512$  Lena image. The binary mask has only 5% of its pixels equal to 1,

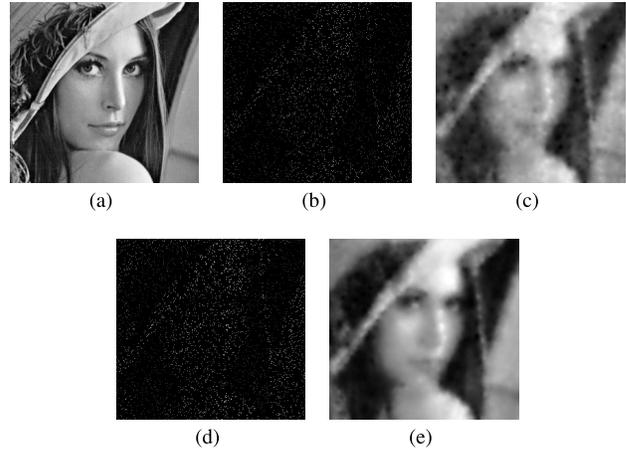


Fig. 1. Inpainting with the Lena image - worst case (95% of pixels missing) scenario. (a) original image; (b) observed image with additive Gaussian noise (SNR 5 dB) and 95% pixels missing; (c) estimate of (b) using the proposed method, ISNR = 15.12 dB; (d) observed image with additive Gaussian noise (SNR 20 dB) and 95% pixels missing; (e) estimate of (d) using the proposed method, ISNR = 15.86 dB.

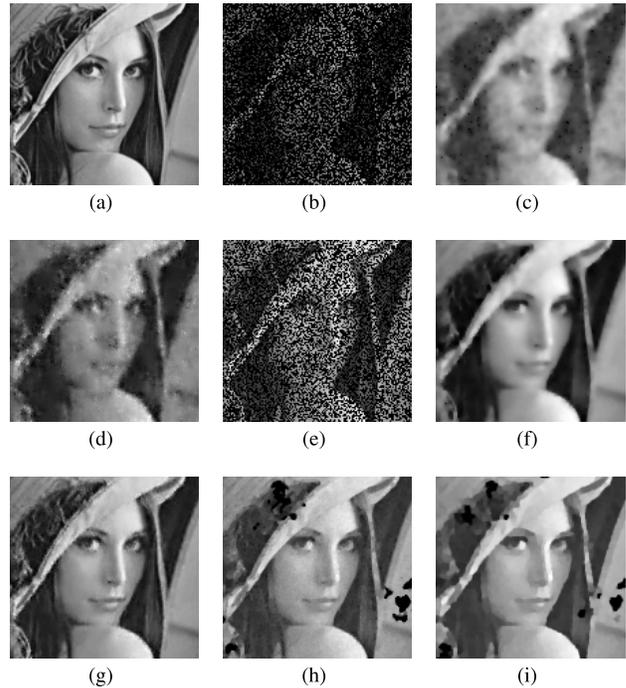


Fig. 2. Inpainting with the Lena image - Comparison. (a) original image; (b) observed image with Gaussian noise (SNR = 5 dB) and 70% pixels missing; (c) estimate from (b) obtained using the proposed method; (d) estimate from (b) obtained using KSVd; (e) observed image with Gaussian noise (SNR = 20 dB) and 50% pixels missing; (f) estimate from (e) obtained using the proposed method; (g) estimate from (e) obtained using KSVd; (h) estimate from (e) obtained using AOP [69]; (i) estimate from (e) obtained using [13].

which means that 95% of the pixels were randomly discarded. Figures 1(b) and 1(d) show the observed images obtained with this mask, with additive Gaussian noise with Signal to Noise Ratios (SNR) of 5 dB and 20 dB, respectively. For the 5 dB Gaussian noise case, the proposed method took 123.88 seconds, and produced an ISNR of 15.12 dB, and an SSIM of 0.634. For the 20 dB case, these values were

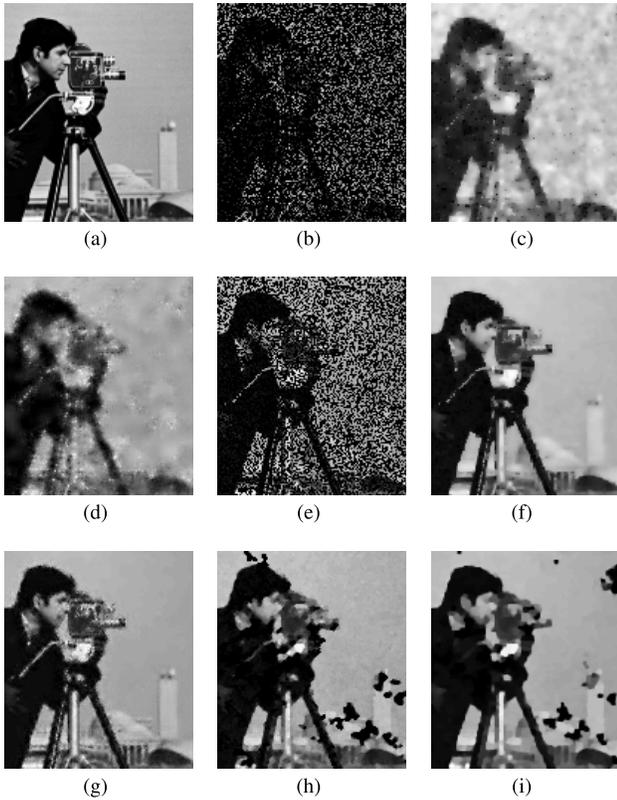


Fig. 3. Inpainting with the Cameraman image - Comparison. (a) original image; (b) observed image with Gaussian noise (SNR = 5 dB) and 70% pixels missing; (c) estimate from (b) obtained using the proposed method; (d) estimate from (b) obtained using KSVD; (e) observed image with Gaussian noise (SNR = 20 dB) and 50% pixels missing; (f) estimate from (e) obtained using the proposed method; (g) estimate from (e) obtained using KSVD; (h) estimate from (e) obtained using AOP [69]; (i) estimate from (e) obtained using [13].

132.74 seconds, ISNR: 15.86 dB, and SSIM: 0.689. The number of mask errors through the XOR operation for the 5 dB case was 89 out of a mask of size  $512^2$ , corresponding to 0.034% of mask pixels estimated incorrectly. There were no mask errors in the estimate for the 20 dB case.

Figure 2 compares the estimate obtained using the proposed method with those obtained using fast two phase deblurring with TV [13],<sup>1</sup> Adaptive Outlier Pursuit (AOP) [69], K-ALS [66],<sup>2</sup> and KSVD [31]<sup>3</sup> for 50% of the pixels missing and with Gaussian noise with SNR = 20 dB, and with 70% of the pixels missing and with Gaussian noise with SNR = 5 dB. Figures 3 and 4 show the estimates for the cameraman and peppers images, under the same experimental conditions.

Table I summarizes the comparison of the proposed method with other methods for blind inpainting, for 3 different levels of Gaussian noise, namely, SNR = 5, 10, and 20, and for three values of the fraction of missing pixels, namely, 0.25, 0.5, and 0.7. The methods [13], [66], and [69] were found to not provide a significant ISNR outside the ranges of fractions of missing pixels, as reported by their authors. We can see from this table that KSVD [31] is computationally

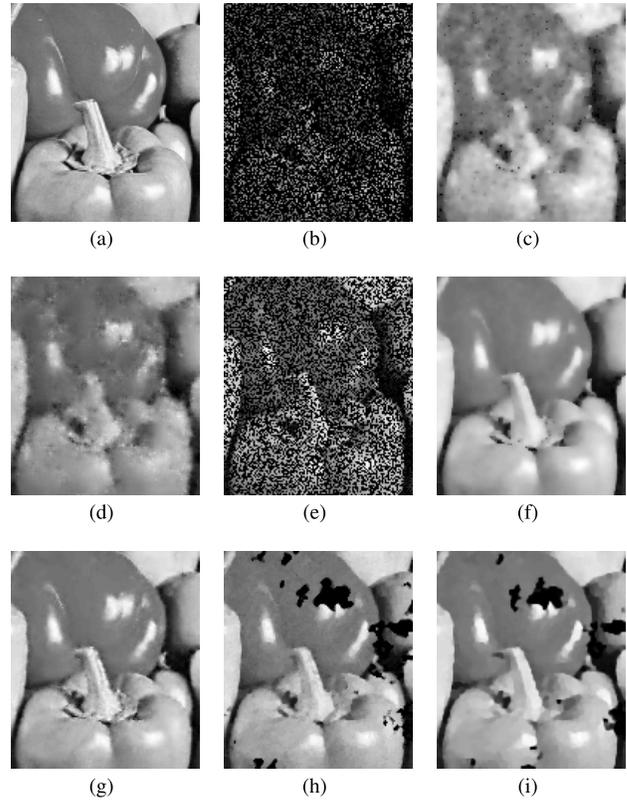


Fig. 4. Inpainting with the Peppers image - Comparison. (a) original image; (b) observed image with Gaussian noise (SNR = 5 dB) and 70% pixels missing; (c) estimate from (b) obtained using the proposed method; (d) estimate from (b) obtained using KSVD; (e) observed image with Gaussian noise (SNR = 20 dB) and 50% pixels missing; (f) estimate from (e) obtained using the proposed method; (g) estimate from (e) obtained using KSVD; (h) estimate from (e) obtained using AOP [69]; (i) estimate from (e) obtained using [13].

heavy but produced the best ISNR for most of the experiments. Our proposed method achieves an ISNR close to the one obtained with KSVD with a much lower computational time, for higher percentages of missing pixels and for higher levels of Gaussian noise. While [13] is more accurate in estimating the observation mask, it produces a lower ISNR.

### B. Impulse Noise Removal

1) *Salt-and-Pepper Noise*: In Figures 5 and 6, we demonstrate salt and pepper noise removal, with 50% of the pixels corrupted and Gaussian noise with an SNR of 5 dB, for the Lena and cameraman images. The estimate obtained with the proposed method and those obtained with [13], [24], [54], [66], and [69] are shown.

The results for different fractions of noisy pixels with the Gaussian noise component with SNR equal to 10, 5 and 2 dB, are presented in the combined Table II. We can see that overall, the method from [54] produces a good ISNR quickly. For higher fractions of missing pixels, [54] was found to be the fastest method, but it produces an ISNR less than that produced by the proposed method.

2) *Random-Valued Impulse Noise*: We present the noisy Lena image with random valued impulse noise and Gaussian noise, and the estimates obtained using our method

<sup>1</sup><http://homepage.math.uiowa.edu/~jiancai/code/FastTV2Phase.zip>

<sup>2</sup><http://www.math.duke.edu/~yiwang/kals.htm>

<sup>3</sup><http://www.cs.technion.ac.il/~elad/Various/Matlab-Package-Book.rar>

TABLE I  
COMPARISON FOR BLIND INPAINTING.  $\kappa$  INDICATES THE FRACTION OF MISSING PIXELS

$\kappa$	SNR (dB)	Method	Lena				Cameraman				Peppers			
			Time (sec)	ISNR (dB)	SSIM	Mask err. (%)	Time (sec)	ISNR (dB)	SSIM	Mask err. (%)	Time (sec)	ISNR (dB)	SSIM	Mask err. (%)
0.25	20	Proposed	13.09	17.88	0.883	<b>0.000</b>	18.3	17.07	0.861	0.46	18.26	15.03	0.915	0.41
		[13]	40.782	19.13	0.903	0.000	59.21	14.53	0.836	0.42	96.76	13.71	0.903	<b>0.32</b>
		AOP	25.878	19.38	0.876	0.829	26.16	13.93	0.735	8.85	26.2	16.05	0.858	2.6
		KALS	151.556	17.62	0.903	1.764	151.97	11.52	0.791	10.4	150.48	16.22	0.889	3.61
		KSVD	972.503	<b>23.94</b>	<b>0.956</b>	<b>0.000</b>	951	<b>19.77</b>	<b>0.924</b>	<b>0.39</b>	986.83	<b>18.06</b>	<b>0.936</b>	0.34
0.25	10	Proposed	15.614	16.94	0.844	0.046	20.57	14.64	0.685	4.02	21.5	14.44	0.788	0.93
		[13]	60.3	17.29	0.834	0.018	81.27	13.68	0.756	3.91	136.48	13.31	0.83	<b>0.88</b>
		AOP	25.776	14.57	0.646	3.872	25.77	11.68	0.499	11.81	25.88	12.97	0.623	6
		KALS	163.956	15.13	0.780	4.680	160.86	11.4	0.663	12.26	162.81	14.52	0.764	6.05
		KSVD	750.984	<b>19.53</b>	<b>0.886</b>	<b>0.015</b>	725.8	<b>15.54</b>	<b>0.810</b>	<b>3.88</b>	740.12	<b>16.62</b>	<b>0.860</b>	0.92
0.25	5	Proposed	18.18	14.79	0.751	0.479	21.27	10.95	0.422	5.81	20.52	11.65	0.533	1.87
		[13]	68.23	15.58	0.759	<b>0.429</b>	90.79	12.18	0.643	<b>5.67</b>	109.94	12.56	0.746	<b>1.79</b>
		AOP	26.342	11.23	0.485	8.856	29.95	9.8	0.366	13.5	24.39	10.33	0.459	10.37
		KALS	159.908	12.37	0.638	10.000	160.73	10.11	0.507	14.36	159.12	11.39	0.591	11.58
		KSVD	641.801	<b>16.87</b>	<b>0.809</b>	0.465	571.86	<b>13.00</b>	<b>0.698</b>	5.73	613.08	<b>14.14</b>	<b>0.787</b>	1.82
0.5	20	Proposed	15.036	18.35	0.831	<b>0.000</b>	21.47	16.7	0.809	<b>0.30</b>	22.92	16.77	0.873	0.23
		[13]	153.93	12.13	0.783	4.259	150.39	10.64	0.71	5.72	177.23	9.94	0.771	5.86
		AOP	67.138	11.95	0.781	4.975	79.24	9.05	0.629	15.89	80.15	8.31	0.715	12.87
		KALS	148.404	-0.46	0.088	62.786	150.44	0.9	0.121	69.84	151.16	0.66	0.099	64.21
		KSVD	1280.12	<b>22.50</b>	<b>0.918</b>	<b>0.000</b>	1219.88	<b>18.76</b>	<b>0.876</b>	<b>0.30</b>	1307.18	<b>17.02</b>	<b>0.904</b>	<b>0.21</b>
0.5	10	Proposed	16.322	17.92	0.804	0.027	24.34	15.79	0.704	2.65	24.9	16.21	0.803	0.61
		[13]	177.834	11.03	0.713	4.645	228.52	8.82	0.541	13.93	228.77	9.17	0.67	7.95
		AOP	68.758	9.40	0.576	12.669	143.11	7.42	0.394	26.54	96.83	6.91	0.517	18.59
		KALS	162.394	1.03	0.060	73.317	159.72	1.66	0.06	80.69	160.97	1.25	0.062	75.34
		KSVD	907.344	<b>20.01</b>	<b>0.851</b>	<b>0.012</b>	889.04	<b>16.70</b>	<b>0.774</b>	<b>2.57</b>	928.8	<b>17.46</b>	<b>0.827</b>	<b>0.56</b>
0.5	5	Proposed	<b>16.354</b>	<b>16.64</b>	0.724	0.363	<b>24.11</b>	<b>13.65</b>	<b>0.500</b>	3.81	23.72	14.57	0.622	1.23
		[13]	153.348	10.28	0.636	5.801	208.97	8.07	0.456	16.07	218.37	8.18	0.576	10.65
		AOP	77.092	7.87	0.439	16.501	158.29	6.55	0.272	30.81	119.25	6.01	0.382	22.87
		KALS	153.73	1.44	0.045	81.457	153.48	2.1	0.024	87.73	153.38	1.68	0.041	84.32
		KSVD	817.908	16.56	<b>0.738</b>	<b>0.328</b>	794.51	13.13	0.433	<b>3.72</b>	806.1	<b>15.56</b>	<b>0.746</b>	<b>1.14</b>
0.7	20	Proposed	18.914	17.31	0.758	<b>0.000</b>	<b>27.98</b>	<b>15.63</b>	<b>0.745</b>	0.17	<b>27.84</b>	<b>16.25</b>	<b>0.811</b>	<b>0.13</b>
		[13]	484.056	1.25	0.011	65.536	485.04	1.26	0.032	65.69	506.8	1.27	0.014	65.47
		AOP	440.584	1.68	0.013	59.952	447.26	1.59	0.004	59.67	440.13	1.7	0.012	59.85
		KALS	150.064	1.54	0.000	59.949	148.55	1.53	0.006	59.78	148.68	1.55	0.001	59.97
		KSVD	1120.89	<b>20.73</b>	<b>0.841</b>	<b>0.000</b>	1023.81	14.74	0.764	<b>0.16</b>	1161.15	15.53	0.789	<b>0.13</b>
0.7	10	Proposed	18.962	17.01	0.739	<b>0.008</b>	<b>28.01</b>	<b>15.29</b>	<b>0.677</b>	1.56	<b>28.73</b>	15.89	<b>0.770</b>	0.36
		[13]	485.324	1.21	0.013	65.414	476.82	1.23	0.022	66.15	491.38	1.23	0.014	65.37
		AOP	436.314	1.56	0.006	60.013	437.64	1.1	0.058	57.42	441.42	1.65	0.012	59.54
		KALS	145.666	1.53	0.000	60.011	146.05	1.51	0.006	58.7	144.64	1.53	0.001	59.82
		KSVD	838.487	<b>17.49</b>	<b>0.745</b>	0.009	767.09	13.54	0.653	<b>1.53</b>	807.81	<b>16.34</b>	0.74	<b>0.35</b>
0.7	5	Proposed	<b>17.838</b>	<b>16.34</b>	<b>0.685</b>	0.206	<b>27.55</b>	<b>14.15</b>	<b>0.545</b>	<b>2.32</b>	<b>27.11</b>	<b>15.03</b>	<b>0.663</b>	<b>0.75</b>
		[13]	416.842	1.19	0.012	65.700	424.85	1.17	0.02	66.46	431.73	1.19	0.013	65.88
		AOP	444.732	1.54	0.007	59.682	428.75	0.98	0.067	55.99	432.56	1.59	0.014	58.8
		KALS	146.772	1.47	0.000	59.756	141.41	1.43	0.006	58.02	138.97	1.46	0.001	59.36
		KSVD	713.631	14.85	0.669	<b>0.201</b>	615.3	12.58	0.512	2.39	594.52	<b>15.16</b>	<b>0.668</b>	<b>0.76</b>

and [24], [54], [66], and [69] in Figure 7, with Gaussian noise with an SNR of 5 dB and 50% of pixels corrupted. The results with the cameraman image under the same conditions are presented in Figure 8. Once again, the combined Table II contains a summary of the results obtained for different fractions of noisy pixels with the Gaussian noise component with SNR equal to 10, 5 or 2 dB. Overall, our method is the significantly faster than the other methods, and produces the best ISNR for higher levels of Gaussian noise.

For the inpainting problem however, this  $\ell_1$  norm relaxation, was experimentally found to produce an ISNR significantly lower than the one obtained using the  $\ell_0$  norm.

### C. Inpainting With Non-Gaussian Noise Models

In this section, we present results for blind inpainting with the Rayleigh multiplicative and Poisson noise models.

For the Lena image, we report the accuracy of reconstruction in terms of the ISNR or the normalized Mean Absolute Error (NMEA) [29] for Rayleigh noise, which is defined as,  $\|\mathbf{x} - \hat{\mathbf{x}}\|_1 / \|\mathbf{x}\|_1$ .

1) *Poisson Noise*: For the Poisson noise case, we first multiplied the image by 0.5 to increase the level of noise, and then corrupted it with Poisson noise. We then multiplied the noisy image with random masks. We compare our proposed method for blind inpainting with Poisson noise from Algorithm 2 with a method for blind inpainting assuming the additive noise model after Anscombe transformation [47]. The additive Gaussian noise model method used was AOP, because of its computational speed and accuracy. We summarize our results for the lena and cameraman images for three values of fractions of missing pixels, in Table III. We can see that taking into account the statistical model offers an improvement in ISNR. As with the additive and Gaussian noise case,

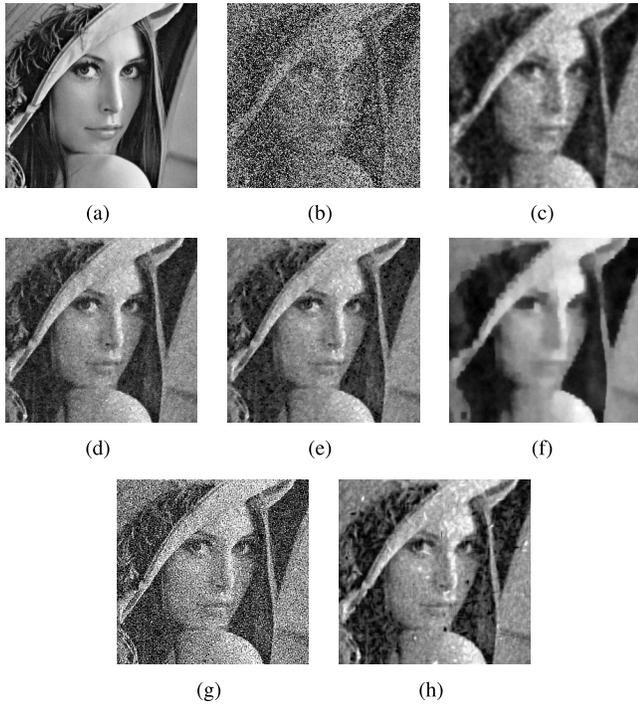


Fig. 5. Salt and pepper noise removal with the Lena image. (a) cropped original image, (b) observed image with 50% noisy pixels and SNR = 5 dB; estimates of (b) using: (c) proposed method, (d) [54], (e) AOP [69], (f) [24], (g) [13], and (h) KALS [66].

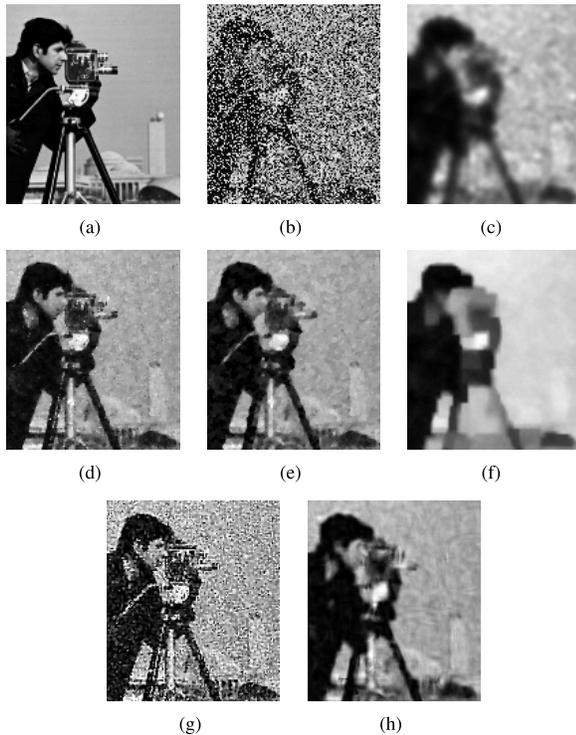


Fig. 6. Salt and pepper noise removal with the Cameraman image. (a) cropped original image, (b) observed image with 50% noisy pixels and SNR = 5 dB; estimates of (b) using: (c) proposed method, (d) [54], (e) AOP [69], (f) [24], (g) [13], and (h) KALS [66].

the existing methods for blind inpainting do not always work well for large fractions of missing pixels, above 50%, or take a long time over 10 minutes.

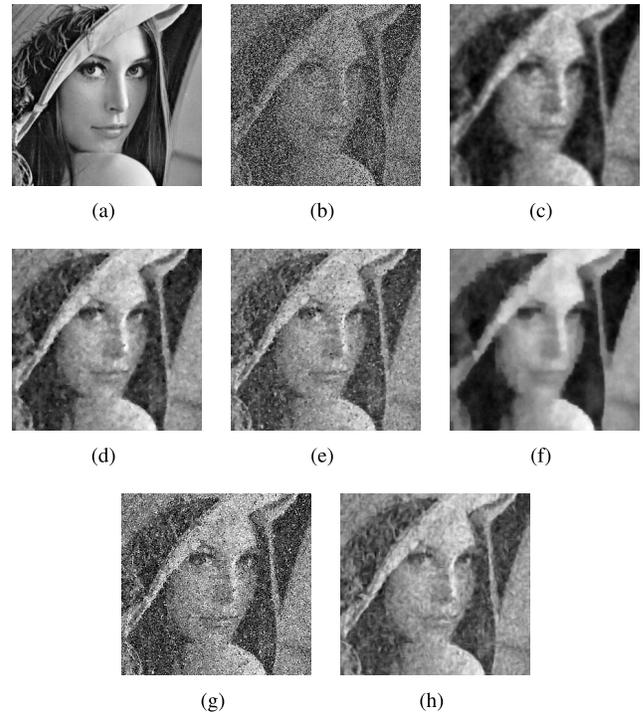


Fig. 7. Random valued impulse noise (noisy pixels have values in the interval  $[0, 255]$ ) removal with the Lena image. (a) cropped original image, (b) observed image with 50% noisy pixels and SNR = 5 dB; estimates of (b) using: (c) proposed method, (d) [54], (e) AOP [69], (f) [24], (g) [13], and (h) KALS [66].

A cropped region from the Lena image is shown in Figure 9(a). A noisy image with 50% of the pixels missing is shown in Figure 9(b) and the respective estimates using the proposed method and AOP after Anscombe transformation are shown in Figures 9(c) and 9(d). For a pixel loss of 90%, the observed image and estimate using the proposed method are shown in Figures 9(e) and 9(f). Blind inpainting with Poisson noise for the cameraman image with 50% of its pixels missing is illustrated in Figure 10.

For the fluorescence microscopy image, we do not have access to the noiseless image. A cropped region of size  $308 \times 380$  from the noisy  $1036 \times 1384$  image is shown in Figure 11(a). We show only the green component, where the information of interest lies. The observed image with 75% pixels missing is shown in Figure 11(b). Figure 11(c) shows the corresponding region of the estimate obtained using the proposed method, and Figure 11(e) shows the diagonal profile. The errors in the estimate of the mask are shown in Figure 11(d) which is the result of the binary XOR operation between the mask and its estimate. We can see that most incorrectly estimated mask bits are in the region that correspond to low pixel values in the noisy image. The total computation time was 842.75 seconds.

2) *Rayleigh Multiplicative Noise*: We follow a similar protocol for the Rayleigh noise case, after normalizing the Lena image by dividing by the maximum pixel value. Once again, to compare our method Algorithm 2 for the Rayleigh noise model noise, we use a logarithmic transformation followed by inpainting using a method for inpainting with additive noise.

TABLE II  
COMPARISON FOR IMPULSE NOISE REMOVAL.  $\kappa$  INDICATES THE FRACTION OF CORRUPTED PIXELS

$\kappa$	SNR (dB)	Method	Salt and Pepper noise						Random valued impulse noise					
			Lena			Cameraman			Lena			Cameraman		
			Time (sec)	ISNR (dB)	SSIM	Time (sec)	ISNR (dB)	SSIM	Time (sec)	ISNR (dB)	SSIM	Time (sec)	ISNR (dB)	SSIM
0.25	10	Proposed	3.882	13	0.687	9.587	10.1	0.583	1.75	8.738	0.694	2.59	7.147	0.583
		[55]	15.22	16.94	0.731	15.14	<b>14.03</b>	0.579	14.59	9.041	0.776	15.43	9.12	0.666
		[24]	118.7	13.49	0.744	123.3	11.05	0.69	122.1	9.479	0.736	123.6	7.568	0.666
		AOP	22.36	15.46	0.657	23.24	12.86	0.511	18.28	10.67	0.619	20.44	8.318	0.452
		[13]	127.7	1.458	0.14	176.3	0.7691	0.132	1324	9.277	0.00541	820.7	8.514	0.00771
		KALS	164.2	<b>17.17</b>	<b>0.796</b>	170.4	13.44	<b>0.68</b>	161.6	<b>12.87</b>	<b>0.784</b>	159.5	<b>10.38</b>	<b>0.668</b>
0.25	5	Proposed	4.49	12.78	0.644	13.66	10.27	0.535	2.073	8.72	0.656	3.682	7.472	0.529
		[55]	11.93	12.13	0.472	13.26	9.559	0.335	12.19	9.334	0.659	15.96	7.805	0.516
		[24]	125.8	12.83	<b>0.703</b>	126.2	10.99	<b>0.658</b>	116.6	8.971	<b>0.691</b>	125.5	7.634	<b>0.623</b>
		AOP	20.16	12.39	0.495	23.71	10.77	0.37	20.37	8.557	0.467	19.76	6.578	0.315
		[13]	147.2	1.091	0.0787	210	1.08	0.0834	672.6	8.503	0.00642	738.8	7.658	0.00878
		KALS	161.9	<b>14.58</b>	0.66	162.7	<b>12.22</b>	0.516	162.9	<b>11.24</b>	0.649	160.5	<b>9.584</b>	0.513
0.25	2	Proposed	<b>5.3</b>	<b>12.59</b>	0.605	17.74	10.4	0.497	2.25	8.91	0.624	3.26	7.892	0.488
		[55]	11.12	9.38	0.334	11.05	7.016	0.224	8.775	7.599	0.562	10.3	5.965	0.41
		[24]	121.8	12.42	<b>0.671</b>	122.1	<b>10.99</b>	<b>0.633</b>	93.73	8.845	<b>0.656</b>	99.8	7.887	<b>0.59</b>
		AOP	21.55	10.53	0.394	24.63	9.274	0.286	17.7	7.524	0.374	17.26	5.88	0.243
		[13]	177.7	1.839	0.0548	267.1	-1.595	0.0594	448.8	7.591	0.00734	619.1	6.649	0.00875
		KALS	161.8	<b>12.67</b>	0.544	162.1	10.39	0.391	150.8	<b>10.04</b>	0.534	155	<b>8.881</b>	0.402
0.5	10	Proposed	7.05	14.6	0.613	13.68	12.09	0.513	2.387	9.702	0.634	2.73	8.137	0.49
		[55]	17.33	<b>19.08</b>	<b>0.728</b>	15	<b>15.75</b>	<b>0.573</b>	18.27	10.91	<b>0.691</b>	20.32	<b>9.384</b>	<b>0.54</b>
		[24]	122.3	14.67	0.686	124.9	12.7	0.644	122	9.526	0.654	123.9	7.877	0.572
		AOP	26.68	18.09	0.682	30.38	15.25	0.539	24.37	10.25	0.54	23.73	7.657	0.374
		[13]	118.5	7.932	0.238	158.9	6.662	0.194	983	6.511	0.00497	791.8	5.87	0.00605
		KALS	160	16.44	0.706	158.3	13.29	0.571	158.2	<b>11.86</b>	0.663	161.6	<b>9.375</b>	0.503
0.5	5	Proposed	7.93	14.19	0.573	18.56	11.97	0.471	2.288	9.081	0.599	2.773	7.535	0.443
		[55]	12.68	14.8	0.483	12.96	11.68	0.327	11.21	<b>10.19</b>	0.576	14.83	<b>8.193</b>	0.416
		[24]	123.2	14.02	<b>0.65</b>	122.5	12.46	<b>0.615</b>	104	8.671	<b>0.616</b>	108.1	7.256	<b>0.538</b>
		AOP	28.09	<b>15.22</b>	0.529	31.83	<b>13.09</b>	0.382	24.67	8.823	0.424	25.29	6.408	0.272
		[13]	148.5	4.429	0.136	197.8	3.7	0.121	619.1	6.235	0.00504	605.3	5.514	0.00617
		KALS	162.4	13.63	0.534	159.1	11.62	0.408	156.5	10.07	0.526	157	8.09	0.365
0.5	2	Proposed	<b>9.195</b>	<b>13.76</b>	0.539	22.37	11.75	0.439	<b>2.22</b>	<b>8.791</b>	0.573	<b>2.543</b>	<b>7.226</b>	0.41
		[55]	12.03	11.9	0.341	12.65	8.802	0.213	8.082	8.281	0.504	10.16	6.934	0.347
		[24]	123.3	<b>13.57</b>	<b>0.624</b>	122.6	<b>12.19</b>	<b>0.592</b>	84.47	8.28	<b>0.59</b>	89.89	6.946	<b>0.517</b>
		AOP	29.01	13.39	0.429	32.07	11	0.281	23.51	7.88	0.352	21.46	5.695	0.22
		[13]	182.2	2.811	0.0958	224.3	2.244	0.0824	412.8	5.837	0.00509	556.7	5.055	0.00609
		KALS	161.6	11.25	0.414	159.8	9.389	0.302	145	<b>8.971</b>	0.436	149.4	<b>7.215</b>	0.288
0.7	10	Proposed	13.5	14.2	0.516	22.82	12.26	0.436	3.11	8.422	0.553	3.1	6.282	0.385
		[55]	13.89	<b>19.07</b>	<b>0.714</b>	15.93	<b>15.77</b>	0.573	18.41	<b>9.403</b>	0.523	18.74	<b>6.876</b>	0.35
		[24]	122.7	14.34	0.621	123.2	12.77	0.592	121.7	7.865	<b>0.563</b>	121.2	5.972	<b>0.487</b>
		AOP	39.2	18.66	0.699	44.64	15.69	<b>0.576</b>	30.49	8.242	0.439	30.44	5.753	0.309
		[13]	131.8	12.2	0.334	182.7	10.35	0.248	846.4	5.13	0.00394	717.6	4.554	0.00464
		KALS	155.2	9.694	0.394	154.3	8.298	0.247	154	8.358	0.461	157.1	6.043	0.329
0.7	5	Proposed	16.04	13.71	0.485	29.1	11.99	0.402	2.575	8.011	0.53	2.302	6.019	0.37
		[55]	12.73	15.92	0.502	14.95	12.24	0.335	11.99	<b>8.621</b>	0.447	11.67	<b>6.344</b>	0.308
		[24]	122.4	13.85	0.591	120	12.47	<b>0.565</b>	105.1	7.421	<b>0.542</b>	99.23	5.798	<b>0.478</b>
		AOP	40.01	<b>16.61</b>	<b>0.573</b>	43.66	<b>13.51</b>	0.417	32.31	7.68	0.387	27.63	5.36	0.265
		[13]	169.4	8.316	0.196	231.3	6.934	0.149	605	5.022	0.00425	602.6	4.398	0.00469
		KALS	159.1	8.271	0.269	156.4	6.918	0.161	148.7	7.755	0.389	149.8	5.614	0.264
0.7	2	Proposed	16.31	13.17	0.457	33.75	11.6	0.378	<b>2.692</b>	<b>7.545</b>	0.504	<b>2.178</b>	<b>5.761</b>	0.355
		[55]	14.5	13.09	0.362	12.41	8.97	0.207	7.828	7.207	0.402	8.495	4.956	0.266
		[24]	119.4	13.31	<b>0.569</b>	109.7	<b>12.23</b>	<b>0.544</b>	83.91	6.968	<b>0.519</b>	85.57	5.571	<b>0.47</b>
		AOP	36.57	<b>14.47</b>	0.471	47.75	10.62	0.294	26.53	7.091	0.339	26.92	5.028	0.237
		[13]	216.5	6.185	0.135	212.8	4.792	0.0985	387.8	4.803	0.00424	442.6	4.16	0.00467
		KALS	157.4	6.852	0.205	156.7	5.56	0.122	138.7	7.052	0.33	142.1	5.298	0.233

In this case, we use fast two phase deblurring [13], since AOP, although faster, also divides the observed image by 255, and in the case of Rayleigh speckle, the image is already normalized. We summarize our results for the lena and cameraman images for different fractions of missing pixels, in Table IV. We can see that taking into account the statistical model offers an improvement in terms of the NMAE. As with the additive and Gaussian noise case, the existing methods for blind inpainting do not always work well for large fractions

of missing pixels, above 50%, or take a long time, over 10 minutes.

For the Lena image, a cropped region from the noisy image with 50% of the pixels missing is shown in Figure 12(b) and the respective estimates using the proposed method and [13] after logarithmic transformation are shown in Figures 12(c) and 12(d). For a pixel loss of 70%, the observed image and estimate using the proposed method are shown in Figures 12(e) and 12(f).

TABLE III

INPAINTING WITH POISSON NOISE.  $\kappa$  INDICATES THE FRACTION OF MISSING PIXELS. (\*) AOP IS USED AFTER ANSCOMBE TRANSFORMATION

$\kappa$	Method	Lena				Cameraman			
		Time (sec.)	ISNR (dB)	SSIM	Mask err. (%)	Time (sec.)	ISNR (dB)	SSIM	Mask err. (%)
0.25	Proposed	<b>23</b>	<b>18.8</b>	<b>0.949</b>	<b>0.0027</b>	<b>14.6</b>	<b>15.71</b>	<b>0.885</b>	<b>2.33</b>
	AOP(*)	195	13.8	0.875	0.141	41.2	12.65	0.712	4.32
0.5	Proposed	<b>36.3</b>	<b>19.8</b>	<b>0.935</b>	<b>9.54e-05</b>	<b>16.6</b>	<b>15.88</b>	<b>0.859</b>	<b>0.87</b>
	AOP(*)	377	11.3	0.798	4.29	84.6	9.41	0.669	13.8
0.7	Proposed	55.4	18.6	0.907	9.54e-05	21.8	14.69	0.82	0.43
0.9	Proposed	141	14.3	0.811	0	53	11.3	0.72	0.048

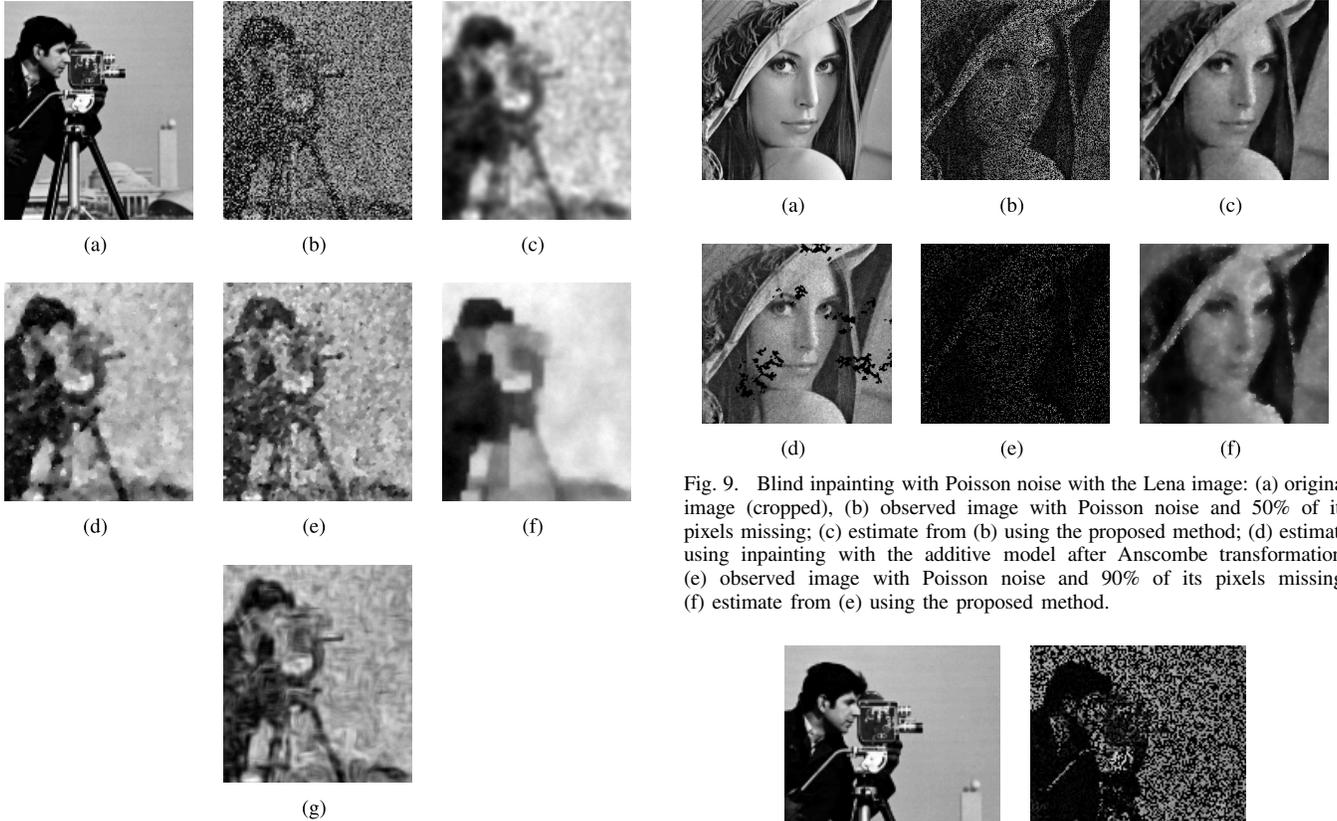


Fig. 9. Blind inpainting with Poisson noise with the Lena image: (a) original image (cropped), (b) observed image with Poisson noise and 50% of its pixels missing; (c) estimate from (b) using the proposed method; (d) estimate using inpainting with the additive model after Anscombe transformation; (e) observed image with Poisson noise and 90% of its pixels missing; (f) estimate from (e) using the proposed method.

Fig. 8. Random valued impulse noise (noisy pixels have values in the interval  $[0, 255]$ ) removal with the Cameraman image. (a) Cropped original image, (b) observed image with 50% noisy pixels and  $\text{SNR} = 5$  dB; estimates of (b) using: (c) proposed method, (d) [54], (e) AOP [69], (f) [24], (g) KALS [66].

Blind inpainting with Poisson noise for the cameraman image with 50% and 70% of its pixels missing is illustrated in Figure 13.

Figure 14 demonstrates blind inpainting with the Rayleigh multiplicative noise model for the transversal ultrasound (US) image of the carotid artery. The noisy radio frequency (RF) envelope image is shown in Figure 14(a). The observed image with 75% of the pixels missing is shown in Figure 14(b). The estimate using the proposed method is shown in Figure 14(c), with the diagonal profiles shown in Figure 14(e). We can see from the result of the binary XOR operation between the sampling mask and its estimate shown in Figure 14(d), that the incorrectly estimated bits are in the region of low pixel values in the RF image. The computation time was 119.3 seconds for the image of size  $201 \times 201$ .

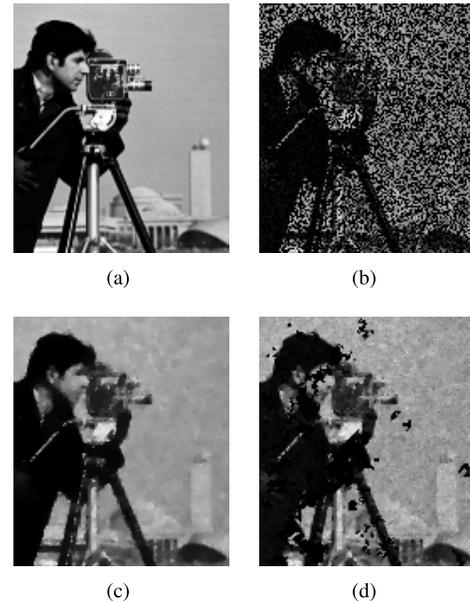


Fig. 10. Blind inpainting with Poisson noise with the Cameraman image: (a) original image (cropped), (b) observed image with Poisson noise and 50% of its pixels missing; (c) estimate using the proposed method; (d) estimate using inpainting with the additive model after Anscombe transformation.

#### D. Maximum Extent of Missing Pixels and Sparse Recovery

We present some results of blind inpainting with the Lena image, for different percentages of missing pixels,

TABLE IV  
INPAINTING WITH RAYLEIGH NOISE.  $\kappa$  INDICATES THE FRACTION OF MISSING PIXELS.  
(\* ) THE ADDITIVE MODEL IS USED AFTER LOGARITHMIC TRANSFORMATION

$\kappa$	Method	Lena				Cameraman			
		Time (sec.)	NMAE	SSIM	Mask err. (%)	Time (sec.)	NMAE	SSIM	Mask err. (%)
0.25	Proposed	41	<b>3.12e-06</b>	<b>0.973</b>	0.664	<b>15.5</b>	<b>1.23e-05</b>	<b>0.975</b>	11.6
	Additive(*)	11.3	3.81e-06	0.958	<b>0.00019</b>	17.3	1.53e-05	0.961	<b>0.00114</b>
0.5	Proposed	<b>66.8</b>	<b>3.16e-06</b>	<b>0.972</b>	<b>0.317</b>	24.4	<b>1.26e-05</b>	<b>0.974</b>	<b>2.13</b>
	Additive(*)	75.8	3.82e-06	0.958	4.9	<b>19.1</b>	1.53e-05	0.961	4.29
0.7	Proposed	116	3.17e-06	0.972	0.0835	156	1.27e-05	0.973	0.484

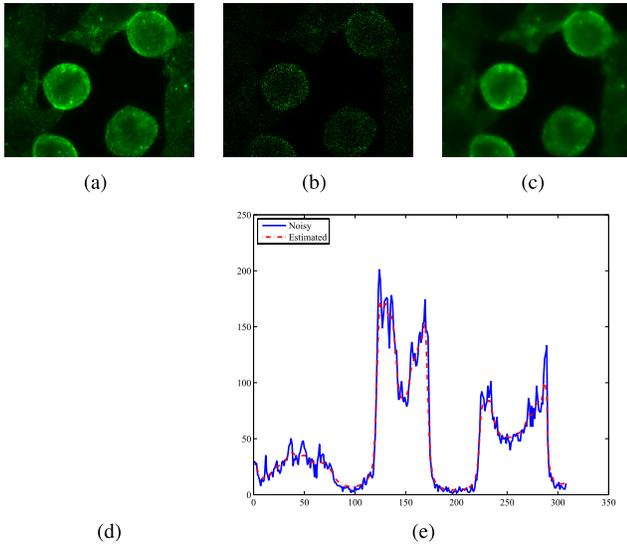


Fig. 11. Blind inpainting with (Poisson) fluorescence microscopy images: (a) noisy image, (b) observed image with 75% of the pixels missing, (c) estimate, (d) result of the binary XOR operation between the mask and its estimate, (e) diagonal profiles of the green component of the noisy and estimated images.

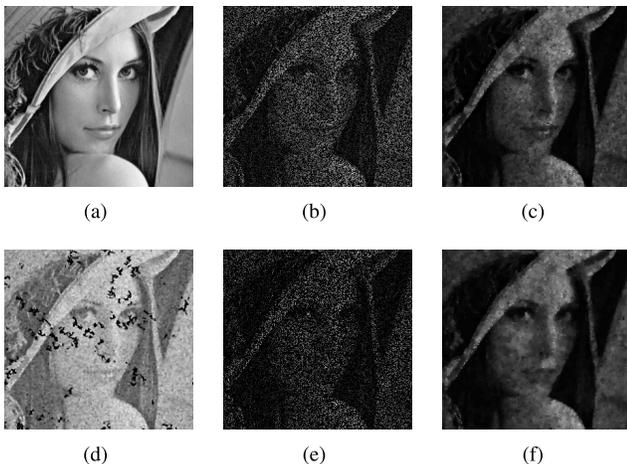


Fig. 12. Blind inpainting with Rayleigh noise with the Lena image: (a) original image (cropped), (b) observed image with Rayleigh noise and 50% of its pixels missing; (c) estimate from (b) using the proposed method; (d) estimate using inpainting with the additive model after logarithmic transformation; (e) observed image with Rayleigh noise and 70% of its pixels missing; (f) estimate from (e) using the proposed method.

to study the fractions of missing pixels for which our method works and the effect of  $\lambda_2$ , the regularization parameter of the  $\ell_0$ -norm regularizer on the mask term.

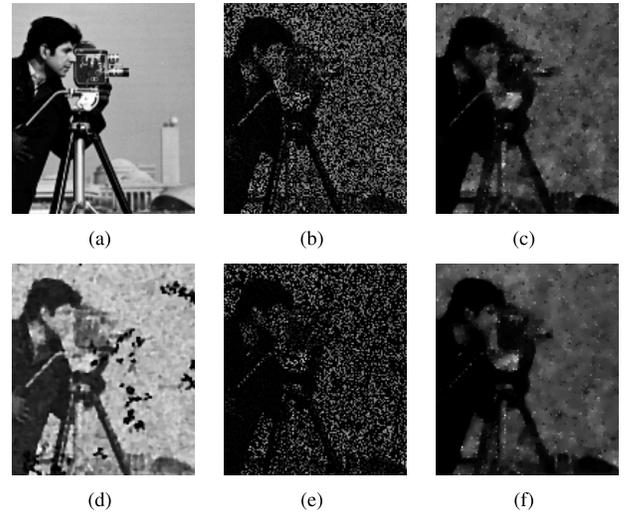


Fig. 13. Blind inpainting with Rayleigh noise with the Cameraman image: (a) original image (cropped), (b) observed image with Rayleigh noise and 50% of its pixels missing; (c) estimate using the proposed method; (d) estimate using inpainting with the additive model after logarithmic transformation; (e) observed image with Rayleigh noise and 70% of its pixels missing; (f) estimate from (e) using the proposed method.

For the case when  $\|\mathbf{v}\|_0 = 0$ , all elements of  $\mathbf{g}$  correspond to observed pixels, and the mask estimate  $\hat{\mathbf{h}}$  is a matrix of binary ones. Thus all elements of the observed image are considered valid pixels, even if they are zeros. Therefore with few missing pixels, with a small value of the parameter  $\lambda_2$ , the hard threshold on  $\mathbf{v}$  allows more elements to be different from zero, and therefore equal to  $-K$ , thereby corresponding to zeros (missing pixels) in the estimate of the mask after the exponential operation, which is incorrect. A high value of  $\lambda_2$  forces most elements of  $\mathbf{v}$  to be equal to zero, and consequently correspond to ones in the mask, leading to fewer errors in the mask estimation. We can observe this in the left hand side of the plot of the mask errors (divided by the number of pixels), as shown in Figure 15(a).

In the opposite case, when all elements of  $\mathbf{v}$  are negative, none of the elements of  $\mathbf{g}$  would contribute to the estimation of the image and the estimate of the image  $\hat{\mathbf{x}}$  would be a constant value. Thus, with a large fraction of the pixels missing, with a small value of the parameter  $\lambda_2$ , the hard threshold on  $\mathbf{v}$  allows more elements to be different from zero, and therefore equal to  $-K$ , thereby corresponding to more zeros (missing pixels) in the estimate of the mask. A high value of  $\lambda_2$  forces most elements of  $\mathbf{v}$  to be equal

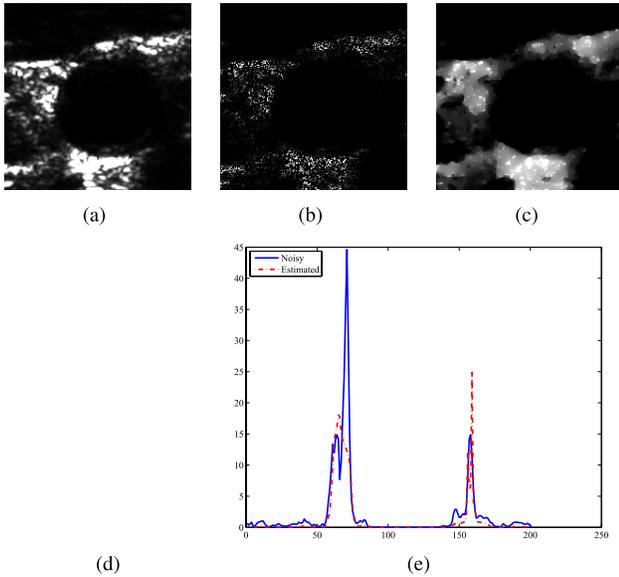


Fig. 14. Blind inpainting with (Rayleigh) transversal ultrasound image of the carotid artery: (a) noisy radio frequency (RF) image, (b) observed image with 75% of the pixels missing, (c) estimate, (d) result of the binary XOR operation between the mask and its estimate, (e) diagonal profiles of the noisy and estimated images.

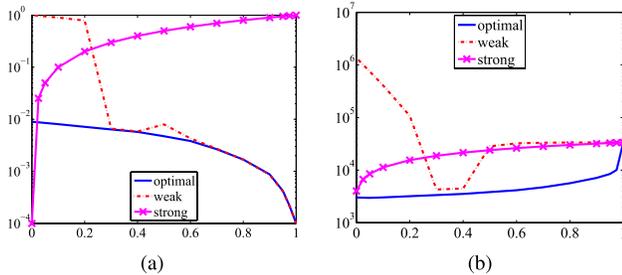


Fig. 15. Evolution in errors of (a) mask estimate; (b) image estimate (MSE), as a function of the fraction of missing pixels, for  $\lambda_2 = 0.1$  (optimal),  $10^{-3}$  (weak), 100 (strong).

to zero, and consequently corresponding to more ones after the exponential transformation, which leads to more errors in the mask estimation. We can observe this in the right hand side of the plot in Figure 15(a).

Notice from the plots of the errors in the mask estimate and the MSE shown in Figure 15(b), that our choice of  $\lambda_2$ , which was arrived at by hand tuning, led to low values of both errors roughly over the entire range of the fraction of missing pixels. Therefore, through an optimal choice of the regularization parameter, outliers are rejected and valid observations are retained.

## V. CONCLUSIONS AND FUTURE WORK

We have presented an iterative method for image inpainting without knowing the locations of the missing pixels, based on alternating minimization to simultaneously estimate the image and observation mask. The method is computationally faster than existing methods and can deal better with a larger fraction of missing pixels than most existing methods. The proposed method has been formulated for the Rayleigh

speckle and Poisson noise models as well, and was found to be more accurate than using transforming the model into an additive one without taking into account the respective statistics. Overall, the method is computationally fast for the removal of impulse noise.

Based on the results obtained with real immunofluorescence and ultrasound images, current and future research includes using the estimation of masks to help in obtaining optimal sampling patterns and possibly even segmenting medical images, by using our method to determine which pixel values are reliable and which ones are outliers. This could be relevant for speckle decomposition.

## ACKNOWLEDGMENTS

The authors thank the associate editor and the anonymous reviewers for their useful feedback, and the authors of [24], [54], and [69], for sharing the implementations of their algorithms with them.

## REFERENCES

- [1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, Sep. 2010.
- [2] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–695, Mar. 2011.
- [3] M. V. Afonso and J. M. R. Sanches, "A total variation based reconstruction algorithm for 3D ultrasound," in *Pattern Recognition and Image Analysis (Lecture Notes in Computer Science)*, vol. 7887. Berlin, Germany: Springer-Verlag, 2013, pp. 149–156.
- [4] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [5] F. J. Anscombe, "The transformation of Poisson, binomial and negative-binomial data," *Biometrika*, vol. 35, nos. 3–4, pp. 246–254, 1948.
- [6] C. A. Z. Barcelos and M. A. Batista, "Image restoration using digital inpainting and noise removal," *Image Vis. Comput.*, vol. 25, no. 1, pp. 61–69, 2007.
- [7] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
- [8] S. Becker, J. Bobin, and E. J. Candès, "NESTA: A fast and accurate first-order method for sparse recovery," *SIAM J. Imag. Sci.*, vol. 4, no. 1, pp. 1–39, 2011.
- [9] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, 2000, pp. 417–424.
- [10] J. M. Bioucas-Dias and M. A. T. Figueiredo, "An iterative algorithm for linear inverse problems with compound regularizers," in *Proc. 15th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2008, pp. 685–688.
- [11] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *J. Fourier Anal. Appl.*, vol. 14, nos. 5–6, pp. 629–654, 2008.
- [12] A. C. Bovik, *Handbook of Image and Video Processing*. New York, NY, USA: Academic, 2010.
- [13] J.-F. Cai, R. H. Chan, and M. Nikolova, "Fast two-phase image deblurring under impulse noise," *J. Math. Imag. Vis.*, vol. 36, no. 1, pp. 46–53, 2010.
- [14] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [15] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.
- [16] A. Chambolle, "An algorithm for total variation minimization and applications," *J. Math. Imag. Vis.*, vol. 20, nos. 1–2, pp. 89–97, 2004.

- [17] R. H. Chan, C. Hu, and M. Nikolova, "An iterative procedure for removing random-valued impulse noise," *IEEE Signal Process. Lett.*, vol. 11, no. 12, pp. 921–924, Dec. 2004.
- [18] T. Chan, S. Esedoglu, F. Park, and A. Yip, "Recent developments in total variation image restoration," in *Handbook of Mathematical Models in Computer Vision*. 2005, pp. 17–30.
- [19] T. Chen and H. R. Wu, "Adaptive impulse detection using center-weighted median filters," *IEEE Signal Process. Lett.*, vol. 8, no. 1, pp. 1–3, Jan. 2001.
- [20] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [21] J. Dahl, P. C. Hansen, S. H. Jensen, and T. L. Jensen, "Algorithms and software for total variation image reconstruction via first-order methods," *Numer. Algorithms*, vol. 53, no. 1, pp. 67–92, 2010.
- [22] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1715–1728, Apr. 2012.
- [23] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, 2004.
- [24] B. Dong, H. Ji, J. Li, Z. Shen, and Y. Xu, "Wavelet frame based blind image inpainting," *Appl. Comput. Harmon. Anal.*, vol. 32, no. 2, pp. 268–279, 2012.
- [25] Y. Dong, R. H. Chan, and S. Xu, "A detection statistic for random-valued impulse noise," *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1112–1120, Apr. 2007.
- [26] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.
- [27] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [28] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Statist. Assoc.*, vol. 90, no. 432, pp. 1200–1224, 1995.
- [29] S. Durand, J. Fadili, and M. Nikolova, "Multiplicative noise removal using  $\ell_1$  fidelity on frame coefficients," *J. Math. Imag. Vis.*, vol. 36, no. 3, pp. 201–226, 2010.
- [30] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Program.*, vol. 55, no. 3, pp. 293–318, 1992.
- [31] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer-Verlag, 2010.
- [32] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, p. 947, 2007.
- [33] E. Esser, "Applications of Lagrangian-based alternating direction methods and connections to split Bregman," Dept. Math., Univ. California, Los Angeles, Los Angeles, CA, USA, Tech. Rep. 09-31, 2009.
- [34] E. Esser and X. Zhang, "Nonlocal patch-based image inpainting through minimization of a sparsity promoting nonconvex functional," Dept. Math., Univ. California Irvine, Irvine, CA, USA, Tech. Rep., 2014.
- [35] J. A. Fessler, "Hybrid Poisson/polynomial objective functions for tomographic image reconstruction from transmission scans," *IEEE Trans. Image Process.*, vol. 4, no. 10, pp. 1439–1450, Oct. 1995.
- [36] M. A. T. Figueiredo and J. M. Bioucas-Dias, "Restoration of Poissonian images using alternating direction optimization," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3133–3145, Dec. 2010.
- [37] G. Gilboa and S. Osher, "Nonlocal linear image regularization and supervised segmentation," *Multiscale Model. Simul.*, vol. 6, no. 2, pp. 595–630, 2007.
- [38] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Model. Simul.*, vol. 7, no. 3, pp. 1005–1028, 2008.
- [39] R. Giryes and M. Elad. (2013). "Sparsity based Poisson denoising with dictionary learning." [Online]. Available: <http://arxiv.org/abs/1309.4306>
- [40] R. Giryes and M. Elad, "Sparsity based poisson inpainting," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, 2014.
- [41] T. Goldstein and S. Osher, "The split Bregman method for  $\ell_1$  regularized problems," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 323–343, 2009.
- [42] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB*, vol. 2. hboUpper Saddle River, NJ, USA: Prentice-Hall, 2009.
- [43] M. R. Hestenes, "Multiplier and gradient methods," *J. Optim. Theory Appl.*, vol. 4, no. 5, pp. 303–320, 1969.
- [44] S. Lefkimmatis and M. Unser, "Poisson image reconstruction with Hessian Schatten-norm regularization," *IEEE Trans. Image Process.*, vol. 22, no. 11, pp. 4314–4327, Nov. 2013.
- [45] Q. Li, W. Yin, and Z. Deng, "Image-based face illumination transferring using logarithmic total variation models," *Vis. Comput.*, vol. 26, no. 1, pp. 41–49, 2010.
- [46] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, 2007.
- [47] M. Maktalo and A. Foi, "Optimal inversion of the Anscombe transformation in low-count Poisson image denoising," *IEEE Trans. Image Process.*, vol. 20, no. 1, pp. 99–109, Jan. 2011.
- [48] S. Masnou and J.-M. Morel, "Level lines based disocclusion," in *Proc. Int. Conf. Image Process. (ICIP)*, Oct. 1998, pp. 259–263.
- [49] J. Nocedal and S. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer-Verlag, 2006.
- [50] A. Oh, Z. Harmany, and R. Willett, "Logarithmic total variation regularization for cross-validation in photon-limited imaging," in *Proc. 20th IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2013, pp. 484–488.
- [51] J. Portilla, "Image restoration through  $\ell_0$  analysis-based sparse optimization in tight frames," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 3909–3912.
- [52] M. J. D. Powell, "A method for nonlinear constraints in minimization problems," in *Optimization*. New York, NY, USA: Academic, 1969, pp. 283–298.
- [53] I. C. Rodrigues and J. M. R. Sanches, "Convex total variation denoising of Poisson fluorescence confocal images with anisotropic filtering," *IEEE Trans. Image Process.*, vol. 20, no. 1, pp. 146–160, Jan. 2011.
- [54] P. Rodriguez, R. Rojas, and B. Wohlberg, "Mixed Gaussian-impulse noise image restoration via total variation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 1077–1080.
- [55] P. Rodriguez and B. Wohlberg, "Efficient minimization method for a generalized total variation functional," *IEEE Trans. Image Process.*, vol. 18, no. 2, pp. 322–332, Feb. 2009.
- [56] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning sparse dictionaries for sparse signal approximation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1553–1564, Mar. 2010.
- [57] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, nos. 1–4, pp. 259–268, 1992.
- [58] J. Salmon, Z. Harmany, C.-A. Deledalle, and R. Willett, "Poisson noise reduction with non-local PCA," *J. Math. Imag. Vis.*, vol. 48, no. 2, pp. 279–294, 2014.
- [59] J. C. R. Seabra, "Medical ultrasound B-mode modeling, de-speckling and tissue characterization assessing the atherosclerotic disease," Ph.D. dissertation, Dept. Biomed. Eng., Instituto Superior Técnico, Lisbon, Portugal, May 2011.
- [60] J. Seabra, J. Xavier, and J. Sanches, "Convex ultrasound image reconstruction with log-Euclidean priors," in *Proc. Eng. Med. Biol. Soc. (EMBS)*, 2008, pp. 435–438.
- [61] J. Shen and T. F. Chan, "Mathematical models for local nontexture inpaintings," *SIAM J. Appl. Math.*, vol. 62, no. 3, pp. 1019–1043, 2002.
- [62] O. V. Solberg, F. Lindseth, H. Torp, R. E. Blake, and T. A. N. Hernes, "Freehand 3D ultrasound reconstruction algorithms—A review," *Ultrasound Med. Biol.*, vol. 33, no. 7, pp. 991–1009, 2007.
- [63] J.-L. Starck, M. Elad, and D. L. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1570–1582, Oct. 2005.
- [64] J.-L. Starck, F. D. Murtagh, and A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
- [65] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, 2008.
- [66] Y. Wang, A. Szlam, and G. Lerman, "Robust locally linear analysis with applications to image denoising and blind inpainting," *SIAM J. Imag. Sci.*, vol. 6, no. 1, pp. 526–562, 2013.
- [67] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [68] Y. Xiao, T. Zeng, J. Yu, and M. K. Ng, "Restoration of images corrupted by mixed Gaussian-impulse noise via  $\ell_1$ - $\ell_0$  minimization," *Pattern Recognit.*, vol. 44, no. 8, pp. 1708–1720, 2011.

- [69] M. Yan, "Restoration of images corrupted by impulse noise and mixed Gaussian impulse noise using blind inpainting," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1227–1245, 2013.
- [70] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for  $\ell_1$  minimization with applications to compressed sensing," *SIAM J. Imag. Sci.*, vol. 1, no. 1, pp. 143–168, 2008.



**Manya V. Afonso** received the B.E. degree in electronics and telecommunication engineering from Goa University, India, in 2003, and the M.Tech. degree in communication engineering from IIT Delhi, in 2005.

He received the Ph.D. degree from the Instituto Superior Tecnico, in 2011. He was a Researcher with the Instituto de Telecomunicacoes, Lisboa. He holds a post-doctoral position with the Instituto de Sistemas e Robótica, Instituto Superior Técnico.

Afonso's research interests include image processing and analysis, inverse problems, optimization, machine learning, computer vision, and video surveillance.



**Joao Miguel Raposo Sanches** was with the Department of Electrical and Computer Engineering, where he has taught in the area of signal processing, systems, and control. He has been actively involved in the course of biomedical engineering, and advising master's and Ph.D. thesis in the area of bioengineering, in particular, on biomedical engineering (BME). He currently teaches with the Department of Bioengineering, Instituto Superior Técnico.

He is a Researcher with the Signal and Image Processing Group, Institute for Systems and Robotics. His work has been focused on BME, namely, in biological and medical image processing and statistical signal processing of physiological and behavioral data. Morphological and textural characterization of tissues from ultrasound images is central issues but he is also working on fluorescence images of microscopy for biological quantification purposes. He is also involved in the development of signal processing algorithms for polysomnography data and smartphones applications for long-term monitoring for sleep disorders diagnosis purposes. In this scope, heart rate variability analysis is today one of his main interests.

Almost all of his research work is in collaboration with medical and biological institutions, namely, the Medical School, University of Lisbon, the Institute of Molecular Pathology and Immunology, University of Porto, and the Electroencephalography and Clinical Neurophysiology Center. Several publications and patents were already produced in the scope of this collaborative work.

He has been a Senior Member of the IEEE Engineering in Medicine and Biology Society since 2011, and a member of the Bio Imaging and Signal Processing Technical Committee of the IEEE Signal Processing Society.