

# Two-Phase Approach for Deblurring Images Corrupted by Impulse Plus Gaussian Noise

Jian-Feng Cai, Raymond H. Chan, and Mila Nikolova

## Abstract

The restoration of blurred images corrupted with impulse noise is a difficult problem which has been considered in a series of recent papers. These papers tackle the problem by using variational methods involving an L1-shaped data-fidelity term. Because of this term, the relevant methods exhibit systematic errors at the corrupted pixel locations and require a cumbersome optimization stage. In this work we propose and justify a much simpler alternative approach which overcomes the above-mentioned systematic errors and leads to much better results. Following a theoretical derivation based on a simple model, we decouple the problem into two phases. First, we identify the outlier candidates—the pixels that are likely to be corrupted by the impulse noise, and we remove them from our data set. In a second phase, the image is deblurred and denoised simultaneously using essentially the outlier-free data. The resultant optimization stage is much simpler in comparison with the current full variational methods and the outlier contamination is more accurately corrected. The experiments show that we obtain a 2 to 6 dB improvement in PSNR. We emphasize that our method can be adapted to deblur images corrupted with mixed impulse plus Gaussian noise, and hence it can address a much wider class of practical problems.

## I. INTRODUCTION

Image deblurring [9] from noisy data is a fundamental problem in image processing. Let the true image  $x$  belong to a proper function space  $\mathbb{S}(\Omega)$  on  $\Omega = [0, 1]^2$ , and the observed digital image  $y$  be a matrix in  $\mathbb{R}^{m \times m}$  indexed by  $\mathcal{A} = \{1, 2, \dots, m\}^2$ . Image deblurring usually is modeled by  $\tilde{y} = Hx + \sigma n$

J.-F. Cai is with Temasek Laboratories and Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543. Email: [tslcai@nus.edu.sg](mailto:tslcai@nus.edu.sg).

R.H. Chan is with Department of Mathematics, The Chinese University of Hong Kong, Shatin, NT, Hong Kong. E-mail: [rchan@math.cuhk.edu.hk](mailto:rchan@math.cuhk.edu.hk). This work was supported by HKRGC Grant CUHK 400405 and CUHK DAG 2060257.

M. Nikolova is with Centre de Mathématiques et de Leurs Applications, ENS de Cachan, 61 av. du Président Wilson, 94235 Cachan Cedex, France. E-mail: [nikolova@cmla.ens-cachan.fr](mailto:nikolova@cmla.ens-cachan.fr).

where  $H : \mathbb{S}(\Omega) \rightarrow \mathbb{R}^{m \times m}$  is a known linear operator that represents blurring and  $\sigma n \in \mathbb{R}^{m \times m}$  is the additive zero-mean Gaussian noise with standard deviation  $\sigma \geq 0$ . In real applications, practical systems can sometimes suffer from few or more pixels, called outliers, which are much noisier than others. Such perturbations are typically caused by malfunctioning arrays in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel, and are modeled as impulse noise. For an overview, see [9]. Taking these into account, a realistic model for the recorded data  $y$  can be modeled as

$$\begin{cases} \tilde{y} = Hx + \sigma n, \\ y = N_p(\tilde{y}), \end{cases} \quad (1)$$

where  $N_p : \mathbb{R}^{m \times m} \rightarrow \mathbb{R}^{m \times m}$  represents outliers which take their values in the dynamic range  $[d_{\min}, d_{\max}]$  of  $\tilde{y}$ , namely  $d_{\min} \leq \tilde{y}_{ij} \leq d_{\max}$  for all  $(i, j)$ . Outliers are usually modeled as either salt-and-pepper or random-valued impulse noise:

- *Salt-and-pepper noise*: the gray level of  $y$  at pixel location  $(i, j)$  is

$$y_{ij} = \begin{cases} d_{\min}, & \text{with probability } s/2, \\ d_{\max}, & \text{with probability } s/2, \\ \tilde{y}_{ij}, & \text{with probability } 1 - s, \end{cases} \quad (2)$$

where  $s$  determines the level of the salt-and-pepper noise.

- *Random-valued noise*: the gray level of  $y$  at pixel location  $(i, j)$  is

$$y_{ij} = \begin{cases} d_{ij}, & \text{with probability } r, \\ \tilde{y}_{ij}, & \text{with probability } 1 - r, \end{cases}$$

where  $d_{ij}$  are uniformly distributed random numbers in  $[d_{\min}, d_{\max}]$  and  $r$  defines the level of the random-valued noise.

Clearly, random-valued impulse noise are more difficult to clean than salt-and-pepper noise since the noise can be arbitrary numbers in  $[d_{\min}, d_{\max}]$ . Examples of images blurred with an out-of-focus kernel of radius 3 and corrupted with different noise patterns are shown in Figure 1. When the blurring operator  $H$  is equal to  $I$ , the identity operator, the problem reduces to a *denoising problem* and a variety of techniques have been proposed to tackle it, see, for example, [7], [10], [11], [18], [19]. In this work, we consider the general case when  $H$  is any smoothing linear operator and the problem is an inverse problem under impulse plus Gaussian noise. We focus on the most common situation when  $H$  is a blurring operator. Let us emphasize that deblurring is a fundamentally harder problem than denoising.

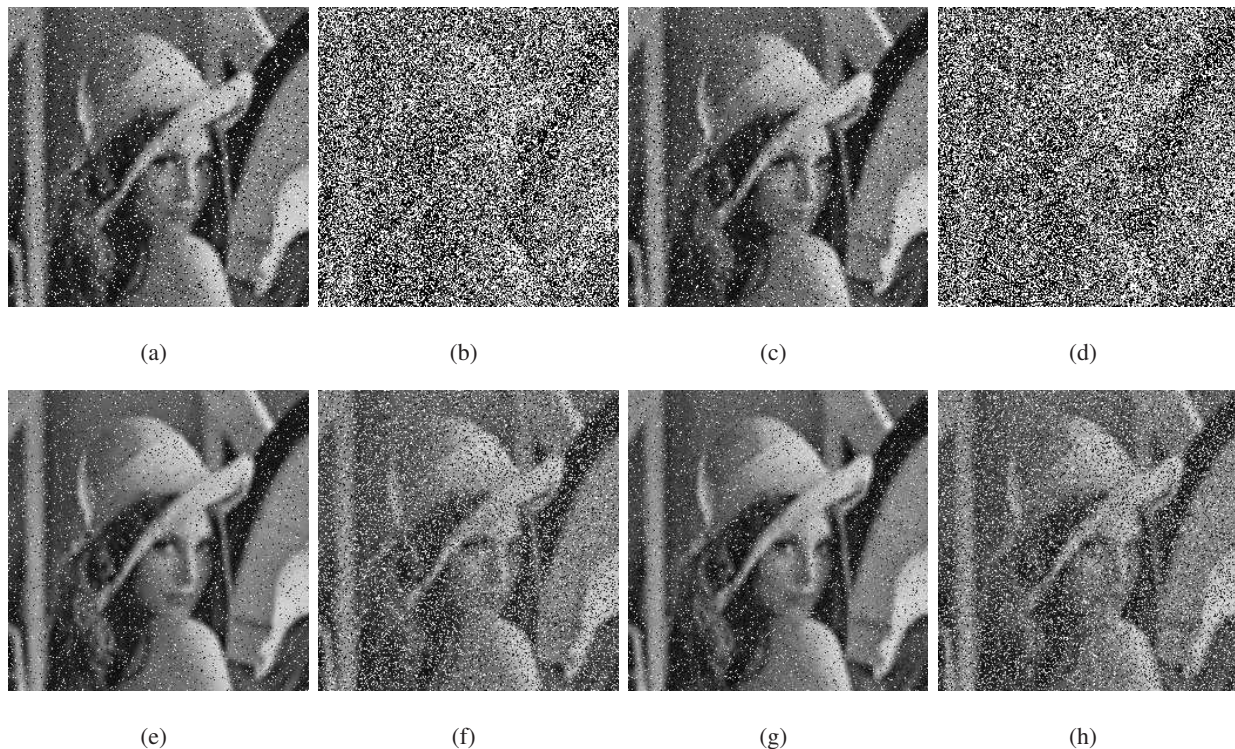


Fig. 1. Lena image blurred by the out-of-focus kernel of radius 3 and contaminated by different noise patterns where  $\sigma$  is the standard deviation of the Gaussian noise, and  $s$  and  $r$  are the levels of the salt-and-pepper and the random-valued noise, respectively. The figures correspond to: (a)  $\sigma = 0$  and  $s = 10\%$ ; (b)  $\sigma = 0$  and  $s = 70\%$ ; (c)  $\sigma = 10$  (SNR=20.8dB) and  $s = 10\%$ ; (d)  $\sigma = 10$  (SNR=20.8dB) and  $s = 70\%$ ; (e)  $\sigma = 0$  and  $r = 10\%$ ; (f)  $\sigma = 0$  and  $r = 40\%$ ; (g)  $\sigma = 10$  (SNR=20.8dB) and  $r = 10\%$ ; (h)  $\sigma = 10$  (SNR=20.8dB) and  $r = 40\%$ .

Blurring smooths the image and thus it entails a loss of high-frequency information. It is well-known that the inverse problem — the inversion of  $H$  — is ill-posed [13], [29], [31]. Since [30], a large variety of regularization methods have been conceived in order to cope with perturbations due to numerical errors and noise. Usually they are based on an  $\ell_2$  data-fitting term which from a statistical point of view means that they are adapted to deal with Gaussian noise. The standard and the current methods used to restore blurred images corrupted by impulse plus Gaussian noise are discussed in Section II.

Our approach is to do the job in two phases. Part of the inspiration comes from the papers [10], [11] where only denoising problems ( $H = I$ ) were considered, even though the present problem is much more complex ( $H \neq I$  and  $\sigma > 0$ ). In the first phase, we locate the data samples which are likely to be corrupted by impulse noise. We call them *outlier candidates*. Such a task can be done easily by comparing the data  $y$  with the output from a properly chosen median-type filter [20], [21] — the data samples modified by the filter are likely to be outliers. Both outliers and their filtered values do not carry

proper information in the sought-after image, so outlier candidates are removed from the data set. In the second phase, we deblur the image based only on the data samples that are *not* outlier candidates, and the deblurring is done by a variational method where the prior information for locally homogeneous images involving sharp edges is introduced using the Mumford-Shah regularization function. We emphasize that what we do here is *totally different* from what was done in [10], [11], where the papers only considered denoising and the denoising was done *only* on the outlier candidates set. Numerical simulations show that our method is 2 to 6 dB better than the full variational deblurring methods in [4], [5], [6], where a functional consisting of a 1-norm data fidelity and the Mumford-Shah regularization term is minimized. Our method outperforms them by at least 2 to 6 dB in PSNR and gives satisfactory results even for noise level as high as  $s = 90\%$  or  $r = 55\%$ .

The rest of the paper is organized as follows. In Section II, we briefly review the existing deblurring methods for solving (1). Our two-phase deblurring approach is presented in Section III. The choice of the data fidelity term and the numerical implementation of our method are discussed in Sections IV and V respectively. Numerical simulation results are presented in Section VI and conclusions are given in Section VII.

## II. CRITICAL REVIEW OF CURRENT METHODS

The main approaches to deblur images corrupted by impulse noise are briefly described below and illustrated by numerical experiments.

### A. Deblurring with no special care to the outliers

One can try to deblur images corrupted by impulse noise by applying classical methods developed for Gaussian noise. These usually amount to defining the restored image as a minimizer of a functional of the form

$$F_y(x) = \|Hx - y\|_2^2 + \beta\Phi(x), \quad (3)$$

where  $\Phi$  is a regularization term and  $\beta > 0$  is the regularization parameter. Frequently,

$$\Phi(x) = \sum_{(i,j) \in \mathcal{A}} \sum_{(k,l) \in \mathcal{V}_{ij}} \varphi(|x_{ij} - x_{kl}|), \quad (4)$$

where  $\mathcal{V}_{ij}$  is the set of the four or the eight closest neighbors of pixel location  $(i, j)$  and  $\varphi$  is an increasing function like those considered e.g. in [12], [22], [28]. Figure 2 shows the restored image obtained by minimizing the discretized formulation of (3) with  $\varphi(t) = \sqrt{t^2 + \alpha}$ , for  $\alpha = 10^{-4}$ . Such  $\varphi(t)$  corresponds to the popular smoothly approximated TV regularization term. Even for very small noise ratio say 1%

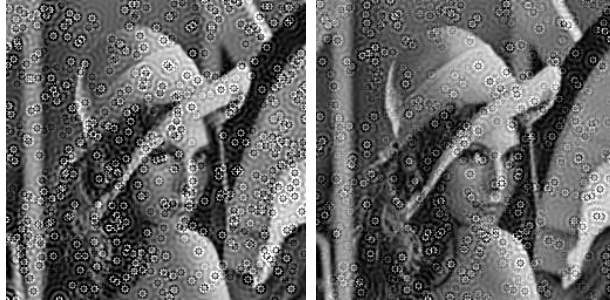


Fig. 2. Lena image blurred with out-of-focus kernel of radius 3, and then corrupted by impulse noise. The image is restored by minimizing (3)–(4) with  $\varphi(t) = \sqrt{t^2 + 10^{-4}}$  and  $\beta = 0.01$ . The left is the restored image when  $s = 1\%$  and the right is when  $r = 1\%$ .

of impulse noise, the method gives very poor results containing numerous spurious concentric rings. Essentially, the impulse noise got deblurred. We note that no improvement can be expected even if one takes the original TV regularization term or any other function  $\varphi$  satisfying  $\varphi'(0) > 0$  since these functions give rise to stair-casing effect.

### B. Outlier smoothing followed by deblurring

Median-type filters are well known to smooth outliers efficiently at a low computational cost [9], [20], [21]. Hence a straightforward deblurring approach is to first restore the outliers using a median-type filter and then to deblure the image using a variational method of the form (3)–(4). This approach is illustrated in Figure 3. The salt-and-pepper noise and the random impulse noise were first smoothed using an adaptive median filter (AMF) [20] and an adaptive center-weighted median filter (ACWMF) [21], respectively, See Figures 3(a) and (c). Then a variational method of the form (3) is applied onto the images to obtain Figures 3(b) and (d). As in Figure 2, spurious circles occur in Figures 3(b) and (d), especially near the edges.

Other approaches for smoothing the impulse noise are the bilateral filter based on ROAD statistic in [15], and the “despike” method in [23]. However, as we will see in Section III-B, whatever filter is used, the restored outliers do not fit the Gaussian noise assumption involved in (3), and hence the variational method (3) will fail.



Fig. 3. Lena image blurred with out-of-focus kernel of radius 3, and then corrupted by impulse noise with  $s = 30\%$  for Figs (a) and (b), and  $r = 25\%$  for Figs (c) and (d). Fig (a) is the result of AMF and Fig (b) is the result by first applying AMF and then minimizing (3)–(4) with  $\varphi(t) = \sqrt{t^2 + 10^{-4}}$  and  $\beta = 0.01$ . Fig (c) is the result of ACWMF and Fig. (d) is the result by first applying ACWMF and then minimizing (3)–(4) with  $\varphi(t) = \sqrt{t^2 + 10^{-4}}$  and  $\beta = 0.01$ .

### C. Simultaneous denoising and deblurring by a variational method

In [4], [5], [6], the authors focus on the restoration of images degraded by blur and impulse noise which corresponds to taking  $\sigma = 0$  in our degradation model (1). To this end, they minimize a functional  $F_y$  of the form

$$F_y(x) = \|Hx - y\|_1 + \beta\Phi(x), \quad (5)$$

where  $\Phi(x)$  is the Mumford-Shah regularization term [22], [1], [3], [14]:

$$\Phi(x) = \int_{\Omega \setminus \Gamma} |\nabla x|^2 + \frac{\alpha}{\beta} \int_{\Gamma} d\sigma, \quad (6)$$

and  $\Gamma$  is the edge set. Using the  $\Gamma$ -convergence functional for  $\Phi$ , see [1], and the smoothly regularized  $L_1$ -norm,  $F_y$  is approximated by

$$F_y(x, w) = \sum_{(i,j) \in \mathcal{A}} \sqrt{[Hx - y]_{ij}^2 + \eta} + \beta \int_{\Omega} w^2 |\nabla x|^2 + \alpha \int_{\Omega} \left( \epsilon |\nabla w|^2 + \frac{(w-1)^2}{4\epsilon} \right), \quad (7)$$

where  $\eta > 0$  and  $\epsilon > 0$  are close to 0. The Euler-Lagrange equation of the above functional is nonlinear. It is solved in [5] by alternate minimization: in each step of the iterative procedure, (7) is minimized with respect to only one of the variables  $x$  or  $w$  with the other being kept fixed. Even though the obtained results are good (see Figures 4 and 8), this full variational approach inevitably involves an intrinsic drawback that we are going to explain below.

#### D. Critical analysis of the method in Section II-C

Ideally, an outlier must be restored using *only* information from its neighbors which are not outliers. For instance, if a region in an image is of constant intensity  $c$  but contains an outlier at pixel location  $(u, v)$ , we naturally require the restored  $\hat{x}_{uv}$  to be equal to  $c$ . This cannot be achieved by any  $F_y$  of the form

$$F_y(x) = \|Hx - y\|_1 + \beta \sum_{(i,j) \in \mathcal{A}} \sum_{(k,l) \in \mathcal{V}_{ij}} \varphi(|x_{ij} - x_{kl}|)$$

with a potential function  $\varphi$  such that  $\varphi'(0) = 0$ . Let us verify this fact when  $H = I$ , i.e. there is no blurring. Suppose that  $\beta$  satisfies  $\beta \geq (4 \max_{t \in \mathbb{R}_+} \varphi'(t))^{-1}$  and that  $\delta > 0$  is the solution to

$$\varphi'(\delta) = 1/(4\beta) \quad \text{with} \quad \varphi''(\delta) > 0. \quad (8)$$

Note that (8) has a unique positive solution under standard assumptions on  $\varphi$ . For example, for the discrete version of the Mumford-Shah regularization [8],

$$\varphi(t) = \begin{cases} t^2 & \text{if } |t| \leq \sqrt{\alpha}, \\ \alpha & \text{otherwise,} \end{cases} \quad (9)$$

we have  $\max_{t \in \mathbb{R}_+} \varphi'(t) = 2\sqrt{\alpha}$  and  $\delta = 1/(8\beta)$ .

Now for simplicity, consider the example where the true image  $x$  is constantly zero and that the observed data  $y$  contains a single outlier at location  $(u, v)$  with magnitude larger than  $\delta$ , that is

$$y_{uv} > \delta \quad \text{and} \quad y_{ij} = x_{ij} = 0, \quad \forall (i, j) \in \mathcal{A} \setminus \{(u, v)\}.$$

According to the full variational method described in Section II-C, the denoised  $\hat{x}$  minimizes  $F_y$  as given in (5) with  $H = I$  and  $\Phi$  of the form (4). Following a derivations similar to [25], we will get

$$\hat{x}_{uv} = \delta > 0, \quad \text{and} \quad \hat{x}_{ij} = 0, \quad \forall (i, j) \in \mathcal{A} \setminus \{(u, v)\}. \quad (10)$$

In other words, the outlier is not removed, and its value is only reduced to  $\delta > 0$ . Notice that this result does not fit the prior since the prior recommends that  $\hat{x}_{uv}$  equals its neighbors. We conclude that the full variational approach cannot restore the outliers correctly in general. We note that although we establish this conclusion for the case  $H = I$ , since deblurring is generally an ill-posed process, these errors can only be amplified in the restoration when  $H \neq I$ .

### III. TWO-PHASE APPROACH

As sketched in the introduction, our approach consists of two phases:

1. Accurate detection of the location of impulse noise (the outlier candidates) using a median-type filter.
2. Edge-preserving restoration that *deblur* and *denoise* simultaneously the data samples which are not outlier candidates.

These phases are explained in details below.

#### A. Outlier detection

Whenever  $H \neq I$ , edges and other high frequency features are smoothed out. Hence they are not as prominent as the outliers in the blurred image  $Hx$ . This suggest that median-type filtering can efficiently detect the locations of the outliers. See [2] for the review of median-type filters. Which median-type filter to choose as an outlier detector depends on the kind of the impulse noise. Based on the experiments in [10], [11], we use the adaptive median filter (AMF) [20] to detect salt-and-pepper noise and the adaptive center-weighted median filter (ACWMF) [21] for random-valued impulse noise. We emphasize that other impulse noise filters, e.g., ROAD statistic [15], can also be used as long as they can provide good outlier detection.

Denote by  $z \in \mathbb{R}^{m \times m}$  the result obtained by applying the median-type filter to the blurred and noisy image  $y$ . As seen already in Figures 3(b) and (d), if we just deblur  $z$ , we will get spurious circles. Instead the filtered data  $z$  will only be used to determine the *outlier candidate set*  $\mathcal{N}$ —the data samples that are likely to be contaminated with impulse noise.

- For salt-and-pepper noise:

$$\mathcal{N} = \{(i, j) \in \mathcal{A} : z_{ij} \neq y_{ij} \text{ and } y_{ij} \in \{d_{\min}, d_{\max}\}\}, \quad (11)$$

- For random-valued impulse noise:

$$\mathcal{N} = \{(i, j) \in \mathcal{A} : z_{ij} \neq y_{ij}\}. \quad (12)$$

Accordingly, the set of data samples that are likely to be uncorrupted with impulse noise is defined as  $\mathcal{U} = \mathcal{A} \setminus \mathcal{N}$ .



### B. Restoration from outlier-free data using a variational method

The example developed in Section II-D clearly shows that outliers should be replaced in accordance with the prior. In fact, the data samples  $y_{ij}$  with  $(i, j) \in \mathcal{N}$  do not carry information of the true image. Their estimates  $z_{ij}$  provided by a median-type filter inevitably only combine information between outliers and their neighbors. They do not carry any information and in addition they contain errors that do not fit the model for Gaussian noise in  $\tilde{y}$  assumed in (1). Hence they can only disrupt the deblurring stage. Indeed, the harmful effect they produce on the solution can be observed in Figures 3(b) and (d). The best we can do is to ignore all  $y_{i,j}$  with  $(i, j) \in \mathcal{N}$  since they are harmful for the subsequent inversion step. The restoration is then done using only the incomplete data set  $y_{ij}$  with  $(i, j) \in \mathcal{U}$ .

These data samples may still contain a few outliers of small amplitude as no median-type filters are perfect impulse noise (outlier) detectors. They may also be corrupted with Gaussian noise if  $\sigma > 0$  in (1). The resultant inverse problem is heavily ill-posed. We solve it by minimizing a functional of the form

$$\sum_{(i,j) \in \mathcal{U}} |[Hx - y]_{ij}|^p + \beta \int_{\Omega \setminus \Gamma} |\nabla x|^2 + \alpha \int_{\Gamma} d\sigma, \quad p = 1, 2. \quad (13)$$

An essential difference with (3) and (5) is that the data-fidelity term here involves only data samples indexed by  $\mathcal{U}$ . The choice of the norm ( $p = 1$  or  $2$ ) will be discussed in the next section. Following [4], [6], we use the Mumford-Shah regularization functional which is well known to produce solutions involving neat edges separating smoothly varying regions. It was shown in [6] that the Mumford-Shah regularizer can be viewed as an extended line process. It reflects spatial organization properties of the image edges that do not appear in the common line process or anisotropic diffusion. This allows one to distinguish outliers from edges and hence leads to superior experimental results. An easy mathematical explanation for discrete images can be found e.g. in [26]. So it describes real-world images better than total variation or other convex regularizers. The price to pay is that the energy in (13) is non-convex and may exhibit numerous local minimizers.

*Remark 1:* Let us notice that the functional (13) is not suited for denoising (when  $H = I$ ). In the case of denoising under impulse noise, *all noisy pixels* must be restored—these are most of the pixels belonging to  $\mathcal{N}$  which is the complement of  $\mathcal{U}$ ! Indeed, in [10], [11], we used for the restoration step a functional of the form

$$\sum_{(i,j) \in \mathcal{N}} |x_{ij} - y_{ij}| + \beta \sum_{(i,j) \in \mathcal{N}} \left( 2 \sum_{(k,l) \in \mathcal{V}_{ij} \cap \mathcal{U}} \varphi(|x_{ij} - y_{kl}|) + \sum_{(k,l) \in \mathcal{V}_{ij} \cap \mathcal{N}} \varphi(|x_{ij} - x_{kl}|) \right). \quad (14)$$

TABLE I  
CHOICE OF THE DATA-FIDELITY IN THE SECOND PHASE.

		<i>Impulse Noise</i>	
		Salt-and-Pepper	Random-valued
<i>Gaussian Noise</i>	$\sigma = 0$	$p = 1$	$p = 1$
	$\sigma > 0$	$p = 2$	$p = 1$ or $p = 2$

The first term in (14) is an  $\ell_1$  norm which helps to retrieve some useful pixels remaining in  $\mathcal{N}$  while the regularization term is restricted only on  $\mathcal{N}$  (observe that the first sum in the regularization has terms like  $(\varphi(|x_{ij} - y_{kl}|))$ ) in order to fit noisy pixels  $x_{ij}$  of  $\mathcal{N}$  to neighboring noise-free samples  $y_{kl} \in \mathcal{U}$ .

In the actual context of deblurring, the second phase is to solve an ill-posed inverse problem in presence of noise, which would be more instable if we include the outliers or some (inevitably) wrong estimates of them. On the other hand, the regularization have to hold on the whole image since  $H$  is non local operator, hence each data sample results from the contribution of a large number of pixels of the underlying image.

#### IV. CHOICE OF THE DATA FIDELITY

We choose  $p = 1$  or  $p = 2$  in (13) depending on the kind of noise remaining in  $\{y_{ij} : (i, j) \in \mathcal{U}\}$ . They are summarized in Table I and are explained in the following.

##### A. Salt-and-pepper noise

The AMF filter is a good detector for salt-and-pepper noise [10], so the data indexed by  $\mathcal{U}$  are almost free of outliers.

1) *Data without Gaussian Noise ( $\sigma = 0$  in (1))*: Almost all data samples in  $\mathcal{U}$  are clean, so we wish to have exact data fitting for them. This can be done by  $\|\cdot\|_1$  because it was shown in [25] that under mild assumptions and a pertinent choice of  $\beta$ , the minimizer  $\hat{x}$  ensures  $(H\hat{x})_{ij} = y_{ij}$  when  $y_{ij}$  is not corrupted while the remaining pixels  $(H\hat{x})_{ij} \neq y_{ij}$  correspond to the regularization term. It is known that  $\|\cdot\|_2^2$  can not do this, see [24] for a detailed explanation. Furthermore,  $\|\cdot\|_1$  is more sensitive to small errors than  $\|\cdot\|_2^2$ , which will lead the data-fitting errors of  $\|\cdot\|_1$  to be smaller than that of  $\|\cdot\|_2^2$ .

2) *Data with Gaussian Noise ( $\sigma > 0$  in (1))*: All pixels in  $\mathcal{U}$  are noisy but the noise is almost Gaussian and white. Therefore the  $\|\cdot\|_2^2$  data fidelity should be used.

### B. Random-valued impulse noise

For the random-valued impulse noise, we use the adaptive center-weighted median filter (ACWMF) to detect the outlier candidates in the first phase. Random-valued impulse noise is much more difficult to discriminate than salt-and-pepper noise. Thus there will still be some outliers undetected and contained in  $\mathcal{U}$  after the detection by ACWMF.

1) *Without Gaussian noise* ( $\sigma = 0$ ): The set  $\mathcal{U}$  contains basically data samples that are free of noise, but also possibly some outliers with values close to those of their neighbors. In order to exactly fit the outlier free data,  $\|\cdot\|_1$  is a better choice due to the property stated above in case A.1. Moreover, it is more insensitive to the exact value of the outliers than  $\|\cdot\|_2^2$ , see [24], [25] for details.

2) *With Gaussian noise* ( $\sigma > 0$ ): Detecting the impulse noise samples in this case is really challenging since small outliers are similar to the Gaussian noise. The set  $\mathcal{U}$  obtained at the output of the ACWMF still contains undetected outliers. Even though they may be negligible, they alter the distribution of the Gaussian noise. On the one hand, the remaining outliers being small amplitude require us to use  $\|\cdot\|_1$  as data fidelity. On the other hand, it is better to use  $\|\cdot\|_2^2$  to handle the Gaussian noise. According to the levels of the impulse noise and the Gaussian noise, better results can be obtained using either  $\|\cdot\|_1$  or  $\|\cdot\|_2^2$ . To guide our choice, we experimentally compare these two data fidelities for an image with edges as shown on the left side of Table II. In Table II, we report the differences in PSNRs (see (20)) between these two data fidelities. As expected,  $\|\cdot\|_1$  is better when  $r$  is big and  $\sigma$  is small, while  $\|\cdot\|_2^2$  is better when  $r$  is small and  $\sigma$  is big. We choose the data fidelity according to the signs in Table II. When the sign is positive we use  $\|\cdot\|_1$ , and  $\|\cdot\|_2^2$  otherwise. However, it still remains an open question to develop a better impulse noise detector as well as a better fidelity for this case. A possible choice is to use  $\|\cdot\|_p^p$  for  $1 < p < 2$  to compromise between  $\|\cdot\|_1$  and  $\|\cdot\|_2^2$ . However, the behavior of a data-fidelity for  $1 < p < 2$  along with regularization is not yet understood in the literature, so we only employ either  $\|\cdot\|_1$  or  $\|\cdot\|_2^2$  as the data fidelity in our experiments.

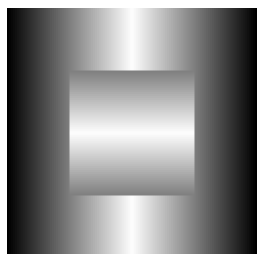
## V. NUMERICAL IMPLEMENTATION

Let  $\chi$  be the characteristic function of the set  $\mathcal{U}$  defined as

$$\chi_{ij} = \begin{cases} 1 & \text{if } (i, j) \in \mathcal{U}, \\ 0 & \text{otherwise,} \end{cases}$$

TABLE II

The left is the image for testing. The right is the table showing the values of  $\text{PSNR}_1 - \text{PSNR}_2$ , where  $\text{PSNR}_i$ ,  $i = 1, 2$  are the PSNR of the restoration by minimizing (13) for  $p = 1$  and  $p = 2$  respectively.

	$\sigma \setminus r$	10%	25%	40%	55%
	1	3.5	6.2	6.9	5.2
	5	-1.2	2.8	2.0	4.3
	10	-0.4	-0.9	1.0	2.1
	15	-0.8	-0.3	1.1	1.9
	20	-1.0	-0.1	-0.1	2.1

and  $\circ$  stand for the Hadamard product (entry-wise product). The functional in (13) can be approximated by

$$\sum_{(i,j) \in \mathcal{A}} \chi_{ij} [Hx - y]_{ij}^2 + \beta \int_{\Omega} w^2 |\nabla x|^2 + \alpha \int_{\Omega} \left( \epsilon |\nabla w|^2 + \frac{(w-1)^2}{4\epsilon} \right), \quad \text{for } p = 2 \quad (15)$$

and

$$\sum_{(i,j) \in \mathcal{A}} \sqrt{\chi_{ij} [Hx - y]_{ij}^2 + \eta} + \beta \int_{\Omega} w^2 |\nabla x|^2 + \alpha \int_{\Omega} \left( \epsilon |\nabla w|^2 + \frac{(w-1)^2}{4\epsilon} \right), \quad \text{for } p = 1 \quad (16)$$

respectively, where  $\eta, \epsilon \gtrsim 0$ . Here the Mumford-Shah functional is approximated by the  $\Gamma$ -convergence. It is explained in [14] that why such an approximation becomes more ideal for image restoration than for the original segmentation task of the Mumford-Shah functional. Moreover, as explained in [6], the Mumford-Shah regularization with  $\Gamma$ -convergence has the theoretical and mathematical advantages of being robust to large gradients (and noise) while preferring structured or smooth edges. However, the alternative edge-preserving stabilizer, for example, the total variation approach, is less robust to outliers. Another advantage of the Mumford-Shah regularization terms with  $\Gamma$ -convergence is that they do not induce nonlinearity. Note in (16), we have smoothed the 1-norm by the parameter  $\eta$ .

The Euler-Lagrange equation of (15) is

$$\begin{cases} 2\beta w |\nabla x|^2 + \alpha \left( \frac{w-1}{2\epsilon} \right) - 2\epsilon \alpha \Delta w = 0, \\ 2H^*(\chi \circ (Hx - y)) - 2\beta \nabla \cdot (w^2 \nabla x) = 0, \end{cases} \quad (17)$$

where  $H^*$  is the adjoint operator of  $H$ . Following the examples of [5], [14], the two equations are solved alternately. For the first equation, we fix  $x$  and solve a linear elliptic equation with respect to  $w$ . For the

second equation, we fix  $w$ , and solve a linear equation with respect to  $x$ . The above process is iterated until convergence. The two equations are discretized by finite difference schemes.

In order to solve the discretized linear equations effectively, preconditioners should be applied. For the first equation in (17), we use the modified incomplete LU (MILU) preconditioner, which is very effective in solving elliptic equations. For the second equation, however, good preconditioner can not be found, since local information (differential operator) and global information (blurring operator) are mixed together. We do not use any preconditioner in the solver of the second equation.

The Euler-Lagrange equation for (16) is

$$\begin{cases} 2\beta w |\nabla x|^2 + \alpha \left( \frac{w-1}{2\epsilon} \right) - 2\epsilon\alpha \Delta w = 0, \\ H^*(\chi \circ W(x) \circ (Hx - y)) - 2\beta \nabla \cdot (w^2 \nabla x) = 0, \end{cases} \quad (18)$$

where

$$[W(x)]_{ij} = \frac{1}{\sqrt{\chi_{ij} [Hx - y]_{ij}^2 + \eta}}.$$

Similarly, the two equations in (18) are solved alternatively. However, the second equation is no longer linear with respect to  $x$  when  $w$  is fixed. Though there are other popular ways such as half-quadratic minimization [16], [17], [27] to linearize this equation, we solve it by a simple fixed point iteration: given  $x^k$ , we get  $x^{k+1}$  by solving

$$H^*(\chi \circ W(x^k) \circ (Hx^{k+1} - y)) - 2\beta \nabla \cdot (w^2 \nabla x^{k+1}) = 0. \quad (19)$$

The equations are again discretized by finite difference schemes. MILU preconditioner is used in solving the first equation in (18), and no preconditioner is used in solving (19).

## VI. SIMULATIONS

In this section, numerical examples are presented to illustrate the effectiveness of our two-phase deblurring method by comparing it with the full variational methods in [4], [5], [6]. The simulations are performed in Matlab 7.01 (R14) on a PC. To assess the restoration performance quantitatively, we evaluate the peak signal to noise ratio [9] defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{n^2} \sum_{(i,j) \in \mathcal{A}} (\hat{x}_{ij} - x_{ij})^2}, \quad (20)$$

where  $\hat{x}_{ij}$  and  $x_{ij}$  are the pixel values of the restored image and of the original image, respectively.

The test images are all 256-by-256 gray level images. Note that there are several parameters to be tuned in both our method and the full variational deblurring method, and we must choose optimal parameters

in order to make the comparisons fair. We have tried our best to determine the optimal parameters. We first fix the 1-norm stabilizer  $\eta$  in (16) to 0.0001. The remaining three parameters  $\alpha, \beta, \epsilon$  are determined by fixing two parameters and adjusting the remaining one such that it gives the best restoration measured in PSNR. The adjusted parameters are tuned one by one. This procedure is repeated several times for each parameter until they become stable. We use the same parameters for different images blurred by the same convolution kernel and corrupted by the same noise level. We note that it is an old and open problem for choosing the optimal parameters, even in the simpler case where there is no impulse noise.

First we discuss the case with salt-and-pepper noise. The comparisons of our method and the full variational deblurring method [4], [5], [6] are shown in Figure 4 and Table III. In the first phase of our method, the outlier candidate set  $\mathcal{N}$ , defined in (11), is detected by the AMF algorithm [20]. The maximum window size we used in AMF is 19 throughout the test. Obviously from Figure 4, our two-phase deblurring method is better than the variational method. In general, the PSNR of the restoration by our method is about 2 to 6 dB higher than that by the variational method, and our two-phase method can handle noise level as high as 90%, while the variational method fails. In Figures 5 and 6, we show the results of our method for other images and for other blurring kernels. We see that our method works well for a wide range of images and blurring kernels. In Figure 7, we show the results when both Gaussian noise and salt-and-pepper noise are presented. We also report  $\tilde{\sigma} = \sqrt{\frac{\sum_{(i,j) \in \mathcal{U}} [H\hat{x} - y]_{ij}^2}{|\mathcal{U}|}}$ , which is supposed to be comparable to  $\sigma$ , the standard deviation of the Gaussian noise. We see that  $\tilde{\sigma}$  is comparable to  $\sigma$  when the impulse noise ratio is low, and increases with the impulse noise ratio. In our experiments the parameters are chosen to optimize the quality of restored images. If we want  $\tilde{\sigma}$  to be close to  $\sigma$ , we may need to use other parameters and the resulting deblurred image will not be as good.

Next we discuss the case of random-valued impulse noise. The outlier is detected by ACWMF [21], which is successively performed four times for every image. The parameters required in ACWMF are chosen to be those used in [11]. Once ACWMF is performed four times, we define the outlier candidate set  $\mathcal{N}$  by (12), and then perform the second phase. Again we compare our two-phase method with the full variational method in [4], [5], [6]. The results are shown in Figure 8 and Table IV. We can see from the figures that our method is again much better than the variational method. The PSNR of the restoration by our method is about 2 to 4 dB higher than that by the variational method. Even for blurred images corrupted by 55% random-valued noise, our method can give a very good restoration, while the variational method fails. In Figures 9 and 10, our method for other images and for other blurring kernels are given. Again our method works well for a wide range of images and blurring kernels. In Figure 11, we give the restoration of blurred images with both random impulse noise and Gaussian noise. We also



Fig. 4. Lena image blurred with out-of-focus kernel of radius 3, and then corrupted by salt-and-pepper noise with noise levels 30%, 50%, 70%, and 90% respectively. *Top*: The restored image by our method, and the parameters we used are  $[\alpha = 0.0002, \beta = 0.0002, \epsilon = 0.001]$ ,  $[\alpha = 0.0002, \beta = 0.0002, \epsilon = 0.0005]$ ,  $[\alpha = 0.0005, \beta = 0.0005, \epsilon = 0.0002]$ ,  $[\alpha = 0.001, \beta = 0.001, \epsilon = 0.0001]$  respectively. *Bottom*: The restored image by the full variational method [4], [5], [6], and the parameters we used are  $[\alpha = 0.005, \beta = 0.002, \epsilon = 0.0002]$ ,  $[\alpha = 0.01, \beta = 0.01, \epsilon = 0.0001]$ ,  $[\alpha = 0.01, \beta = 0.01, \epsilon = 0.00005]$ ,  $[\alpha = 0.01, \beta = 0.01, \epsilon = 0.00005]$  respectively.

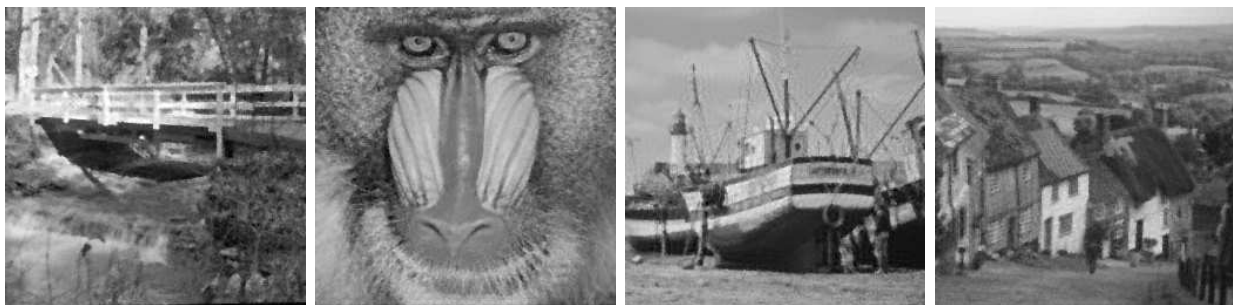


Fig. 5. Restoration of our method for images blurred with out-of-focus kernel of radius 3, and then corrupted by salt-and-pepper noise  $s = 70\%$ . The parameters used are the same as in Figure 4 for  $s = 70\%$ .

TABLE III

The PSNR (dB), computing time (seconds), and the number of iterations of the two-phase method and the full variational method. The blurring kernel is the out-of-focus kernel of radius 3.

Image	$s$	Two-Phase Method				Full Variational Method		
		PSNR	Time		# of iter	PSNR	Time	# of iter
			1st Phase	2nd Phase				
<i>Lena</i>	30%	35.9	0.2	504	3	30.0	629	4
	50%	32.7	0.3	496	3	27.3	721	7
	70%	30.1	0.5	488	3	25.3	625	8
	90%	26.7	10.5	623	4	21.5	730	9
<i>bridge</i>		26.2	0.6	514	3	22.7	716	10
<i>baboon</i>		24.7	0.7	452	3	22.5	409	7
<i>boat</i>	70%	26.7	0.6	488	3	23.4	553	6
<i>goldhill</i>		28.4	0.5	402	3	25.1	558	7



Fig. 6. Lena image blurred with different kernels, and then corrupted by salt-and-pepper noise  $s = 70\%$ . From left to right: the blurred image with Gaussian kernel (generated by MATLAB command `fspecial('Gaussian', [7 7], 1)`) with no noise added yet, and the restored image by our method with PSNR 30.6dB; the blurred image with motion kernel (generated by MATLAB command `fspecial('motion', 9, 1)`) with no noise added yet, and the restored image by our method with PSNR 28.0dB. The parameters used are  $[\alpha = 0.002, \beta = 0.002, \epsilon = 0.001]$ ,  $[\alpha = 0.005, \beta = 0.005, \epsilon = 0.001]$  respectively.

report  $\tilde{\sigma}$  in the figure. Again, we see that  $\tilde{\sigma}$  is comparable to  $\sigma$  when the impulse noise ratio is low, and increases with the impulse noise ratio.

For the computational efficiency, in Tables III and IV we show the computational times and the numbers of iterations for both the proposed two-phase method and the full variational method. We see that in the two-phase method, the second phase consumes majority of the CPU times. Compared to the full variational method, our proposed two-phase method has similar computational efficiency. In fact, for salt-



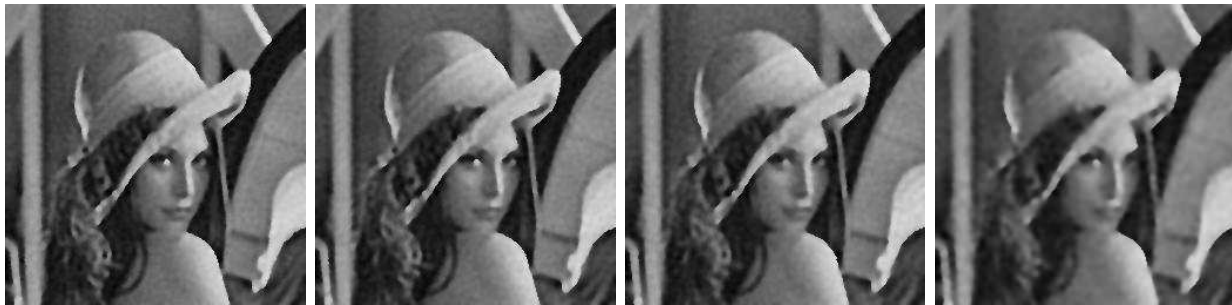


Fig. 7. Lena image blurred with out-of-focus kernel of radius 3, and then corrupted by Gaussian noise with  $\sigma = 5$  (SNR=26.9dB) and salt-and-pepper noise with  $s = 30\%$ ,  $50\%$ ,  $70\%$ , and  $90\%$  respectively. *From left to right*: The restored image by our method with PSNRs 27.2dB, 26.9dB, 26.4dB, and 24.7dB respectively. The parameters we used are all  $[\alpha = 0.05, \beta = 0.05, \epsilon = 0.0002]$ . The  $\tilde{\sigma}$  is 6.5, 6.8, 7.5 and 11.9 respectively.



Fig. 8. Lena image blurred with out-of-focus kernel of radius 3, and then corrupted by random-valued noise with noise levels are 10%, 25%, 40%, and 55% respectively. *Top*: The restored image by our method, and the parameters we used are  $[\alpha = 0.0005, \beta = 0.0005, \epsilon = 0.001]$ ,  $[\alpha = 0.001, \beta = 0.001, \epsilon = 0.0005]$ ,  $[\alpha = 0.002, \beta = 0.002, \epsilon = 0.0005]$ ,  $[\alpha = 0.005, \beta = 0.005, \epsilon = 0.0001]$  respectively. *Bottom*: The restored image by the full variational method, and the parameters we used are  $[\alpha = 0.001, \beta = 0.001, \epsilon = 0.0005]$ ,  $[\alpha = 0.005, \beta = 0.005, \epsilon = 0.0005]$ ,  $[\alpha = 0.005, \beta = 0.005, \epsilon = 0.00005]$ ,  $[\alpha = 0.01, \beta = 0.01, \epsilon = 0.0001]$  respectively.



Fig. 9. The restorations of our method for images blurred with out-of-focus kernel of radius 3, and then corrupted by random-valued impulse noise with  $r = 40\%$ . The parameters used are the same as in Figure 8 when  $s = 40\%$ .

TABLE IV

The PSNR (dB), computing time (second), and the number of iterations of the two-phase method and the full variational method. The blurring kernel is the out-of-focus kernel of radius 3.

Image	$r$	Two-Phase Method				Full Variational Method		
		PSNR	Time		# of iter	PSNR	Time	# of iter
			1st Phase	2nd Phase				
<i>Lena</i>	10%	38.7	7.1	584	3	34.5	625	3
	25%	34.4	7.1	606	3	30.6	854	5
	40%	31.2	7.1	739	4	27.4	684	6
	55%	27.8	7.1	784	6	24.8	861	8
<i>bridge</i>		27.3	7.0	726	4	24.1	619	7
<i>baboon</i>		25.3	7.1	635	4	23.5	478	6
<i>boat</i>	40%	28.2	7.1	709	4	24.7	538	5
<i>goldhill</i>		29.5	7.0	615	4	26.6	539	6

and-pepper noise, the two-phase method is even faster than the full variational method. For both methods, the numbers of iterations are all very small. As we have pointed out in Section V, in each iteration the most time-consuming step lies in the solver for the second equation in (17) or (18). Therefore, in order to improve the computational efficiency, one possible way is to find good preconditioners for the solver.

We note that in all the cases tested, there are no circles appearing in our restored images which are common in other approaches (see Figures 2 and 3). We can also see that in general the two-phase method for salt-and-pepper noise performs better than for random-valued noise: it can handle salt-and-pepper noise as high as 90% but random-valued noise for about 55%. One reason is that the former is more easy to



Fig. 10. Lena image blurred with different kernels, and then corrupted by random-valued impulse noise  $r = 40\%$ . *From left to right*: The restoration of blurred image with Gaussian kernel (generated by MATLAB command `fspecial('Gaussian', [7 7], 1)`), and the PSNR is 31.5dB; the restoration of blurred image with motion kernel (generated by MATLAB command `fspecial('motion', 9, 1)`), and the PSNR is 29.8dB. The parameters used are  $[\alpha = 0.002, \beta = 0.002, \epsilon = 0.0001]$ ,  $[\alpha = 0.002, \beta = 0.002, \epsilon = 0.0002]$  respectively.



Fig. 11. Lena image blurred with out-of-focus kernel of radius 3, and then corrupted by Gaussian noise with  $\sigma = 5$  and random-valued impulse noise with  $r = 10\%, 25\%, 40\%$ , and  $55\%$  respectively. *From left to right*: The restored image by our method with PSNRs 27.2dB, 27.0dB, 26.7dB, and 25.6dB respectively. The parameters we used are  $[\alpha = 0.05, \beta = 0.05, \epsilon = 0.0002]$ ,  $[\alpha = 0.01, \beta = 0.005, \epsilon = 0.0002]$ ,  $[\alpha = 0.01, \beta = 0.005, \epsilon = 0.0002]$ ,  $[\alpha = 0.01, \beta = 0.01, \epsilon = 0.0001]$  respectively. The functional minimized is (13) for  $p = 2$  for  $s = 10\%$  and (13) for  $p = 1$  for others according to Table II. The  $\tilde{\sigma}$  is 6.5, 7.0, 8.4 and 11.9 respectively.

detect than the latter. In fact, for salt-and-pepper noise, most of the noisy pixels are much more dissimilar to the uncorrupted pixels, hence filters like AMF can detect almost all the outlier positions even when the noise ratio is very high. However, there is no good detector for random-valued noise when the noise ratio is high. The performance for random-valued noise can be improved if a better outlier detector can be found. Finally, we remark that for both types of impulse noise, we can use ACWMF as noise detector. However, since AMF can already give a good detection for salt-and-pepper noise, we choose it due to its efficiency when compared with ACWMF; see Tables I and II also the CPU times for the first phase.

## VII. CONCLUSIONS

In this paper, we propose a powerful approach for restoring images blurred and corrupted with Gaussian and impulse noise. Our two-phase method can give excellent restored images, with 2 to 6 dB higher than a full variational method, and can handle extremely high noise level, with  $s = 90\%$  or  $r = 55\%$ . In the case of random-valued impulse noise, works are underway to find better outlier detectors and better data fidelity so as to improve our method further. Also the two-phase deblurring method developed in this paper may be extended to blind convolution and to segmentation under impulse noise plus Gaussian noise based on the nature of Mumford-Shah functional. Furthermore, some theoretical aspects of the proposed method, such as the convergence and convergence rate of the Euler-Lagrange equations, and the preconditioners for the linear systems, can be explored. Another possible extension is deblurring color images under impulse plus Gaussian noise. One difficulty is how to modify the regularizer—the Mumford-Shah functional—for color images. It has been discussed in [4]. With this, the two-phase deblurring method can be extended to color images. Another difficulty is the number of possible noise models for color images, e.g. whether the impulse noise affects one channel or all, and whether the noise level are the same for all channels, etc. These are future research topics.

## REFERENCES

- [1] L. Ambrosio and V.M. Tortorelli, “Approximation of Functionals Depending on Jumps by Elliptic Functionals via  $\Gamma$ -Convergence”, *Communications on Pure and Applied Mathematics*, 43 (1990), pp. 999–1036.
- [2] J. Astola and P. Kuosmanen, *Fundamentals of Nonlinear Digital Filtering*, Boca Rator, CRC, 1997.
- [3] G. Aubert and P. Kornprobst, *Mathematical problems in images processing*, 2002, Springer-Verlag.
- [4] L. Bar, A. Brook, N. Sochen, and N. Kiryati, “Deblurring of Color Images Corrupted by Salt-and-Pepper Noise,” *IEEE Transactions on Image Processing*, 16 (2007), pp. 1101–1111.
- [5] L. Bar, N. Sochen, and N. Kiryati, “Image Deblurring in the Presence of Salt-and-Pepper Noise,” in *Proceeding of 5th International Conference on Scale Space and PDE methods in Computer Vision*, LNCS, vol. 3439, 2005, pp. 107–118.
- [6] L. Bar, N. Sochen, and N. Kiryati, “Image Deblurring in the Presence of Impulsive Noise,” *International Journal of Computer Vision*, 70 (2006), pp. 279–298.
- [7] A. Ben Hamza, H. Krim, “Image denoising: a nonlinear robust statistical approach,” *IEEE Transactions on Signal Processing*, 49 (2001), pp. 3045–3054.
- [8] Blake, A. and Zisserman, A., *Visual reconstruction*, 1987, The MIT Press, Cambridge.
- [9] A. Bovik, *Handbook of Image and Video Processing*, Academic Press, 2000.
- [10] R. H. Chan, C. W. Ho, and M. Nikolova, “Salt-and-Pepper Noise Removal by Median-type Noise Detector and Edge-preserving Regularization,” *IEEE Transactions on Image Processing*, 14 (2005), 1479–1485.
- [11] R. H. Chan, C. Hu, and M. Nikolova, “An Iterative Procedure for Removing Random-Valued Impulse Noise,” *IEEE Signal Processing Letters*, 11 (2004), pp. 921–924.

- [12] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, “Deterministic Edge-Preserving Regularization in Computed Imaging,” *IEEE Transactions on Image Processing*, 6 (1997), pp. 298–311.
- [13] G. Demoment, “Image Reconstruction and Restoration : Overview of Common Estimation Structure and Problems”, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37 (1989), pp. 2024–2036.
- [14] S. Esedoglu, and J. Shen, “Digital Inpainting Based on the Mumford-Shah-Euler Image Model”, *European Journal of Applied Mathematics*, 13 (2002), pp. 353–370.
- [15] R. Garnett, T. Huegerich, C. Chui, and W. He, “A Universal Noise Removal Algorithm with an Impulse Detector,” *IEEE Transactions on Image Processing*, 14 (2005), pp. 1747–1754.
- [16] D. Geman and G. Reynolds, “Constrained restoration and recovery of discontinuities,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14 (1992), pp. 367–383.
- [17] D. Geman and C. Yang, “Nonlinear image recovery with half-quadratic regularization,” *IEEE Transactions on Image Processing*, 4 (1995), pp. 932–946.
- [18] J. G. Gonzalez and G. R. Arce, “Optimality of the myriad filter in practical impulsive-noise environments,” *IEEE Transactions on Signal Processing*, 49 (2001), pp. 438–441.
- [19] R. C. Hardie and K. E. Barner, “Rank conditioned rank selection filters for signal restoration,” *IEEE Transactions on Image Processing*, 3 (1994), pp. 192–206.
- [20] H. Hwang and R. A. Haddad, “Adaptive Median Filters: New Algorithms and Results,” *IEEE Transactions on Image Processing*, 4 (1995), pp. 499–502.
- [21] S.-J. Ko and Y. H. Lee, “Center Weighted Median Filters and their Applications to Image Enhancement,” *IEEE Transactions on Circuits and Systems*, 38 (1991), pp. 984–993.
- [22] D. Mumford and J. Shah, “Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems”, *Communications on Pure and Applied Mathematics*, 42 (1989), pp. 577–684.
- [23] NASA, “Help for DESPIKE, The VICAR Image Processing System,” <http://www-mipl.jpl.nasa.gov/vicar/vicar260/html/vichelp/despikes.html>, 1999.
- [24] M. Nikolova, “Minimizers of Cost-Functions Involving Nonsmooth Data-Fidelity Terms. Application to the Processing of Outliers,” *SIAM Journal on Numerical Analysis*, 40 (2002), pp. 965–994.
- [25] M. Nikolova, “A Variational Approach to Remove Outliers and Impulse Noise,” *Journal of Mathematical Imaging and Vision*, 20 (2004), pp. 99–120.
- [26] M. Nikolova, “Analysis of the recovery of edges in images and signals by minimizing nonconvex regularized least-squares”, *SIAM Journal on Multiscale Modeling and Simulation*, 4(3), 2005, pp.960–991.
- [27] M. Nikolova and R.H. Chan, “The Equivalence of Half-Quadratic Minimization and the Gradient Linearization Iteration,” *IEEE Transactions on Image Processing*, 16 (2007), pp. 1623–1627.
- [28] L. Rudin, S. Osher and E. Fatemi, “Nonlinear Total Variation Based Noise Removal Algorithms”, *Physica D*, 60 (1992), pp. 259–268.
- [29] A. Tarantola, *Inverse problem theory : Methods for data fitting and model parameter estimation*, Elsevier Science Publishers, 1987.
- [30] A. Tikhonov and V. Arsenin, *Solutions of Ill-Posed Problems*, Winston 1977.
- [31] C. Vogel, *Computational Methods for Inverse Problems*, SIAM (Frontiers in Applied Mathematics Series, Number 23), 2002.