# $\ell_0$TV: A Sparse Optimization Method for Impulse Noise Image Restoration

Ganzhao Yuan, Bernard Ghanem

**Abstract**—Total Variation (TV) is an effective and popular prior model in the field of regularization-based image processing. This paper focuses on total variation for removing impulse noise in image restoration. This type of noise frequently arises in data acquisition and transmission due to many reasons, e.g. a faulty sensor or analog-to-digital converter errors. Removing this noise is an important task in image restoration. State-of-the-art methods such as Adaptive Outlier Pursuit(AOP) [57], which is based on TV with $\ell_{02}$-norm data fidelity, only give sub-optimal performance. In this paper, we propose a new sparse optimization method, called $\ell_0TV$-PADMM, which solves the TV-based restoration problem with $\ell_0$-norm data fidelity. To effectively deal with the resulting non-convex non-smooth optimization problem, we first reformulate it as an equivalent biconvex Mathematical Program with Equilibrium Constraints (MPEC), and then solve it using a proximal Alternating Direction Method of Multipliers (PADMM). Our $\ell_0TV$-PADMM method finds a desirable solution to the original $\ell_0$-norm optimization problem and is proven to be convergent under mild conditions. We apply $\ell_0TV$-PADMM to the problems of image denoising and deblurring in the presence of impulse noise. Our extensive experiments demonstrate that $\ell_0TV$-PADMM outperforms state-of-the-art image restoration methods.

**Index Terms**—Total Variation, Image Restoration, MPEC, $\ell_0$ Norm Optimization, Proximal ADMM, Impulse Noise.

## 1 Introduction

Image restoration is an inverse problem, which aims at estimating the original *clean* image $\mathbf{u}$ from a blurry and/or noisy observation $\mathbf{b}$. Mathematically, this problem is formulated as:

$$\mathbf{b} = ((\mathbf{K}\mathbf{u}) \odot \boldsymbol{\varepsilon}_m) + \boldsymbol{\varepsilon}_a, \tag{1}$$

where $\mathbf{K}$ is a linear operator, $\boldsymbol{\varepsilon}_m$ and $\boldsymbol{\varepsilon}_a$ are the noise vectors, and $\odot$ denotes an elementwise product. Let $\mathbf{1}$ and $\mathbf{0}$ be column vectors of all entries equal to one and zero, respectively. When $\boldsymbol{\varepsilon}_m = \mathbf{1}$ and $\boldsymbol{\varepsilon}_a \neq \mathbf{0}$ (or $\boldsymbol{\varepsilon}_m \neq \mathbf{0}$ and $\boldsymbol{\varepsilon}_a = \mathbf{0}$), (1) corresponds to the additive (or multiplicative) noise model. For convenience, we adopt the vector representation for images, where a 2D $M \times N$ image is column-wise stacked into a vector $\mathbf{u} \in \mathbb{R}^{n\times 1}$ with $n = M \times N$. So, for completeness, we have $\mathbf{1}, \mathbf{0}, \mathbf{b}, \mathbf{u}, \boldsymbol{\varepsilon}_a, \boldsymbol{\varepsilon}_m \in \mathbb{R}^n$, and $\mathbf{K} \in \mathbb{R}^{n\times n}$. Before proceeding, we present an image restoration example on the well-known 'barbara' image using our proposed method for solving impulse noise removal in Figure 1.

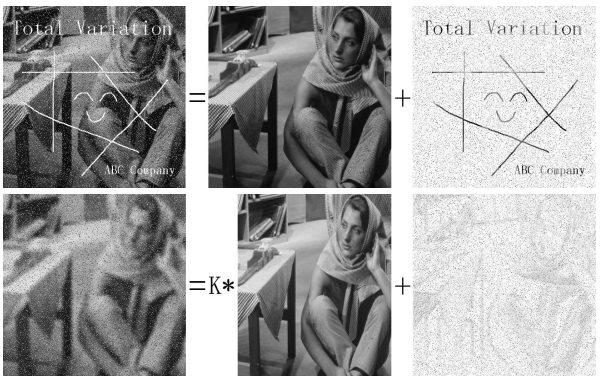



Figure 1: An example of an image recovery result using our proposed $\ell_0$TV-PADMM method. Left column: corrupted image. Middle column: recovered image. Right column: absolute residual between these two images.

In general image restoration problems, $\mathbf{K}$ represents a certain linear operator, e.g. convolution, wavelet transform, etc., and recovering $\mathbf{u}$ from $\mathbf{b}$ is known as image deconvolution or image deblurring. When $\mathbf{K}$ is the identity operator, estimating $\mathbf{u}$ from $\mathbf{b}$ is referred to as image denoising [48]. The problem of estimating $\mathbf{u}$ from $\mathbf{b}$ is called a linear inverse problem which, for most scenarios of practical interest, is ill-posed due to the singularity and/or the ill-conditioning of $\mathbf{K}$. Therefore, in order to stabilize the recovery of $\mathbf{u}$, it is necessary to incorporate prior-enforcing regularization on the solution.

- *Ganzhao Yuan (yuanganzhao@gmail.com) is with School of Data and Computer Science, Sun Yat-sen University (SYSU), China.*
- *Bernard Ghanem is with Visual Computing Center, King Abdullah University of Science and Technology (KAUST), Saudi Arabia.*

*Manuscript received April 19, 2005; revised August 26, 2015.*

Table 1: Data Fidelity Models

| Data Fidelity Function | Noise and References |
|---|---|
| $\ell_2(\mathbf{Ku},\mathbf{b}) = \|\mathbf{Ku}-\mathbf{b}\|_2^2$ | add. Gaussian noise [12], [45] |
| $\ell_1(\mathbf{Ku},\mathbf{b}) = \|\mathbf{Ku}-\mathbf{b}\|_1$ | add. Laplace noise [21], [58] |
| $\ell_\infty(\mathbf{Ku},\mathbf{b}) = \|\mathbf{Ku}-\mathbf{b}\|_\infty$ | add. uniform noise [20], [49] |
| $\ell_p(\mathbf{Ku},\mathbf{b}) = \langle \mathbf{Ku}-\mathbf{b}\odot\log(\mathbf{Ku}),\mathbf{1}\rangle$ | mul. Poisson noise [34], [47] |
| $\ell_g(\mathbf{Ku},\mathbf{b}) = \langle \log(\mathbf{Ku})+\mathbf{b}\odot\frac{1}{\mathbf{Ku}},\mathbf{1}\rangle$ | mul. Gamma noise [3], [51] |
| $\ell_r(\mathbf{Ku},\mathbf{b}) = \langle \log(\mathbf{Ku})+\mathbf{b}\odot\mathbf{b}\odot\frac{1}{\mathbf{Ku}^2},\mathbf{1}\rangle$ | mul. Rayleigh noise [2], [46] |
| $\ell_{02}(\mathbf{Ku},\mathbf{b}) = \|\mathbf{Ku}-\mathbf{b}+\mathbf{z}\|_2^2, s.t. \|\mathbf{z}\|_0 \le k$ | mixed Gaussian impulse noise [57] |
| $\ell_0(\mathbf{Ku},\mathbf{b}) = \|\mathbf{Ku}-\mathbf{b}\|_0$ | add./mul. impulse noise **[ours]** |

Table 2: Regularization Models

| Regularization Function | Description and References |
|---|---|
| $\Omega_{\text{tik}}(\mathbf{g},\mathbf{h}) = \sum_{i=1}^n \mathbf{g}_i^2+\mathbf{h}_i^2$ | Tikhonov-like [1] |
| $\Omega_{\text{tv}_2}(\mathbf{g},\mathbf{h}) = \sum_{i=1}^n (\mathbf{g}_i^2+\mathbf{h}_i^2)^{\frac{1}{2}}$ | Isotropic [45], [51] |
| $\Omega_{\text{tv}_1}(\mathbf{g},\mathbf{h}) = \sum_{i=1}^n \|\mathbf{g}_i\|+\|\mathbf{h}_i\|$ | Anisotropic [48], [58] |
| $\Omega_{\text{stv}}(\mathbf{g},\mathbf{h}) = \sum_{i=1}^n (\mathbf{g}_i^2+\mathbf{h}_i^2+\varepsilon^2)^{\frac{1}{2}}$ | smooth TV [16], [49] |
| $\Omega_{\text{pot}}(\mathbf{g},\mathbf{h}) = \sum_{i=1}^n \|\mathbf{g}_i\|_0+\|\mathbf{h}_i\|_0$ | Potts model [54], [55] |
| $\Omega_{\text{hub}}(\mathbf{g},\mathbf{h}) = \sum_{i=1}^n \varphi(\mathbf{g}_i;\mathbf{h}_i)$, $\varphi(\mathbf{g}_i;\mathbf{h}_i) = \begin{cases} \varepsilon\|\mathbf{g}_i;\mathbf{h}_i\|_2^2/2; & \|\mathbf{g}_i;\mathbf{h}_i\|_2 \le 1/\varepsilon \\ \|\mathbf{g}_i;\mathbf{h}_i\|_2 - \varepsilon/2; & \text{otherwise} \end{cases}$ | Huber-Like [44] |

Therefore, image restoration can be modelled globally as the following optimization problem:

$$\min_{\mathbf{u}} \ \ell(\mathbf{Ku},\mathbf{b}) + \lambda\,\Omega(\boldsymbol{\nabla}_x\mathbf{u},\boldsymbol{\nabla}_y\mathbf{u}), \tag{2}$$

where $\ell(\mathbf{Ku},\mathbf{b})$ measures the data fidelity between $\mathbf{Ku}$ and the observation $\mathbf{b}$, $\boldsymbol{\nabla}_x \in \mathbb{R}^{n\times n}$ and $\boldsymbol{\nabla}_y \in \mathbb{R}^{n\times n}$ are two suitable linear transformation matrices such that $\boldsymbol{\nabla}_x\mathbf{u} \in \mathbb{R}^n$ and $\boldsymbol{\nabla}_y\mathbf{u} \in \mathbb{R}^n$ compute the discrete gradients of the image $\mathbf{u}$ along the $x$-axis and $y$-axis, respectively[1], $\Omega(\boldsymbol{\nabla}_x\mathbf{u},\boldsymbol{\nabla}_y\mathbf{u})$ is the regularizer on $\boldsymbol{\nabla}_x\mathbf{u}$ and $\boldsymbol{\nabla}_y\mathbf{u}$, and $\lambda$ is a positive parameter used to balance the two terms for minimization. Apart from regularization, other prior information such as bound constraints [5], [68] or hard constraints can be incorporated into the general optimization framework in (2).

### 1.1 Related Work

This subsection presents a brief review of existing TV methods, from the viewpoint of data fidelity models, regularization models and optimization algorithms.

**Data Fidelity Models:** The fidelity function $\ell(\cdot,\cdot)$ in (2) usually penalizes the difference between $\mathbf{Ku}$ and $\mathbf{b}$ by using different norms/divergences. Its form depends on the assumed distribution of the noise model. Some typical noise models and their corresponding fidelity terms are listed in Table 1. The classical TV model [45] only considers TV minimization involving the squared $\ell_2$-norm fidelity term for recovering images corrupted by additive Gaussian noise. However, this model is far from optimal when the noise is not Gaussian. Other works [21], [58] extend classical TV to use the $\ell_1$-norm in the fidelity term. Since the $\ell_1$-norm fidelity term coincides with the probability density function of Laplace distribution, it is suitable for image restoration in the presence of Laplace noise. Moreover, additive uniform noise [20], [49], multiplicative Poisson noise [34], and multiplicative Gamma noise [51] have been considered in the literature. Recently, a sparse noise model using an $\ell_{02}$-norm for data fidelity has been investigated in [57] to remove impulse and mixed Gaussian impulse noise. Some extensions have been made to deal with mixed Rayleigh impulse noise and mixed Poisson impulse noise in [2]. In this paper, we consider $\ell_0$-norm data fidelity and show that it is particularly suitable for reconstructing images corrupted with additive/multiplicative [2] impulse noise.

**Regularization Models:** Several regularization models have been studied in the literature (see Table 2). The Tikhonov-like regularization [1] function $\Omega_{\text{tik}}$ is quadratic and smooth, therefore it is relatively inexpensive to minimize with first-order smooth optimization methods. However, since this method tends to overly smooth images, it often erodes strong edges and texture details. To address this issue, the total variation (TV) regularizer was proposed by Rudin, Osher and Fatemi in [45] for image denoising. Several other variants of TV have been extensively studied. The original TV norm $\Omega_{\text{tv}_2}$ in [45] is isotropic, while an anisotropic variation $\Omega_{\text{tv}_1}$ is also used. From a numerical point of view, $\Omega_{\text{tv}_2}$ and $\Omega_{\text{tv}_1}$ cannot be directly minimized since they are not differentiable. A popular method is to use their smooth approximation $\Omega_{\text{stv}}$ and $\Omega_{\text{hub}}$ (see [44] for details). Very recently, the Potts model $\Omega_{\text{pot}}$ [7], [27], [40], which is based on the $\ell_0$-norm, has received much attention. It has been shown to be particularly effective for image smoothing [54] and motion deblurring [55].

**Optimization Algorithms:** The optimization problems involved in TV-based image restoration are usually difficult due to the non-differentiability of the TV norm and the high dimensionality of the image data. In the past several decades, a plethora of approaches have been proposed, which include PDE methods based on the Euler-Lagrange equation [45], the interior-point method [16], the semi-smooth Newton method [43], the second-order cone optimization method [29], the splitting Bregman method [30], [67], the fixed-point iterative method [19], Nesterov's first-order optimal method [5],

1. In practice, one does not need to compute and store the matrices $\boldsymbol{\nabla}_x$ and $\boldsymbol{\nabla}_y$ explicitly. Since the adjoint of the gradient operator $\boldsymbol{\nabla}$ is the negative divergence operator $-\text{div}$, i.e., $\langle \mathbf{r},\boldsymbol{\nabla}_x\mathbf{u}\rangle = \langle -\text{div}_x\mathbf{r},\mathbf{u}\rangle, \langle \mathbf{s},\boldsymbol{\nabla}_y\mathbf{u}\rangle = \langle -\text{div}_y\mathbf{s},\mathbf{u}\rangle$ for any $\mathbf{r},\mathbf{s} \in \mathbb{R}^n$, the inner product between vectors can be evaluated efficiently. Fore more details on the computation of $\boldsymbol{\nabla}$ and div operators, please refer to [4], [12], [49].

2. The impulse noise has a discrete nature (corrupted or uncorrupted), thus it can be viewed as additive noise or multiplicative noise.

[42], and alternating direction methods [18], [48], [51]. Among these methods, some solve the TV problem in its primal form [48], while others consider its dual or primal-dual forms [16], [21]. In this paper, we handle the TV problem with $\ell_0$-norm data fidelity using a primal-dual formulation, where the resulting equality constrained optimization is solved using proximal Alternating Direction Method of Multipliers (PADMM). It is worthwhile to note that the Penalty Decomposition Algorithm (PDA) in [37] can also solve our problem, however, it lacks numerical stability. This motivates us to design a new $\ell_0$-norm optimization algorithm in this paper.

### 1.2 Contributions and Organization

The main contributions of this paper are two-fold. **(1)** $\ell_0$-norm data fidelity is proposed to address the TV-based image restoration problem[3]. Compared with existing models, our model is particularly suitable for image restoration in the presence of impulse noise. **(2)** To deal with the resulting NP-hard [4] $\ell_0$ norm optimization, we propose a proximal ADMM to solve an equivalent MPEC form of the problem. A preliminary version of this paper appeared in [61].

The rest of the paper is organized as follows. Section 2 presents the motivation and formulation of the problem for impulse noise removal. Section 3 presents the equivalent MPEC problem and our proximal ADMM solution. Section 4 discusses the connection between our method and prior work. Section 5 provides extensive and comparative results in favor of our $\ell_0$TV method. Finally, Section 6 concludes the paper.

## 2 Motivation and Formulations

### 2.1 Motivation

This work focuses on image restoration in the presence of impulse noise, which is very common in data acquisition and transmission due to faulty sensors or analog-to-digital converter errors, etc. Moreover, scratches in photos and video sequences can be also viewed as a special type of impulse noise. However, removing this kind of noise is not easy, since corrupted pixels are randomly distributed in the image and the intensities at corrupted pixels are usually indistinguishable from those of their neighbors. There are two main types of impulse noise in the literature [21], [33]: random-valued and salt-and-pepper impulse noise. Let $[u_{\min}, u_{\max}]$ be the dynamic range of an image, where $u_{\min} = 0$ and $u_{\max} = 1$ in this paper. We also denote the original and corrupted intensity values at position $i$ as $\mathbf{u}_i$ and $\mathcal{T}(\mathbf{u}_i)$, respectively.

**Random-valued impulse noise**: A certain percentage of pixels are altered to take on a uniform random number $d_i \in [u_{\min}, u_{\max}]$:

$$\mathcal{T}(\mathbf{u}_i) = \begin{cases} d_i, & \text{with probability } r_{rv}; \\ (\mathbf{K}\mathbf{u})_i, & \text{with probability } 1 - r_{rv}. \end{cases} \tag{3}$$

**Salt-and-pepper impulse noise**: A certain percentage of pixels are altered to be either $u_{\min}$ or $u_{\max}$:

$$\mathcal{T}(\mathbf{u}_i) = \begin{cases} u_{\min}, & \text{with probability } r_{sp}/2; \\ u_{\max}, & \text{with probability } r_{sp}/2; \\ (\mathbf{K}\mathbf{u})_i, & \text{with probability } 1 - r_{sp}. \end{cases} \tag{4}$$

The above definition means that impulse noise corrupts a portion of pixels in the image while keeping other pixels unaffected. Expectation maximization could be used to find the MAP estimate of $\mathbf{u}$ by maximizing the conditional posterior probability $p(\mathbf{u}|\mathcal{T}(\mathbf{u}))$, the probability that $\mathbf{u}$ occurs when $\mathcal{T}(\mathbf{u})$ is observed. By the Bayes' theorem, we have that

$$p(\mathbf{u}|\mathcal{T}(\mathbf{u})) = p(\mathbf{u}) \cdot p(\mathcal{T}(\mathbf{u})|\mathbf{u}) \;/\; p(\mathcal{T}(\mathbf{u})).$$

Taking the negative logarithm of the above equation, the estimate is a solution of the following minimization problem:

$$\max_{\mathbf{u}} \; \log p(\mathcal{T}(\mathbf{u})|\mathbf{u}) + \log p(\mathbf{u}). \tag{5}$$

We now focus on the two terms in (5). (i) The expression $p(\mathcal{T}(\mathbf{u})|\mathbf{u})$ can be viewed as a fidelity term measuring the discrepancy between the estimate $\mathbf{u}$ and the noisy image $\mathcal{T}(\mathbf{u})$. The choice of the likelihood $p(\mathcal{T}(\mathbf{u})|\mathbf{u})$ depends upon the property of noise. From the definition of impulse noise given above, we have that

$$p(\mathcal{T}(\mathbf{u})|\mathbf{u}) \quad = \quad 1 - r = 1 - \|\mathcal{T}(\mathbf{u}) - \mathbf{b}\|_0/n,$$

where $r$ is the noise density level as defined in (3) and (4) and $\|\cdot\|_0$ counts the number of non-zero elements in a vector. (ii) The term $p(\mathbf{u})$ in (5) is used to regularize a solution that has a low probability. We use a prior which has the Gibbs form: $p(\mathbf{u}) = \frac{1}{\vartheta}\exp(-E(\mathbf{u}))$ with $E(\mathbf{u}) = \sigma \cdot \Omega_{\text{tv}}(\boldsymbol{\nabla}_x\mathbf{u}, \boldsymbol{\nabla}_y\mathbf{u})$. Here, $E(\mathbf{u})$ is the TV prior energy functional, $\vartheta$ is a normalization factor such that the TV prior is a probability, and $\sigma$ is the free parameter of the Gibbs measure. Replacing $p(\mathcal{T}(\mathbf{u})|\mathbf{u})$ and $p(\mathbf{u})$ into (5) and ignoring a constant, we obtain the following $\ell_0TV$ model:

$$\min_{\mathbf{u}} \; \|\mathbf{K}\mathbf{u} - \mathbf{b}\|_0 + \lambda \sum_{i=1}^{n} \left[ |(\boldsymbol{\nabla}_x\mathbf{u})_i|^p + |(\boldsymbol{\nabla}_y\mathbf{u})_i|^p \right]^{1/p},$$

where $\lambda$ is a positive number related to $n$, $\sigma$ and $r$. The parameter $p$ can be 1 (anisotropic TV) or 2 (isotropic TV), and $(\boldsymbol{\nabla}_x\mathbf{u})_i$ and $(\boldsymbol{\nabla}_y\mathbf{u})_i$ denote the $i$th

3. We are also aware of Ref. [17] where $\ell_0$-norm data fidelity is considered. However, their interpretation from the MAP viewpoint is not correct.

4. The $\ell_0$ norm problem is known to be NP-hard [41], since it is equivalent to NP-complete subset selection problems.

component of the vectors $\boldsymbol{\nabla}_x\mathbf{u}$ and $\boldsymbol{\nabla}_y\mathbf{u}$, respectively. For convenience, we define $\forall \mathbf{x} \in \mathbb{R}^{2n}$:

$$\|\mathbf{x}\|_{p,1} \triangleq \sum_{i=1}^{n}(|\mathbf{x}_i|^p + |\mathbf{x}_{n+i}|^p)^{\frac{1}{p}};\ \boldsymbol{\nabla} \triangleq \begin{bmatrix}\boldsymbol{\nabla}_x \\ \boldsymbol{\nabla}_y\end{bmatrix} \in \mathbb{R}^{2n\times n}.$$

In order to make use of more prior information, we consider the following box-constrained model:

$$\min_{\mathbf{0}\le\mathbf{u}\le\mathbf{1}}\ \|\mathbf{o}\odot(\mathbf{K}\mathbf{u}-\mathbf{b})\|_0 + \lambda\|\boldsymbol{\nabla}\mathbf{u}\|_{p,1}, \tag{6}$$

where $\mathbf{o} \in \{0,1\}^n$ is specified by the user. When $\mathbf{o}_i$ is 0, it indicates the pixel in position $i$ is an outlier, while when $\mathbf{o}_i$ is 1, it indicates the pixel in position $i$ is a potential outlier. For example, in our experiments, we set $\mathbf{o} = \mathbf{1}$ for the random-valued impulse noise and $\mathbf{o}_i = \begin{cases}0, & \mathbf{b}_i = u_{\min} \text{ or } u_{\max}\\ 1, & \text{otherwise}\end{cases}$ for the salt-and-pepper impulse noise. In what follows, we focus on optimizing the general formulation in (6).

### 2.2 Equivalent MPEC Reformulations

In this section, we reformulate the problem in (6) as an equivalent MPEC from a primal-dual viewpoint. First, we provide the variational characterization of the $\ell_0$-norm using the following lemma.

**Lemma 1.** *For any given $\mathbf{w} \in \mathbb{R}^n$, it holds that*

$$\|\mathbf{w}\|_0 = \min_{\mathbf{0}\le\mathbf{v}\le\mathbf{1}}\ \langle\mathbf{1}, \mathbf{1}-\mathbf{v}\rangle,\ s.t.\ \mathbf{v}\odot|\mathbf{w}| = \mathbf{0}, \tag{7}$$

*and $\mathbf{v}^* = \mathbf{1} - \mathrm{sign}(|\mathbf{w}|)$ is the unique optimal solution of the problem in (7). Here, the standard signum function sign is applied componentwise, and* $\mathrm{sign}(0) = 0$.

*Proof.* The total number of zero elements in $\mathbf{w}$ can be computed as $n - \|\mathbf{w}\|_0 = \max_{\mathbf{v}\in\{0,1\}}\ \sum_{i=1}^{n}\mathbf{v}_i,\ s.t.\ \mathbf{v} \in \Phi$, where $\Phi \triangleq \{\mathbf{v} \mid \mathbf{v}_i\cdot|\mathbf{w}_i| = 0,\ \forall i \in [n]\}$. Note that when $\mathbf{w}_i = 0$, $\mathbf{v}_i = 1$ will be achieved by maximization, when $\mathbf{w}_i \neq 0$, $\mathbf{v}_i = 0$ will be enforced by the constraint. Thus, $\mathbf{v}_i^* = 1 - \mathrm{sign}(|\mathbf{w}_i|)$. Since the objective function is linear, maximization is always achieved at the boundaries of the feasible solution space. Thus, the constraint of $\mathbf{v}_i \in \{0,1\}$ can be relaxed to $0 \le \mathbf{v}_i \le 1$, we have: $\|\mathbf{w}\|_0 = n - \max_{\mathbf{0}\le\mathbf{v}\le\mathbf{1},\ \mathbf{v}\in\Phi}\ \sum_{i=1}^{n}\mathbf{v}_i = \min_{\mathbf{0}\le\mathbf{v}\le\mathbf{1},\ \mathbf{v}\in\Phi}\ \langle\mathbf{1}, \mathbf{1}-\mathbf{v}\rangle$. □

The result of Lemma 1 implies that the $\ell_0$-norm minimization problem in (6) is equivalent to

$$\begin{aligned}\min_{\mathbf{0}\le\mathbf{u},\mathbf{v}\le\mathbf{1}}\ &\langle\mathbf{1}, \mathbf{1}-\mathbf{v}\rangle + \lambda\|\boldsymbol{\nabla}\mathbf{u}\|_{p,1}\\ \text{s.t.}\quad &\mathbf{v}\odot|\mathbf{o}\odot(\mathbf{K}\mathbf{u}-\mathbf{b})| = \mathbf{0}.\end{aligned} \tag{8}$$

If $\mathbf{u}^*$ is a global optimal solution of (6), then $(\mathbf{u}^*, \mathbf{1} - \mathrm{sign}(|\mathbf{K}\mathbf{u}^* - \mathbf{b}|))$ is globally optimal to (8). Conversely, if $(\mathbf{u}^*, \mathbf{1} - \mathrm{sign}(|\mathbf{K}\mathbf{u}^* - \mathbf{b}|))$ is a global optimal solution of (8), then $\mathbf{u}^*$ is globally optimal to (6).

Although the MPEC problem in (8) is obtained by increasing the dimension of the original $\ell_0$-norm problem in (6), this does not lead to additional local optimal solutions. Moreover, compared with (6), (8) is a non-smooth non-convex minimization problem and its non-convexity is only caused by the complementarity constraint $\mathbf{v}\odot|\mathbf{o}\odot(\mathbf{K}\mathbf{u}-\mathbf{b})| = \mathbf{0}$.

Such a variational characterization of the $\ell_0$-norm is proposed in [23], [25], [32], but it is not used to develop any optimization algorithms for $\ell_0$-norm problems. We argue that, from a practical perspective, improved solutions to (6) can be obtained by reformulating the $\ell_0$-norm in terms of complementarity constraints [38], [61], [62], [63], [64], [65]. In the following section, we will develop an algorithm to solve (8) based on proximal ADMM and show that such a "lifting" technique can achieve a desirable solution of the original $\ell_0$-norm optimization problem.

## 3 Proposed Optimization Algorithm

This section is devoted to the solution of (8). This problem is rather difficult to solve, because it is neither convex nor smooth. Our solution is based on the proximal ADM method, which iteratively updates the primal and dual variables of the augmented Lagrangian function of (8).

First, we introduce two auxiliary vectors $\mathbf{x} \in \mathbb{R}^{2n}$ and $\mathbf{y} \in \mathbb{R}^n$ to reformulate (8) as:

$$\begin{aligned}\min_{\mathbf{0}\le\mathbf{u},\mathbf{v}\le\mathbf{1},\ \mathbf{x},\ \mathbf{y}}\ &\langle\mathbf{1}, \mathbf{1}-\mathbf{v}\rangle + \lambda\|\mathbf{x}\|_{p,1}\\ \text{s.t.}\quad &\boldsymbol{\nabla}\mathbf{u} = \mathbf{x},\ \mathbf{K}\mathbf{u}-\mathbf{b} = \mathbf{y},\ \mathbf{v}\odot\mathbf{o}\odot|\mathbf{y}| = \mathbf{0}.\end{aligned} \tag{9}$$

Let $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{2n} \times \mathbb{R}^n \times \mathbb{R}^{2n} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be the augmented Lagrangian function of (9).

$$\begin{aligned}\mathcal{L}(\mathbf{u},\mathbf{v},\mathbf{x},\mathbf{y},\boldsymbol{\xi},\boldsymbol{\zeta},\boldsymbol{\pi}) := \langle\mathbf{1}, \mathbf{1}-\mathbf{v}\rangle + \lambda\|\mathbf{x}\|_{p,1} +\\ \langle\boldsymbol{\nabla}\mathbf{u}-\mathbf{x}, \boldsymbol{\xi}\rangle + \tfrac{\beta}{2}\|\boldsymbol{\nabla}\mathbf{u}-\mathbf{x}\|^2 + \langle\mathbf{K}\mathbf{u}-\mathbf{b}-\mathbf{y}, \boldsymbol{\zeta}\rangle +\\ \tfrac{\beta}{2}\|\mathbf{K}\mathbf{u}-\mathbf{b}-\mathbf{y}\|^2 + \langle\mathbf{v}\odot\mathbf{o}\odot|\mathbf{y}|, \boldsymbol{\pi}\rangle + \tfrac{\beta}{2}\|\mathbf{v}\odot\mathbf{o}\odot|\mathbf{y}|\|^2,\end{aligned}$$

where $\boldsymbol{\xi}$, $\boldsymbol{\zeta}$ and $\boldsymbol{\pi}$ are the Lagrange multipliers associated with the constraints $\boldsymbol{\nabla}\mathbf{u} = \mathbf{x}$, $\mathbf{K}\mathbf{u} - \mathbf{b} = \mathbf{y}$ and $\mathbf{v}\odot\mathbf{o}\odot|\mathbf{y}| = 0$, respectively, and $\beta > 0$ is the penalty parameter. The detailed iteration steps of the proximal ADM for (9) are described in Algorithm 1. In simple terms, ADM updates are performed by optimizing for a set of primal variables at a time, while keeping all other primal and dual variables fixed. The dual variables are updated by gradient ascent on the resulting dual problem.

Next, we focus our attention on the solutions of the subproblems in (10) and (11) arising in Algorithm 1. We will show that the computation required in each iteration of Algorithm 1 is insignificant.

(i) $(\mathbf{u}, \mathbf{v})$-subproblem. Proximal ADM introduces a convex proximal term to the objective. The specific form of $\mathbf{D}$ is chosen to expedite the computation of the closed form solution. The introduction of $\mu$ is to guarantee strongly convexity of the subproblems.

**Algorithm 1** ($\ell_0TV$**-ADMM) A Proximal ADMM for Solving the Biconvex MPEC Problem (8)**

**(S.0)** Choose a starting point $(\mathbf{u}^0, \mathbf{v}^0, \mathbf{x}^0, \mathbf{y}^0, \boldsymbol{\xi}^0, \boldsymbol{\zeta}^0)$. Set $k = 0$. Select step size $\gamma \in (0, 2)$, $\mu > 0$, $\beta = 1$, and $L = \mu + \beta\|\boldsymbol{\nabla}\|^2 + \beta\|\mathbf{K}\|^2$.

**(S.1)** Solve the following minimization problems with $\mathbf{D} := L\mathbf{I} - (\beta\boldsymbol{\nabla}^T\boldsymbol{\nabla} + \beta\mathbf{K}^T\mathbf{K})$ and $\mathbf{E} := \mu\mathbf{I}$:

$$\begin{bmatrix} \mathbf{u}^{k+1} \\ \mathbf{v}^{k+1} \end{bmatrix} = \underset{\mathbf{0}\le\mathbf{u},\mathbf{v}\le\mathbf{1}}{\arg\min}\, \mathcal{L}(\mathbf{u}, \mathbf{v}, \mathbf{x}^k, \mathbf{y}^k, \boldsymbol{\xi}^k, \boldsymbol{\zeta}^k, \boldsymbol{\pi}^k) + \tfrac{1}{2}\|\mathbf{u} - \mathbf{u}^k\|^2_{\mathbf{D}} + \tfrac{1}{2}\|\mathbf{v} - \mathbf{v}^k\|^2_{\mathbf{E}} \tag{10}$$

$$\begin{bmatrix} \mathbf{x}^{k+1} \\ \mathbf{y}^{k+1} \end{bmatrix} = \underset{\mathbf{x},\mathbf{y}}{\arg\min}\, \mathcal{L}(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}, \mathbf{x}, \mathbf{y}, \boldsymbol{\xi}^k, \boldsymbol{\zeta}^k, \boldsymbol{\pi}^k) \tag{11}$$

**(S.2)** Update the Lagrange multipliers:

$$\boldsymbol{\xi}^{k+1} = \boldsymbol{\xi}^k + \gamma\beta(\boldsymbol{\nabla}\mathbf{u}^k - \mathbf{x}^k), \tag{12}$$

$$\boldsymbol{\zeta}^{k+1} = \boldsymbol{\zeta}^k + \gamma\beta(\mathbf{K}\mathbf{u}^k - \mathbf{b} - \mathbf{y}^k), \tag{13}$$

$$\boldsymbol{\pi}^{k+1} = \boldsymbol{\pi}^k + \gamma\beta(\mathbf{o} \odot \mathbf{v}^k \odot |\mathbf{y}^k|). \tag{14}$$

**(S.3)** if ($k$ is a multiple of 30), then $\beta = \beta \times \sqrt{10}$

**(S.4)** Set $k := k + 1$ and then go to Step (S.1).

$\mathbf{u}$-subproblem in (10) reduces to the following minimization problem:

$$\mathbf{u}^{k+1} = \arg\min\nolimits_{\mathbf{0}\le\mathbf{u}\le\mathbf{1}}\ \tfrac{\beta}{2}\|\boldsymbol{\nabla}\mathbf{u} - \mathbf{x}^k + \boldsymbol{\xi}^k/\beta\|^2 + \tfrac{\beta}{2}\|\mathbf{K}\mathbf{u} - \mathbf{b} - \mathbf{y}^k + \boldsymbol{\zeta}^k/\beta\|^2 + \tfrac{1}{2}\|\mathbf{u} - \mathbf{u}^k\|^2_{\mathbf{D}}. \tag{15}$$

After an elementary calculation, subproblem (15) can be simplified as

$$\mathbf{u}^{k+1} = \arg\min\nolimits_{\mathbf{0}\le\mathbf{u}\le\mathbf{1}}\ \tfrac{1}{2}\|\mathbf{u} - (\mathbf{u}^k - \mathbf{g}^k/L)\|^2$$

with $\mathbf{g}^k = \boldsymbol{\nabla}^T\xi^k + \mathbf{K}^T\boldsymbol{\zeta}^k + \beta\boldsymbol{\nabla}^T(\mathbf{x}^k - \boldsymbol{\nabla}\mathbf{u}^k) + \beta\mathbf{K}^T(\mathbf{b} + \mathbf{y}^k - \mathbf{K}\mathbf{u}^k)$. Then, the solution $\mathbf{u}^k$ of (10) has the following closed form expression:

$$\mathbf{u}^{k+1} = \min(\mathbf{1}, \max(\mathbf{0}, \mathbf{u}^k - \mathbf{g}^k/L)).$$

Here the parameter $L$ depends on the spectral norm of the linear matrices $\boldsymbol{\nabla}$ and $\mathbf{K}$. Using the definition of $\boldsymbol{\nabla}$ and the classical finite differences that $\|\boldsymbol{\nabla}_y\| \le 2$ and $\|\boldsymbol{\nabla}_y\| \le 2$ (see [4], [12], [68]), the spectral norm of $\boldsymbol{\nabla}$ can be computed by: $\|\boldsymbol{\nabla}\| = \|\begin{pmatrix} \boldsymbol{\nabla}_x \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \boldsymbol{\nabla}_y \end{pmatrix}\| \le \|\begin{pmatrix} \boldsymbol{\nabla}_x \\ \mathbf{0} \end{pmatrix}\| + \|\begin{pmatrix} \mathbf{0} \\ \boldsymbol{\nabla}_y \end{pmatrix}\| = \|\boldsymbol{\nabla}_x\| + \|\boldsymbol{\nabla}_y\| \le 4$.

$\mathbf{v}$-subproblem in (10) reduces to the following minimization problem:

$$\mathbf{v}^{k+1} = \arg\min\nolimits_{\mathbf{0}\le\mathbf{v}\le\mathbf{1}}\ \tfrac{1}{2}\sum_{i=1}^n \mathbf{s}_i^k\mathbf{v}_i^2 + \langle\mathbf{v}, \mathbf{c}^k\rangle,$$

where $\mathbf{c}^k = o\odot\boldsymbol{\pi}^k\odot|\mathbf{y}^k| - \mathbf{1} - \mu\mathbf{v}^k$, $\mathbf{s}^k = \beta\mathbf{o}\odot\mathbf{y}^k\odot\mathbf{y}^k + \mu$. Therefore, the solution $\mathbf{v}^k$ can be computed as:

$$\mathbf{v}^{k+1} = \min(\mathbf{1}, \max(\mathbf{0}, -\tfrac{\mathbf{c}^k}{\mathbf{s}^k})).$$

(iii) $(\mathbf{x}, \mathbf{y})$-subproblem. Variable $\mathbf{x}$ in (11) is updated by solving the following problem:

$$\mathbf{x}^{k+1} = \arg\min\nolimits_{\mathbf{x}\in\mathbb{R}^{2n}}\ \tfrac{\beta}{2}\|\mathbf{x} - \mathbf{h}^k\|^2 + \lambda\|\mathbf{x}\|_{p,1},$$

where $\mathbf{h}^k := \boldsymbol{\nabla}\mathbf{u}^{k+1} + \boldsymbol{\xi}^k/\beta$. It is not difficult to check that for $p = 1$,

$$\mathbf{x}^{k+1} = \mathrm{sign}\big(\mathbf{h}^k\big) \odot \max\big(|\mathbf{h}^k| - \lambda/\beta, 0\big),$$

and when $p = 2$,

$$\begin{bmatrix} \mathbf{x}_i^{k+1} \\ \mathbf{x}_{i+n}^{k+1} \end{bmatrix} = \big(\max(0, 1 - \tfrac{\lambda/\beta}{\|(\mathbf{h}_i^k; \mathbf{h}_{i+n}^k)\|})\big) \begin{bmatrix} \mathbf{h}_i^k \\ \mathbf{h}_{i+n}^k \end{bmatrix}$$

Variable $\mathbf{y}$ in (11) is updated by solving the following problem:

$$\mathbf{y}^{k+1} = \arg\min\nolimits_{\mathbf{y}}\ \tfrac{\beta}{2}\|\mathbf{y} - \mathbf{q}^k\|^2 + \tfrac{\beta}{2}\|\mathbf{w}^k \odot |\mathbf{y}| + \boldsymbol{\pi}^k/\beta\|^2,$$

where $\mathbf{q}^k = \mathbf{K}\mathbf{u}^{k+1} - \mathbf{b} + \boldsymbol{\zeta}^k/\beta$ and $\mathbf{w}^k = \mathbf{o} \odot \mathbf{v}^{k+1}$. A simple computation yields that the solution $\mathbf{y}^k$ can be computed in closed form as:

$$\mathbf{y}^{k+1} = \mathrm{sign}(\mathbf{q}^k) \odot \max\big(0, \tfrac{|\mathbf{q}^k| - \boldsymbol{\pi}^k\odot\mathbf{w}^k/\beta}{1 + \mathbf{v}^k\odot\mathbf{w}^k}\big),$$

Proximal ADM has excellent convergence in practice. The global convergence of ADM for convex problems was given by He and Yuan in [18], [31] under the variation inequality framework. However, since our optimization problem in (8) is non-convex, the convergence analysis for ADM needs additional conditions. By imposing some mild conditions, Wen et al. [50] managed to show that the sequence generated by ADM converges to a KKT point. Along a similar line, we establish the convergence property of proximal ADM. Specifically, we have the following convergence result.

**Theorem 1.** ***Convergence of Algorithm 1.*** *Let $X \triangleq (\mathbf{u}, \mathbf{v}, \mathbf{x}, \mathbf{y})$, $Y \triangleq (\boldsymbol{\xi}, \boldsymbol{\zeta}, \boldsymbol{\pi})$ and $\{X^k, Y^k\}_{k=1}^\infty$ be the sequence generated by Algorithm 1. Assume that $\{Y^k\}_{k=1}^\infty$ is bounded and satisfies $\sum_{k=0}^\infty \|Y^{k+1} - Y^k\|_F^2 < \infty$. Then any accumulation point of sequence satisfies the KKT conditions of (9).*

*Proof.* Please refer to **Appendix A**. □

**Remark 1.** The condition $\sum_{k=0}^\infty \|Y^{k+1} - Y^k\|_F^2 < \infty$ holds when the multiplier does not change in two consecutive iterations. By the boundedness of the penalty parameter $\beta$ and Eqs (12-14), this condition also indicates that the equality constraints in (9) are satisfied. This assumption can be checked by measuring the violation of the equality constraints. Theorem 1 indicates that when the equality constraint holds, PADMM converges to a KKT point. Though not satisfactory, it provides some assurance on the convergence of Algorithm 1.

**Remark 2.** Two reasons explain the good performance of our method. (i) It targets a solution to the original problem in (6). (ii) It has monotone and self-penalized

properties owing to the complimentarity constraints brought on by the MPEC. Our method directly handles the complimentary constraints in (9): $\mathbf{v} \odot \mathbf{o} \odot |\mathbf{y}| = \mathbf{0}$ with $\mathbf{v} \geq \mathbf{0}$. These constraints are the only sources of non-convexity for the optimization problem and they characterize the optimality of the KKT solution of (6). These special properties of MPEC distinguish it from general nonlinear optimization [62], [63], [64], [65]. We penalize the complimentary error of $\mathbf{v} \odot \mathbf{o} \odot |\mathbf{y}|$ (which is always non-negative) and ensure that the error is decreasing in every iteration.

## 4 Connection with Existing Work

In this section, we discuss the connection between the proposed method $\ell_0TV$-PADM and prior work.

### 4.1 Sparse Plus Low-Rank Matrix Decomposition

Sparse plus low-rank matrix decomposition [33], [52] is becoming a powerful tool that effectively corrects large errors in structured data in the last decade. It aims at decomposing a given corrupted image $\mathbf{B}$ (which is of matrix form) into its sparse component ($\mathbf{S}$) and low-rank component ($\mathbf{L}$) by solving: $\min_{\mathbf{B},\mathbf{L}} \ \|\mathbf{S}\|_0 + \lambda\, rank(\mathbf{L}),\ s.t.\ \mathbf{B} = \mathbf{L} + \mathbf{S}$. Here the sparse component represents the foreground of an image which can be treated as outliers or impulse noise, while the low-rank component corresponds to the background, which is highly correlated. This is equivalent to the following optimization problem:

$$\min_{\mathbf{L}} \ \|\mathbf{B} - \mathbf{L}\|_0 + \lambda\, rank(\mathbf{L}),$$

which is also based on $\ell_0$-norm data fidelity. While they consider the low-rank prior in their objective function, we consider the Total Variation (TV) prior in ours.

### 4.2 Convex Optimization Method $\ell_1TV$

The goal of image restoration in the presence of impulse noise has been pursued by a number of authors (see, e.g., [21], [58]) using $\ell_1TV$, which can be formulated as follows:

$$\min_{\mathbf{0}\leq\mathbf{u}\leq\mathbf{1}} \ \|\mathbf{K}\mathbf{u} - \mathbf{b}\|_1 + \lambda\, \|\boldsymbol{\nabla}\mathbf{u}\|_{p,1}. \tag{16}$$

It is generally believed that $\ell_1TV$ is able to remove the impulse noise properly. This is because $\ell_1$-norm provides the tightest convex relaxation for the $\ell_0$-norm over the unit ball in the sense of $\ell_\infty$-norm. It is shown in [10] that the problem of minimizing $\|\mathbf{K}\mathbf{u} - \mathbf{b}\|_1$ is equivalent to $\|\mathbf{K}\mathbf{u} - \mathbf{b}\|_0$ with high probability under the assumptions that (i) $\mathbf{K}\mathbf{u} - \mathbf{b}$ is sparse at the optimal solution $\mathbf{u}^*$ and (ii) $\mathbf{K}$ is a random Gaussian matrix and sufficiently "incoherent" (i.e., number of rows in $\mathbf{K}$ is greater than its number of columns). However, these two assumptions required in [10] do not necessarily hold true for our $\ell_0TV$ optimization problem. Specifically, when the noise level of the impulse noise is high, $\mathbf{K}\mathbf{u} - \mathbf{b}$ may not be sparse at the optimal solution $\mathbf{u}^*$. Moreover, the matrix $\mathbf{K}$ is a square identity or ill-conditioned matrix. Generally, $\ell_1TV$ will only lead to a sub-optimal solution.

### 4.3 Adaptive Outlier Pursuit Algorithm

Very recently, Yan [57] proposed the following new model for image restoration in the presence of impulse noise and mixed Gaussian impulse noise:

$$\min_{\mathbf{u},\mathbf{z}} \ \chi\|\mathbf{K}\mathbf{u} - \mathbf{b} - \mathbf{z}\|_2^2 + \|\boldsymbol{\nabla}\mathbf{u}\|_{p,1},\ s.t.\ \|\mathbf{z}\|_0 \leq k, \tag{17}$$

where $\chi > 0$ is the regularization parameter. They further reformulate the problem above into $\min_{\mathbf{u},\,\mathbf{v}} \ \|\mathbf{v}\odot(\mathbf{K}\mathbf{u} - \mathbf{b})\|_2^2 + \lambda\, \|\boldsymbol{\nabla}\mathbf{u}\|_{p,1},\ s.t.\ \mathbf{0} \leq \mathbf{v} \leq \mathbf{1},\ \langle \mathbf{v}, \mathbf{1}\rangle \leq n - k$ and then solve this problem using an Adaptive Outlier Pursuit(AOP) algorithm. The AOP algorithm is actually an alternating minimization method, which separates the minimization problem over $\mathbf{u}$ and $\mathbf{v}$ into two steps. By iteratively restoring the images and updating the set of damaged pixels, it is shown that AOP algorithm outperforms existing state-of-the-art methods for impulse noise denoising, by a large margin.

Despite the merits of the AOP algorithm, we must point out that it incurs three drawbacks, which are unappealing in practice. First, the formulation in (17) is only suitable for mixed Gaussian impulse noise, i.e. it produces a sub-optimal solution when the observed image is corrupted by pure impulse noise. (ii) Secondly, AOP is a multiple-stage algorithm. Since the minimization sub-problem over $\mathbf{u}$[5] needs to be solved exactly in each stage, the algorithm may suffer from slow convergence. (iii) As a by-product of (i), AOP inevitably introduces an additional parameter (that specifies the Gaussian noise level), which is not necessarily readily available in practical impulse denoising problems.

In contrast, our proposed $\ell_0$TV method is free from these problems. Specifically, (i) as have been analyzed in Section 2, i.e. our $\ell_0$-norm model is optimal for impulse noise removal. Thus, our method is expected to produce higher quality image restorations, as seen in our results. (ii) Secondly, we have integrated $\ell_0$-norm minimization into a unified proximal ADM optimization framework, it is thus expected to be faster than the multiple stage approach of AOP. (iii) Lastly, while the optimization problem in (17) contains two parameters, our model only contains one single parameter.

### 4.4 Other $\ell_0$-Norm Optimization Techniques

Actually, the optimization technique for the $\ell_0$-norm regularization problem is the key to removing impulse

5. It actually reduces to the $\ell_2TV$ optimization problem.

noise. However, existing solutions are not appealing. The $\ell_0$-norm problem can be reformulated as a 0-1 mixed integer programming [6]problem which can be solved by a tailored branch-and-bound algorithm but it involves high computational complexity. The simple projection methods are inapplicable to our model since they assume the objective function is smooth. Similar to the $\ell_1$ relaxation, the convex methods such as $k$-support norm relaxation [39], $k$-largest norm relaxation [60], QCQP and SDP relaxations [13] only provide loose approximation of the original problem. The non-convex methods such as Schatten $\ell_p$ norm [26], [35], re-weighted $\ell_1$ norm [11], $\ell_{1\text{-}2}$ norm DC (difference of convex) approximation [59], the Smoothly Clipped Absolute Deviation (SCAD) penalty method [66], the Minimax Concave Plus (MCP) penalty method [24] only produce sub-optimal results since they give approximate solutions for the $\ell_0TV$ problem or incur high computational overhead. For example, the $\ell_p$ norm method may suffer two issues. First, it involves an additional hyper-parameter $p$ which may not be appealing in practice. Second, the $\ell_p$ regularized norm problem for general $p$ could be difficult to solve. This includes the iterative re-weighted least square method [36] and proximal point method. The former approximates $\|\mathbf{x}\|_p^p$ by $\sum_{i=1}^n(\mathbf{x}_i^2+\epsilon)^{p/2}$ with a small parameter $\epsilon$ and solves the resulting re-weighted least squares subproblem which reduces to a weighted $\ell_2TV$ problem. The latter needs to evaluate a relatively expensive proximal operator $\Pi(\mathbf{a}) = \min_{\mathbf{x}} \ \frac{1}{2}\|\mathbf{x}-\mathbf{a}\|_2^2 + \lambda\|\mathbf{x}\|_p^p$ in general, except that it has a closed form solution for some special values such as $p=\frac{1}{2}$ and $p=\frac{2}{3}$ [56].

Recently, Lu et al. propose a Penalty Decomposition Algorithm (PDA) for solving the $\ell_0$-norm optimization algorithm [37]. As has been remarked in [37], direct ADM on the $\ell_0$ norm problem can also be used for solving $\ell_0TV$ minimization simply by replacing the quadratic penalty functions in the PDA by augmented Lagrangian functions. Nevertheless, as observed in our preliminary experiments and theirs, the practical performance of direct ADM is worse than that of PDA.

Actually, in our experiments, we found PDA is unstable. The penalty function can reach very large values ($\geq 10^8$), and the solution can be degenerate when the minimization problem of the augmented Lagrangian function in each iteration is not exactly solved. This motivates us to design a new $\ell_0$-norm optimization algorithm in this paper. We consider a proximal ADM algorithm to the MPEC formulation of $\ell_0$-norm since it has a primal-dual interpretation. Extensive experiments have demonstrated that proximal ADM for solving the "lifting" MPEC formulation for $\ell_0TV$ produces better image restoration qualities.

## 5 Experimental Validation

In this section, we provide empirical validation for our proposed $\ell_0TV$-PADMM method by conducting extensive image denoising experiments and performing a thorough comparative analysis with the state-of-the-art.

In our experiments, we use 5 well-known test images of size $512 \times 512$. All code is implemented in MATLAB using a 3.20GHz CPU and 8GB RAM. Since past studies [9], [19] have shown that the isotropic TV model performs better than the anisotropic one, we choose $p=2$ as the order of the TV norm here. In our experiments, we apply the following algorithms:

**(i) BM3D** is an image denoising strategy based on an enhanced sparse representation in transform-domain. The enhancement of the sparsity is achieved by grouping similar 2D image blocks into 3D data arrays [22].

**(ii) MFM**, Median Filter Methods. We utilize adaptive median filtering to remove salt-and-pepper impulse noise and adaptive center-weighted median filtering to remove random-valued impulse noise.

**(iii)** $\ell_1TV$**-SBM**, the Split Bregman Method (SBM) of [30], which has been implemented in [28]. We use this convex optimization method as our baseline implementation.

**(iv) TSM**, the Two Stage Method [8], [14], [15]. The method first detects the damaged pixels by MFM and then solves the TV image inpainting problem.

**(v)** $\ell_pTV$**-ADMM (direct)**. We directly use ADMM (Alternating Direction Method of Multipliers) to solve the non-smooth non-convex $\ell_p$ problem with proximal operator being computed analytically. We only consider $p=\frac{1}{2}$ in our experiments [56].

**(vi)** $\ell_{02}TV$**-AOP**, the Adaptive Outlier Pursuit (AOP) method described in [57]. We use the implementation provided by the author. Here, we note that AOP iteratively calls the $\ell_1TV$-$SBM$ procedure, mentioned above.

**(vii)** $\ell_0TV$**-PDA**, the Penalty Decomposition Algorithm (PDA) [37] for solving the $\ell_0TV$ optimization problem in (6).

**(viii)** $\ell_0TV$**-PADMM**, the proximal ADMM described in Algorithm 1 for solving the $\ell_0TV$ optimization problem in (6). We set the relaxation parameter to 1.618 and the strongly convex parameter $\mu$ to 0.01. All MATLAB codes to reproduce the experiments of this paper are available online at the authors' research webpages.

### 5.1 Experiment Setup

For the denoising and deblurring test, we use the following strategies to generate artificial noisy images.

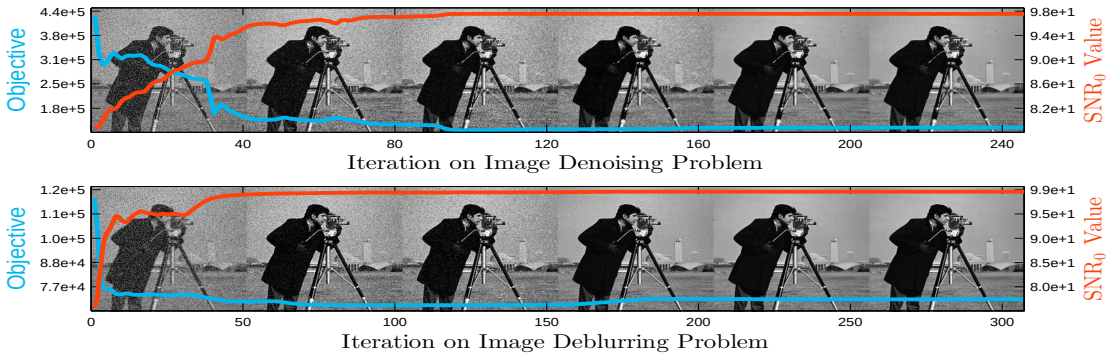



Figure 2: Asymptotic behavior for optimizing (6) to denoise and deblur the corrupted 'cameraman' image. We plot the value of the objective function (solid blue line) and the SNR value (dashed red line) against the number of optimization iterations. At specific iterations (i.e. 1, 10, 20, 40, 80, and 160), we also show the denoised and deblurred image. Clearly, the corrupting noise is being effectively removed throughout the optimization process.

**(a) Denoising problem**. We corrupt the original image by injecting random-value, salt-and-pepper noise, and mixed noise (half random-value and half salt-and-pepper) with different densities (10% to 90%) to the images.

**(b) Deblurring problem**. Although blurring kernel estimation has been pursued by many studies (e.g. [53]), here we assume that the blurring kernel is known beforehand. We blur the original images with a $9 \times 9$ Gaussian blurring kernel and add impulse noise with different densities (10% to 90%). We use the following MATLAB scripts to generate a blurring kernel of radius $r$ ($r$ is set to 7 in the experiments):

$$\begin{array}{c} \text{[x,y] = meshgrid (}-\text{r:r, }-\text{r:r),} \\ \text{K=double(x.\^{}2 + y.\^{}2 <= r.\^{}2), P=K/sum(K(:)).} \end{array} \tag{18}$$

We run all the previously mentioned algorithms on the generated noisy and blurry images. For $\ell_{02}TV$-AOP, we adapt the author's image denoising implementation to the image deblurring setting. Since both BM3D and Median Filter Methods (MFM) are not convenient to solve the deblurring problems, we do not test them in the deblurring problem. We terminate $\ell_0 TV$-PADMM whenever $\|\boldsymbol{\nabla}\mathbf{u}^k - \mathbf{x}^k\|_2 \leq \frac{1}{255}$ and $\|\mathbf{K}\mathbf{u}^k - \mathbf{b} - \mathbf{y}^k\|_2 \leq \frac{1}{255}$ and $\|\mathbf{o} \odot \mathbf{v}^k \odot |\mathbf{y}^k|\|_2 \leq \frac{1}{255}$. For $\ell_p TV$-PADMM, $\ell_0 TV$-PDA, and $\ell_0 TV$-PADMM, we use the same stopping criterion to terminate the optimization. For $\ell_1 TV$-SBM and $\ell_{02}TV$-AOP, we adopt the default stopping conditions provided by the authors. For the regularization parameter $\lambda$, we swept over $\{0.1, 0.6, 1.1, ..., 9.6\}$. For the regularization parameter $\chi$ in $\ell_{02}TV$-AOP, we swept over $\{10, 50, 100, 500, 1000, 5000, 10000, 50000\}$ and set $k$ to the number of corrupted pixels.

To evaluate these methods, we compute their Signal-to-Noise Ratios (SNRs). Since the corrupted pixels follow a Bernoulli-like distribution, it is generally hard to measure the data fidelity between the original images and the recovered images. Therefore, we consider three ways to measure SNR.

$$\begin{aligned} SNR_0(\boldsymbol{u}) &\triangleq \frac{n - \|\boldsymbol{u}^0 - \boldsymbol{u}\|_{0\text{-}\epsilon}}{n - \|\boldsymbol{u}^0 - \boldsymbol{u}^0\|_{0\text{-}\epsilon}} \times 100, \\ SNR_1(\boldsymbol{u}) &\triangleq 10 \log_{10} \frac{\|\boldsymbol{u}^0 - \bar{\boldsymbol{u}}\|_1}{\|\boldsymbol{u} - \bar{\boldsymbol{u}}\|_1}, \\ SNR_2(\boldsymbol{u}) &\triangleq 10 \log_{10} \frac{\|\boldsymbol{u}^0 - \bar{\boldsymbol{u}}\|_2^2}{\|\boldsymbol{u} - \bar{\boldsymbol{u}}\|_2^2}, \end{aligned}$$

where $\boldsymbol{u}^0$ is the original clean image and $\bar{\boldsymbol{u}}$ is the mean intensity value of $\boldsymbol{u}^0$, and $\|\cdot\|_{0\text{-}\epsilon}$ is the soft $\ell_0$-norm which counts the number of elements whose magnitude is greater than a threshold $\epsilon$. We adopt $\epsilon = \frac{20}{255}$ in our experiments.

### 5.2 Convergence of $\ell_0 TV$-PADMM

Here, we verify the convergence property of our $\ell_0 TV$-PADMM method on denoising and deblurring problems by considering the 'cameraman' image subject to 30% random-valued impulse noise. We set $\lambda = 8$ for this problem. We record the objective and SNR values for $\ell_0 TV$-PADMM at every iteration $k$ and plot these results in Figure 2.

We make two important observations from these results. **(i)**) The objective value (or the SNR value) does not necessarily decrease (or increase) monotonically, and we attribute this to the non-convexity of the optimization problem and the dynamic updates of the penalty factor in Algorithm 1. **(ii)** The objective and SNR values stabilize after the 120th iteration, which means that our algorithm has converged, and the increase of the SNR value is negligible after the 80th iteration. This implies that one may use a looser stopping criterion without sacrificing much restoration quality.

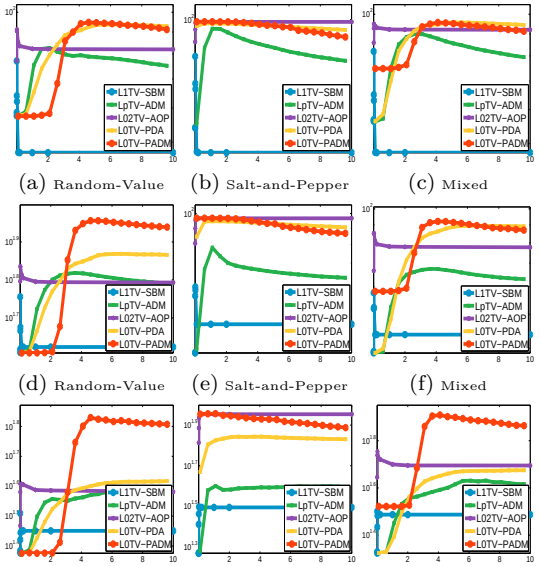
### 5.3 General Image Denoising Problems

In this subsection, we compare the performance of all 6 methods on general denoising problems. Table 3 shows image recovery results when random-value or salt-and-pepper or mixed impulse noise is added. Figure 3 shows

Table 3: General denoising problems. The results separated by '/' are $SNR_0$, $SNR_1$ and $SNR_2$, respectively. The $1^{st}$, $2^{nd}$, and $3^{rd}$ best results are colored with **red**, blue and green, respectively.

| Img. \ Alg. | BM3D | $\ell_1TV$-SBM | MFM | TSM | $\ell_{02}TV$-AOP | $\ell_pTV$-PADMM | $\ell_0TV$-PDA | $\ell_0TV$-PADM |
|---|---|---|---|---|---|---|---|---|
| **Random-Value Impulse Noise** | | | | | | | | |
| walkbridge+10% | 93/7.1/11.0 | 95/12.3/15.6 | 92/7.7/12.3 | 95/11.8/12.9 | 96/12.8/16.6 | 95/12.1/13.8 | **97/14.1/16.9** | **97**/13.8/15.9 |
| walkbridge+30% | 76/3.7/7.1 | 89/8.6/11.0 | 82/6.1/10.3 | 85/5.8/7.8 | 89/8.4/12.1 | 89/7.8/11.5 | **91/9.6/12.8** | **91**/9.5/11.9 |
| walkbridge+50% | 59/2.2/4.3 | 76/4.9/5.7 | 67/4.1/7.0 | 69/2.7/4.8 | 76/5.4/8.1 | 79/5.4/8.7 | 84/**7.0/10.1** | **85/7.0**/9.2 |
| walkbridge+70% | 42/1.0/1.9 | 56/2.0/1.7 | 45/2.0/3.3 | 50/1.3/2.2 | 53/2.5/4.0 | 59/3.0/5.0 | 65/4.0/6.2 | **76/5.1/7.0** |
| walkbridge+90% | 26/-0.1/-0.1 | 32/-0.2/-1.1 | 28/0.3/0.5 | 30/0.0/-0.0 | 31/0.4/0.8 | 30/0.4/0.8 | 34/0.7/1.3 | **57/2.7/3.9** |
| pepper+10% | 67/5.0/9.9 | 99/19.1/21.5 | 99/15.0/22.2 | 97/13.5/15.8 | 74/5.4/11.3 | 99/13.6/20.3 | **100/20.2/24.6** | 99/18.0/21.0 |
| pepper+30% | 55/3.7/7.0 | 96/12.3/13.6 | 96/11.4/16.3 | 87/6.3/9.5 | 72/5.2/10.7 | **98**/12.0/16.8 | **98/15.1/19.7** | **98**/14.6/18.3 |
| pepper+50% | 44/2.4/4.5 | 85/6.7/6.7 | 85/7.0/9.7 | 71/3.5/5.5 | 65/4.5/8.9 | 94/9.7/13.1 | **96/11.8/15.7** | **96**/11.6/14.4 |
| pepper+70% | 33/1.2/2.1 | 63/2.8/2.1 | 59/3.1/4.4 | 52/1.6/2.4 | 51/2.7/4.7 | 79/5.2/6.2 | 84/6.8/8.9 | **93/9.0/11.4** |
| pepper+90% | 24/0.2/0.1 | 35/0.1/-1.0 | 30/0.6/0.6 | 31/0.3/0.1 | 28/0.7/1.1 | 35/0.9/1.0 | 39/1.3/1.7 | **76/4.2/4.8** |
| mandrill+10% | 74/3.3/6.0 | 89/8.1/9.0 | 92/6.9/6.9 | 93/9.6/9.6 | 84/3.7/7.4 | 93/9.6/9.6 | **95/11.1/11.5** | **95**/10.8/10.3 |
| mandrill+30% | 63/2.0/3.6 | 83/5.9/6.6 | 76/3.8/5.9 | 83/4.7/4.9 | 73/3.0/5.5 | 85/5.8/6.8 | **87/6.8/7.4** | 86/6.4/6.5 |
| mandrill+50% | 50/1.1/2.2 | 73/3.6/3.7 | 65/2.9/4.6 | 69/2.0/3.4 | 61/2.2/4.0 | 74/3.6/5.0 | 77/**4.6/5.6** | **78**/4.4/4.6 |
| mandrill+70% | 36/0.4/0.8 | 57/1.4/0.6 | 51/1.5/2.4 | 52/0.9/1.5 | 47/1.2/2.2 | 62/2.3/3.4 | 64/2.9/**3.9** | **70/3.1**/3.5 |
| mandrill+90% | 28/-0.3/-0.6 | 36/-0.6/-1.9 | 37/0.2/0.4 | 34/-0.1/-0.4 | 33/0.1/0.3 | 39/0.5/0.9 | 42/0.8/1.2 | **58/1.9/2.5** |
| lake+10% | 92/6.9/12.5 | **98**/16.9/**21.3** | 96/11.3/17.7 | 97/14.0/15.0 | 97/8.7/16.1 | **98**/14.3/19.2 | **98/17.2**/21.1 | **98**/16.7/19.5 |
| lake+30% | 75/4.3/8.1 | 93/11.3/13.9 | 91/9.3/14.4 | 86/7.1/10.0 | 92/7.9/13.9 | **95**/10.5/15.0 | **95/12.7/16.7** | **95**/12.0/14.3 |
| lake+50% | 58/2.6/4.9 | 79/6.5/7.2 | 71/5.9/9.4 | 69/3.7/5.9 | 78/6.2/10.2 | 88/8.3/11.7 | **91/10.0/13.7** | 90/9.5/11.5 |
| lake+70% | 41/1.3/2.3 | 54/2.9/2.6 | 42/2.5/4.1 | 47/1.8/2.8 | 43/2.8/4.6 | 60/4.7/7.0 | 68/5.8/8.6 | **84/7.4/9.0** |
| lake+90% | 24/0.3/0.3 | 26/0.5/-0.4 | 25/0.6/0.8 | 26/0.5/0.4 | 24/0.6/1.0 | 13/0.7/1.1 | 26/1.1/1.7 | **62/4.2/5.3** |
| jetplane+10% | 39/2.5/6.1 | **99/17.5/21.0** | 98/11.5/17.5 | 98/12.8/13.3 | 39/3.4/8.3 | **99**/13.1/19.1 | **99**/17.0/20.0 | 98/15.6/17.0 |
| jetplane+30% | 32/0.7/2.6 | 95/10.3/11.5 | 94/9.0/13.3 | 87/5.0/7.3 | 38/3.2/7.5 | **97**/10.4/15.0 | **97/12.4/15.7** | **97**/11.5/12.6 |
| jetplane+50% | 27/-0.6/-0.1 | 80/4.5/4.0 | 75/4.2/6.7 | 69/1.5/2.8 | 34/2.4/5.2 | 92/7.9/10.6 | **94/9.3/12.2** | **94**/9.0/10.0 |
| jetplane+70% | 22/-1.7/-2.4 | 53/0.6/-0.7 | 42/0.2/0.9 | 47/-0.5/-0.5 | 23/-0.6/-0.3 | 67/3.2/4.8 | 74/4.4/6.4 | **90/6.7/7.4** |
| jetplane+90% | 18/-2.5/-4.1 | 25/-1.8/-3.6 | 25/-1.7/-2.5 | 26/-1.8/-2.9 | 18/-2.3/-3.4 | 14/-1.6/-2.2 | 26/-1.2/-1.5 | **74/3.4/3.7** |
| **Salt-and-Pepper Impulse Noise** | | | | | | | | |
| walkbridge+10% | 90/5.4/9.9 | 96/12.9/17.3 | 90/7.6/12.4 | 98/15.8/19.9 | 98/16.3/20.7 | 98/15.8/19.9 | **99**/17.2/22.7 | **99/17.5/23.2** |
| walkbridge+30% | 71/3.0/4.5 | 94/10.4/14.3 | 83/6.3/9.8 | 96/11.7/16.4 | 94/10.5/15.2 | 96/11.7/16.4 | 96/12.0/17.1 | **97/12.3/17.5** |
| walkbridge+50% | 51/-0.1/-1.7 | 89/8.1/11.4 | 71/4.0/5.4 | 92/9.3/14.0 | 88/7.8/11.8 | 92/9.3/13.9 | 92/9.2/13.8 | **93/9.5/14.3** |
| walkbridge+70% | 32/-2.0/-4.6 | 82/6.1/8.7 | 49/1.4/2.7 | **87**/7.3/11.5 | 69/4.4/6.9 | **87**/7.3/11.5 | 85/6.9/11.0 | **87/7.4/11.6** |
| walkbridge+90% | 15/-3.2/-6.2 | 67/3.7/5.1 | 26/0.2/0.6 | 73/**4.8**/**7.8** | 36/0.9/1.6 | 73/**4.8**/7.7 | 56/3.3/5.8 | **74/4.8/7.8** |
| pepper+10% | 68/4.9/9.6 | 99/14.8/20.1 | 99/15.0/21.8 | **100**/20.5/24.9 | 74/5.4/11.4 | **100**/20.5/24.9 | **100**/23.2/30.5 | **100/23.9/31.0** |
| pepper+30% | 52/3.1/4.8 | 98/14.6/18.3 | 95/10.8/13.6 | 99/16.8/22.9 | 73/5.4/11.2 | 99/16.8/22.9 | 99/17.7/24.8 | **100/18.5/25.6** |
| pepper+50% | 38/0.3/-1.1 | 97/12.9/16.1 | 84/6.1/7.0 | **99**/14.9/21.5 | 71/5.2/10.6 | **99**/14.8/21.5 | **99**/14.5/21.1 | **99/15.4/22.4** |
| pepper+70% | 25/-1.5/-3.9 | 95/10.6/13.3 | 57/2.1/3.4 | **98**/12.5/18.5 | 61/3.9/7.4 | **98**/12.5/18.5 | 96/11.4/16.9 | **98/12.7/18.7** |
| pepper+90% | 14/-2.7/-5.5 | 89/7.2/8.5 | 27/0.4/0.6 | **93**/8.8/12.7 | 32/1.2/1.9 | **93**/8.8/12.5 | 75/4.8/7.9 | **93/9.0/12.9** |
| mandrill+10% | 77/2.7/4.9 | 93/9.8/11.3 | 90/4.5/6.9 | 97/13.1/14.3 | 87/4.2/9.2 | 97/13.1/14.3 | **98**/14.4/17.1 | **98/14.5/17.2** |
| mandrill+30% | 61/1.5/2.3 | 90/7.8/9.0 | 75/4.0/5.9 | 92/8.9/10.7 | 79/3.6/7.2 | 92/8.9/10.7 | **93**/9.3/11.8 | **93/9.4/11.9** |
| mandrill+50% | 44/-0.9/-2.8 | 84/5.7/6.6 | 67/2.7/3.3 | 87/6.6/8.5 | 68/2.8/5.2 | 87/6.6/8.5 | 87/6.7/**8.8** | **88/6.8/8.8** |
| mandrill+70% | 27/-2.7/-5.6 | 76/3.8/4.3 | 48/1.1/1.9 | **80/4.9**/6.5 | 54/2.0/3.6 | **80/4.9**/6.5 | 79/4.8/**6.6** | **80/4.9**/6.5 |
| mandrill+90% | 10/-3.8/-7.2 | 63/2.0/1.9 | 36/0.3/0.6 | **69/3.1**/4.3 | 35/0.4/0.8 | **69/3.1**/4.3 | 59/2.4/3.8 | **69/3.1/4.4** |
| lake+10% | 91/6.6/11.9 | 99/16.4/22.9 | 96/11.3/17.6 | 99/19.6/25.9 | 99/9.0/17.2 | 99/19.6/25.7 | **100**/20.3/27.5 | **100/20.6/27.9** |
| lake+30% | 71/3.9/5.6 | 97/13.6/18.7 | 90/9.1/12.8 | 98/15.0/21.4 | 97/8.6/16.0 | 98/15.0/21.3 | 98/15.1/21.7 | **99/15.4/22.3** |
| lake+50% | 52/1.2/-0.4 | 94/11.2/15.3 | 76/5.7/6.8 | **97**/12.5/18.3 | 91/7.7/13.6 | **97**/12.5/18.2 | 96/12.2/17.9 | **97/12.7/18.6** |
| lake+70% | 33/-0.5/-3.0 | 90/9.0/12.1 | 52/2.4/3.7 | 93/**10.4/15.2** | 63/5.0/8.2 | 93/**10.4/15.2** | 91/9.7/14.4 | **94/10.4/15.2** |
| lake+90% | 18/-1.6/-4.5 | 80/6.2/7.5 | 26/0.5/0.9 | **84**/7.3/10.1 | 25/1.1/1.9 | 83/7.3/10.1 | 51/4.3/7.3 | **84/7.4/10.2** |
| jetplane+10% | 49/2.5/6.0 | **100**/17.0/23.4 | 98/11.6/17.3 | **100**/20.4/26.8 | 39/3.4/8.5 | **100**/20.4/26.8 | **100**/20.7/28.0 | **100/21.3/29.2** |
| jetplane+30% | 39/0.6/1.2 | 98/13.6/17.9 | 93/8.3/10.4 | **99**/15.5/21.9 | 40/3.4/8.3 | **99**/15.5/21.6 | **99**/15.3/21.6 | **99/15.9/22.7** |
| jetplane+50% | 33/-1.4/-4.1 | 96/10.9/14.1 | 79/4.0/5.1 | **98**/12.7/18.4 | 39/3.1/7.2 | **98**/12.7/18.4 | **98**/12.1/17.3 | **98/12.9/18.5** |
| jetplane+70% | 30/-2.8/-6.4 | 93/8.5/10.5 | 53/0.3/1.2 | **96**/10.2/**14.6** | 32/1.2/3.0 | **96**/10.2/**14.6** | 94/9.2/13.3 | **96/10.3/14.6** |
| jetplane+90% | 28/-3.7/-7.9 | 87/5.6/6.0 | 26/-1.7/-2.1 | **89**/6.6/8.6 | 29/-1.9/-2.8 | **89**/6.6/8.6 | 54/2.4/4.8 | **89/6.8/8.7** |
| **Mixed Impulse Noise (Half Random-Value Noise and Half Salt-and-Pepper Noise)** | | | | | | | | |
| walkbridge+10% | 91/6.1/10.1 | 93/10.6/14.7 | 91/7.5/12.3 | 96/12.6/13.3 | 96/12.5/16.0 | 96/12.6/13.3 | **98**/14.8/17.8 | **98/15.1/17.9** |
| walkbridge+30% | 73/3.6/6.7 | 90/8.4/11.8 | 83/6.3/10.3 | 88/6.6/8.3 | 89/8.6/12.2 | 92/8.6/12.2 | **93/10.2/13.5** | **93/10.2**/12.9 |
| walkbridge+50% | 55/1.5/1.9 | 81/5.7/7.0 | 70/4.3/6.8 | 76/3.5/5.7 | 78/5.7/8.7 | 85/6.3/10.0 | 86/**7.6/10.8** | **87/7.6**/10.1 |
| walkbridge+70% | 37/-0.5/-1.8 | 63/2.4/1.9 | 50/2.0/2.9 | 58/1.9/3.2 | 56/2.8/4.9 | 72/4.4/7.2 | 74/5.1/**7.9** | **80/5.7/7.9** |
| walkbridge+90% | 21/-1.9/-4.0 | 34/-0.6/-2.1 | 30/0.1/0.4 | 34/0.3/0.5 | 31/0.6/1.3 | 38/1.2/2.0 | 40/1.3/2.3 | **63/3.3/4.9** |
| pepper+10% | 68/5.0/9.7 | 98/13.9/19.5 | 99/15.0/22.0 | 98/14.3/16.0 | 74/5.4/11.3 | 99/14.4/19.9 | **100/21.0/25.6** | 99/19.9/23.4 |
| pepper+30% | 54/3.7/6.8 | 97/12.7/16.0 | 96/11.4/15.4 | 91/7.5/10.8 | 72/5.3/10.8 | 98/12.8/18.5 | **99/15.8/20.7** | 98/14.9/18.4 |
| pepper+50% | 41/1.8/2.3 | 92/8.5/8.6 | 86/7.0/8.9 | 80/4.5/7.0 | 68/4.8/9.5 | **97**/11.2/16.1 | **97/12.6/17.1** | **97/12.6**/15.7 |
| pepper+70% | 29/-0.1/-1.2 | 73/3.6/2.4 | 62/3.0/3.6 | 63/2.5/3.8 | 54/3.3/5.9 | 90/8.1/10.7 | 92/9.1/12.5 | **94/10.1/12.8** |
| pepper+90% | 19/-1.4/-3.4 | 39/-0.2/-2.0 | 33/0.4/0.5 | 37/0.6/0.7 | 31/1.0/1.5 | 53/2.1/2.5 | 49/2.2/2.9 | **82/5.6/6.6** |
| mandrill+10% | 76/3.0/5.3 | 86/6.8/8.3 | 91/5.5/6.8 | 95/10.4/10.1 | 83/3.6/7.3 | 95/10.5/10.3 | **96/12.1/12.4** | **96**/11.7/11.2 |
| mandrill+30% | 63/1.8/3.4 | 82/5.4/6.6 | 74/3.9/6.0 | 85/5.3/5.1 | 73/2.9/5.3 | 88/6.5/7.4 | **89/7.3/8.1** | **89/7.3**/7.5 |
| mandrill+50% | 47/0.6/0.6 | 75/3.7/4.0 | 67/3.0/4.4 | 74/2.5/3.8 | 61/2.2/3.9 | 78/4.4/5.6 | 80/**5.0/5.9** | **81/5.0**/5.3 |
| mandrill+70% | 32/-1.0/-2.6 | 60/1.3/0.2 | 53/1.5/1.8 | 58/1.3/2.1 | 48/1.4/2.6 | 68/2.9/4.2 | 69/3.3/**4.3** | **73/3.5**/3.9 |
| mandrill+90% | 20/-2.4/-4.9 | 35/-1.2/-3.3 | 36/0.3/0.5 | 37/0.2/0.1 | 33/0.3/0.6 | 46/1.1/1.8 | 45/1.0/1.3 | **62/2.2/2.8** |
| lake+10% | 91/6.8/12.0 | 98/14.6/20.5 | 96/11.3/17.7 | 98/15.0/15.5 | 97/8.7/16.1 | 98/15.0/19.4 | **99/18.0/22.2** | **99**/17.9/21.2 |
| lake+30% | 73/4.3/7.6 | 95/11.7/15.7 | 91/9.3/13.7 | 90/8.0/10.5 | 92/7.9/13.8 | **96**/11.0/16.4 | **96/13.1/17.2** | **96**/12.8/15.6 |
| lake+50% | 55/2.3/2.7 | 87/7.9/9.0 | 75/6.1/8.6 | 78/4.8/7.2 | 82/6.6/11.0 | **92**/9.3/13.1 | **92/10.4/14.1** | **92**/10.0/12.2 |
| lake+70% | 37/0.6/-0.6 | 66/3.7/3.1 | 44/2.8/3.7 | 58/2.6/4.1 | 48/3.7/6.2 | 82/7.0/9.8 | 83/7.7/**10.8** | **87/7.9**/9.4 |
| lake+90% | 22/-0.6/-2.7 | 34/0.4/-1.1 | 20/0.6/0.7 | 30/0.8/1.0 | 24/0.8/1.5 | 22/1.5/2.4 | 33/2.0/3.1 | **74/5.3/6.0** |
| jetplane+10% | 44/2.6/6.0 | **99**/15.4/**20.8** | 98/11.6/17.5 | **99**/13.9/13.3 | 39/3.4/8.3 | **99**/13.9/19.3 | **99/17.6/20.8** | **99**/16.8/18.5 |
| jetplane+30% | 36/0.8/2.5 | 97/11.6/14.2 | 94/8.8/12.3 | 91/6.3/8.2 | 38/3.2/7.7 | **98**/11.0/**16.6** | **98/13.1**/16.4 | **98**/12.6/14.1 |
| jetplane+50% | 30/-0.8/-1.6 | 90/6.6/6.2 | 79/4.5/5.8 | 79/2.8/4.2 | 37/2.8/6.1 | **95**/9.0/12.7 | **95/10.0/13.0** | **95**/9.7/10.7 |
| jetplane+70% | 25/-2.1/-4.5 | 68/1.7/-0.1 | 45/0.6/0.5 | 60/0.4/0.9 | 25/0.7/1.9 | 88/6.3/7.9 | 87/6.6/**8.8** | **91/7.3**/8.0 |
| jetplane+90% | 22/-3.1/-6.4 | 34/-1.8/-4.4 | 19/-1.8/-2.4 | 30/-1.5/-2.3 | 16/-2.1/-3.0 | 19/-0.8/-0.9 | 32/-0.2/-0.1 | **79/4.2/4.4** |

image recovery results with varying the regularization parameter $\lambda$. For $\ell_{02}TV$ model in (17), the parameter $\chi$ is scaled to the range $[0, 10]$ for better visualization. We make the following interesting observations. **(i)** The $\ell_{02}TV$-AOP method greatly improves upon $\ell_1TV$-SBM, MFM and TSM, by a large margin. These results are consistent with the reported results in [57]. **(ii)** The $\ell_0TV$-PDA method outperforms $\ell_{02}TV$-AOP in most test cases because it adopts the $\ell_0$-norm in the data fidelity term. **(iii)** In the case of random-value impulse noise, our $\ell_0TV$-PADMM method is better than $\ell_0TV$-PDA in $SNR_0$ value while it is comparable to $\ell_0TV$-PDA in $SNR_1$ and $SNR_2$. On the other hand, when salt-and-pepper impulse noise is added, we find that $\ell_0TV$-PADMM outperforms $\ell_0TV$-PDA in most test cases. Interestingly, the performance gap between $\ell_0TV$-PADMM and $\ell_0TV$-PDA grows larger, as the noise level increases. **(iv)** For the same noise level, $\ell_0TV$-PADMM

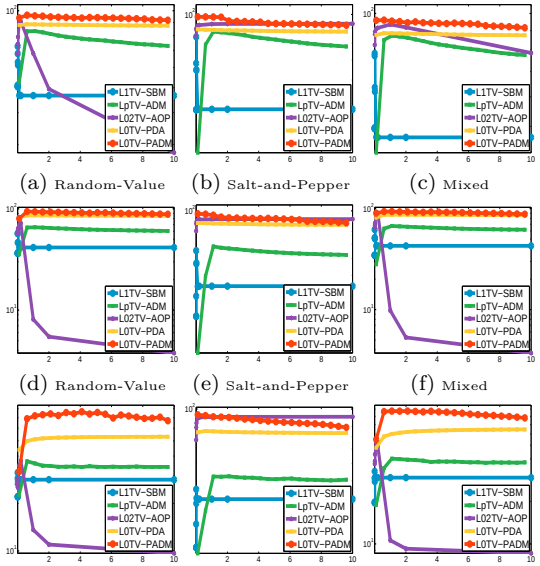



(a) Random-Value (b) Salt-and-Pepper (c) Mixed

(d) Random-Value (e) Salt-and-Pepper (f) Mixed

(g) Random-Value (h) Salt-and-Pepper (i) Mixed

Figure 3: Image denoising with varying the tuning parameter $\lambda$ in (6) on 'cameraman' image. First row: noise level = 50%. Second row: noise level = 70%. Third row: noise level = 90%.




(a) Random-Value (b) Salt-and-Pepper (c) Mixed

(d) Random-Value (e) Salt-and-Pepper (f) Mixed

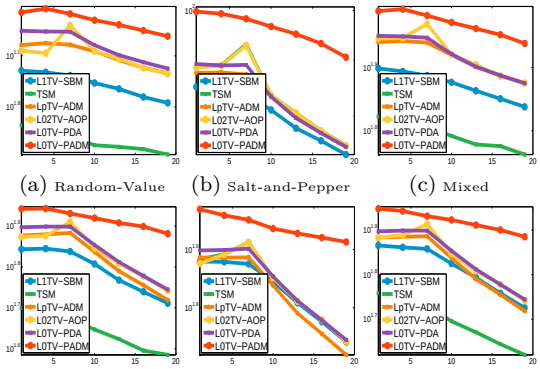
(g) Random-Value (h) Salt-and-Pepper (i) Mixed

Figure 4: Image deblurring with varying the tuning parameter $\lambda$ in (6) on ' cameraman' image. First row: noise level = 50%. Second row: noise level = 70%. Third row: noise level = 90%.

achieves better recovery performance in the presence of salt-and-pepper impulse noise than random-valued impulse noise. This is primarily due to the fact that random-valued noise can take any value between 0 and 1, thus, making it more difficult to detect which pixels are corrupted.

### 5.4 General Image Deblurring Problems

In this subsection, we demonstrate the performance of all methods with their optimal regularization parameters on general deblurring problems. Table 4 shows the recovery results for random-valued impulse noise, salt-and-pepper impulse noise, and mixed impulse noise, respectively. Figure 4 shows image recovery results with varying the regularization parameter. We have the following interesting observations. (**i**) $\ell_{02}TV$-AOP significantly outperforms $\ell_1 TV$-SBM, and the performance gap becomes larger as the noise level increases. This is because the key assumption in the $\ell_1$ model is that $Ku - b$ is sparse at the optimal solution $u^*$. This does not hold when the noise level is high. (**ii**) $\ell_0 TV$-PDA outperforms $\ell_{02}TV$-AOP for high level ($\geq$ 30%) random-valued impulse noise. However, for salt-and-pepper impulse noise, $\ell_0 TV$-PDA gives worse performance than $\ell_{02}TV$-AOP in most cases. This phenomenon indicates that the Penalty Decomposition Algorithm is not stable for deblurring problems. (**iii**) By contrast, our $\ell_0 TV$-PADMM consistently outperforms all methods, especially when the noise level is large. We attribute this result to




(a) Random-Value (b) Salt-and-Pepper (c) Mixed

(d) Random-Value (e) Salt-and-Pepper (f) Mixed

Figure 5: Image deblurring with varying the radius parameter $r$ in (18). First row: 'cameraman' image. Second row: 'barbara' image.

the "lifting" technique that is used in our optimization algorithm.

Finally, we also report the performance of all methods with sweeping the radius parameter $r$ as in (18) over $\{1, 4, 7, ..., 20\}$ in Figure 5. We notice that the restoration quality degenerates as the radius of the kernel increases for all methods. However, our method consistently gives the best performance.

### 5.5 Scratched Image Denoising Problems

In this subsection, we demonstrate the superiority of the proposed $\ell_0 TV$-PADMM in real-world image restoration problems. Specifically, we corrupt the images with

Table 4: General deblurring problems. The results separated by '/' are $SNR_0$, $SNR_1$ and $SNR_2$, respectively. The $1^{st}$, $2^{nd}$, and $3^{rd}$ best results are colored with **red**, blue and green, respectively.

| Img. \ Alg. | Corrupted | $\ell_1 TV$-$SBM$ | $TSM$ | $\ell_p TV$-$PADMM$ | $\ell_{02} TV$-$AOP$ | $\ell_0 TV$-$PDA$ | $\ell_0 TV$-$PADM$ |
|---|---|---|---|---|---|---|---|
| Random-Valued Impulse Noise | | | | | | | |
| walkbridge+10% | 63/2.9/3.4 | 74/4.8/8.6 | 72/4.6/8.2 | 77/5.1/9.2 | 81/5.6/10.1 | 76/5.0/9.0 | **91/7.0/13.2** |
| walkbridge+30% | 52/1.1/0.0 | 72/4.6/8.1 | 61/3.7/6.8 | 75/4.9/8.7 | 79/5.4/9.7 | 74/4.8/8.7 | **86/6.4/11.7** |
| walkbridge+50% | 42/-0.2/-1.9 | 63/3.8/6.9 | 46/2.4/4.6 | 71/4.5/8.0 | 75/4.9/8.6 | 73/4.7/8.3 | **84/6.0/11.0** |
| walkbridge+70% | 31/-1.2/-3.2 | 46/2.1/3.8 | 33/1.1/2.3 | 55/2.9/5.1 | 65/3.3/4.8 | 69/4.3/7.7 | **81/5.6/10.1** |
| walkbridge+90% | 21/-2.0/-4.2 | 28/0.3/0.8 | 25/0.2/0.5 | 31/0.6/1.2 | 33/0.4/0.6 | 42/1.7/3.0 | **67/3.7/5.8** |
| pepper+10% | 81/4.9/4.5 | 94/9.3/14.7 | 93/8.3/13.6 | 70/5.1/10.1 | 96/9.7/15.8 | 94/9.0/14.7 | **99/11.1/19.8** |
| pepper+30% | 66/2.1/0.3 | 92/8.5/13.3 | 82/5.7/9.9 | 68/4.9/9.7 | 96/9.7/15.8 | 93/8.8/14.1 | **98/10.7/18.8** |
| pepper+50% | 52/0.4/-1.8 | 83/6.4/9.9 | 58/3.4/6.0 | 65/4.6/8.9 | 95/9.3/14.9 | 92/8.5/13.5 | **98/10.4/17.8** |
| pepper+70% | 37/-0.8/-3.2 | 58/3.1/4.7 | 37/1.6/2.9 | 52/3.0/5.4 | 82/5.1/5.4 | 90/7.8/12.1 | **97/9.8/16.4** |
| pepper+90% | 23/-1.8/-4.3 | 29/0.6/1.0 | 24/0.4/0.7 | 29/0.9/1.3 | 38/0.9/0.7 | 54/2.5/3.5 | **85/6.1/7.2** |
| mandrill+10% | 59/1.6/1.3 | 67/2.9/4.7 | 65/2.7/4.3 | 54/2.1/3.8 | 68/3.0/4.5 | 68/3.1/5.0 | **78/4.3/7.3** |
| mandrill+30% | 50/0.0/-1.7 | 66/2.9/4.6 | 60/2.3/3.9 | 52/2.1/3.7 | 68/3.0/4.6 | 67/3.0/4.8 | **76/4.0/6.8** |
| mandrill+50% | 40/-1.1/-3.4 | 64/2.7/4.3 | 50/1.6/2.9 | 51/2.0/3.5 | 68/2.9/4.5 | 66/2.9/4.6 | **73/3.6/6.0** |
| mandrill+70% | 30/-2.0/-4.7 | 53/1.8/3.1 | 40/0.9/1.7 | 46/1.6/2.9 | 64/2.5/3.6 | 65/2.7/4.4 | **70/3.3/5.4** |
| mandrill+90% | 21/-2.7/-5.6 | 38/0.5/0.9 | 36/0.3/0.6 | 34/0.4/0.7 | 42/0.6/0.8 | 49/1.5/2.5 | **65/2.7/4.2** |
| lake+10% | 71/4.8/4.9 | 84/7.6/11.6 | 83/7.3/11.3 | 83/6.7/11.3 | 89/8.6/13.8 | 84/7.7/12.1 | **96/10.0/17.4** |
| lake+30% | 59/2.6/1.2 | 81/7.1/10.8 | 65/5.2/8.9 | 80/6.4/10.7 | 89/8.5/13.2 | 83/7.4/11.6 | **94/9.5/15.9** |
| lake+50% | 46/1.1/-0.7 | 68/5.5/8.8 | 35/3.2/5.6 | 76/6.0/9.8 | 86/7.9/11.9 | 82/7.2/11.1 | **92/9.1/15.1** |
| lake+70% | 34/0.0/-2.1 | 35/2.6/4.5 | 22/1.6/2.9 | 39/3.3/5.6 | 66/4.3/5.4 | 79/6.7/10.2 | **89/8.5/13.8** |
| lake+90% | 22/-0.9/-3.1 | 22/0.6/1.0 | 16/0.4/0.8 | 22/0.7/1.3 | 21/0.6/0.8 | 31/2.1/3.5 | **74/5.6/7.2** |
| jetplane+10% | 76/3.3/2.1 | 88/6.7/9.9 | 88/6.1/9.7 | 63/2.8/6.5 | 93/7.9/12.5 | 89/6.8/10.5 | **98/9.1/16.6** |
| jetplane+30% | 63/0.7/-1.9 | 86/6.2/9.1 | 68/3.2/6.3 | 66/2.7/6.2 | 93/7.8/12.0 | 88/6.6/10.0 | **97/8.8/15.6** |
| jetplane+50% | 49/-0.9/-3.9 | 74/3.9/6.6 | 34/0.9/2.6 | 55/2.5/5.6 | 91/7.0/9.7 | 87/6.3/9.4 | **95/8.4/14.2** |
| jetplane+70% | 36/-2.1/-5.3 | 37/0.3/1.3 | 22/-0.7/-0.3 | 35/-0.1/0.6 | 64/1.5/1.9 | 84/5.8/8.5 | **93/7.8/12.4** |
| jetplane+90% | 23/-3.0/-6.3 | 23/-1.7/-2.3 | 14/-1.9/-2.5 | 16/-2.2/-3.3 | 20/-1.7/-2.5 | 30/0.0/0.6 | **80/4.5/5.1** |
| Salt-and-Pepper Impulse Noise | | | | | | | |
| walkbridge+10% | 61/2.0/0.8 | 73/4.8/8.5 | 80/5.6/10.1 | 76/5.1/9.1 | 80/5.6/10.1 | 76/5.0/9.0 | **94/7.4/14.3** |
| walkbridge+30% | 48/-0.5/-3.2 | 71/4.5/7.9 | 79/5.4/9.7 | 74/4.8/8.5 | 79/5.4/9.7 | 75/4.9/8.8 | **92/7.2/13.7** |
| walkbridge+50% | 35/-2.1/-5.3 | 67/4.1/7.3 | 77/5.2/9.3 | 72/4.5/8.1 | 77/5.2/9.3 | 73/4.8/8.5 | **90/6.8/12.9** |
| walkbridge+70% | 22/-3.3/-6.7 | 53/2.8/5.2 | 75/5.0/8.8 | 61/3.5/6.4 | 75/4.9/8.8 | 71/4.5/8.1 | **86/6.4/11.8** |
| walkbridge+90% | 8/-4.2/-7.7 | 31/0.6/1.0 | 73/4.7/8.3 | 34/0.9/1.7 | 73/4.7/8.3 | 59/3.4/6.3 | **79/5.4/9.9** |
| pepper+10% | 79/3.6/1.3 | 94/8.9/14.2 | 96/9.7/15.8 | 69/5.0/10.0 | 96/9.6/15.8 | 94/9.1/14.8 | **99/11.4/20.3** |
| pepper+30% | 62/0.2/-3.2 | 92/8.5/13.2 | 96/9.6/15.7 | 69/4.9/9.6 | 96/9.6/15.7 | 94/8.9/14.4 | **99/11.2/19.7** |
| pepper+50% | 45/-1.7/-5.4 | 87/7.3/11.2 | 95/9.4/15.4 | 66/4.7/9.1 | 95/9.4/15.4 | 93/8.6/13.8 | **99/10.9/19.1** |
| pepper+70% | 28/-3.0/-6.8 | 70/4.3/6.5 | 95/9.2/14.8 | 56/3.7/6.8 | 95/9.2/14.9 | 91/8.3/13.0 | **98/10.3/18.2** |
| pepper+90% | 11/-4.1/-7.9 | 33/0.8/1.1 | 94/8.8/14.1 | 32/1.1/1.8 | 94/8.8/14.1 | 79/5.6/8.8 | **96/9.5/15.8** |
| mandrill+10% | 58/0.7/-1.3 | 67/2.9/4.7 | 67/2.9/4.4 | 53/2.1/3.8 | 67/2.9/4.4 | 68/3.1/5.0 | **86/5.2/9.5** |
| mandrill+30% | 45/-1.7/-5.2 | 65/2.8/4.4 | 67/2.9/4.5 | 52/2.1/3.6 | 67/2.9/4.5 | 68/3.0/4.9 | **83/4.9/8.7** |
| mandrill+50% | 32/-3.2/-7.2 | 64/2.6/4.2 | 66/2.8/4.4 | 51/2.0/3.5 | 66/2.8/4.4 | 67/3.0/4.7 | **80/4.5/7.9** |
| mandrill+70% | 19/-4.4/-8.6 | 56/2.0/3.3 | 65/2.7/4.2 | 48/1.8/3.1 | 65/2.7/4.2 | 66/2.8/4.5 | **75/4.0/6.7** |
| mandrill+90% | 7/-5.2/-9.6 | 39/0.5/0.8 | 65/2.7/4.2 | 35/0.5/1.0 | 65/2.7/4.2 | 60/2.4/3.9 | **70/3.3/5.3** |
| lake+10% | 69/3.9/2.4 | 83/7.4/11.4 | 90/8.7/13.8 | 82/6.6/11.2 | 90/8.7/13.8 | 85/7.7/12.1 | **98/10.3/18.5** |
| lake+30% | 54/1.0/-1.8 | 81/7.1/10.6 | 89/8.5/13.4 | 80/6.3/10.6 | 89/8.5/13.4 | 84/7.6/11.8 | **97/10.1/17.9** |
| lake+50% | 38/-0.7/-3.9 | 76/6.4/9.6 | 87/8.2/12.9 | 77/6.0/9.8 | 87/8.2/12.8 | 82/7.3/11.3 | **96/9.8/17.0** |
| lake+70% | 23/-1.9/-5.3 | 49/3.9/6.3 | 86/7.9/12.2 | 56/4.4/7.3 | 86/7.9/12.2 | 81/7.0/10.7 | **94/9.3/15.9** |
| lake+90% | 8/-2.8/-6.4 | 24/0.9/1.4 | 83/7.4/11.2 | 21/1.0/1.8 | 84/7.5/11.1 | 63/5.0/8.1 | **88/8.2/13.3** |
| jetplane+10% | 75/2.3/-0.4 | 88/6.5/9.7 | 93/8.0/12.6 | 67/2.8/6.5 | 93/8.0/12.6 | 89/6.9/10.6 | **99/9.5/17.8** |
| jetplane+30% | 58/-0.9/-4.8 | 86/6.2/9.0 | 93/7.7/11.9 | 64/2.7/6.1 | 92/7.6/11.8 | 88/6.7/10.2 | **99/9.4/17.2** |
| jetplane+50% | 42/-2.7/-7.0 | 82/5.4/7.8 | 91/7.5/11.4 | 54/2.5/5.7 | 91/7.5/11.5 | 87/6.5/9.7 | **98/9.0/16.2** |
| jetplane+70% | 25/-3.9/-8.4 | 48/1.9/3.8 | 90/7.1/10.7 | 39/1.2/2.9 | 90/7.1/10.6 | 86/6.1/9.0 | **96/8.7/14.9** |
| jetplane+90% | 8/-4.9/-9.5 | 24/-1.3/-1.8 | 89/6.7/9.7 | 21/-1.9/-2.8 | 89/6.7/9.9 | 72/3.6/6.1 | **92/7.2/11.8** |
| Mixed Impulse Noise (Half Random-Value Noise and Half Salt-and-Pepper Noise) | | | | | | | |
| walkbridge+10% | 62/2.4/1.9 | 74/4.8/8.5 | 72/4.6/8.2 | 77/5.1/9.2 | 81/5.6/10.1 | 76/5.0/9.0 | **93/7.4/14.0** |
| walkbridge+30% | 50/0.2/-1.9 | 71/4.5/7.9 | 65/3.9/7.2 | 74/4.8/8.6 | 79/5.4/9.6 | 74/4.8/8.7 | **87/6.5/12.0** |
| walkbridge+50% | 38/-1.3/-3.9 | 64/3.8/6.9 | 52/2.9/5.5 | 71/4.5/8.0 | 78/5.2/8.8 | 73/4.7/8.3 | **84/6.1/11.0** |
| walkbridge+70% | 27/-2.3/-5.3 | 48/2.4/4.4 | 38/1.6/3.2 | 59/3.3/6.0 | 74/4.5/7.1 | 70/4.4/7.8 | **81/5.6/10.1** |
| walkbridge+90% | 15/-3.2/-6.3 | 29/0.5/1.0 | 27/0.4/0.9 | 33/0.8/1.6 | 43/1.1/1.4 | 50/2.3/3.7 | **71/4.3/7.2** |
| pepper+10% | 80/4.2/2.6 | 94/9.1/14.5 | 93/8.5/13.7 | 69/5.1/10.0 | 96/9.7/15.9 | 94/9.0/14.7 | **99/11.1/19.8** |
| pepper+30% | 64/1.0/-1.8 | 91/8.4/13.0 | 87/6.4/10.9 | 68/4.9/9.6 | 96/9.7/15.8 | 93/8.8/14.1 | **99/10.9/19.3** |
| pepper+50% | 49/-0.8/-3.9 | 84/6.7/10.2 | 68/4.2/7.5 | 66/4.7/9.1 | 96/9.4/15.0 | 92/8.5/13.5 | **98/10.5/18.2** |
| pepper+70% | 33/-2.1/-5.4 | 61/3.5/5.2 | 43/2.3/4.0 | 54/3.4/6.3 | 94/8.4/11.4 | 90/7.9/12.2 | **97/10.0/16.8** |
| pepper+90% | 17/-3.1/-6.4 | 31/0.9/1.3 | 27/0.7/1.2 | 32/1.1/1.8 | 55/2.0/1.5 | 60/3.2/4.9 | **92/8.2/11.7** |
| mandrill+10% | 58/1.1/-0.2 | 67/2.9/4.7 | 65/2.7/4.3 | 53/2.1/3.8 | 67/2.9/4.6 | 68/3.1/5.0 | **85/5.0/9.4** |
| mandrill+30% | 47/-0.9/-3.7 | 66/2.8/4.5 | 62/2.4/4.0 | 52/2.1/3.7 | 68/3.0/4.6 | 67/3.0/4.8 | **76/4.0/6.8** |
| mandrill+50% | 36/-2.3/-5.7 | 64/2.6/4.2 | 54/1.9/3.3 | 51/2.0/3.4 | 68/2.9/4.6 | 66/2.9/4.6 | **74/3.7/6.3** |
| mandrill+70% | 25/-3.3/-7.0 | 54/1.9/3.2 | 43/1.1/2.2 | 47/1.7/3.1 | 67/2.8/4.2 | 65/2.7/4.4 | **71/3.4/5.4** |
| mandrill+90% | 14/-4.2/-8.1 | 38/0.4/0.7 | 36/0.4/0.8 | 35/0.5/0.9 | 50/1.3/1.4 | 48/1.0/1.1 | **66/2.8/4.3** |
| lake+10% | 70/4.3/3.5 | 83/7.5/11.5 | 83/7.4/11.4 | 82/6.6/11.3 | 89/8.6/13.8 | 84/7.7/12.1 | **97/10.0/17.9** |
| lake+30% | 56/1.7/-0.5 | 80/7.0/10.6 | 74/5.8/9.6 | 80/6.3/10.6 | 88/8.4/13.3 | 83/7.5/11.6 | **94/9.5/16.2** |
| lake+50% | 42/0.1/-2.6 | 73/6.0/9.3 | 45/4.0/7.0 | 77/6.0/9.8 | 88/8.1/11.8 | 82/7.2/11.1 | **92/9.1/15.1** |
| lake+70% | 29/-1.0/-4.0 | 40/2.9/5.1 | 27/2.3/4.0 | 51/4.1/6.8 | 84/7.4/10.8 | 79/6.8/10.3 | **89/8.5/13.5** |
| lake+90% | 15/-2.0/-5.0 | 18/0.7/1.2 | 17/0.7/1.3 | 18/0.9/1.6 | 32/1.4/1.5 | 55/3.8/5.4 | **81/6.8/9.8** |
| jetplane+10% | 76/2.8/0.6 | 88/6.7/9.9 | 89/6.4/9.8 | 66/2.8/6.5 | 93/7.9/12.5 | 89/6.8/10.5 | **98/9.1/16.6** |
| jetplane+30% | 60/-0.2/-3.6 | 86/6.2/8.9 | 79/4.1/7.5 | 66/2.7/6.1 | 93/7.8/11.8 | 88/6.6/9.9 | **97/8.8/15.6** |
| jetplane+50% | 45/-1.9/-5.7 | 81/5.0/7.5 | 44/1.9/4.2 | 51/2.5/5.6 | 91/7.1/10.3 | 87/6.4/9.5 | **95/8.4/14.1** |
| jetplane+70% | 30/-3.1/-7.1 | 39/0.7/2.2 | 25/0.0/1.0 | 32/0.8/2.2 | 89/6.4/8.4 | 85/5.9/8.7 | **93/7.7/12.2** |
| jetplane+90% | 15/-4.1/-8.2 | 16/-1.6/-2.1 | 16/-1.6/-2.0 | 22/-2.0/-3.0 | 30/-1.1/-1.8 | 56/2.0/3.1 | **86/5.8/7.8** |

scratches which can be viewed as impulse noise[6], see Figure 6. We only consider recovering images using $\ell_{02}TV$-AOP, $\ell_0 TV$-PDA and $\ell_0 TV$-PADMM. We show the recovered results in Figure 7. For better visualization of the images recovered by all methods, we also show auxiliary images $\mathbf{c}$ in Figure 8, which show the complement of the absolute residual between the recovered image $\mathbf{u}$ and the corrupted image $\mathbf{b}$ (i.e., $\mathbf{c} = \{\mathbf{1} - |\mathbf{b} - \mathbf{u}|\}$). Note that when $\mathbf{c}_i$ is approximately equal to 1, the color of the corresponding pixel at position $i$ in the image is white. A conclusion can be drawn that our method $\ell_0 TV$-PADMM generates more 'white' images $\mathbf{c}$ than the other two methods, since it can identify the 'right' outliers in the corrupted image and make the correction using their neighborhood information.

6. Note that this is different from the classical image inpainting problem that assumes the mask is known. In our scratched image denoising problem, we assume the mask is unknown.

Figure 6: Sample images in scratched image denoising problems.

Figure 7: Recovered images in scratched image denoising problems. First column: $\ell_{02}TV$-AOP, second column: $\ell_0TV$-PDA, third column: $\ell_0TV$-PADMM.

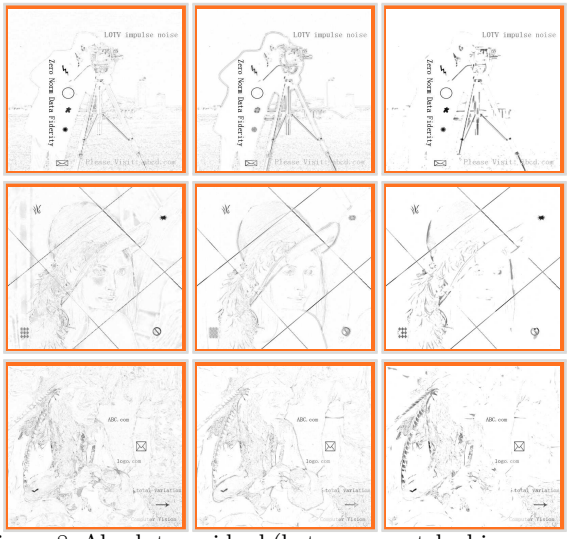


Figure 8: Absolute residual (between scratched image and recovered image) in scratched image denoising problems. First column: $\ell_{02}TV$-AOP, second column: $\ell_0TV$-PDA, third column: $\ell_0TV$-PADMM.

(a) clean 'lenna' (b) corrupted 'lenna' (c) recovered 'lenna'

Figure 9: Colored image denoising problems.

### 5.6 Colored Image Denoising Problems

Our proposed method can be directly extended to its color version. Since color total variation is not the main theme of this paper, we only provide a basic implementation of it. Specifically, we compute the color total variation channel-by-channel, and take a $\ell_1$-norm of the resulting vectors. Suppose we have RGB channels, then we have the following optimization problem:

$$\min_{\mathbf{0}\leq\mathbf{u}^1,\mathbf{u}^2,\mathbf{u}^3\leq\mathbf{1}} \sum_{k=1}^{3}(\|\mathbf{o}^k\odot(\mathbf{K}\mathbf{u}^k-\mathbf{b}^k)\|_0+\lambda\|\boldsymbol{\nabla}\mathbf{u}^k\|_{p,1}),$$

where $\mathbf{o}^k$ and $\mathbf{u}^k$ are the prior and the solution of the $k$th channel. The grayscale proximal ADM algorithm in Algorithm 1 can be directly extended to solve the optimization above. We demonstrate its applicability in colored image denoising problems in Figure 9. The regularization parameter $\lambda$ is set to 8 for the three images in our experiments.

### 5.7 Running Time Comparisons

We provide some running time comparisons for the methods $\ell_1TV$-SBM, TSM, $\ell_pTV$-ADMM, $\ell_{02}TV$-AOP, $\ell_0TV$-PDA, and $\ell_0TV$-PADMM on grayscale image 'cameraman' corrupted by 50% random-value impulse noise. For RGB color images, the running time is three times the amount of grayscale images since the colored image recovery problem can be decomposed into dependent subproblems. Table 5 shows the average CPU time for five runs. Generally, our method is efficient and comparable with existing solutions. This is expected since our method is an alternating optimization algorithm.

Table 5: CPU time (in seconds) comparisons. First row: image denoising; second row: image deblurring.

| $\ell_1TV$-SBM | TSM | $\ell_pTV$-ADMM | $\ell_{02}TV$-AOP | $\ell_0TV$-PDA | $\ell_0TV$-PADMM |
|---|---|---|---|---|---|
| 5±4 | 6±4 | 15±4 | 30±5 | 17±3 | 14±4 |
| 15±8 | 16±7 | 38±8 | 62±4 | 39±7 | 35±8 |

## 6 Conclusions

In this paper, we propose a new method for image restoration based on total variation (TV) with $\ell_0$-norm data fidelity, which is particularly suitable for removing impulse noise. Although the resulting optimization model is non-convex, we design an efficient and effective

proximal ADM method for solving the equivalent MPEC problem of the original $\ell_0$-norm minimization problem. Extensive numerical experiments indicate that the proposed $\ell_0$TV model significantly outperforms the state-of-the-art in the presence of impulse noise. In particular, our proposed proximal ADM solver is more effective than the penalty decomposition algorithm used for solving the $\ell_0$TV problem [37].

**Acknowledgments.** We would like to thank Prof. Shaohua Pan for her helpful discussions on this paper. We also thank Prof. Ming Yan for sharing his code with us. This work was supported by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research and, in part, by the NSF-China (61772570, 61402182).

## References

[1] V. A. A. Tikhonov. *Solution of ill-posed problems*. Winston, Washington, DC, 1977.

[2] M. V. Afonso and J. M. Raposo Sanches. Blind inpainting using and total variation regularization. *IEEE Transactions on Image Processing*, 24(7):2239–2253, 2015.

[3] G. Aubert and J.-F. Aujol. A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4):925–946, 2008.

[4] J.-F. Aujol. Some first-order algorithms for total variation based image restoration. *Journal of Mathematical Imaging and Vision*, 34(3):307–327, 2009.

[5] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.

[6] D. Bienstock. Computational study of a family of mixed-integer quadratic programming problems. *Mathematical programming*, 74(2):121–140, 1996.

[7] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 23(11):1222–1239, 2001.

[8] J.-F. Cai, R. H. Chan, and M. Nikolova. Fast two-phase image deblurring under impulse noise. *Journal of Mathematical Imaging and Vision*, 36(1):46–53, 2010.

[9] J.-F. Cai, B. Dong, S. Osher, and Z. Shen. Image restoration: Total variation, wavelet frames, and beyond. *Journal of the American Mathematical Society*, 25(4):1033–1089, 2012.

[10] E. J. Candès and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

[11] E. J. Candes, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted $\ell_1$ minimization. *Journal of Fourier Analysis and Applications*, 14(5-6):877–905, 2008.

[12] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.

[13] A. B. Chan, N. Vasconcelos, and G. R. G. Lanckriet. Direct convex relaxations of sparse svm. In *International Conference on Machine Learning*, pages 145–153, 2007.

[14] R. H. Chan, C. Ho, and M. Nikolova. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on Image Processing*, 14(10):1479–1485, 2005.

[15] R. H. Chan, C. Hu, and M. Nikolova. An iterative procedure for removing random-valued impulse noise. *IEEE Signal Processing Letters*, 11(12):921–924, 2004.

[16] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.

[17] R. Chartrand and V. Staneva. A quasi-newton method for total variation regularization of images corrupted by non-gaussian noise. *IET Image Processing*, 2:295–303, 2008.

[18] C. Chen, B. He, and X. Yuan. Matrix completion via an alternating direction method. *IMA Journal of Numerical Analysis*, 32(1):227–245, 2011.

[19] D.-Q. Chen, H. Zhang, and L.-Z. Cheng. A fast fixed point algorithm for total variation deblurring and segmentation. *Journal of Mathematical Imaging and Vision*, 43(3):167–179, 2012.

[20] C. Clason. $\ell_\infty$ fitting for inverse problems with uniform noise. *Inverse Problems*, 28(10):104007, 2012.

[21] C. Clason, B. Jin, and K. Kunisch. A duality-based splitting method for $\ell_1$-tv image restoration with automatic regularization parameter choice. *SIAM Journal Scientific Computing*, 32(3):1484–1505, 2010.

[22] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.

[23] A. d'Aspremont. A semidefinite representation for some minimum cardinality problems. In *IEEE Conference on Decision and Control*, volume 5, pages 4985–4990, 2003.

[24] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[25] M. Feng, J. E. Mitchell, J.-S. Pang, X. Shen, and A. Wächter. Complementarity formulations of $\ell$0-norm optimization problems. 2013.

[26] D. Ge, X. Jiang, and Y. Ye. A note on the complexity of $\ell_p$ minimization. *Mathematical Programming*, 129(2):285–299, 2011.

[27] S. GEMAN and D. GEMAN. Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *The IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[28] P. Getreuer. tvreg v2: Variational imaging methods for denoising, deconvolution, inpainting, and segmentation, matlab code: http://www.mathworks.com/matlabcentral/fileexchange/29743. 2010.

[29] D. Goldfarb and W. Yin. Second-order cone programming methods for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 27(2):622–645, 2005.

[30] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[31] B. He and X. Yuan. On the $\mathcal{O}(1/n)$ convergence rate of the douglas-rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2):700–709, 2012.

[32] J. Hu. On linear programs with linear complementarity constraints. pages 1–129, 2008.

[33] H. Ji, S. Huang, Z. Shen, and Y. Xu. Robust video restoration by joint sparse and low rank matrix approximation. *SIAM Journal on Imaging Sciences*, 4(4):1122–1142, 2011.

[34] T. Le, R. Chartrand, and T. J. Asaki. A variational approach to reconstructing images corrupted by poisson noise. *Journal of Mathematical Imaging and Vision*, 27(3):257–263, 2007.

[35] C. Lu, J. Tang, S. Yan, and Z. Lin. Nonconvex nonsmooth low rank minimization via iteratively reweighted nuclear norm. *IEEE Transactions Image Processing*, 25(2):829–839, 2016.

[36] Z. Lu. Iterative reweighted minimization methods for $\ell_p$ regularized unconstrained nonlinear programming. *Mathematical Programming*, 147(1):277–307, 2014.

[37] Z. Lu and Y. Zhang. Sparse approximation via penalty decomposition methods. *SIAM Journal on Optimization*, 23(4):2448–2478, 2013.
[38] Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical programs with equilibrium constraints*. Cambridge University Press, 1996.
[39] A. M. McDonald, M. Pontil, and D. Stamos. Spectral $k$-support norm regularization. In *Neural Information Processing Systems*, pages 3644–3652, 2014.
[40] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.
[41] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, Apr. 1995.
[42] Y. E. Nesterov. *Introductory lectures on convex optimization: a basic course*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, 2003.
[43] M. K. Ng, L. Qi, Y.-F. Yang, and Y.-M. Huang. On semismooth newton's methods for total variation minimization. *Journal of Mathematical Imaging and Vision*, 27(3):265–276, 2007.
[44] M. Nikolova and M. K. Ng. Analysis of half-quadratic minimization methods for signal and image recovery. *SIAM Journal on Scientific Computing*, 27(3):937–966, 2005.
[45] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.
[46] J. Seabra, J. Xavier, and J. Sanches. Convex ultrasound image reconstruction with log-euclidean priors. In *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2008.
[47] G. Steidl and T. Teuber. Removing multiplicative noise by douglas-rachford splitting methods. *Journal of Mathematical Imaging and Vision*, 36(2):168–184, 2010.
[48] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008.
[49] P. Weiss, G. Aubert, and L. Blanc-Féraud. Some application of $\ell_\infty$ constraints in image processing. *INRIA Research Report*, 6115, 2006.
[50] Z. Wen, C. Yang, X. Liu, and S. Marchesini. Alternating direction methods for classical and ptychographic phase retrieval. *Inverse Problems*, 28(11):115010, 2012.
[51] H. Woo and S. Yun. Proximal linearized alternating direction method for multiplicative denoising. *SIAM Journal on Scientific Computing*, 35(2):B336–B358, 2013.
[52] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *Neural Information Processing Systems*, pages 2080–2088, 2009.
[53] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *European Conference on Computer Vision*, pages 157–170. Springer, 2010.
[54] L. Xu, C. Lu, Y. Xu, and J. Jia. Image smoothing via $\ell_0$ gradient minimization. *ACM Transactions on Graphics*, 30(6):174, 2011.
[55] L. Xu, S. Zheng, and J. Jia. Unnatural $\ell_0$ sparse representation for natural image deblurring. In *Computer Vision and Pattern Recognition*, 2013.
[56] Z. Xu, X. Chang, F. Xu, and H. Zhang. $l_{1/2}$ regularization: A thresholding representation theory and a fast solver. *IEEE Transactions on Neural Networks and Learning Systems*, 23(7):1013–1027, 2012.
[57] M. Yan. Restoration of images corrupted by impulse noise and mixed gaussian impulse noise using blind inpainting. *SIAM Journal on Imaging Sciences*, 6(3):1227–1245, 2013.
[58] J. Yang, Y. Zhang, and W. Yin. An efficient tvl1 algorithm for deblurring multichannel images corrupted by impulsive noise. *SIAM Journal on Scientific Computing*, 31(4):2842–2865, 2009.
[59] P. Yin, Y. Lou, Q. He, and J. Xin. Minimization of $\ell_{1-2}$ for compressed sensing. *SIAM Journal on Scientific Computing*, 37(1), 2015.
[60] J. Yu, A. Eriksson, T.-J. Chin, and D. Suter. An adversarial optimization approach to efficient outlier removal. In *International Conference on Computer Vision*, pages 399–406, 2011.
[61] G. Yuan and B. Ghanem. $\ell_0$tv: A new method for image restoration in the presence of impulse noise. In *Computer Vision and Pattern Recognition*, pages 5369–5377, 2015.
[62] G. Yuan and B. Ghanem. Binary optimization via mathematical programming with equilibrium constraints. *arXiv preprint*, 2016.
[63] G. Yuan and B. Ghanem. A proximal alternating direction method for semi-definite rank minimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016.
[64] G. Yuan and B. Ghanem. Sparsity constrained minimization via mathematical programming with equilibrium constraints. *arXiv preprint*, 2016.
[65] G. Yuan and B. Ghanem. An exact penalty method for binary optimization based on mpec formulation. In *AAAI*, pages 2867–2875, 2017.
[66] C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.
[67] X. Zhang, M. Burger, X. Bresson, and S. Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM Journal on Imaging Sciences*, 3(3):253–276, 2010.
[68] W. Zuo and Z. Lin. A generalized accelerated proximal gradient approach for total-variation-based image restoration. *IEEE Transactions on Image Processing*, 20(10):2748–2759, 2011.

**Ganzhao Yuan** was born in Guangdong, China. He received his Ph.D. in School of Computer Science and Engineering, South China University of Technology (SCUT) in 2013. He is currently a research associate professor at School of Data and Computer Science in Sun Yat-sen University (SYSU). His research interests primarily center around large-scale nonlinear optimization and its applications in computer vision and machine learning. He has published papers in ICML, SIGKDD, AAAI, CVPR, VLDB, and ACM Transactions on Database System (TODS).

**Bernard Ghanem** was born in Betroumine, Lebanon. He received his Ph.D. in Electrical and Computer Engineering from the University of Illinois at Urbana-Champaign (UIUC) in 2010. He is currently an assistant professor at King Abdullah University of Science and Technology (KAUST), where he leads the Image and Video Understanding Lab (IVUL). His research interests focus on designing, implementing, and analyzing approaches to address computer vision problems (e.g. object tracking and action recognition/detection in video), especially at large-scale.

# $\ell_0$TV: A Sparse Optimization Method for Impulse Noise Image Restoration (Appendix)

Ganzhao Yuan, Bernard Ghanem

## Appendix A
## Proof of Theorem 1

*Proof.* We define $Z \triangleq (X, Y)$ and denote $I(\cdot)$ as the indicator function on the constrained set $\Delta \triangleq \{z \mid \mathbf{0} \leq \mathbf{z} \leq \mathbf{1}\}$. First of all, we present the first-order KKT conditions of the MPEC reformulation. Based on the augmented Lagrangian function $\mathcal{L}$, we naturally derive the following KKT conditions for $\{\mathbf{u}^*, \mathbf{v}^*, \mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\xi}^*, \boldsymbol{\zeta}^*, \boldsymbol{\pi}^*\}$:

$$\begin{aligned} 0 &\in \boldsymbol{\nabla}^T\boldsymbol{\xi}^* + \mathbf{K}^T\boldsymbol{\zeta}^* + \partial I(\mathbf{u}^*) \\ 0 &\in \boldsymbol{\pi}^* \odot \mathbf{o} \odot |\mathbf{y}^*| - \mathbf{1} + \partial I(\mathbf{v}^*) \\ 0 &\in \partial\lambda\|\mathbf{x}^*\|_{p,1} - \boldsymbol{\xi}^* \\ 0 &\in \boldsymbol{\pi}^* \odot \mathbf{v}^* \odot \mathbf{o} \odot \partial\|\mathbf{y}^*\|_1 - \boldsymbol{\zeta}^* \\ 0 &= \boldsymbol{\nabla}\mathbf{u}^* - \mathbf{x}^* \\ 0 &= \mathbf{K}\mathbf{u}^* - \mathbf{b} - \mathbf{y}^* \\ 0 &= \mathbf{o} \odot \mathbf{v}^* \odot |\mathbf{y}^*|. \end{aligned} \tag{1}$$

Secondly, we prove that the solution is convergent: $Z^{k+1} - Z^k \to 0$. We observe that $\mathcal{L}$ can be rewritten as:

$$\begin{aligned} \mathcal{L}(Z) \triangleq\ & \langle \mathbf{1}, \mathbf{1} - \mathbf{v}\rangle + \lambda\|\mathbf{x}\|_{p,1} + \tfrac{\beta}{2}\|\boldsymbol{\nabla}\mathbf{u} - \mathbf{x} + \boldsymbol{\xi}/\beta\|^2 \\ & - \tfrac{1}{2\beta}\|\boldsymbol{\xi}\|^2 + \frac{\beta}{2}\|\mathbf{K}\mathbf{u} - \mathbf{b} - \mathbf{y} + \boldsymbol{\zeta}/\beta\|^2 - \tfrac{1}{2\beta}\|\boldsymbol{\zeta}\|^2 \\ & + \tfrac{\beta}{2}\|\mathbf{v} \odot \mathbf{o} \odot |\mathbf{y}| + \boldsymbol{\pi}/\beta\|^2 - \tfrac{1}{2\beta}\|\boldsymbol{\pi}\|^2. \end{aligned}$$

Since $Y \triangleq (\boldsymbol{\xi}, \boldsymbol{\zeta}, \boldsymbol{\pi})$ is bounded by assumption, $\mathcal{L}(Z)$ is bounded below for all $Z$. We now define $\mathcal{J}(Z)$ as:

$$\mathcal{J}(Z) = \mathcal{L}(Z) + \tfrac{1}{2}\|\mathbf{u} - \mathbf{u}'\|^2_{\mathbf{D}} + \tfrac{1}{2}\|\mathbf{v} - \mathbf{v}'\|^2_{\mathbf{E}},$$

where $\mathbf{u}'$ and $\mathbf{v}'$ denote the values of $\mathbf{u}$ and $\mathbf{v}$ in the previous iteration. We define $Z^{-1} = Z^0$, and the variable $Z$ in $\mathcal{J}(Z)$ is in the range of $\{Z^0,\ Z^1,\ Z^2, ...\}$. Since $\mathcal{J}(Z)$ is strongly and jointly convex with respect to $\{\mathbf{u}, \mathbf{v}\}$ and $\{\mathbf{u}^{k+1},\ \mathbf{v}^{k+1}\}$ is the minimizer of $\min_{\mathbf{u},\mathbf{v}} \mathcal{J}(\mathbf{u},\ \mathbf{v},\ \mathbf{x}^k,\ \mathbf{y}^k,\ Y^k)$ which is based on $\{\mathbf{u}^k,\ \mathbf{v}^k\}$, using the second order growth condition, we have:

$$\begin{aligned} &\mathcal{J}(\mathbf{u}^k, \mathbf{v}^k, \mathbf{x}^k, \mathbf{y}^k, Y^k) - \mathcal{J}(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}, \mathbf{x}^k, \mathbf{y}^k, Y^k) \\ &\geq \tfrac{\mu}{2}\|\mathbf{u}^k - \mathbf{u}^{k+1}\|^2 + \tfrac{\mu}{2}\|\mathbf{v}^k - \mathbf{v}^{k+1}\|^2. \end{aligned} \tag{2}$$

Using the same methodology for the variable $\mathbf{x}$ and $\mathbf{y}$, we have the following inequalities:

$$\begin{aligned} &\mathcal{J}(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}, \mathbf{x}^k, \mathbf{y}^k, Y^k) \\ &-\mathcal{J}(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}, \mathbf{x}^{k+1}, \mathbf{y}^{k+1}, Y^k) \\ &\geq \tfrac{\beta}{2}\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 + \tfrac{\beta}{2}\|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2. \end{aligned} \tag{3}$$

Denoting $\rho = \frac{1}{2}\min(\mu, \beta)$ and combining (2) and (3), we obtain:

$$\mathcal{J}(X^k, Y^k) - \mathcal{J}(X^{k+1}, Y^k) \geq \rho\|X^k - X^{k+1}\|^2_F. \tag{4}$$

Using the definition of $\mathcal{J}$ and the update rule of the multipliers, we have:

$$\begin{aligned} & \mathcal{J}(X^{k+1}, Y^{k+1}) - \mathcal{J}(X^{k+1}, Y^k) \\ = \ & \langle \boldsymbol{\nabla}\mathbf{u}^{k+1} - \mathbf{x}^{k+1}, \boldsymbol{\xi}^{k+1} - \boldsymbol{\xi}^k\rangle + \\ & \langle \mathbf{K}\mathbf{u}^{k+1} - \mathbf{b} - \mathbf{y}^{k+1}, \boldsymbol{\zeta}^{k+1} - \boldsymbol{\zeta}^k\rangle + \\ & \langle \mathbf{v}^{k+1} \odot \mathbf{o} \odot |\mathbf{y}^{k+1}|, \boldsymbol{\pi}^{k+1} - \boldsymbol{\pi}^k\rangle \\ = \ & \tfrac{1}{\gamma\beta}\|Y^{k+1} - Y^k\|^2. \end{aligned} \tag{5}$$

Combining (4) and (5), we have:

$$\begin{aligned} & \mathcal{J}(X^k, Y^k) - \mathcal{J}(X^{k+1}, Y^{k+1}) \\ \geq \ & \rho\|X^k - X^{k+1}\|^2_F - \tfrac{1}{\gamma\beta}\|Y^k - Y^{k+1}\|^2_F. \end{aligned}$$

Taking summation of the above inequality and using the boundedness of $\mathcal{J}(Z)$, we have that:

$$\begin{aligned} & \textstyle\sum_{k=0}^{\infty}(\rho\|X^k - X^{k+1}\|^2_F - \tfrac{1}{\gamma\beta}\|Y^k - Y^{k+1}\|^2_F) \\ \leq\ & \mathcal{J}(X^0, Y^0) - \mathcal{J}(X^\infty, Y^\infty) < \infty. \end{aligned}$$


- *Ganzhao Yuan (yuanganzhao@gmail.com) is with School of Data and Computer Science, Sun Yat-sen University (SYSU), China.*
- *Bernard Ghanem is with Visual Computing Center, King Abdullah University of Science and Technology (KAUST), Saudi Arabia.*


*Manuscript received April 19, 2005; revised August 26, 2015.*

Since the second term in the inequality above is bounded, i.e. $\sum_{k=0}^{\infty} \lim_{k\to\infty} \|Y^k - Y^{k+1}\|_F^2 = 0$, we obtain that $\sum_{k=0}^{\infty} \lim_{k\to\infty} \|X^k - X^{k+1}\|_F^2 = 0$ and $X^k - X^{k+1} \to 0$.

Finally, we are ready to prove the result of the theorem. By the update rule of $Y^k$, we have:

$$\begin{aligned}
\boldsymbol{\xi}^{k+1} - \boldsymbol{\xi}^k &= \gamma\beta(\boldsymbol{\nabla}\mathbf{u}^k - \mathbf{x}^k) \\
\boldsymbol{\zeta}^{k+1} - \boldsymbol{\zeta}^k &= \gamma\beta(\mathbf{K}\mathbf{u}^k - \mathbf{b} - \mathbf{y}^k) \\
\boldsymbol{\pi}^{k+1} - \boldsymbol{\pi}^k &= \gamma\beta(\mathbf{o} \odot \mathbf{v}^k \odot |\mathbf{y}^k|).
\end{aligned}$$

Using the convergence of $Y$ that $Y^k - Y^{k+1} \to 0$ and the optimality of $X^{k+1}$ with respect to $\mathcal{J}(\cdot)$, we have:

$$\begin{aligned}
0 &= \boldsymbol{\nabla}^T\boldsymbol{\xi}^k + \mathbf{K}^T\boldsymbol{\zeta}^k + \partial I(\mathbf{u}^{k+1}) + \mu(\mathbf{u}^{k+1} - \mathbf{u}^k) \\
0 &= \boldsymbol{\pi}^k \odot \mathbf{o} \odot |\mathbf{y}^k| - \mathbf{1} + \partial I(\mathbf{v}^{k+1}) + \mu(\mathbf{v}^{k+1} - \mathbf{v}^k) \\
0 &\in \partial\lambda\|\mathbf{x}^{k+1}\|_{p,1} - \boldsymbol{\xi}^k \\
0 &\in \boldsymbol{\pi}^k \odot \mathbf{v}^{k+1} \odot \mathbf{o} \odot \partial\|\mathbf{y}^{k+1}\|_1 - \boldsymbol{\zeta}^k.
\end{aligned}$$

Combining the convergence of $X$ that: $X^k - X^{k+1} \to 0$, we have

$$\begin{aligned}
0 &\in \boldsymbol{\nabla}^T\boldsymbol{\xi}^{k+1} + \mathbf{K}^T\boldsymbol{\zeta}^{k+1} + \partial I(\mathbf{u}^{k+1}) \\
0 &\in \boldsymbol{\pi}^{k+1} \odot \mathbf{o} \odot |\mathbf{y}^{k+1}| - \mathbf{1} + \partial I(\mathbf{v}^{k+1}) \\
0 &\in \partial\lambda\|\mathbf{x}^{k+1}\|_{p,1} - \boldsymbol{\xi}^{k+1} \\
0 &\in \boldsymbol{\pi}^{k+1} \odot \mathbf{v}^{k+1} \odot \mathbf{o} \odot \partial\|\mathbf{y}^{k+1}\|_1 - \boldsymbol{\zeta}^{k+1} \\
0 &= \boldsymbol{\nabla}\mathbf{u}^{k+1} - \mathbf{x}^{k+1} \\
0 &= \mathbf{K}\mathbf{u}^{k+1} - \mathbf{b} - \mathbf{y}^{k+1} \\
0 &= \mathbf{o} \odot \mathbf{v}^{k+1} \odot |\mathbf{y}^{k+1}|,
\end{aligned}$$

which coincides with the KKT condition in (1). Therefore, $Z^{k+1}$ asymptotically converges to the KKT point. □