

Received July 12, 2020, accepted August 2, 2020, date of publication August 7, 2020, date of current version August 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3015140

L_p -Method-Noise Based Regularization Model for Image Restoration

YINGJUN WANG¹, CHENPING ZHAO^{1,2,3}, HONGWEI JIAO², AND XUDONG WANG⁴

¹School of Information Engineering, Henan Institute of Science and Technology, Xinxiang 453003, China

²School of Mathematical Science, Henan Institute of Science and Technology, Xinxiang 453003, China

³College of Mathematics and Information Science, Henan Normal University, Xinxiang 453007, China

⁴School of Computer and Information Engineering, Nanning Normal University, Nanning 530001, China

Corresponding author: Chenping Zhao (zcp0378@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 11871196, Grant 61941111, and Grant 61772389; in part by the China Postdoctoral Science Foundation under Grant 2019M652545 and Grant 2017M622340; in part by the Key Scientific and Technological Research Projects in Henan Province under Grant 192102210084, Grant 192102210114, and Grant 202102210147; and in part by the Higher School Key Scientific Research Projects of Henan Province under Grant 19A110015 and Grant 18A110019.

ABSTRACT Various regularization techniques have been sufficiently developed to improve the quality of the image restoration. By utilizing existing image smoothing operators, method noise provides a new way to formulate regularization functions. The so-called method noise refers to the difference of an image and its smoothed version, obtained by an image smoothing operator. It is concluded that the method noise of a clear image mainly contains edges and small scaled details, and should be as sparse as possible. Based on this conclusion, we introduce L_p -norm penalty on the method noise, which can accurately describe its sparse prior distribution. We formulate an L_p -method-noise based regularization model and analyze its advantages in terms of its solution and performance in image restoration. Specifically, the L_p -norm penalty of the method noise is better than other forms of norm in removing noise and keeping the details. Moreover, a modified Bregmanized operator splitting algorithm is designed for the proposed model. Experimental results show that the proposed method can obtain better results than other method noise based regularization methods.

INDEX TERMS Image restoration, method noise, regularization model, modified Bregmanized operator splitting.

I. INTRODUCTION

Image restoration, including image denoising, deblurring, inpainting, etc., is one of the most important areas in imaging science. It aims at recovering an image from the degraded version by making use of some prior information such as sparsity, smoothness, and so on [1]–[5]. The degradation model of image restoration is usually given as

$$f = Hu + \eta, \quad (1)$$

where $f \in R^N$ ($N = m \times n$ denotes the size of image) is the observed image, H is some linear operator, u is expected as the ground-truth, and η is i.i.d. white Gaussian noise with variance σ^2 . Due to the fact that image restoration is an ill-posed problem, it needs some regularization indicated by the prior knowledge of the ground-truth, so that image restoration can get a stable solution. The image restoration problem is equivalent to solving

$$\min_u \|Hu - f\|_2^2 + \lambda J(u), \quad (2)$$

The associate editor coordinating the review of this manuscript and approving it for publication was Gerardo Di Martino.

here $J(u)$ is called regularization function, which aims to guarantee some reasonable smooth or some kind of structure of the resulting image [6]–[8].

Well-known selections of the regularization term $J(u)$ include total variation (TV) regularization, non-local TV or some sparse norm in transform domain [9]–[12]. However, each of these regularities exit some kind of short-coming. How to further improve the restore quality, and preserve important structures like edges and details is still challenging. Buades *et al.* used the residual of the true image and its non-local means (NLM) estimation as the regularization term and obtained good results [13]. Specifically, the regularization term is

$$J(u) = \|u - \text{NLM}_f(u)\|_2^2, \quad (3)$$

where $\text{NLM}_f(u)$ denotes the non-local means of u and the weight is computed from the observed image f . Zhang *et al.* [14] proposed Bregmanized operator splitting (BOS) algorithm to solve (2) and obtained convergent results under the assumption that the regularization $J(u)$ is convex. Inspired by BOS algorithm, authors in [15] developed a non-local

means regularization model which utilizes $\|u - \text{NLM}_u(u)\|_2^2$ as regularization term, this regularization term utilizes the ground-truth rather than the observed image to compute the weight. The model was solved by BOS algorithm where a restrictive stopping criterion is used. Authors in [16] suggested a general form of this kind of regularity, pointed out that L₁-norm penalty is better than that of L₂-norm, and formulated the regularization function $J(u) = \|u - D_\tau(u)\|_1$, where $D_\tau(\cdot)$ is some kind of filtering operator. At the same time, Plug-and-Play scheme is developed to improve the image restoration performance [17]–[20]. The main idea is to regard the proximal operator of the regularization function as an image denoiser $D_\tau(u)$, and introduce any off-the-shelf image denoising operator into the image restoration problems.

Generally speaking, $u - D_\tau(u)$ is called method noise, which is defined as the difference of an image and its smoothed version, obtained by an image smoothing operator. Therefore, $u - D_\tau(u)$ mainly contains edges, small scaled details, noise and so on.

In this work, we propose a novel regularization term. Basically, we impose an L_p-norm penalty on the method noise, $\|u - D_\tau(u)\|_p^p$ ($0 < p \leq 1$), to enforce its sparsity. The main contributions of this paper can be summarized as follows:

- 1) The method noise $u - D_\tau(u)$ is considered, which allows other denoiser $D_\tau(u)$, such as TV, block matching and 3D filtering (BM3D) [21], neighborhood filtering (NF) [22] and so on. For example, BM3D outperforms NLM method in most case, and it can restores better results than NLM as denoiser.
- 2) Since that method noise $u - D_\tau(u)$ of a clean image should be as sparse as possible, L_p-norm constraint of $u - D_\tau(u)$ is more reasonable than L₂-norm and L₁-norm, which can accurately depict the sparse prior distribution of image. Therefore, we propose L_p-denoiser based regularization model for image restoration.
- 3) We present a modified BOS algorithm by adding an adaptive adjusting scheme on the parameters, which can force the residual error to decline in whole, and improve the convergence performance of the algorithm. From the point of algorithm structure, the modified BOS algorithm exquisitely combines the Plug-and-Play algorithm with the residual shrinkage method, so it can also be considered as extension version of Plug-and-Play algorithm.

The remainder of the paper is organized as follows. In the next section we review some related work, including Plug-and-Play framework and BOS algorithm. We propose our model and numerical algorithm in Section 3. In section 4 we present experimental results obtained by our method, in which we discuss the key parameters, and evaluate our method by both objective metrics and visual effects. We conclude the paper and present some guidelines for future work in Section 5.

II. RELATED WORK AND DISCUSSION

A. PLUG-AND-PLAY ALGORITHM

The Plug-and-Play (PnP) algorithm is a powerful framework to solve image restoration. For the minimization problem (2), the main idea of PnP based on the alternating direction method of multiplier (ADMM) is as follows [17]. Firstly, the minimization problem (2) can be equivalently converted into a constrained problem

$$\min_{u,v} \|Hu - f\|_2^2 + \lambda J(v) \quad \text{s.t. } u = v, \quad (4)$$

and the corresponding augmented Lagrangian function is

$$L(u, v; w) = \|Hu - f\|_2^2 + \lambda J(v) + \frac{\rho}{2} \|u - v + w\|_2^2, \quad (5)$$

where w is scaled Lagrangian multiplier. The saddle point of $L(u, v; w)$ includes the minimizer of (4), which can be obtained by solving a sequence of sub-problems

$$\begin{cases} u^{(k+1)} = \arg \min_u \|Hu - f\|_2^2 + \frac{\rho}{2} \|u - v + w\|_2^2, & (5) \\ v^{(k+1)} = \arg \min_v \lambda J(v) + \frac{\rho}{2} \|u - v + w\|_2^2, & (6) \\ w^{(k+1)} = w^{(k)} + (u^{(k+1)} - v^{(k+1)}). & (7) \end{cases}$$

It is observed that the sub-problem (6) can be regarded as a denoising step as it involves the prior regularization function $J(v)$. For example, if $J(v) = \|v\|_{\text{TV}}$, the minimization problem in (6) corresponds to the well-known ROF denoising method [2]. Based on this intuition, authors in [17] proposed a variant of above ADMM by suggesting that one does not need to specify $J(v)$ before running ADMM. Instead, they replaced (6) by using an off-the-shelf image denoising algorithm or a filtering, denoted by $D_\tau(\cdot)$, to yield

$$v^{(k+1)} = D_\tau(u^{(k+1)} + w^{(k)}). \quad (8)$$

Examples of $D_\tau(\cdot)$ include a wide variety of patch based approaches such as TV, NLM, BM3D, NF and so on.

B. METHOD NOISE AND BOS ALGORITHM

By introducing the regularization $J(u) = \|u - \text{NLM}_f(u)\|_2^2$, authors in [13] gave NLM based regularization model as

$$\min_u \|u - \text{NLM}_f(u)\|_2^2 \quad \text{s.t. } \|Hu - f\|_2^2 < \sigma^2. \quad (9)$$

Since the observed image f has been blurred and maybe corrupted by noise, the weight computed from it is unreliable. Authors in [14] proposed to improve the above model by computing the weight from the restored image u , that is

$$\min_u \|u - \text{NLM}_u(u)\|_2^2 \quad \text{s.t. } \|Hu - f\|_2^2 < \sigma^2. \quad (10)$$

To solve (10), they introduced Bregmanized operator splitting (BOS) algorithm, which combines the Bregman iteration and operator splitting into a unified framework

$$\begin{cases} v^{(k+1)} = u^{(k)} - \delta H^T (Hu^{(k)} - v^{(k)}), \\ u^{(k+1)} = \arg \min_u \left(\|u - \text{NLM}_{v^{(k+1)}}(v^{(k+1)})\|_2^2 \right. \\ \quad \left. + \lambda \|u - v^{(k+1)}\|_2^2 \right), \\ w^{(k+1)} = w^{(k)} + f - Hu^{(k+1)}. \end{cases}$$

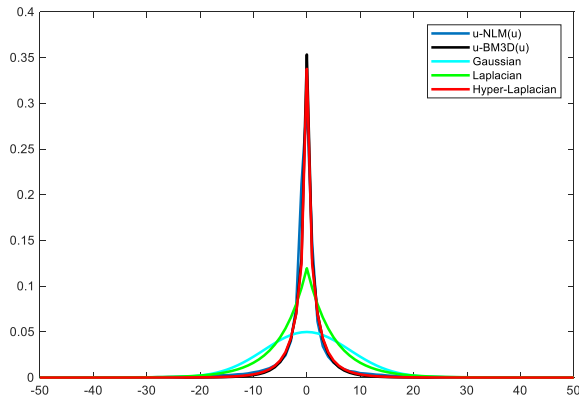


FIGURE 1. Histograms of $u - D_\tau(u)$ with the denoisers: NLM and BM3D, the empirical distribution of Gaussian, Laplacian, and Hyper-Laplacian Magnetization.

Direct extension of problem (10) is replacing the NLM operator with a general image smoothing operator $D_\tau(\cdot)$. Authors in [16] analyzed that the method noise mainly contains edges, small scaled details and noise (if exists), concluded that the L₁-norm penalty of the method noise is better than the L₂-norm penalty. They proposed a weighted L₁-method-noise regularization model, denoted as WL₁-denoiser method,

$$\min_u \|W(u - D_\tau(u))\|_1 \quad \text{s.t.} \quad \|Hu - f\|_2^2 < \sigma^2. \quad (11)$$

III. PROPOSED MODEL AND ALGORITHM

A. THE PROPOSED MODEL

Based on above observation and analysis, it is concluded that $u - D_\tau(u)$ reflects some prior information of the ground-truth. Here, $D_\tau(\cdot)$ can be regarded as the filtering result of some off-the-shelf operator such as TV, NLM, BM3D, and BF and so on. In Fig.1, we plot the histograms of $u - D_\tau(u)$ and the empirical distribution of Gaussian, Laplacian, and Hyper-Laplacian, where $D_\tau(\cdot)$ selected as NLM and BM3D, respectively. It is observed that the method noise of a clean image should be as sparse as possible. The distribution of it is close to Hyper-Laplace distribution [23]–[25], and the better-matched regularization function is L_p-norm ($0 < p < 1$) function. Based on this, we impose L_p-norm on the method noise and propose the sparse regularization model as

$$\min_u \|u - D_\tau(u)\|_p^p \quad \text{s.t.} \quad \|Hu - f\|_2^2 < \sigma^2, \quad (12)$$

where p is an important parameter. In this paper we model it in the range $0 < p \leq 1$, and further discuss its impact on the image restoration quality in the experimental section.

B. THE MINIMIZATION ALGORITHM

By applying BOS technique, the numerical algorithm to solve problem (12) can be formulated as

$$\begin{cases} v^{(k+1)} = u^{(k)} - \delta H^T (Hu^{(k)} - v^{(k)}), & (13) \\ u^{(k+1)} = \arg \min_u \left(\|u - \text{NLM}_{v^{(k+1)}}(v^{(k+1)})\|_p^p \right. \\ \quad \left. + \lambda \|u - v^{(k+1)}\|_2^2 \right), & (14) \\ w^{(k+1)} = w^{(k)} + f - Hu^{(k+1)}. & (15) \end{cases}$$

Algorithm 1 Generalized Soft-Thresholding Algorithm for (17)

Input: Given p, s, x, J , we calculate $y = \text{GST}_s^p(x)$ as following process.

- 1: $\mu_s^p = (2s(1-p))^{2-p} + s \cdot p \cdot (2s(1-p))^{p-1}$,
- 2: if $|x| \leq \mu_s^p(x)$, then $y = 0$,
- 3: if $|x| > \mu_s^p(x)$, then we set $y^{(0)} = |x|$, and calculate y through the following iterative:
- 4: for $k = 0, 1, 2, \dots, J - 1$
- 5: $y^{(k+1)} = |x| - p \cdot s \cdot (y^{(k)})^{p-1}$
- 6: end
- 7: $y = \text{sgn}(x)y^{(J)}$,

Output: y

Algorithm 2 BOS Algorithm for (12)

Input: Choose a group of initial point, and generate new iteration via the following scheme.

- 1: for $k = 0, 1, 2, \dots$, do
- 2: update v :
 $v^{(k+1)} = u^{(k)} - \delta H^T (Hu^{(k)} - w^{(k)})$,
- 3: update u :
 $u^{(k+1)} = D_\tau(v^{(k+1)}) + \text{GST}_{\frac{1}{2\lambda}}^p(v^{(k+1)} - D_\tau(v^{(k+1)}))$,
- 4: update w :
 $w^{(k+1)} = w^{(k)} + f - Hu^{(k+1)}$,
- 5: end the iteration if some stopping criterion is satisfied,

Output: u .

To solve the sub-problem (14), we let $\eta = u - D_\tau(v^{(k+1)})$, and rewrite the sub-problem as

$$\eta^{(k+1)} = \arg \min_\eta \|\eta\|_p^p + \lambda \|\eta - (v^{(k+1)} - D_\tau(v^{(k+1)}))\|_2^2. \quad (16)$$

The minimization problem (16) can be solved by the generalized soft-thresholding method

$$\eta^{(k+1)} = \text{GST}_{\frac{1}{2\lambda}}^p(v^{(k+1)} - D_\tau(v^{(k+1)})), \quad (17)$$

where GST refers to the generalized soft-thresholding operator as expanded in Algorithm 1. Noting that the iteration number in it is empirically selected as $J = 2$ or 3 [24], with which we can obtain satisfactory results. As a result, we can update u as

$$\begin{aligned} u^{(k+1)} &= D_\tau(v^{(k+1)}) + \eta^{(k+1)} \\ &= D_\tau(v^{(k+1)}) + \text{GST}_{\frac{1}{2\lambda}}^p(v^{(k+1)} - D_\tau(v^{(k+1)})). \end{aligned} \quad (18)$$

Combining equation (13), (18) and (15), we formulate the numerical algorithm for (12) in Algorithm 2. Comparing it with PnP, one can observe that, Algorithm 2 add the residual shrinkage $\text{GST}_{\frac{1}{2\lambda}}^p(v^{(k+1)} - D_\tau(v^{(k+1)}))$ back into the restored image, which includes useful edges and details. So it is better than Plug-and-Play algorithm in protecting the edge and detail information.



FIGURE 2. Test images.

C. ADVANTAGE OF THE PROPOSED MODEL

It is noted that, if $p = 2$ and $D_\tau(\cdot)$ is selected as NLM denoiser, the proposed model will be degenerated to L_2 -NLM method (10), which can be solved by using the BOS algorithm.

If $p = 1$, it is observed that, in Algorithm 2, the updating schemes for v and w have no changes, while the updating for u will be degenerated to soft-thresholding method. This just corresponds the solution of the L_1 -denoiser model. So the proposed algorithm is the generalized method of L_1 -denoiser and WL_1 -denoiser method.

Furthermore, $\frac{1}{2\lambda} \rightarrow +\infty$ when $\lambda \rightarrow 0$, the updating of u degenerates into

$$u^{(k+1)} = D_\tau(v^{(k+1)}).$$

This leads to the following iteration

$$\begin{cases} v^{(k+1)} = u^{(k)} - \delta H^T (Hu^{(k)} - w^{(k)}), \\ u^{(k+1)} = D_\tau(v^{(k+1)}), \\ w^{(k+1)} = w^{(k)} + f - Hu^{(k+1)}, \end{cases}$$

which corresponds to PnP algorithm. In this sense, Algorithm 2 can be considered as extension version of the Plug-and-Play algorithm.

Comparing with the L_2 -NLM, WL_1 -denoiser, and PnP algorithm, one can appreciate the advantages of our method over others from the solution process. In the iteration scheme of all methods, $v^{(k+1)}$ is essentially an intermediate image and $u^{(k+1)}$ further refines $v^{(k+1)}$. Different models lead to different refining strategies. Our method improves the quality of $u^{(k+1)}$ in this way: it first applies an image smoothing operator $D_\tau(\cdot)$ to the intermediate blurred image $v^{(k+1)}$, yielding a smoothed clean version $D_\tau(v^{(k+1)})$ and a residual or the method noise $v^{(k+1)} - D_\tau(v^{(k+1)})$, which mainly contains the edges, small scaled details and noise of $v^{(k+1)}$. To restore the edges back while discarding the noise, it applies the

generalized soft-thresholding operator on the method noise, and obtains a refined version $u^{(k+1)}$. The L_1 -method-noise model refines $v^{(k+1)}$ in a similar way, the major difference is that it only corresponds the case $p = 1$ of our method, which is lack of adaptability and cannot always simulate the edges well. The PnP algorithm does not restore edges back. Actually, it refines $v^{(k+1)}$ directly by applying a smoothing operator $D_\tau(\cdot)$, thus may cause over smoothing of the edges. The L_2 -NLM method refines $v^{(k+1)}$ by the weighted average of $v^{(k+1)}$ and its smoothed clean version $NLM(v^{(k+1)})$, which may also cause edges to be over smoothed.

IV. IMPLEMENTATION DETAILS AND EXPERIMENTAL RESULTS

To validate the performance of the proposed model, we test on several standard test images as shown in Fig. 2. We first discuss the parameters and their choosing criterions, then compare the proposed method with several state-of-the-art methods. We utilize the normalized mean square error (NMSE), peak signal-to-noise ratio (PSNR) and structure similarity index (SSIM) as performance measures, which are defined as

$$NMSE = E \left(\frac{\|u - u_0\|_2^2}{\|u_0\|_2^2} \right), \tag{19}$$

$$PSNR = 10 \log_{10} \left(\frac{255^2 N}{\|u - u_0\|_2^2} \right), \tag{20}$$

and

$$SSIM = \frac{2\mu_u \mu_{u_0} (2\sigma_0 + c_2)}{(\mu_u^2 + \mu_{u_0}^2 + c_1) (\sigma_u^2 + \sigma_{u_0}^2 + c_2)}, \tag{21}$$

respectively. Here, u_0 is the original image, u is the restored image, E is the expectation function, μ_u and μ_{u_0} denote there means, σ_u^2 and $\sigma_{u_0}^2$ are their variances, respectively. σ_0 is the covariance of u and u_0 , $c_1 > 0$ and $c_2 > 0$ are constants.

Algorithm 3 Modified BOS Algorithm for (12)

Input: Choose a group of initial point, and generate new iteration via the following scheme.

- 1: for $k = 0, 1, 2, \dots$, do
- 2: update λ_k by formula (22- 23),
- 3: update τ_k and δ_k :

$$\begin{cases} \tau_k = \sqrt{\alpha/\lambda_k}, \\ \delta_k = 1/(\beta\lambda_k), \end{cases}$$
- 4: update v :

$$v^{(k+1)} = u^{(k)} - \delta_k H^T (Hu^{(k)} - w^{(k)}),$$
- 5: update u :

$$u^{(k+1)} = D_{\tau_k}(v^{(k+1)}) + GST_{\frac{1}{2\lambda_k}}^p (v^{(k+1)} - D_{\tau_k}(v^{(k+1)})),$$
- 6: update w :

$$w^{(k+1)} = w^{(k)} + f - Hu^{(k+1)},$$
- 7: end the iteration if some stopping criterion is satisfied,

Output: u .

It is clear that the restoration results is better when NMSE is smaller, while higher PSNR and SSIM imply better quality of the restored image. It is worth noting that, we conduct ten random realizations for each group of experiment. In this case, the reported NMSE/PSNR/SSIM values are their average numbers.

A. CRITERION OF CHOOSING PARAMETERS

To start up the proposed Algorithm 2, some necessary parameters λ , τ , δ and p are needed to be given. p is model parameters, which will be particular analyzed in the following subsection. λ is the regularization parameters to keep balance among the terms in problem (11). τ and δ are the penalty parameters, respectively.

To further better the convergence performance of Algorithm 2, we introduce discriminative increasing scheme for the parameters. Instead of choosing constant parameter, we update λ by $\lambda_{k+1} = \gamma\lambda_k$, and simultaneously, set $\tau_k = \sqrt{\alpha/\lambda_k}$ and $\delta_k = 1/(\beta\lambda_k)$. The adaptive updating rule for λ_k is based on the residual error Δ_k , which is defined as

$$\Delta_{k+1} = \frac{1}{\sqrt{n}} (\|u^{k+1} - u^k\|_2 + \|v^{k+1} - v^k\|_2).$$

By introducing constant $\gamma > 1$ and $0 < \eta < 1$, we conditionally update λ_k according to the followings scheme:

- If $\Delta_{k+1} \geq \eta\Delta_k$, then $\lambda_{k+1} = \gamma\lambda_k$, (22)
- If $\Delta_{k+1} < \eta\Delta_k$, then $\lambda_{k+1} = \lambda_k$. (23)

With above adaptive increasing parameters, Algorithm 2 can be enriched as a modified BOS algorithm, as shown in Algorithm 3. To evaluate the modified scheme, we test the variation tendency of Δ_k in the scenario of image deblurring for Lena, seen as in Fig.3. The results show that the modified scheme in Algorithm 3 avoids the residual error from bouncing too much, and force it to keep decline tendency in whole. To this extent, this can stop the algorithm from falling into some bad local minimum solution.

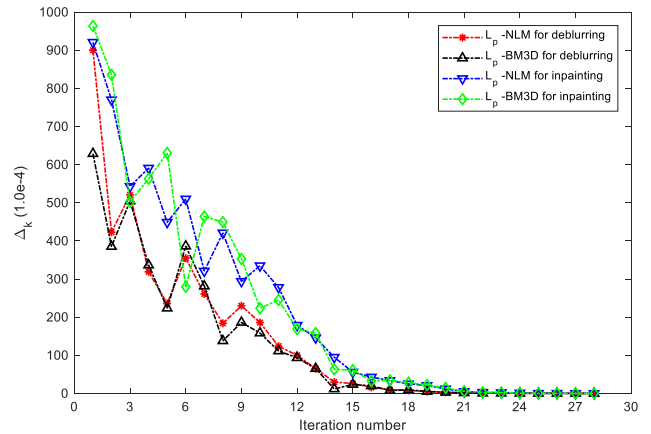


FIGURE 3. Trends of Δ_k by L_p -denoiser with $p = 0.7$.

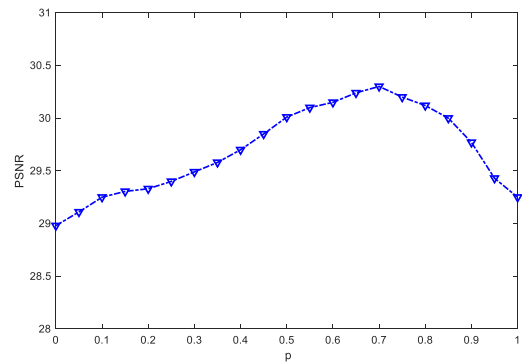


FIGURE 4. Plot of PSNR versus parameter p for image deblurring.

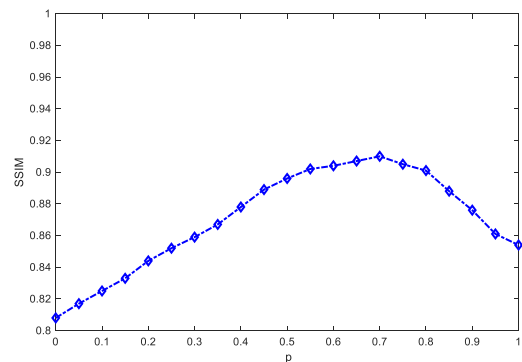


FIGURE 5. Plot of SSIM versus parameter p for image deblurring.

In the following experiments, we need explicitly value four parameters: λ_0 , α , β and γ . A large number of experiments show that our method works well with $\lambda_0 \in (10^{-3}, 10^{-1})$, so we empirically set $\lambda_0 = 10^{-2}$. Other three parameters can be found in each group of deblurring and inpainting experiments.

B. DISCUSSION ON PARAMETER P

Since p is an important parameter in the proposed model, we evaluate its influence on the image restoration performance through plenty of experiments. Taking the deblurring task for the image Lena as example, we report the average PSNRs and SSIMs of the restored images in Fig.4 and Fig.5.

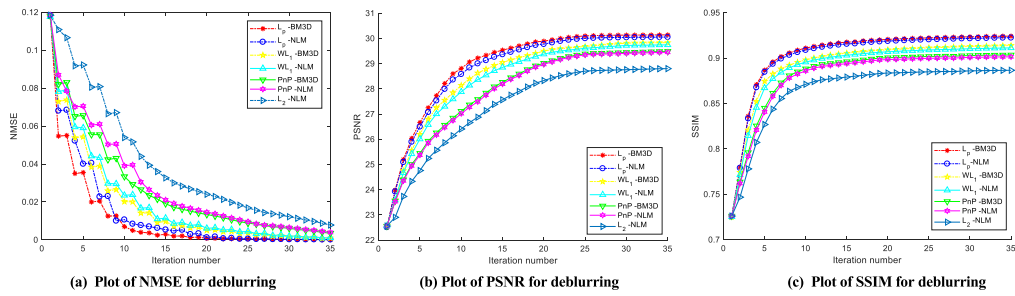


FIGURE 6. Variation trends of NMSE, PSNR and SSIM for image deblurring, where $p = 0.7$ in L_p -NLM and L_p -BM3D.

TABLE 1. Deblurring performance of different methods for average blur.

Average blur	Image	Index	L2-NLM	WL1-NLM	WL1-BM3D	PnP-NLM	PnP-BM3D	L_p -NLM	L_p -BM3D
5×5	Baboon	PSNR	27.13	28.21	28.46	27.93	28.13	28.41	28.56
		SSIM	0.757	0.820	0.833	0.815	0.825	0.843	0.850
	Barbara	PSNR	27.23	28.23	28.40	28.17	28.31	28.62	28.66
		SSIM	0.788	0.803	0.817	0.801	0.813	0.823	0.831
	Boat	PSNR	27.30	27.97	28.05	27.86	27.93	28.22	28.36
		SSIM	0.797	0.823	0.835	0.820	0.828	0.861	0.866
	Building	PSNR	27.60	28.50	28.66	28.47	28.63	28.89	29.02
		SSIM	0.822	0.890	0.911	0.899	0.910	0.921	0.932
	Cameraman	PSNR	28.02	28.97	29.10	28.85	29.02	29.35	29.46
		SSIM	0.851	0.911	0.925	0.910	0.926	0.932	0.943
	Couple	PSNR	27.93	28.76	29.15	28.61	29.02	29.16	29.28
		SSIM	0.822	0.897	0.909	0.895	0.905	0.919	0.927
	Hill	PSNR	27.97	28.78	28.97	28.65	28.78	29.04	29.28
		SSIM	0.834	0.899	0.912	0.893	0.907	0.921	0.932
	House	PSNR	28.54	29.31	29.55	29.24	29.62	29.78	29.79
		SSIM	0.890	0.943	0.954	0.936	0.950	0.966	0.972
Lena	PSNR	28.88	29.53	29.89	29.51	29.87	30.23	30.31	
	SSIM	0.870	0.900	0.917	0.898	0.911	0.931	0.940	
Peppers	PSNR	28.87	29.55	29.90	29.49	29.89	30.19	30.29	
	SSIM	0.871	0.902	0.912	0.892	0.905	0.919	0.927	
7×7	Baboon	PSNR	27.01	27.89	28.07	27.88	28.07	28.32	28.50
		SSIM	0.737	0.811	0.825	0.810	0.822	0.834	0.846
	Barbara	PSNR	26.82	28.15	28.33	28.07	28.25	28.60	28.63
		SSIM	0.777	0.800	0.809	0.803	0.808	0.817	0.829
	Boat	PSNR	27.35	27.82	27.93	27.83	27.88	28.20	28.30
		SSIM	0.790	0.820	0.833	0.818	0.826	0.855	0.861
	Building	PSNR	27.67	28.47	28.63	28.46	28.26	28.85	29.00
		SSIM	0.820	0.887	0.908	0.892	0.910	0.918	0.920
	Cameraman	PSNR	28.00	28.74	28.94	28.64	28.90	29.32	29.45
		SSIM	0.852	0.908	0.923	0.907	0.919	0.930	0.935
	Couple	PSNR	27.92	28.64	28.94	28.60	28.90	29.10	29.15
		SSIM	0.819	0.895	0.906	0.895	0.903	0.914	0.925
	Hill	PSNR	27.87	28.69	28.88	28.60	28.71	29.00	29.20
		SSIM	0.825	0.895	0.910	0.890	0.902	0.920	0.927
	House	PSNR	28.50	29.28	29.40	29.20	29.60	29.72	29.70
		SSIM	0.888	0.940	0.951	0.932	0.951	0.965	0.967
Lena	PSNR	28.85	29.50	29.85	29.50	29.85	30.15	30.26	
	SSIM	0.870	0.897	0.900	0.894	0.903	0.916	0.925	
Peppers	PSNR	28.85	29.52	29.81	29.45	29.85	30.16	30.26	
	SSIM	0.872	0.901	0.910	0.889	0.901	0.914	0.920	

It is observed that the deblurring results perform better with $p \in (0.5, 0.8)$ than other scenarios. Especially, the maximum values of both PSNR and SSIM are approximately

at $p = 0.7$. Without loss of generality, we take $p = 0.7$ as the representative selection of p in the following experimental tests.

TABLE 2. Deblurring performance of different methods for motion blur.

Motion blur	Image	Index	L2-NLM	WL1-NLM	WL1-BM3D	PnP-NLM	PnP-BM3D	L _p -NLM	L _p -BM3D
H_1	Baboon	PSNR	27.32	27.99	28.09	27.89	28.09	28.33	28.50
		SSIM	0.743	0.823	0.825	0.812	0.822	0.842	0.850
	Barbara	PSNR	26.90	28.20	28.42	28.15	28.30	28.64	28.65
		SSIM	0.785	0.803	0.815	0.802	0.811	0.820	0.823
	Boat	PSNR	27.53	27.86	27.90	27.72	27.94	28.20	28.35
		SSIM	0.789	0.825	0.836	0.824	0.833	0.866	0.877
	Building	PSNR	27.59	28.51	28.64	28.49	28.60	28.88	29.03
		SSIM	0.825	0.889	0.910	0.895	0.906	0.916	0.930
	Cameraman	PSNR	28.06	28.70	28.92	28.64	28.87	29.36	29.51
		SSIM	0.850	0.910	0.922	0.912	0.923	0.930	0.940
	Couple	PSNR	28.11	28.52	28.90	28.60	28.70	29.12	29.20
		SSIM	0.817	0.892	0.906	0.890	0.902	0.915	0.925
	Hill	PSNR	27.93	28.75	28.95	28.62	28.79	29.06	29.18
		SSIM	0.832	0.895	0.910	0.890	0.904	0.921	0.930
	House	PSNR	28.55	29.30	29.44	29.25	29.60	29.72	29.73
		SSIM	0.887	0.941	0.947	0.935	0.944	0.961	0.966
	Lena	PSNR	28.80	29.51	29.77	29.50	29.76	30.20	30.22
		SSIM	0.871	0.903	0.908	0.900	0.903	0.916	0.922
	Peppers	PSNR	28.88	29.50	29.80	29.53	29.82	30.10	30.20
		SSIM	0.877	0.906	0.907	0.899	0.905	0.912	0.925
H_2	Baboon	PSNR	26.80	27.79	28.00	27.78	28.01	28.36	28.45
		SSIM	0.735	0.816	0.820	0.811	0.818	0.830	0.834
	Barbara	PSNR	26.89	28.17	28.30	28.11	28.23	28.51	28.59
		SSIM	0.787	0.812	0.820	0.808	0.815	0.820	0.823
	Boat	PSNR	27.30	27.86	27.97	27.80	27.90	28.19	28.28
		SSIM	0.787	0.820	0.840	0.822	0.838	0.852	0.861
	Building	PSNR	27.59	28.42	28.60	28.45	28.61	28.93	29.01
		SSIM	0.826	0.878	0.900	0.893	0.901	0.912	0.919
	Cameraman	PSNR	28.08	28.66	28.88	28.59	28.80	29.12	29.24
		SSIM	0.851	0.889	0.901	0.900	0.902	0.912	0.921
	Couple	PSNR	28.01	28.56	28.85	28.54	28.70	29.01	29.11
		SSIM	0.822	0.890	0.900	0.891	0.895	0.912	0.919
	Hill	PSNR	27.91	28.60	28.81	28.56	28.79	28.93	29.07
		SSIM	0.833	0.890	0.899	0.887	0.901	0.911	0.919
	House	PSNR	28.46	29.01	29.03	29.00	29.06	29.22	29.43
		SSIM	0.878	0.890	0.909	0.910	0.902	0.934	0.938
	Lena	PSNR	28.81	29.00	29.06	29.05	29.16	29.65	29.66
		SSIM	0.876	0.891	0.900	0.894	0.897	0.902	0.917
	Peppers	PSNR	28.83	29.00	29.23	29.40	29.65	30.00	30.16
		SSIM	0.880	0.900	0.903	0.890	0.901	0.911	0.912

C. COMPARISON WITH OTHER REGULARIZATION METHODS

The proposed model can be widely applied in image restoration. In this subsection, we evaluate its performance in two applications: image deblurring and inpainting. For the denoiser, we mainly consider NLM and BM3D. As a result, we compare the proposed method with the following state-of-the-art methods: L₂-NLM, WL₁-NLM, WL₁-BM3D, PnP-NLM, and PnP-BM3D [15]–[17].

1) IMAGE DEBURRING

As for image deblurring, all images are gray-scaled with size 256 × 256. We mainly report experimental results under

two forms of blur: average blur and motion blur. The average blur is simulated by applying box filter of size 5 × 5 and 7 × 7, respectively. The larger the size, the heavier the image is blurred. The motion blur consists of two causes: rotation un-clockwise by an angle θ (in degree) and shifting by L (in pixel). We consider $H_1 : (L, \theta) = (5, 10^\circ)$, and $H_2 : (L, \theta) = (7, 20^\circ)$, respectively. Furthermore, we also add i.i.d. white Gaussian noise in the experiments in order to show the robustness of our method to noise. During the experiment, the stopping criteria is set as $\Delta_{k+1} < 10^{-4}$ or the iteration number $k \geq 50$. By plenty of experiments, we empirically set $\alpha = 10^{-2}$, $\beta = 1$, $\gamma = 1.2$.



FIGURE 7. Restored results from 5×5 average blurred image Lena by different methods.



FIGURE 8. Restored results for motion blurred ($L = 5, \theta = 10^\circ$) image Lena by different methods.

To evaluate the convergence performance of the proposed method, we report the variation trends of NMSE, PSNR and SSIM, respectively. Seen as in Fig.6. From the plots, it can be found that the proposed L_p -denoiser method significantly outperforms several related methods. Specifically, the NMSE

index of L_p -BM3D ($p = 0.7$) decays the most rapidly, also, the PSNR and SSIM of it rise the fastest.

We report the average PSNR and SSIM of the restored images in Table 1 and Table 2, where Table 1 corresponds to the results on average blur, and Table 2 on motion blur.

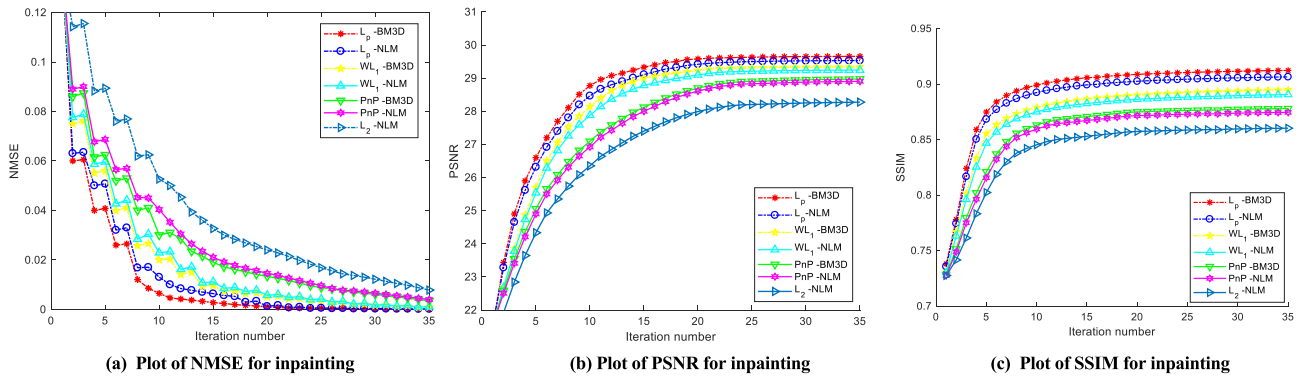


FIGURE 9. Variation trends of NMSE, PSNR and SSIM for image inpainting, where $p = 0.7$ in L_p -NLM and L_p -BM3D.

TABLE 3. Inpainting performance of different methods on 50% random loss.

Image	Index	L2-NLM	WL1-NLM	WL1-BM3D	PnP-NLM	PnP-BM3D	L_p -NLM	L_p -BM3D
Baboon	PSNR	27.56	28.11	28.09	27.99	28.09	28.25	28.49
	SSIM	0.766	0.826	0.822	0.828	0.832	0.842	0.851
Barbara	PSNR	27.62	28.22	28.44	28.18	28.47	28.60	28.62
	SSIM	0.799	0.801	0.817	0.804	0.821	0.826	0.829
Boat	PSNR	27.75	27.87	27.90	27.77	27.85	28.00	28.09
	SSIM	0.791	0.815	0.830	0.821	0.835	0.850	0.853
Building	PSNR	27.98	28.32	28.55	28.41	28.58	28.69	28.89
	SSIM	0.867	0.887	0.895	0.890	0.901	0.909	0.912
Cameraman	PSNR	28.23	28.66	28.88	28.65	28.89	29.07	29.11
	SSIM	0.891	0.900	0.912	0.908	0.916	0.912	0.927
Couple	PSNR	27.99	28.32	28.70	28.40	28.67	28.78	28.90
	SSIM	0.891	0.894	0.901	0.897	0.902	0.911	0.919
Hill	PSNR	27.91	28.35	28.46	28.39	28.54	28.89	29.01
	SSIM	0.855	0.887	0.900	0.890	0.894	0.909	0.915
House	PSNR	28.55	28.91	29.03	29.00	29.06	29.17	29.22
	SSIM	0.889	0.921	0.932	0.929	0.930	0.939	0.943
Lena	PSNR	28.98	29.09	29.21	29.00	29.18	29.32	29.40
	SSIM	0.889	0.903	0.911	0.900	0.909	0.917	0.925
Peppers	PSNR	28.95	29.30	29.45	29.40	29.50	29.61	29.63
	SSIM	0.897	0.906	0.910	0.899	0.905	0.911	0.926

The best results are in bold font. It can be observed that the PSNRs and SSIMs of the proposed method are higher than that of other methods. Taking the average blur 7×7 as an example, in terms of PSNR, our method L_p -BM3D improves L_2 -NLM, WL_1 -NLM, WL_1 -BM3D, PnP-NLM, PnP-NLM around 1.30dB, 0.30dB, 0.55dB, 0.35dB and 0.40dB, respectively. In terms of SSIM, the proposed L_p -BM3D improves these methods around 0.09, 0.05, 0.03, 0.06 and 0.04. These results confirm that, by introducing sparse penalty on the method noise, the proposed model performs better than others in the aspect of recovering useful image information.

For visual assessment, we show the restored images of Lena in Figs.7 for average blur, and Figs.8 for motion blur. The size of the average blur mask is 5×5 , and the parameters of the motion blur are $L = 5, \theta = 10^\circ$. Using either NLM or BM3D as the image smoothing operator, the proposed L_p -denoiser obtains better results than L_2 -NLM,

PnP-denoiser and WL_1 -denoiser. For example, the image Fig.7(h) restored by our method L_p -BM3D is visually much better than the image Fig.7(d) restored by WL_1 -BM3D, and the image Fig.7(f) restored by PnP-BM3D. In the smooth areas of the image in figure Fig.7 (h), one can notice that the noise is removed better than that in figure Fig.7 (d) and Fig.7 (f). Moreover, the brim (marked in yellow box) looks more natural than that obtained by other methods. Similarly, the image Fig.8 (g) and Fig.8 (h) restored better results than that of others.

2) IMAGE INPAINTING

Now we consider the image restoration on an image with missing pixels. For a binary mark H , the indices where the entries are zeros, represent the locations of missing pixels. In this group of experiments, each of an incomplete image is presented in the form of 50% random missing pixels.



FIGURE 10. Restored results for 50% random missing pixels image Lena by different methods.

Parameters are empirically fixed as $\alpha = 0.8 \times 10^{-2}$, $\beta = 1$, and $\gamma = 1.1$.

We report the variation trends of NMSE, PSNR and SSIM, respectively, see as in Fig.9. From the results, it can be found that the NMSE of L_p -denoiser methods, including L_p -NLM L_p -BM3D, decay rapidly than others, and the PSNR and SSIM of them increase faster than others.

Table 3 reports the experimental results on all test images. From the data, it is clear that the L_p -denoiser has superior performance than L_2 -NLM, WL_1 -denoiser, and PnP-denoiser. In terms of PSNR, L_p -BM3D improves L_2 -NLM, WL_1 -NLM, WL_1 -BM3D, PnP-NLM, PnP-BM3D around 1.0dB, 0.35dB, 0.30dB, 0.50dB and 0.40dB, respectively. In terms of SSIM, the proposed L_p -BM3D improves these methods around 0.06, 0.05, 0.02, 0.04 and 0.03.

In Fig.10, we show the restored images of Lena for inpainting. Either using NLM or BM3D as the denoiser, our L_p -denoiser method can obtain better results than L_2 -NLM, PnP-denoiser, and WL_1 -denoiser. For example, the image (Fig.10 (h)) restored by our method L_1 -BM3D is visually much better than the images restored by PnP-BM3D (Fig.10 (d)) and WL_1 -BM3D (Fig.10 (f)). Especially, the brim (marked in yellow box) looks more natural than that of other methods.

V. CONCLUSION

The character of method noise is distinctive. It mainly contains the edge, small scaled details of the image, and the corresponding histogram is sparse. The L_p ($0 < p < 1$) norm can better describe the sparsity. Therefore the proposed model with regularization function $\|u - D_\tau(u)\|_p^p$ is well-motivated

and reasonable though its convexity is unclear. Furthermore, we design a modified BOS algorithm in which parameters are adaptive updated to ensure the downward trend of the residual error. Finally, we evaluate the proposed algorithm by experiments. The results show that the proposed L_p -denoiser method has superior performance and gets better visual effects than L_2 -NLM, L_1 -denoiser and PnP-denoiser algorithm. In our future work, we will strengthen theoretical analysis on the convergence and stability of the algorithm, together with the study on the parameter learning methods.

REFERENCES

- [1] H. Zhang, L. Tang, Z. Fang, C. Xiang, and C. Li, "Nonconvex and nonsmooth total generalized variation model for image restoration," *Signal Process., Off. Publication Eur. Assoc. Signal Process.*, vol. 143, pp. 69–85, Feb. 2018.
- [2] M. Nikolova, "An algorithm for total variation minimization and applications," *J. Math. Imag. Vis.*, vol. 20, nos. 1–2, pp. 89–97, Jan. 2004.
- [3] R. H. Chan, M. Tao, and X. Yuan, "Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers," *SIAM J. Imag. Sci.*, vol. 6, no. 1, pp. 680–697, Jan. 2013.
- [4] C. Guillemot and O. Le Meur, "Image inpainting: Overview and recent advances," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 127–144, Jan. 2014.
- [5] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5505–5514.
- [6] H. Namkoong and C. J. Duchi, "Variance-based regularization with convex objectives," *J. Mach. Learn. Res.*, vol. 20, no. 68, pp. 1–55, 2017.
- [7] C.-P. Zhao, X.-C. Feng, X.-X. Jia, R.-Q. He, and C. Xu, "Root-transformation based multiplicative denoising model and its statistical analysis," *Neurocomputing*, vol. 275, pp. 2666–2680, Jan. 2018.
- [8] I. Selesnick, "Sparse regularization via convex analysis," *IEEE Trans. Signal Process.*, vol. 65, no. 17, pp. 4481–4494, Sep. 2017.

- [9] M. Vauhkonen, D. Vadasz, P. A. Karjalainen, E. Somersalo, and J. P. Kaipio, "Tikhonov regularization and prior information in electrical impedance tomography," *IEEE Trans. Med. Imag.*, vol. 17, no. 2, pp. 285–293, Apr. 1998.
- [10] S. Tang, W. Gong, W. Li, and W. Wang, "Non-blind image deblurring method by local and nonlocal total variation models," *Signal Process.*, vol. 94, pp. 339–349, Jan. 2014.
- [11] L. Xu, S. Zheng, and J. Jia, "Unnatural L_0 sparse representation for natural image deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1107–1114.
- [12] M. Kang, M. Jung, and M. Kang, "Rician denoising and deblurring using sparse representation prior and nonconvex total variation," *J. Vis. Commun. Image Represent.*, vol. 54, pp. 80–99, Jul. 2018.
- [13] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 60–65.
- [14] X. Zhang, M. Burger, X. Bresson, and S. Osher, "Bregmanized nonlocal regularization for deconvolution and sparse reconstruction," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 253–276, 2010.
- [15] Z. M. Wang and H. Bao, "A new regularization model based on non-local means for image deblurring," *Appl. Mech. Mater.*, vols. 411–414, pp. 1164–1169, Sep. 2013.
- [16] C. Yang, W. Wang, X. Feng, and X. Liu, "Weighted- l_1 -method-noise regularization for image deblurring," *Signal Process.*, vol. 157, pp. 14–24, Apr. 2019.
- [17] S. H. Chan, X. Wang, and O. A. Elgandy, "Plug-and-play ADMM for image restoration: Fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, Mar. 2017.
- [18] S. V. Venkatakrisnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2013, pp. 945–948.
- [19] X.-C. Feng, C.-P. Zhao, S.-L. Peng, X.-Y. Hu, and Z.-W. Ouyang, "Plug-and-play based optimization algorithm for new crime density estimation," *J. Comput. Sci. Technol.*, vol. 34, no. 2, pp. 476–493, Mar. 2019.
- [20] Y. Sun, B. Wohlberg, and U. S. Kamilov, "An online plug-and-play algorithm for regularized image reconstruction," *IEEE Trans. Comput. Imag.*, vol. 5, no. 3, pp. 395–408, Sep. 2019.
- [21] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [22] S. Parameswaran, E. Luo, and T. Q. Nguyen, "Patch matching for image denoising using neighborhood-based collaborative filtering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 392–401, Feb. 2018.
- [23] M. Shi, T. Han, and S. Liu, "Total variation image restoration using hyper-Laplacian prior with overlapping group sparsity," *Signal Process.*, vol. 126, pp. 65–76, Sep. 2016.
- [24] W. Zuo, D. Meng, L. Zhang, X. Feng, and D. Zhang, "A generalized iterated shrinkage algorithm for non-convex sparse coding," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 217–224.
- [25] C. Zhao, Y. Wang, H. Jiao, J. Yin, and X. Li, " L_p -norm-based sparse regularization model for license plate deblurring," *IEEE Access*, vol. 8, pp. 22072–22081, 2020.



YINGJUN WANG received the B.S. and M.S. degrees from the School of Computer and Information Engineering, Henan University, Kaifeng, in 2003 and 2006, respectively. He is currently teaching with the School of Information Engineering, Henan Institute of Science and Technology. His research interests include optimization algorithms, computer vision, machine learning, and embedded system design.



CHENPING ZHAO received the B.S. and M.S. degrees from the School of Mathematics and Statistics, Henan University, Kaifeng, in 2003 and 2006, respectively, and the Ph.D. degree in applied mathematics from the School of Mathematics and Statistics, Xidian University, Xi'an, in 2018. She is currently teaching with the School of Mathematical Science, Henan Institute of Science and Technology. Her research interests include variation method and optimization algorithm for image processing.



HONGWEI JIAO received the B.S. degree in mathematics and applied mathematics and the M.S. degree in applied mathematics from Henan Normal University, China, in 2003 and 2006, respectively, and the Ph.D. degree in applied mathematics from Xidian University. He is currently an Associate Professor with the School of Mathematical Science, Henan Institute of Science and Technology, China. He has published extensively and holds numerous patents in his research areas.

His current research interests include optimization algorithms, system engineering, and optimal control.



XUDONG WANG received the B.S. degree in mathematics from Shandong Normal University, China, in 1997, the M.S. degree in applied mathematics from the Guilin University of Electronic Technology, in 2007, and the Ph.D. degree in applied mathematics from Xidian University, China, in 2013. He is currently an Associate Researcher with the School of Computer and Information Engineering, Nanning Normal University, China. His current research interests include image processing and analysis and data processing.

• • •