Simultaneous Fidelity and Regularization Learning for Image Restoration

Dongwei Ren, Wangmeng Zuo, David Zhang, Lei Zhang, and Ming-Hsuan Yang

Abstract—Most existing non-blind restoration methods are based on the assumption that a precise degradation model is known. As the degradation process can only be partially known or inaccurately modeled, images may not be well restored. Rain streak removal and image deconvolution with inaccurate blur kernels are two representative examples of such tasks. For rain streak removal, although an input image can be decomposed into a scene layer and a rain streak layer, there exists no explicit formulation for modeling rain streaks and the composition with scene layer. For blind deconvolution, as estimation error of blur kernel is usually introduced, the subsequent non-blind deconvolution process does not restore the latent image well. In this paper, we propose a principled algorithm within the maximum a posterior framework to tackle image restoration with a partially known or inaccurate degradation model. Specifically, the residual caused by a partially known or inaccurate degradation model is spatially dependent and complexly distributed. With a training set of degraded and ground-truth image pairs, we parameterize and learn the fidelity term for a degradation model in a task-driven manner. Furthermore, the regularization term can also be learned along with the fidelity term, thereby forming a simultaneous fidelity and regularization learning model. Extensive experimental results demonstrate the effectiveness of the proposed model for image deconvolution with inaccurate blur kernels, deconvolution with multiple degradations and rain streak removal.

Index Terms—Image restoration, blind deconvolution, rain streak removal, task-driven learning.

✦

1 INTRODUCTION

IMAGE restoration that aims to recover the latent clean image from a degraded observation is a fundamental problem in low-level vision. However, the degradation generally is irreversible, making image restoration an ill-posed inverse problem. While significant advances have been made in the past decades, it is challenging to develop proper models for various image restoration tasks.

In general, the linear degradation process of a clean image $\mathbf{x}$ can be modeled as

$$\mathbf{y} = \mathcal{A}\mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{n}$ is additive noise, $\mathcal{A}$ is degradation operator, and $\mathbf{y}$ is degraded observation. By changing the settings of the degradation operator and noise type, they can be applied to different image restoration tasks. For example, $\mathcal{A}$ can be an identity matrix for denoising, a blur kernel convolution for deconvolution, and a downsampling operator for super-resolution, to name a few. The maximum a posterior (MAP) model for image restoration can then be formulated as

$$\mathbf{x} = \arg\min_{\mathbf{x}} \frac{\lambda}{2}\|\mathcal{A}\mathbf{x} - \mathbf{y}\|_2^2 + \mathcal{R}(\mathbf{x}), \tag{2}$$

where λ is a trade-off parameter, $\mathcal{R}(\mathbf{x})$ is the regularization term associated with image prior, and the fidelity term is specified by

D. Ren is with the College of Intelligence and Computing, Tianjin University, Tianjin, 300350, China, and also with the Department of Computing, Hong Kong Polytechnic University, Hong Kong. E-mail: rendongweihit@gmail.com
W. Zuo is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China. E-mail: cswmzuo@gmail.com
D. Zhang is with the School of Science and Engineering, Chinese University of Hong Kong (Shenzhen), China. Email: csdzhang@comp.polyu.edu.hk
L. Zhang is with the Department of Computing, Hong Kong Polytechnic University, Hong Kong. E-mail: cslzhang@comp.polyu.edu.hk
M.-H. Yang is with the School of Engineering, University of California, Merced, CA 95344. E-mail: mhyang@ucmerced.edu
(Corresponding author: Wangmeng Zuo)

degradation $\mathcal{A}$ as well as noise $\mathbf{n}$ [1]–[3]. Assuming the noise $\mathbf{n}$ is additive white Gaussian, the fidelity term can be characterized by the ℓ_2-norm.

When the degradation operator $\mathcal{A}$ is precisely known, noise and image prior models play two key roles in the MAP-based image restoration model. Two widely-used types of noise distributions are Gaussian and Poisson. Other distributions, e.g., hyper-Laplacian [4], Gaussian Mixture Model (GMM) [5] and Mixture of Exponential Power (MoEP) [6], are also introduced for modeling complex noise. For image prior, gradient-based models, e.g., total variation [7] and hyper-Laplacian distribution [1], are first studied due to simplicity and efficiency. Subsequently, patch-based [2] and non-local similarity [8], [9] models are developed to characterize more complex and internal dependence among image patches. Recently, data-driven and task-driven learning methods have also been exploited to learn regularization from training images. The approach based on fields of experts (FoE) [10] is designed to learn the distribution of filter responses on images. Following the FoE framework, numerous discriminative learning approaches, e.g., cascaded shrinkage field (CSF) [3], trainable non-linear reaction diffusion (TNRD) [11], [12] and universal denoising network (UNET) [13], use the stage-wise learning scheme to enhance the restoration performance as well as computational efficiency.

However, the precise degradation process for most restoration tasks is not known and thus the degradation process is modeled as

$$\mathbf{y} = \mathcal{A}\mathbf{x} + g(\mathbf{x}; \mathcal{B}) + \mathbf{n}. \tag{3}$$

In the restoration stage, only the model parameter $\mathcal{A}$ is known, while in the form $g(\mathbf{x}; \cdot)$, the noise type $\mathbf{n}$ or the parameters $\mathcal{B}$ are unknown. Here we define this problem as *image restoration with partially known or inaccurate degradation models*.
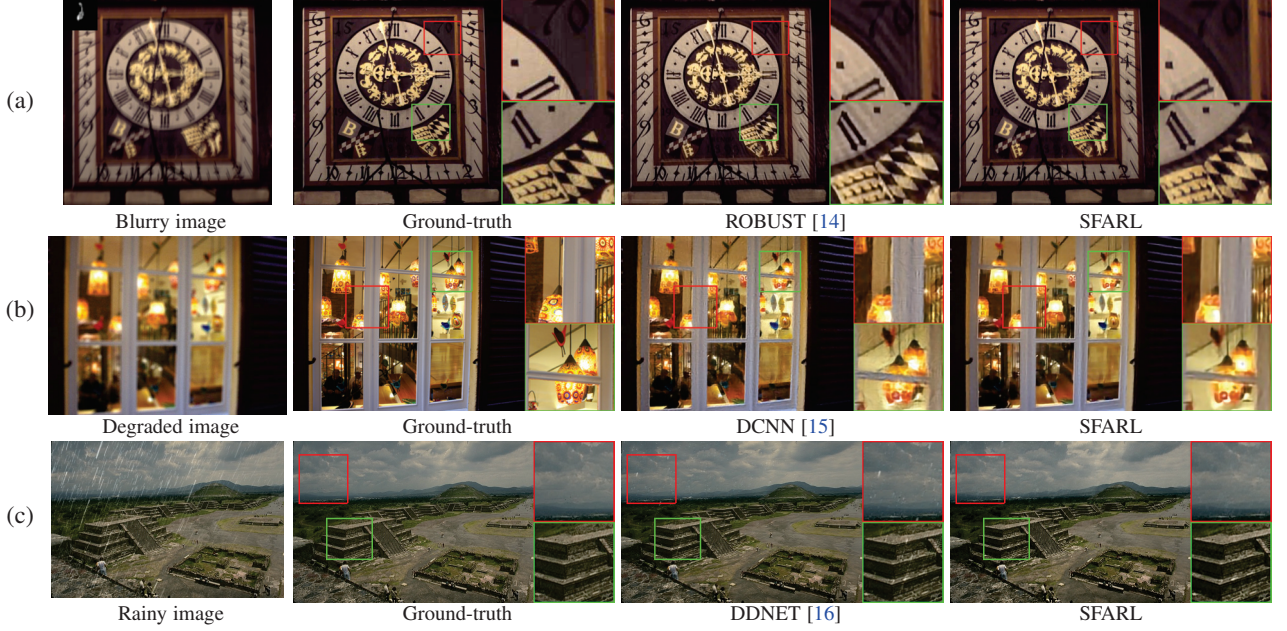
Fig. 1: Illustration of the SFARL model on three restoration tasks. (a) In image deconvolution with inaccurate blur kernels, The SFARL method is effective in relieving the ringing artifacts. (b) For deconvolution along with saturation, Gaussian noise and JPEG compression, the SFARL model can achieve visually plausible result with less noises than DCNN [15]. (c) For rain streak removal, the SFARL model can produce more clean image than DDNET [16].

Image deconvolution with inaccurate blur kernels and rain streak removal are two representative image restoration tasks with partially known or inaccurate degradation models. Image deconvolution with an inaccurate blur kernel is a subproblem of blind deconvolution which generally includes blur kernel estimation and non-blind deconvolution. In the blur kernel estimation stage, the kernel error $\triangle \mathbf{k}$ generally is inevitable to be introduced by a specific method [17]–[22]. In the non-blind deconvolution stage, the degradation model can then be written as

$$\mathbf{y} = \mathbf{k} \otimes \mathbf{x} + \triangle \mathbf{k} \otimes \mathbf{x} + \mathbf{n}, \qquad (4)$$

where $\otimes$ denotes the 2D convolution operator. Thus, the subproblem in the non-blind deconvolution stage is equivalent to *image deconvolution with inaccurate blur kernels*. Based on (3), we have $g(\mathbf{x}; \triangle \mathbf{k}) = \triangle \mathbf{k} \otimes \mathbf{x}$, but $\triangle \mathbf{k}$ is unknown. Existing non-blind deconvolution methods are sensitive to kernel error and usually result in ringing and other artifacts [1], [2], as shown in Figure 1.

For rain streak removal, an input image $\mathbf{y}$ can be represented as the composition of a scene image layer $\mathbf{x}$ and a rain streak layer $\mathbf{x}_r$. However, it remains challenging to model rain streak with any explicit formulation. On one hand, a linear summation $\mathbf{y} = \mathbf{x} + \mathbf{x}_r$ is usually used for combining the scene image and rain streak layers [23], [24]. On the other hand, it has been suggested [25] that a complex model based on screen blend is more effective for combining the scene image and rain streak layers,

$$\mathbf{y} = \mathbf{x} - \mathbf{x} \cdot \mathbf{x}_r + \mathbf{x}_r, \qquad (5)$$

where $\cdot$ denotes the element-wise product. By setting $g(\mathbf{x}; \mathbf{x}_r) = -\mathbf{x} \cdot \mathbf{x}_r$, rain streak removal can be treated as an image restoration problem with a partially known degradation model, i.e., both $g(\mathbf{x}; \mathbf{x}_r)$ and $\mathbf{x}_r$ cannot be explicitly modeled in the deraining stage. As shown in Figure 1, the method [24] is less effective for modeling rainy scenes, resulting in an over-smooth image with visible streaks.

Image restoration with partially known or inaccurate degradation models cannot be simply addressed by noise modeling. From (3), we define the residual image as

$$\mathbf{r} = \mathbf{y} - \mathcal{A}\mathbf{x} = g(\mathbf{x}; \mathcal{B}) + \mathbf{n}. \qquad (6)$$

Due to the introduction of $g(\mathbf{x}; \mathcal{B})$, even $\mathbf{n}$ is white, the residual $\mathbf{r}$ is spatially dependent and complexly distributed. Although several noise models have been suggested for complex noise modeling, these are all based on the independent and identically distributed (i.i.d.) assumption and ineffective for modeling the spatial dependency of the residual. Furthermore, the characteristics of $\mathbf{r}$ is task specific and there exists no universal model that can be applied to all problems, thereby making it more challenging to solve (6).

Recently, deep CNN-based methods have achieved considerable progress on some low level vision tasks [26]–[30], e.g., rain streak removal [16], [31], [32], non-blind deconvolution [15], [33], [34] and Gaussian denosing [35]. These CNN methods, however, either do not take partially known degradations into consideration, or simply address this issue by learning a direct mapping from degraded image to ground-truth. In comparison with CNN-based models, we aim at providing a principled restoration framework for handling partially known or inaccurate degradations.

In this paper, we propose a principled fidelity learning algorithm for image restoration with partially known or inaccurate degradation models. For either kernel error caused by a specific kernel estimation method or rain streaks, the resulting residual $\mathbf{r}$ is not entirely random and can be characterized by spatial dependency and distribution models. Thus, a task-driven scheme is developed to learn the fidelity term from a training set of degraded and ground-truth image pairs. For modeling spatial dependence and complex distribution, the residual $\mathbf{r}$ is characterized by a set of nonlinear penalty functions based on filter responses, leading to a parameterized formulation of the fidelity term. Such a fidelity term

is effective and flexible in modeling complex residual patterns and spatial dependency caused by partially known or inaccurate degradation for a variety of image restoration tasks. Furthermore, for different tasks (e.g., rain streak removal and image deconvolution), the residual patterns are also different. With task-driven learning, the proposed method can adaptively tailor the fidelity term to specific inaccurate or partially known degradation models.

We show that the regularization term can be parameterized and learned along with the fidelity term, resulting in our simultaneous fidelity and regularization learning (SFARL) model. In addition, we characterize the regularizer by a set of nonlinear penalty functions on filters responses of clean image. The SFARL model is formulated as a bi-level optimization problem where a gradient descent scheme is used to solve the inner task and stage-wise parameters are learned from the training data. Experimental results on image deconvolution and rain streak removal demonstrate the effectiveness of the SFARL model in terms of quantitative metrics and visual quality (see Figure 1(a)(b)(c)). Furthermore, for image restoration with precise degradation process, e.g., non-blind Gaussian denoising, the SFARL model can be used to learn the proper fidelity term for optimizing visual perception metrics, and obtain results with better visual quality (see the results in the supplementary material).

In CSF [3], TNRD [12], and UNET [13], similar parametric formulation has been adopted to model natural image prior, and discriminative learning is employed to boost restoration performance. However, the degradation in these methods is assumed as precisely known, and thus the fidelity term is explicitly specified, e.g., ℓ_2-norm for deconvolution with ground-truth kernel. But in practical applications, the degradation process is usually partially known, e.g., inaccurately estimated blur kernel, separation of rain layer and background layer and combination of multiple degradations. In comparison, our SFARL model aims at providing a principled restoration framework, in which fidelity term is flexible and effective to model partially known degradation and can be jointly learned with the regularization terms during training. As a result, when applied to image restoration with partially known or inaccurate degradation models, SFARL can be trained to perform favorably in comparison with TNRD and the state-of-the-arts.

The contributions of this work are summarized as follows:

- We propose a principled algorithm for image restoration with partially known or inaccurate degradation. Give an image restoration task, our model can adaptively learn the proper fidelity term from the training set for modeling the spatial dependency and highly complex distribution of the task-specific residual caused by partially known or inaccurate degradation.
- We present a bi-level optimization model for simultaneous learning of the fidelity term as well as regularization term, and stage-wise model parameters for task-specific image restoration.
- We carry out experiments on rain streak removal, image deconvolution with inaccurate blur kernels and deconvolution with multiple degradations to validate the effectiveness of the SFARL model.

2 RELATED WORK

For specific vision tasks, numerous methods have been proposed for image deconvolution with inaccurate blur kernels and rain streak removal. However, considerably less effort has been made to address image restoration with partially known or inaccurate degradation models. In this section, we review related topics most relevant to this work, including noise modeling, discriminative image restoration, image deconvolution with inaccurate blur kernels, and rain streak removal.

2.1 Noise Modeling

For vision tasks based on robust principal component analysis (RPCA) or low rank matrix factorization (LRMF), noise is often assumed to be sparsely distributed and can be characterized by ℓ_p-norms [4], [36]. However, the noise in real scenarios is usually more complex and cannot be simply modeled using ℓ_p-norms. Consequently, GMM and its variants have been used as universal approximations for modeling complex noise. In RPCA models, Zhao et al. [37] use a GMM model to fit a variety of noise types, such as Gaussian, Laplacian, sparse noise and their combinations. For LRMF, GMM is used to approximate unknown noise, and its effectiveness has been validated in face modeling and structure from motion [5]. In addition, a GMM model is also extended for noise modeling by low rank tensor factorization [38], and generalized to the Mixture of exponential power (MoEP) scheme [6] for modeling complex noise. To determine the parameters of a GMM model, the Dirichlet process has been suggested to estimate the number of Gaussian components under variational Bayesian framework [39]. Recently, the weighted mixture of ℓ_1-norm, ℓ_2-norm [40] and Gaussian [41], [42] models have also been used for blind denoising with unknown noise.

However, noise modeling cannot be readily used to address image restoration with partially known or inaccurate degradation models. The residual $\mathbf{r}$ caused by inaccurate degradation is not i.i.d. Thus, both spatial dependency and complex noise distribution need to be considered to characterize the residual.

2.2 Discriminative Image Restoration

In a MAP-based image restoration model, the regularization term is associated with a statistical prior and assumed to be learned solely based on clean images in a generative manner, e.g., K-SVD [43], GMM [2], and FoE [10]. Recently, discriminative learning has been extensively studied in image restoration. In general, discriminative image restoration aims to learn a fast inference procedure by optimizing an objective function using a training set of the degraded and ground-truth image pairs. One typical discriminative learning approach is to combine existing image prior models with truncated optimization procedures [44], [45]. For example, CSF [3], [46] uses truncated half-quadratic optimization to learn stage-wise model parameters of a modified FoE. On the other hand, TNRD [11], [12] unfolds a fixed number of gradient descent inference steps. Non-parametric methods, such as regression tree fields (RTF) [44], [45] and filter forests [47], are also used for modeling image priors.

Existing discriminative image restoration methods, however, are all based on the precise degradation assumption. These algorithms focus on learning regularization terms in a discriminative framework such that the models can be applied to arbitrary images and blur kernels. In contrast, we propose a discriminative learning algorithm that considers both fidelity and regularization terms, and apply it to image restoration with partially known or inaccurate degradation models.

2.3 Image Deconvolution with Inaccurate Blur Kernels

Typical blind deconvolution approaches consist of two stages: blur kernel estimation and non-blind deconvolution. Existing methods mainly focus on the first stage [18], [19], [22], [48], and considerable attention has been paid to blur kernel estimation. For the second stage, conventional non-blind deconvolution methods usually are used to restore the clean image based on the estimated blur kernels. Despite significant progress has been made in blur kernel estimation, errors are inevitable introduced after the first stage. Furthermore, non-blind deconvolution methods are not robust to kernel errors, and artifacts are likely to be introduced or exacerbated during deconvolution [1], [2].

One intuitive solution is to design specific image priors to suppress artifacts [49]–[52]. To the best of our knowledge, there exists only one attempt [14] to implicitly model kernel error in fidelity term,

$$\mathbf{x} = \arg \min_{\mathbf{x}} \frac{\lambda}{2} \|\mathbf{k} \otimes \mathbf{x} - \mathbf{y} + \mathbf{z}\|^2 + \mathcal{R}(\mathbf{x}) + \tau \|\mathbf{z}\|_1. \quad (7)$$

Here the residual $\mathbf{r}$ is defined as $\mathbf{r} = \mathbf{z} + \mathbf{n}$, where $\mathbf{z}$ is associated with the ℓ_1-norm, and $\mathbf{n}$ is additive white Gaussian noise. However, a method based on $\mathbf{z}$ with the ℓ_1-norm does not model the spatial dependency of residual signals. The method [14] alleviates the effect of kernel errors at the expense of potential over-smoothing restoration results. A recent deep CNN-based approach, i.e., FCN [34], receives multiple inputs with complementary information to produce high quality restoration result. But FCN relies on tuning parameters of non-blind deconvolution method to provide proper network inputs. In this work, we focus on the second stage of blind deconvolution, and propose the SFARL model to characterize the kernel error of a specific kernel estimation method.

2.4 Rain Streak Removal

Rain streak and scene composition models are two important issues for removing rain drops from input images. Based on the linear model $\mathbf{y} = \mathbf{x} + \mathbf{x}_r$, the MAP-based deraining model can be formulated as

$$\mathbf{x} = \arg \min_{\mathbf{x}} \frac{\lambda}{2} \|\mathbf{y} - \mathbf{x} - \mathbf{x}_r\|^2 + \mathcal{R}(\mathbf{x}) + \mathcal{Q}(\mathbf{x}_r)$$
$$\text{s.t. } \forall i, \ 0 \le x_i \le y_i, \ 0 \le x_{ri} \le y_i, \quad (8)$$

where $\mathcal{Q}(\cdot)$ denotes the regularization term of the rain streak layer, and the inequality constraints are introduced to obtain non-negative solutions of $\mathbf{x}$ and $\mathbf{x}_r$ [24].

In [23], hand-crafted regularization is employed to impose smoothness on the image layer and low rank on the rain streak layer. In [24], both image and rain streak layers are modeled as GMMs that are separately trained on clean patches and rain streak patches. Based on the screen blend model, Luo et al. [25] use the discriminative dictionary learning scheme to separate rain streaks by enforcing that two layers need to share fewest dictionary atoms. Recently, specifically designed CNN models [16], [32] have achieved progress in rain streak removal. Instead of using explicit analytic models, the SFARL method is developed based on a data-driven learning approach to accommodate the complexity and diversity of rain streak and scene composition models.

3 PROPOSED ALGORITHM

We consider a class of image restoration problems, where the degradation model is partially known or inaccurate but a training

set of degraded and ground-truth image pairs is available. To handle these problems, we use a flexible model to parameterize the fidelity term caused by partially known or inaccurate degradation. For a given problem, a task-driven learning approach can then be developed to obtain a task-specific fidelity term from training data.

In this section, we first present our method for parameterizing the fidelity term to characterize the spatial dependency and complex distribution of the residual images. In addition, the regularization term is also parameterized, resulting in our simultaneous fidelity and regularization learning model. Finally, we propose a task-driven manner to learn the proposed model from training data.

3.1 Fidelity Term

The fidelity term is used to characterize the spatial dependency and highly complex distribution of the residual image $\mathbf{r} = g(\mathbf{x}; \mathcal{B}) + \mathbf{n}$. On one hand, the popular explicit formulation, e.g., ℓ_2-norm and ℓ_1-norm, cannot model the complex distribution of residual image $\mathbf{r}$. Due to the i.i.d. assumption, the existing noise modeling approaches, e.g., GMM [37] and MoEP [6], also cannot be readily adopted to model spatial dependency in fidelity term. On the other hand, the residual $\mathbf{r}$ generally is spatially dependent and complicatedly distributed. Motivated by the success of discriminative regularization learning [3], [11], we also use a set of linear filters $\{\mathbf{p}_i\}_{i=1}^{N_f}$ with diverse patterns to model the spatial dependency in $g(\mathbf{x}; \mathcal{B})$. Moreover, due to the effect of $\mathbf{n}$ and its combination with $g(\mathbf{x}; \mathcal{B})$, the filter responses $\{\mathbf{r} \otimes \mathbf{p}_i\}_{i=1}^{N_f}$ remain of complex distribution. Therefore, a set of non-linear penalty functions $\{\mathcal{D}_i\}_{i=1}^{N_f}$ is further introduced to characterize the distribution of filter responses.

To sum up, we propose a principled residual modeling in the fidelity term as follows,

$$\mathcal{F}(\mathbf{x}) = \lambda \sum_{j=1}^{N} \sum_{i=1}^{N_f} \mathcal{D}_i \left(\left(\mathbf{p}_i \otimes (\mathcal{A}\mathbf{x} - \mathbf{y}) \right)_j \right), \quad (9)$$

where $\mathcal{A}$ is the degradation operator defined in (1) and $\otimes$ is the 2D convolution operator. In the proposed fidelity term, the parameters include $\Theta_f = \{\lambda, \mathbf{p}_i, \mathcal{D}_i\}_{i=1}^{N_f}$. When $N_f = 1$, $\mathbf{p}_1$ is delta function and $\mathcal{D}_1$ is the squared ℓ_2-norm, the proposed model (9) is equivalent to the standard MAP-based model in (2).

Due to the introduction of linear filters $\{\mathbf{p}_i\}_{i=1}^{N_f}$ and penalty functions $\{\mathcal{D}_i\}_{i=1}^{N_f}$, the proposed fidelity term can describe the complex patterns in residual $\mathbf{r}$ caused by partially known or inaccurate degradation models. Furthermore, our fidelity model is flexible and applicable to different tasks. With proper training, it can be specified to certain image restoration tasks, such as rain streak removal, image deconvolution with inaccurate blur kernels. It is worth noting that the fidelity term in (9) can be regarded as a special form of convolution layer in CNN. Nonetheless, the fidelity term (9) can retain better interpretability and flexibility in characterizing residual $\mathbf{r}$. In particular, the learned $\mathbf{p}_i$s and $\mathcal{D}_i$s are closely related to the characteristics of redidual $\mathbf{r}$ (see an example in the supplementary material). Moreover, the distribution of $\mathbf{p}_i \otimes \mathbf{r}$ generally is much more complex, and cannot be simply characterized by ReLU and its variants in conventional CNN.

3.2 Regularization Term

To increase modeling capacity on image prior, the regularization term is further parameterized as

$$\mathcal{R}(\mathbf{x}) = \sum_{j=1}^{N} \sum_{i=1}^{N_r} \mathcal{R}_i \left((\mathbf{f}_i \otimes \mathbf{x})_j \right), \qquad (10)$$

where $\mathbf{f}_i$ is the i-th linear filter, $\mathcal{R}_i$ is the corresponding non-linear penalty function, and N_r is the number of linear filters and penalty functions for the regularization term. The parameters for the regularization term include $\Theta_r = \{\mathbf{f}_i, \mathcal{R}_i\}_{i=1}^{N_r}$. The proposed model is the generalization of the FoE [10] model by parameterizing the regularization term with both the filters and penalty functions. Similar models have also been used in discriminative non-blind image restoration [3], [11], [13].

3.3 SFARL Model

Given a specific image restoration task, the parameters for the fidelity and regularization terms need to be specified. As a large number of parameters are involved in $\mathcal{F}(\mathbf{x})$ and $\mathcal{R}(\mathbf{x})$, it is not feasible to manually determine proper values. In this work, we propose to learn the parameters of both fidelity and regularization terms in a task-driven manner.

Denote a training set of S samples by $\{\mathbf{y}_s, \mathbf{x}_s^{gt}\}_{s=1}^{S}$, where $\mathbf{y}_s$ is the s-th degraded image and $\mathbf{x}_s^{gt}$ is the corresponding ground-truth image. The parameters $\Theta = \{\Theta_f, \Theta_r\}$ can be learned by solving the following bi-level optimization problem,

$$\min_{\Theta} \mathcal{L}(\Theta) = \sum_{s=1}^{S} \ell \left(\mathbf{x}_s^*, \mathbf{x}_s^{gt} \right)$$
$$\text{s.t.} \ \ \mathbf{x}_s^* = \arg\min_{\mathbf{x} \in \mathcal{X}} \sum_{j=1}^{N} \sum_{i=1}^{N_r} \mathcal{R}_i \left((\mathbf{f}_i \otimes \mathbf{x})_j \right) \qquad (11)$$
$$+ \lambda \sum_{j=1}^{N} \sum_{i=1}^{N_f} \mathcal{D}_i \left((\mathbf{p}_i \otimes (\mathcal{A}\mathbf{x} - \mathbf{y}_s))_j \right),$$

where $\mathcal{X}$ is the feasible solution space. For image deconvolution with an inaccurate blur kernel, the feasible solution is only constrained to be in real number space, i.e., $\mathcal{X} = \{\mathbf{x} \mid \mathbf{x} \in \mathbb{R}^N\}$. For rain streak removal, additional constraints on the feasible solution space are required, i.e., $\mathcal{X} = \{\mathbf{x} \mid \forall i, \ 0 \le x_i \le y_i\}$, where x_i (and y_i) is the i-th element of clean image $\mathbf{x}$ (and rainy image $\mathbf{y}$). In principle, the trade-off parameter λ can be absorbed into the non-linear transform $\mathcal{D}_i$ and removed from the model (11). However, the trade-off between the fidelity and regularization terms cannot be easily made due to that the scales of $\mathcal{D}_i$ and $\mathcal{R}_i$ vary for different restoration tasks, thereby making it necessary to include λ in (11).

The loss function $\ell(\cdot, \cdot)$ measures the dissimilarity between the output of the SFARL model and the ground-truth image. One representative loss used in discriminative image restoration is based on the mean-squared error (MSE) [11],

$$\ell \left(\mathbf{x}, \mathbf{x}^{gt} \right) = \frac{1}{2} \|\mathbf{x} - \mathbf{x}^{gt}\|^2. \qquad (12)$$

For image restoration when the precise degradation process is known, the optimal fidelity term in terms of MSE becomes the negative log-likelihood. The standard MAP model $\mathbf{x} = \arg\min_{\mathbf{x}} \frac{\lambda}{2} \|\mathcal{A}\mathbf{x} - \mathbf{y}\|^2 + \mathcal{R}(\mathbf{x})$ can then be used in the inner loop of the bi-level optimization task. Thus, the MSE loss is only applicable to learning fidelity term for image restoration with partially known or inaccurate degradation models.

In this work, we use the visual perception metric, e.g., negative SSIM [53], [54], as the loss function,

$$\ell \left(\mathbf{x}, \mathbf{x}^{gt} \right) = -\text{SSIM} \left(\mathbf{x}, \mathbf{x}^{gt} \right). \qquad (13)$$

The reason of using negative SSIM is two-fold. On one hand, it is known that SSIM is closely related to visual perception of image quality, and minimizing negative SSIM is expected to benefit the visual quality of restoration result. On the other hand, even for image restoration with precise degradation process, the negative log-likelihood will not be the optimal fidelity term when the negative SSIM loss is used. Thus the residual model (9) can be utilized to learn proper fidelity term from training data for either image deconvolution with inaccurate blur kernels, rain streak removal, or Gaussian denoising. In addition, the experimental results also validate the effectiveness of negative SSIM and residual modeling in terms of both visual quality and perception metric.

4 SFARL TRAINING

In this section, we first present an iterative solution to inner task in the bi-level optimization problem. The SFARL model is then parameterized and gradient-based optimization algorithm can be used for training. The SFARL model is trained by sequentially performing greedy training in Algorithm 2 and joint fine-tuning in Algorithm 3. Finally, the derivations of gradients for the greedy and end-to-end training processes are presented.

4.1 Iterative Solution to Inner Optimization Task

The inner task in (11) implicitly defines a function $\mathbf{x}^*(\Theta)$ on the model parameters. As the optimization problem is non-convex, it is difficult to obtain the explicit analytic form of either $\mathbf{x}^*(\Theta)$ or $\frac{\partial \mathbf{x}^*(\Theta)}{\partial \Theta}$. In this work, we learn Θ by considering the truncation of an iterative optimization algorithm [3], [11], [12], [46]. Furthermore, the stage-wise model parameters are also used to improve image restoration [3], [11].

To solve (11), the updated solution $\mathbf{x}^{t+1}$ can then be written as a function of $\mathbf{x}^t$ and Θ, i.e., $\mathbf{x}^{t+1}(\Theta; \mathbf{x}^t)$. Suppose that $\{(\Theta^1, \mathbf{x}^1), ..., (\Theta^t, \mathbf{x}^t)\}$ are known. The stage-wise parameters Θ^{t+1} can then be learned by solving the following problem,

$$\min_{\Theta^{t+1}} \mathcal{L}(\Theta^{t+1}) = \sum_{s=1}^{S} \ell \left(\mathbf{x}_s^{t+1}(\Theta^{t+1}; \mathbf{x}^t), \mathbf{x}_s^{gt} \right). \qquad (14)$$

Here we use a gradient descent method to solve the inner optimization loop, and $\mathbf{x}^{t+1}(\Theta; \mathbf{x}^t)$ can be written as

$$\mathbf{x}^{t+1}(\Theta^{t+1}; \mathbf{x}^t) = \mathbf{x}^t - \sum_{i=1}^{N_r} \bar{\mathbf{f}}_i^{t+1} \otimes \phi_i^{t+1} (\mathbf{f}_i^{t+1} \otimes \mathbf{x}^t)$$
$$- \lambda^{t+1} \sum_{i=1}^{N_f} \mathcal{A}^{\top} \bar{\mathbf{p}}_i^{t+1} \otimes \varphi_i^{t+1} (\mathbf{p}_i^{t+1} \otimes (\mathcal{A}\mathbf{x}^t - \mathbf{y})), \qquad (15)$$

where the influence functions are defined as $\varphi_i = \mathcal{D}'_i$ and $\phi_i = \mathcal{R}'_i$. These functions are entry-wisely performed on a vector or matrix. In addition, $\bar{\mathbf{p}}_i$ and $\bar{\mathbf{f}}_i$ are filters by rotating $\mathbf{p}_i$ and $\mathbf{f}_i$ 180 degrees, respectively. After each gradient descent step, $\mathbf{x}^{t+1}$ is projected to the feasible solution space $\mathcal{X}$. The inference procedure is shown in Algorithm 1.

We use ADAM [55] to solve the optimization problem in (14). Therefore, we need to present the parameterization of the solution in (15) and derive the gradients for the greedy and end-to-end learning processes.

Algorithm 1 SFARL

Input: Current result $\mathbf{x}^0$, degraded image $\mathbf{y}$, degradation operator $\mathcal{A}$, model parameters $\{\Theta^t\}_{t=1}^T$
Output: Restoration results $\mathbf{x}$
1: **for** $t = 0$ to $T - 1$ **do**
2:     Compute $\mathbf{x}^{t+1}$ using (15)
3: **end for**
4: $\mathbf{x} = \{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^T\}$

Algorithm 2 Greedy Training

Input: Training data $\{\mathbf{y}_i, \mathbf{x}_i^{gt}, \mathcal{A}_i\}_{i=1}^N$
Output: SFARL parameters $\{\Theta^t\}_{t=1}^T$
1: Set stage number T, epoch number E, mini-batch size n, mini-batch number $M = N/n$
2: Initialize: $\{\mathbf{x}_i^0 | \mathbf{x}_i^0 = \mathbf{y}_i\}_{i=1}^N$, $\{\Theta^t\}_{t=1}^T$
3: **for** $t = 0$ to $T - 1$ **do**
4:     **for** $epoch = 1$ to E **do**
5:         **for** $m = 0$ to $M - 1$ **do**
6:             Prepare m-th mini-batch data: $\{\mathbf{y}_i, \mathbf{x}_i^{gt}, \mathcal{A}_i\}_{i=m \times n+1}^{m \times n+n}$
7:             Forward samples in m-th mini-batch:

$$\mathbf{x}_i^{t+1} = \text{SFARL}(\mathbf{x}_i^t, \mathbf{y}_i, \mathcal{A}_i, \Theta^{t+1})$$

8:             Compute gradients for stage $t + 1$: $\frac{1}{n} \sum_i \frac{\partial \ell(\mathbf{x}_i^{t+1}, \mathbf{x}_i^{gt})}{\partial \Theta^{t+1}}$
9:             Use Adam to optimize stage $t + 1$ parameters Θ^{t+1}
10:         **end for**
11:     **end for**
12: **end for**

Algorithm 3 Joint Fine-tuning

Input: Training data $\{\mathbf{y}_i, \mathbf{x}_i^{gt}, \mathcal{A}_i\}_{i=1}^N$, model parameters $\{\Theta^t\}_{t=1}^T$
Output: SFARL parameters $\{\Theta^t\}_{t=1}^T$
1: Set epoch number E, mini-batch size n, mini-batch number $M = N/n$
2: Initialize $\{\mathbf{x}_i^0 | \mathbf{x}_i^0 = \mathbf{y}_i\}_{i=1}^N$
3: **for** $epoch = 1$ to E **do**
4:     **for** $m = 0$ to $M - 1$ **do**
5:         Prepare m-th mini-batch data: $\{\mathbf{y}_i, \mathbf{x}_i^{gt}, \mathcal{A}_i\}_{i=m \times n+1}^{m \times n+n}$
6:         Forward samples in m-th mini-batch:

$$\{\mathbf{x}_i^1, \mathbf{x}_i^2, ..., \mathbf{x}_i^T\} = \text{SFARL}(\mathbf{x}_i^0, \mathbf{y}_i, \mathcal{A}_i, \{\Theta^t\}_{t=1}^T)$$

7:         Compute gradients for each stage: $\{\frac{1}{n} \sum_i \frac{\partial \ell(\mathbf{x}_i^T, \mathbf{x}_i^{gt})}{\partial \Theta^t}\}_{t=1}^T$
8:         Use Adam to end-to-end optimize parameters $\{\Theta^t\}_{t=1}^T$
9:     **end for**
10: **end for**

4.2 Parameterization

Similar to [3], [11], we use the weighted summation of Gaussian RBF functions to parameterize the influence functions in regularization term

$$\phi_i(z) = \sum_{j=1}^M \pi_{ij} \exp\left(-\frac{\gamma}{2}(z - \mu_j)^2\right), \quad (16)$$

and in fidelity term

$$\varphi_i(z) = \sum_{j=1}^M w_{ij} \exp\left(-\frac{\gamma}{2}(z - \mu_j)^2\right), \quad (17)$$

where π_{ij} and w_{ij} are weight coefficients, μ_j is mean value and γ is precision.

The filters $\mathbf{f}_i$ in regularization term and $\mathbf{p}_i$ in fidelity term are specified as linear combination of DCT basis with unit norm constraint,

$$\mathbf{f}_i = \mathcal{B}_r \frac{\mathbf{s}_i}{\|\mathbf{s}_i\|_2} \quad \text{and} \quad \mathbf{p}_i = \mathcal{B} \frac{\mathbf{c}_i}{\|\mathbf{c}_i\|_2}, \quad (18)$$

where $\mathcal{B}$ is complete DCT basis, $\mathcal{B}_r$ is DCT basis by excluding the DC component, $\mathbf{s}_i$ and $\mathbf{c}_i$ are coefficients for regularization term and fidelity term respectively.

In our implementation, we utilize filters with size 7×7 in both regularization term and fidelity term. Thus, the numbers of non-linear functions and filters can be accordingly set, i.e., $N_r = 48$ for regularization term, and $N_f = 49$ for fidelity term. The numbers of Gaussian functions are fixed to 63 for both fidelity and regularization terms, i.e., $M = 63$. To handle the boundary condition in convolution operation, the image is padded for processing and only the valid region is cropped for output.

4.3 Greedy Training

The SFARL model is firstly trained stage-by-stage. To learn the model parameters of stage $t + 1$, we need to compute gradient by the chain rule,

$$\frac{\partial \ell(\mathbf{x}^{t+1}, \mathbf{x}^{gt})}{\partial \Theta^{t+1}} = \frac{\partial \mathbf{x}^{t+1}}{\partial \Theta^{t+1}} \frac{\partial \ell(\mathbf{x}^{t+1}, \mathbf{x}^{gt})}{\partial \mathbf{x}^{t+1}}. \quad (19)$$

4.3.1 Deviation of $\frac{\partial \ell(\mathbf{x}^{t+1}, \mathbf{x}^{gt})}{\partial \mathbf{x}^{t+1}}$

When the loss function is specified as MSE, i.e., $\ell\left(\mathbf{x}^{t+1}, \mathbf{x}^{gt}\right) = \frac{1}{2}\|\mathbf{x}^{t+1} - \mathbf{x}^{gt}\|^2$, the gradient can be simply computed as

$$\frac{\partial \ell(\mathbf{x}^{t+1}, \mathbf{x}^{gt})}{\partial \mathbf{x}^{t+1}} = \mathbf{x}^{t+1} - \mathbf{x}^{gt}. \quad (20)$$

Visual perception metric, i.e., negative SSIM

When the loss function is specified as visual perception metric, i.e., $\ell\left(\mathbf{x}^{t+1}, \mathbf{x}^{gt}\right) = -\text{SSIM}(\mathbf{x}^{t+1}, \mathbf{x}^{gt})$ [53], [54], we give the gradient deviation as follows. To distinct the entire image and small patch, only in this subsection we use $\mathbf{X}$ and $\mathbf{Y}$ as entire image and reference image respectively. The SSIM value is computed based on the small patches $\mathbf{x}_i$ and $\mathbf{y}_i$

$$\text{SSIM}(\mathbf{X}, \mathbf{Y}) = \frac{1}{N_s} \sum_{i=1}^{N_s} \text{ssim}(\mathbf{x}_i, \mathbf{y}_i), \quad (21)$$

where N_s is the number of patches. The value on each patch is computed as

$$\text{ssim}(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x \mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (22)$$

where $\mu_x = \frac{1}{N_p}\left(\mathbf{1}^\top \mathbf{x}\right)$ is mean value of patch $\mathbf{x}$, $\sigma_x^2 = \frac{1}{N_p-1}(\mathbf{x} - \mu_x)^\top(\mathbf{x} - \mu_x)$ is variance of patch $\mathbf{x}$, and $\sigma_{xy} = \frac{1}{N_p-1}(\mathbf{x} - \mu_x)^\top(\mathbf{y} - \mu_y)$ is covariance of pathes $\mathbf{x}$ and $\mathbf{y}$, and C_1, C_2 are some constant values. Let us define $A_1 = 2\mu_x \mu_y + C_1$, $A_2 = 2\sigma_{xy} + C_2$, $B_1 = \mu_x^2 + \mu_y^2 + C_1$ and $B_2 = \sigma_x^2 + \sigma_y^2 + C_2$. Then we have $S(\mathbf{x}, \mathbf{y}) = \frac{A_1 A_2}{B_1 B_2}$.

The gradient of negative SSIM is

$$\frac{\partial(-\text{SSIM}(\mathbf{X}, \mathbf{Y}))}{\partial \mathbf{X}} = -\frac{1}{N_s} \sum_{i=1}^{N_s} \frac{\partial(\text{ssim}(\mathbf{x}_i, \mathbf{y}_i))}{\partial \mathbf{Y}}$$
$$= -\frac{1}{N_s} \sum_{i=1}^{N_s} \frac{\partial(\text{ssim}(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{X}}\bigg|_{\mathbf{x}=\mathbf{x}_i, \mathbf{y}=\mathbf{y}_i}, \quad (23)$$

where

$$\frac{\partial(\text{ssim}(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{x}} = \frac{2}{N_p B_1^2 B_2^2}\left(A_1 B_1 (B_2 \mathbf{x} - A_2 \mathbf{y})\right.$$
$$\left. + B_1 B_2 (A_2 - A_1)\mu_x \mathbf{1} + A_1 A_2 (B_1 - B_2)\mu_y \mathbf{1}\right). \quad (24)$$

For simplicity, we hereafter use $\mathbf{e}$ to denote $\frac{\partial \ell(\mathbf{x}^{t+1}, \mathbf{x}^{gt})}{\partial \mathbf{x}^{t+1}}$ for both MSE and negative SSIM.

4.3.2 Deviation of $\frac{\partial \mathbf{x}^{t+1}}{\partial \Theta^{t+1}}$

Since the parameterization of fidelity term and regularization term is similar, we only use the fidelity term as an example, and it is easy to extend it to the regularization term.

Weight parameter λ

The gradient with respect to λ is

$$\frac{\partial \mathbf{x}^{t+1}}{\partial \lambda^{t+1}} = -\left(\sum_{i=1}^{N_f} \mathcal{A}^\top \bar{\mathbf{p}}_i^{t+1} \otimes \varphi_i^{t+1} \left(\mathbf{p}_i^{t+1} \otimes (\mathcal{A}\mathbf{x}^t - \mathbf{y}) \right) \right)^\top . \quad (25)$$

The overall gradient with respect to λ is

$$\frac{\partial \ell(\mathbf{x}^{t+1}, \mathbf{x}^{gt})}{\partial \lambda^{t+1}} = -\left(\sum_{i=1}^{N_c} \mathcal{A}^\top \bar{\mathbf{p}}_i^{t+1} \otimes \varphi_i^{t+1} \left(\mathbf{p}_i^{t+1} \otimes (\mathcal{A}\mathbf{x}^t - \mathbf{y}) \right) \right)^\top \mathbf{e}. \quad (26)$$

Filter $\mathbf{p}_i$

The function $\mathbf{x}^{t+1}$ with respect to each filter $\mathbf{p}_i$ can be simplified to,

$$\mathbf{x}^{t+1} = -\lambda^{t+1} \mathcal{A}^\top \bar{\mathbf{p}}_i^{t+1} \otimes \varphi_i^{t+1} \left(\mathbf{p}_i^{t+1} \otimes (\mathcal{A}\mathbf{x}^t - \mathbf{y}) \right) + C, \quad (27)$$

where C denotes a constant which is independent with $\mathbf{p}_i$. Let us define $\mathbf{u} = -\mathcal{A}^\top \bar{\mathbf{p}}_i^{t+1}$ and $\mathbf{v} = \varphi_i^{t+1} \left(\mathbf{p}_i^{t+1} \otimes (\mathcal{A}\mathbf{x}^t - \mathbf{y}) \right)$. Thus, we can obtain the gradient deviation as

$$\frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{p}_i^{t+1}} = \frac{\partial \mathbf{u}}{\partial \mathbf{p}_i^{t+1}} \frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{u}} + \frac{\partial \mathbf{v}}{\partial \mathbf{p}_i^{t+1}} \frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{v}}. \quad (28)$$

Based on the convolution theorem [56], we have

$$\mathbf{u} \otimes \mathbf{v} \Leftrightarrow \mathbf{U}\mathbf{v} \Leftrightarrow \mathbf{V}\mathbf{u}, \quad (29)$$

where $\mathbf{U}$ and $\mathbf{V}$ are sparse convolution matrices of $\mathbf{u}$ and $\mathbf{v}$, respectively. Thus, the first term in (28) is

$$\frac{\partial \mathbf{u}}{\partial \mathbf{p}_i^{(t)}} \frac{\partial \mathbf{x}^{(t)}}{\partial \mathbf{u}} = -R_{180}^\top \mathcal{A}\mathbf{V}^\top, \quad (30)$$

where R_{180} rotates matrix by 180 degrees.

For the second term, we introduce an auxiliary variable $\mathbf{b} = \mathcal{A}\mathbf{x}^t - \mathbf{y}$, $\mathbf{z} = \mathbf{p}_i^{t+1} \otimes \mathbf{b}$, and we have $\mathbf{v} = \varphi_i^{t+1}(\mathbf{z})$. We note that

$$\mathbf{z} = \mathbf{p}_i^{t+1} \otimes \mathbf{b} \Leftrightarrow \mathbf{B}\mathbf{p}_i^{t+1}.$$

Therefore, we have

$$\frac{\partial \mathbf{v}}{\partial \mathbf{p}_i^{t+1}} \frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{v}} = \frac{\partial \mathbf{z}}{\partial \mathbf{p}_i^{t+1}} \frac{\partial \mathbf{v}}{\partial \mathbf{z}} \frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{v}} = -\mathbf{B}^\top \Lambda \mathbf{U}^\top, \quad (31)$$

where $\Lambda = \text{diag}\left(\varphi_i^{t+1\prime}(z_1), ..., \varphi_i^{t+1\prime}(z_N) \right)$ is a diagonal matrix. The gradient of $\varphi_i(z)$ is

$$\varphi_i'(z) = -\gamma \sum_{j=1}^M w_{ij} \exp\left(-\frac{\gamma}{2}(z - \mu_j)^2 \right) (z - \mu_j). \quad (32)$$

Since the filter is specified as linear combination of DCT basis, one need to derive the gradient with respect to the combination coefficients $\mathbf{c}$, i.e.,

$$\frac{\partial \ell}{\partial \mathbf{c}} = \frac{\partial \mathbf{p}}{\partial \mathbf{c}} \frac{\partial \ell}{\partial \mathbf{p}}. \quad (33)$$

By introducing $\mathbf{v} = \frac{\mathbf{c}}{\|\mathbf{c}\|_2}$, we then have

$$\begin{aligned} \frac{\partial \mathbf{p}}{\partial \mathbf{c}} &= \frac{\partial \mathbf{v}}{\partial \mathbf{c}} \frac{\partial \mathbf{p}}{\partial \mathbf{v}} = \frac{\partial \mathbf{v}}{\partial \mathbf{c}} \mathcal{B}^\top \\ &= \left(\frac{\mathbf{I}}{\|\mathbf{c}\|_2} + \frac{\partial (\mathbf{c}^\top \mathbf{c})^{-\frac{1}{2}}}{\partial \mathbf{c}} \right) \mathcal{B}^\top \\ &= \left(\frac{\mathbf{I}}{\|\mathbf{c}\|_2} + \frac{\partial (\mathbf{c}^\top \mathbf{c})}{\partial \mathbf{c}} (-\frac{1}{2} \frac{1}{\|\mathbf{c}\|_2^3}) \mathbf{c}^\top \right) \mathcal{B}^\top \\ &= \left(\frac{\mathbf{I}}{\|\mathbf{c}\|_2} + 2\mathbf{c}(-\frac{1}{2} \frac{1}{\|\mathbf{c}\|_2^3}) \mathbf{c}^\top \right) \mathcal{B}^\top \\ &= \frac{1}{\|\mathbf{c}\|_2} \left(\mathbf{I} - \frac{\mathbf{c}}{\|\mathbf{c}\|_2} \frac{\mathbf{c}^\top}{\|\mathbf{c}\|_2} \right) \mathcal{B}^\top. \end{aligned} \quad (34)$$

Finally, the overall gradient with respect to combination coefficients $\mathbf{c}_i^{t+1}$ is given by

$$\frac{\partial \ell}{\partial \mathbf{c}_i^{t+1}} = -\frac{1}{\|\mathbf{c}_i^{t+1}\|_2} \left(\mathbf{I} - \frac{\mathbf{c}_i^{t+1}}{\|\mathbf{c}_i^{t+1}\|_2} \frac{(\mathbf{c}_i^{t+1})^\top}{\|\mathbf{c}_i^{t+1}\|_2} \right) \mathcal{B}^\top \left(\mathbf{B}^\top \Lambda \mathbf{U}^\top + R_{180}^\top \mathcal{A}\mathbf{V}^\top \right) \mathbf{e}. \quad (35)$$

Non-linear function φ_i

We first reformulate the function $\mathbf{x}^{t+1}$ with respect to φ_i into the matrix form

$$\mathbf{x}^{t+1} \sim -\lambda^{t+1} \mathcal{A}^\top (\mathbf{P}_i^{t+1})^\top \varphi_i^{t+1}(\mathbf{b}), \quad (36)$$

where $\mathbf{b} = \mathbf{P}_i^{t+1}(\mathcal{A}\mathbf{x}^t - \mathbf{y})$. Therefore, the column vector $\varphi_i^{t+1}(\mathbf{b})$ can be reformulated into the matrix form,

$$\varphi_i^{t+1}(\mathbf{b}) = \mathbf{G}(\mathbf{b})\mathbf{w}_i^{t+1}, \quad (37)$$

where $\mathbf{w}_i$ is the vectorized version of parameters w_{ij}, matrix $\mathbf{G}(\mathbf{b})$ is

$$\mathbf{G}(\mathbf{b}) = \begin{bmatrix} \exp(-\frac{\gamma}{2}(b_1 - \mu_1)^2) & \exp(-\frac{\gamma}{2}(b_1 - \mu_2)^2) & \cdots & \exp(-\frac{\gamma}{2}(b_1 - \mu_M)^2) \\ \exp(-\frac{\gamma}{2}(b_2 - \mu_1)^2) & \exp(-\frac{\gamma}{2}(b_2 - \mu_2)^2) & \cdots & \exp(-\frac{\gamma}{2}(b_2 - \mu_M)^2) \\ \vdots & \vdots & \ddots & \vdots \\ \exp(-\frac{\gamma}{2}(b_N - \mu_1)^2) & \exp(-\frac{\gamma}{2}(b_N - \mu_2)^2) & \cdots & \exp(-\frac{\gamma}{2}(b_N - \mu_M)^2) \end{bmatrix}.$$

Thus, we can get

$$\frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{w}_i^{t+1}} = -\lambda^{t+1} \mathbf{G}^\top \mathbf{P}_i^{t+1} \mathcal{A}, \quad (38)$$

and finally the overall gradient with respect to $\mathbf{w}_i^{t+1}$ is

$$\frac{\partial \ell}{\partial \mathbf{w}_i^{t+1}} = -\lambda^{t+1} \mathbf{G}^\top \mathbf{P}_i^{t+1} \mathcal{A}\mathbf{e}. \quad (39)$$

In our implementation, we do not explicitly compute the matrix $\mathbf{U}, \mathbf{V}, \mathbf{B}$, since they can be efficiently operated via 2D convolution.

4.4 Joint Fine-tuning

Once the greedy training process for each stage is carried out, an end-to-end training process is used to fine-tune all the parameters across stages. The joint training loss function is defined as

$$\mathcal{L}(\Theta^1, ..., \Theta^T) = \sum_{s=1}^S \ell \left(\mathbf{x}_s^T, \mathbf{x}_s^{gt} \right), \quad (40)$$

where T is the maximum iteration number. The gradient can be computed by the chain rule,

$$\frac{\partial \ell(\mathbf{x}^T, \mathbf{x}^{gt})}{\partial \Theta^t} = \frac{\partial \mathbf{x}^t}{\partial \Theta^t} \frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{x}^t} \cdots \frac{\partial \ell(\mathbf{x}^T, \mathbf{x}^{gt})}{\partial \mathbf{x}^T}. \quad (41)$$

where only $\frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{x}^t}$ need to be additionally computed. By reformulating the solution in the matrix form,

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \sum_{i=1}^{N_r} (\mathbf{F}_i^{t+1})^\top \phi_i^{t+1} (\mathbf{F}_i^{t+1} \mathbf{x}^t)$$
$$-\lambda^{t+1} \sum_{i=1}^{N_f} \mathcal{A}^\top (\mathbf{P}_i^{t+1})^\top \varphi_i^{t+1} \left(\mathbf{P}_i^{t+1} \left(\mathcal{A}\mathbf{x}^t - \mathbf{y} \right) \right), \quad (42)$$

the gradient can be computed as

$$\frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{x}^t} = \mathbf{I} - \sum_{i=1}^{N_r} (\mathbf{F}_i^{t+1})^\top \Gamma_i \mathbf{F}_i^{t+1} - \sum_{i=1}^{N_f} \mathcal{A}^\top (\mathbf{P}_i^{t+1})^\top \Lambda_i \mathbf{P}_i^{t+1} \mathcal{A}, \quad (43)$$

where $\Gamma_i = \mathrm{diag}\left(\phi_i^{t'}(z_1), ..., \phi_i^{t'}(z_N) \right)$ is also a diagonal matrix.

Once $\frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{x}^t}$ is computed, the overall gradient can be computed by the chain rule and the other gradient parts in (41) can be borrowed from greedy training.

4.4.1 Training Procedure

Given a training dataset, the training of SFARL is to sequentially run greedy training as Algorithm 2 and joint fine-tuning as Algorithm 3. Algorithm 1 lists the inference of SFARL given model parameters, in which all the intermediate results are recorded for backward propagation during training. In greedy training Θ^{t+1}, parameters $\{\Theta^i\}_{i=1}^t$ in previous t stages are fixed, and only gradients in stage $t+1$ are computed and are fed to ADAM algorithm. In joint fine-tuning, gradients in each stage are computed, and are fed to ADAM algorithm to optimize the parameters $\{\Theta^t\}_{t=1}^T$ for all the stages.

5 EXPERIMENTAL RESULTS

In this section, we evaluate the proposed SFARL algorithm on several restoration tasks, i.e., image deconvolution either with an inaccurate blur kernel or with multiple degradations, rain streak removal from a single image. SFARL can also be evaluated on Gaussian denoising, and we have presented the results in the supplementary material. In our experiments, 7×7 filters are adopted in both fidelity and regularization terms. As for stage number, we recommend to set it based on the convergence behavior during greedy training, and empirically use 10-stage SFARL for image deconvolution, and 5-stage SFARL for rain streak removal and Gaussian denoising. During training SFARL, greedy training ends with 10 epoches for each stage, and then the parameters are further jointly fine-tuned with 50 epochs. We use ADAM [55] to optimize these SFARL models with learning rate 1×10^{-3}, $\beta_1 = 0.9$ and $\beta_2 = 0.99$. Using rain streak removal as an example, it takes about 19 hours to train a SFARL model on a computer equipped with a GTX 1080Ti GPU. The SFARL models are quantitatively and qualitatively evaluated and compared with state-of-the-art conventional and deep CNN-based approaches.

More experimental settings and results are included in the supplementary material. The source code is available at https://github.com/csdwren/sfarl.

5.1 Deconvolution with Inaccurate Blur Kernels

We consider the blind deconvolution task and use two blur kernel estimation methods, i.e., Cho and Lee [48] and Xu and Jia [18], for experiments. For each estimation approach, we evaluate the performance of SFARL for handling approach-specific blur kernel estimation error. To construct the training dataset, we use eight

blur kernels [57] on 200 clean images from the BSD dataset [58]. The Gaussian noise with $\sigma = 0.25$ is added to generate the blurry images. The methods by Cho and Lee [48] and Xu and Jia [18] are used to estimate blur kernels. Thus, we have 1,600 training samples for each blur kernel estimation approach. To ensure the training sample quality, we randomly select 500 samples with error ratio [57] above 3 for each image deconvolution method.

TABLE 1: Quantitative SSIM results on the dataset by Levin et al. [57].

Kernel estimation	EPLL [2]	ROBUST [14]	IRCNN [33]	SFARL
Cho and Lee [48]	0.8801	0.8659	0.8825	0.8903
Xu and Jia [18]	0.9000	0.8917	0.9023	0.9164

On the widely used synthetic dataset, i.e., Levin et al. [57], we compare our SFARL with EPLL [2], ROBUST [14] and IRCNN [33]. The testing dataset includes 4 clean images and 8 blur kernels. The blur kernels are estimated by Cho and Lee [48] and Xu and Jia [18]. Table 1 lists the average SSIM values of all evaluated methods on the dataset by Levin et al. [57]. Overall, the SFARL algorithm performs favorably against the other methods in terms of SSIM. From Table 1, we also have the following observations. First, the SFARL algorithm models the residual images by specific blur kernel estimation method to improve restoration result. For each blur kernel estimation method, what we need to do is to retrain the SFARL model from the synthetic data. Second, when the estimated blur kernel is more accurate (e.g., Xu and Jia [18]), better quantitative performance indexes are also attained by our SFARL.

We evaluate the SFARL algorithm against the state-of-the-art methods on a synthetic and a real blurry images in Figures 2 and 3. The blur kernels are estimated using the method by Xu and Jia [18]. As the blur kernel can be accurately estimated in Fig. 2, all the evaluated methods perform well and the SFARL algorithm restores more texture details. On the other hand, the estimated blur kernel is less accurately estimated in Fig. 3. Among all the evaluated methods, the deblurred image by the SFARL algorithm is sharper with fewer ringing effects than those by the other methods. We note that IRCNN [33] use the ℓ_2-norm in the fidelity term and the ROBUST scheme [14] introduces an ℓ_1-norm regularizer on the residual $\mathbf{z}$ caused by kernel error. However, both ℓ_2-norm and ℓ_1-norm are limited in modeling the complex distribution of the residual, and neither GMM prior in EPLL nor deep CNN prior in IRCNN cannot well compensate the effect caused by inaccurate blur kernels. Thus, the performance gain of the SFARL model can be attributed to its effectiveness in characterizing the spatial dependency and complex distribution of residual images.

5.2 Deconvolution with Multiple Degradations

We consider a more challenging deconvolution task [15], in which blur convolution is followed by multiple degradations including saturation, Gaussian noise and JPEG compression. SFARL is compared with DCNN [15], Whyte [59], IRCNN [33] and SRN [60]. Following the degradation steps in [15], 500 clean images from BSD dataset [58] are used to synthesize training dataset, on which SFARL and SRN are trained. Since only testing code of DCNN [15] and 30 testing images on a disk kernel with radius 7 (Disk7) are released, SFARL is only evaluated on Disk7 kernel. From Table 2, SFARL performs favorably in terms of average PSNR and SSIM. The results by SFARL are also visually more pleasing, while the results by the other methods suffer from visible noises and artifacts, as shown in Fig. 4. It is worth noting that

| Blurry image | EPLL [2] | ROBUST [14] | IRCNN [33] | SFARL |

Fig. 2: Visual quality comparison on Levin et al.'s dataset [57].

| Blurry images | IRCNN [33] | ROBUST [14] | SFARL |
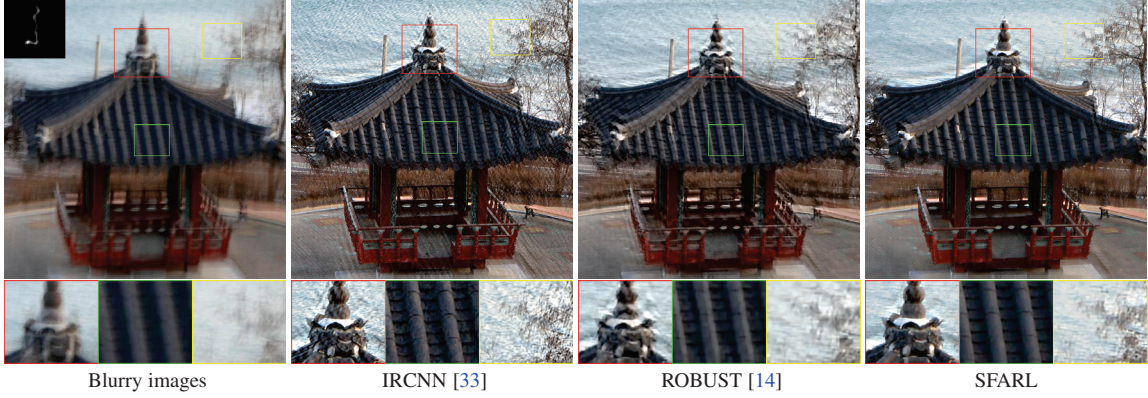
Fig. 3: Deblurring results on real blurry images, in which blur kernels are estimated by Xu and Jia [18].

| Blurry image | Whyte [59] | DCNN [15] |
| IRCNN [33] | SRN [60] | SFARL |
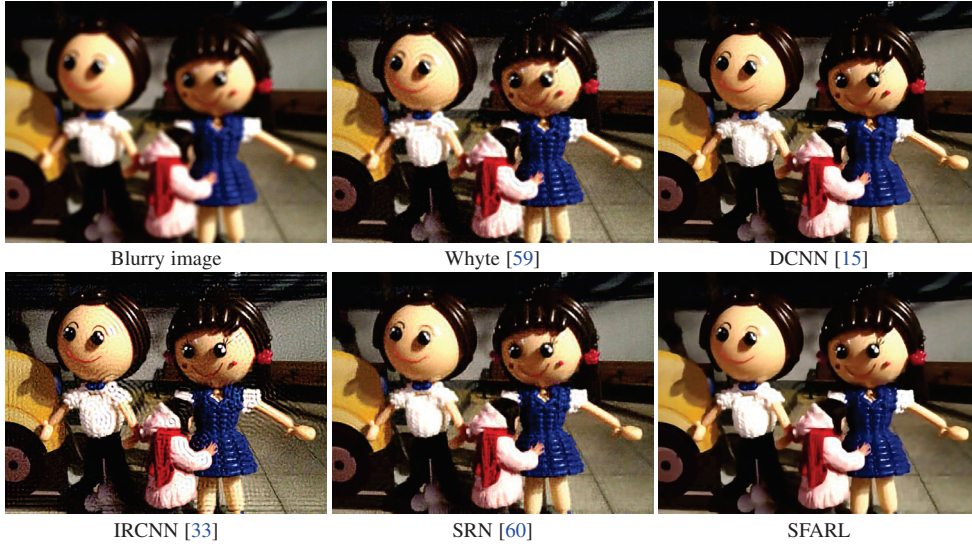
Fig. 4: Visual quality comparison on deconvolution along with Gaussian noise, satature and JPEG compression.

IRCNN works well in reducing blurring, but magnifies other degradations to yield ringing effects and noises. SRN is an up-to-date deep motion deblurring network, but is still suffering from visible noises and artifacts, since the ill-poseness caused by disk blur is usually more severe than motion blur. Thus, we conclude that SFARL is able to model these multiple degradations in fidelity term. Moreover, it should be noted that DCNN needs to initialize deconvolution sub-network using inverse kernels, while our SFARL is much easier to train given proper training dataset.

TABLE 2: Quantitative comparison on deconvolution with multiple degradations [15].

Method	Whyte [59]	DCNN [15]	IRCNN [33]	SRN [60]	SFARL
PSNR	26.35	26.50	23.84	26.46	26.66
SSIM	0.8307	0.8442	0.6673	0.8447	0.8532

5.3 Singe Image Rain Streak Removal

To train the SFARL model for rain streak removal, we construct a synthetic rainy dataset. We randomly select 100 clean outdoor images from the UCID dataset [63], and use the Photoshop function (http://www.photoshopessentials.com/photo-effects/rain/) to generate 7 rainy images at 7 random rain scales and different orientations ranged from 60 to 90 degrees. The training dataset contains 700 images with different rain orientations and scales.

We evaluate the SFARL method with the state-of-the-art algorithms including SR [61], LRA [23], GMM [24], and the CNN [62], on a the synthetic dataset [24]. The dataset consists of 12 rainy images with orientation ranged from left to right. Table 3 shows that the SFARL algorithm achieves the highest SSIM values for each test image. Fig. 5 shows rain streak removal results by all
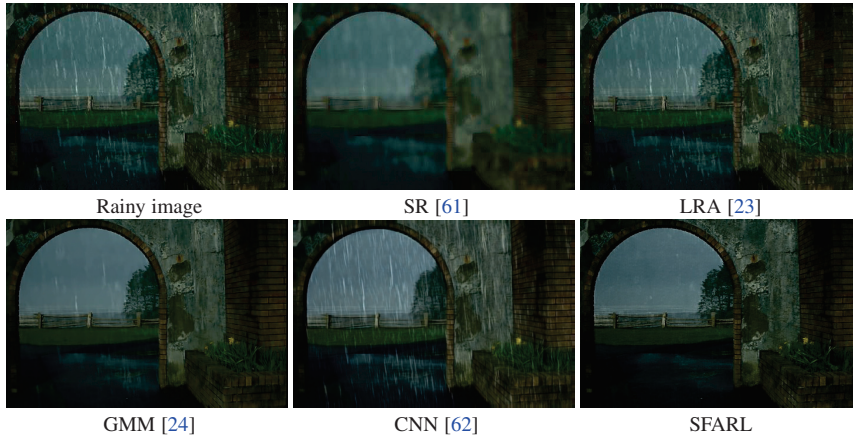
Fig. 5: Rain streak removal results of five evaluated methods on a synthetic image in [24].

TABLE 3: Deraining results on synthetic rainy images in [24] in terms of SSIM

Method	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	Avg.
SR [61]	0.74	0.79	0.84	0.77	0.63	0.73	0.82	0.77	0.74	0.74	0.65	0.77	0.75
LRA [23]	0.83	0.88	0.76	0.96	0.92	0.93	0.94	0.81	0.90	0.82	0.85	0.80	0.87
GMM [24]	0.89	0.93	0.92	0.94	0.90	0.95	0.96	0.90	0.91	0.90	0.86	0.92	0.91
CNN [62]	0.75	0.79	0.71	0.89	0.76	0.80	0.85	0.77	0.81	0.76	0.79	0.73	0.78
SFARL	0.93	0.93	0.92	0.95	0.97	0.94	0.98	0.95	0.97	0.98	0.95	0.97	0.95

the evaluated algorithms on a synthetic rainy image. The results by the SFARL and GMM algorithms are significantly better than the other methods. However, the result by the GMM method still has visible rain streaks, while the SFARL model recovers satisfying clean image.

Furthermore, we compare SFARL with a recent deep CNN-based method, i.e., DDNET [16]. The authors [16] provide a training dataset of 12,600 rainy images and a testing dataset of 1,400 rainy images (Rain1400). We train SFARL on the training dataset, and on the testing dataset, SFARL is quantitatively and qualitatively compared with DDNET. From Table 4, SFARL obtains better PSNR and SSIM values on Rain1400. In Fig. 6, SFARL produces satisfactory deraining results, while rain streaks are still visible in the results by DDNET.

TABLE 4: Average PSNR/SSIM comparison on Rain1400 [16] and Rain100L [32]. Both SFARL and DDNET are trained for Rain1400, and are directly used to process Rain100L to validate generalization ability.

Method	DDNET [16]	SFARL
Rain1400 [16]	29.91/0.9099	31.37/0.9188
Rain100L [32]	29.12/0.9012	29.73/0.9181

Moreover, we evaluate the SFARL model on real world rainy images against the state-of-the-art methods. Since the rain in second image of Fig. 7 is too heavy to see rain streaks, we first use the dehazing method [64] before applying a deraining algorithm. On both test images, the SFARL algorithm performs better than DDNET [16] and GMM [24]. For real rainy images, the image formation process is complex and may not be well characterized by either linear additive model nor screen blend model. Nevertheless, due to the flexibility of the fidelity term in modeling spatially dependent and highly complex patterns, the SFARL model is more effective in modeling the complex degradation process and achieving satisfactory deraining result.

5.4 Ablation Study

In this section, we take rain streak removal as an example to analyze training convergence and effect of negative SSIM loss. We also evaluate the generalization and transferring ability of SFARL. Besides, the visualization of learned filters, interpretability and flexibility of fidelity term as well as more discussions on stage number setting are presented in the supplementary material.

5.4.1 Convergence
As shown in Fig. 8, average PSNR of each epoch is computed to form the converge curves in the 5 stages of greedy training and the final joint fine-tuning. In greedy training, SFARL can stably converge in every stage, in which notable performance gains can be attained in the first two stages, while the PSNR increases marginally in the last 3 stages. After greedy training, SFARL is further jointly fine-tuned, and empirically converge to a much better solution.
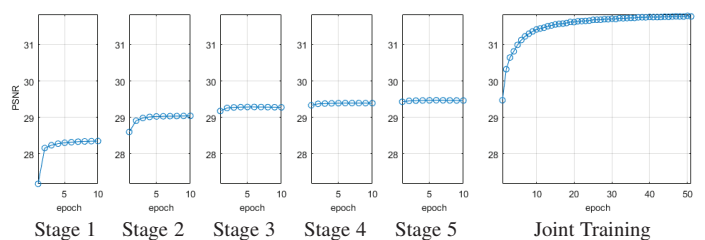

Fig. 8: Empirical convergence of a 5-stage SFARL for rain streak removal during greedy training and joint fine-tuning.

5.4.2 Loss function
To verify the effect of negative SSIM loss, we train two SFARL models, which share the same settings except training loss, i.e., one is trained by minimizing MSE loss (SFARL-MSE), while the other one by minimizing negative SSIM loss. These two SFARL models are trained and tested on the datasets provided by [16]. Form Table 5, it is reasonable to see that SFARL-MSE leads to a higher average PSNR value, while SFARL-SSIM performs better in terms of SSIM metric. Moreover, SFARL-SSIM can better remove rain streaks than SFARL-MSE, e.g., *sky* region in Fig. 9, indicating that negative SSIM loss is effective in attaining result with higher visual quality.
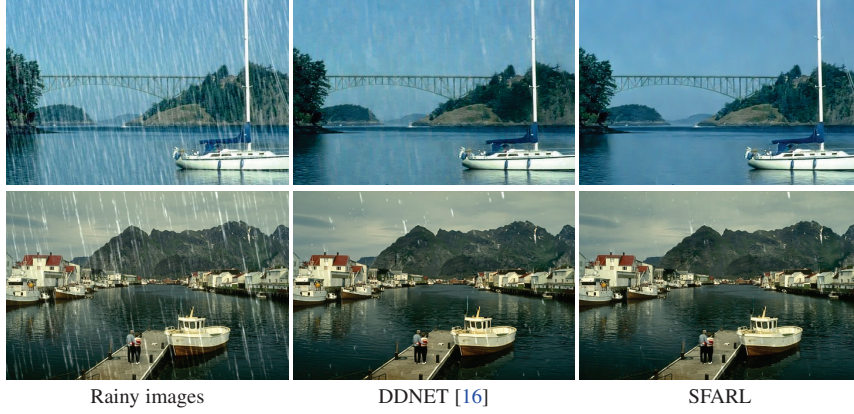
Fig. 6: Visual quality comparison for rain streak removal. Both SFARL and DDNET are trained for Rain1400 [16]. The first row is from Rain1400 [16], while the second row is from Rain100L [32] for evaluating generalization ability.
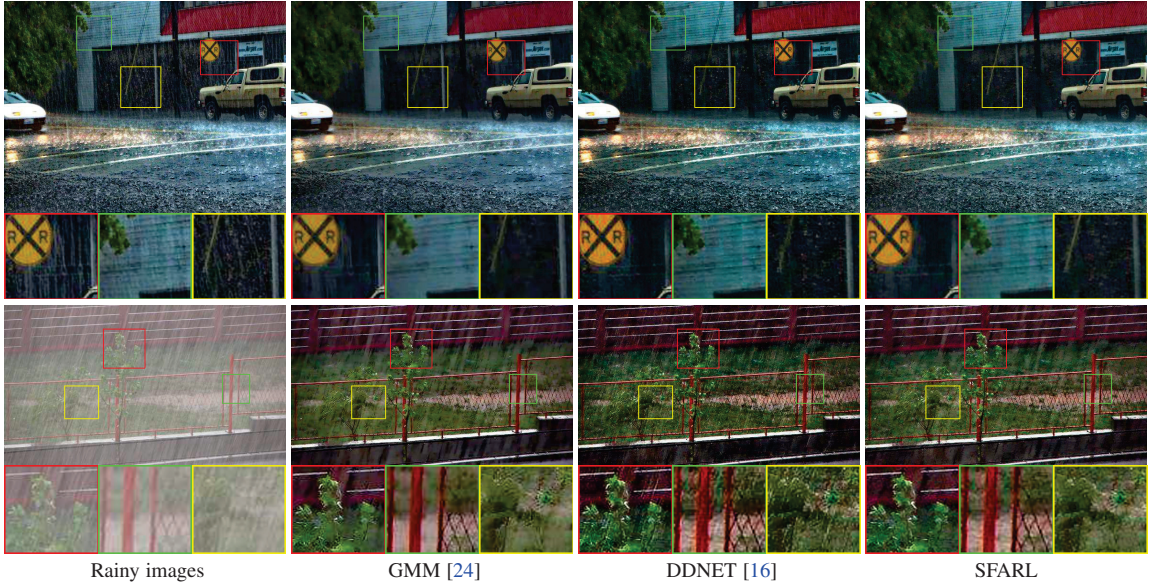

Rainy images · GMM [24] · DDNET [16] · SFARL

Fig. 7: Results on real rainy images. The rain in second image is very heavy and we first dehaze [64] it to make rain streaks more visible.
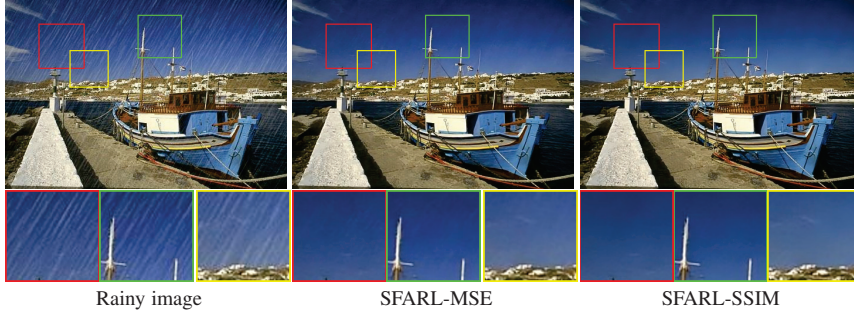

Rainy image · SFARL-MSE · SFARL-SSIM

Fig. 9: Visual quality comparison of SFARL trained by MSE loss and negative SSIM loss.

TABLE 5: Average PSNR and SSIM on testing dataset [16] of SFARL models for rain streak removal trained by MSE loss and SSIM loss

Training Loss	SFARL-MSE	SFARL-SSIM
PSNR	31.48	31.37
SSIM	0.9153	0.9188

5.4.3 Generalization evaluation

We use the trained model of SFARL for Rain1400 [16] in Section 5.3 to directly process rainy images in another dataset Rain100L [32] for evaluating the generalization ability, and compare it with the deep deraining method DDNET [16]. The rain streaks in Rain1400 and Rain100L are quite different, where those in Rain1400 are dense but gentle, and those in Rain100L are sparse but bright. And Fig. 6 shows two rainy images for an intuitive illustration. From Table 4, our SFARL can be well generalized to Rain100L. As shown in Fig. 6, remaining bright rain streaks on Rain100L can still be observed from the results by SFARL and DDNET, indicating that the learning-based methods are limited in

handling the cases that are very different from training samples. Even though, our SFARL exhibits satisfying generalization ability, and there are less remaining rain streaks in the deraining results.

5.4.4 Transferring filters across different tasks

We discuss the transferring ability of SFARL by applying the learned fidelity and regularization filters across different restoration tasks. In particular, the filters learned for deconvoluton and denoising tasks are applied to rain streak removal on the Rain12 dataset. For both denoising and deraining, there are 5 stages for the learned SFARL models, and thus the filters can be transferred in a stage-to-stage manner. As for deconvolution, the learned SFARL model has 10 stages. Considering that the first 5 stages are more correlated with deblurring, we apply the filters in the last 5 stages to rain streak removal. In the following, we respectively discuss the transferring ability of fidelity and regularization filters.

First, we apply the regularization filters learned for deconvolution and denoising to the SFARL model for deraining, denoted by SFARL$_{BlurReg}$ and SFARL$_{NoiseReg}$, respectively. From Table 6, both SFARL$_{BlurReg}$ and SFARL$_{NoiseReg}$ are notably inferior to SFARL specified for deraning. As shown in Fig. 10, most rain streaks can still be removed by SFARL$_{BlurReg}$ and SFARL$_{NoiseReg}$, but some fine-scale details may be blurry or smoothed out. From the generative learning perspective, the regularization filters are used to model clean images, and can be transferred freely across tasks. Nonetheless, due to the effect of discriminative learning, the regularization filters of SFARL are also tailored to the specific degradation type. To sum up, regularization filters exhibit moderate generalization ability across different tasks, especially the two degradation types (e.g., deraining and denoising) are more similar.

Then, we transfer fidelity filters from deconvolution and denoising to deraining, denoted as SFARL$_{BlurFid}$ and SFARL$_{NoiseFid}$, respectively. As shown in Table 6 and Fig. 10, SFARL$_{BlurFid}$ and SFARL$_{NoiseFid}$ fail in removing rain streaks quantitatively and qualitatively. Due to the correlation between fidelity filters and kernel estimation error, the result by SFARL$_{BlurFid}$ suffers from ringing effects. The fidelity filters in SFARL$_{NoiseFid}$ are learned to model noises, and perform poor in removing rain streaks from rainy image. Thus, fidelity filters are highly task-dependent, and cannot be transferred across tasks.

TABLE 6: Quantitative results on rainy Rain12 dataset [24] by transferring filters from SFARL models for deconvolution and denoising.

Method	SFARL	SFARL$_{BlurReg}$	SFARL$_{NoiseReg}$	SFARL$_{BlurFid}$	SFARL$_{NoiseFid}$
PSNR	35.97	29.66	32.02	27.03	27.79
SSIM	0.9581	0.8344	0.9074	0.6769	0.7508

6 CONCLUSION

In this paper, we propose an algorithm to effectively handle image restoration with partially known or inaccurate degradation. We present a flexible model to parameterize the fidelity term for characterizing spatial dependency and complex residual distribution of the residual image. The simultaneous fidelity and regularization learning model is developed by incorporating with the parameterized regularization term. With a set of degraded and ground-truth image pairs, task-specific and stage-wise model parameters of SFARL can then be learned in a task driven manner. Experimental results on two image restoration tasks, i.e., image deconvolution and rain streak removal, show that the SFARL model performs favorably against the state-of-the-art methods in terms of quantitative metrics and visual quality. Experiments on Gaussian denoising show that the SFARL method is effective in improving visual perception metrics and visual quality of the denoising results. Our future work includes extending the SFARL model to other restoration tasks, and developing training methods within the unsupervised learning framework.

REFERENCES

[1] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Advances in Neural Information Processing Systems*, 2009, pp. 1033–1041. 1, 2, 4

[2] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *IEEE International Conference on Computer Vision*. IEEE, 2011, pp. 479–486. 1, 2, 3, 4, 8, 9

[3] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2774–2781. 1, 3, 4, 5, 6

[4] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Advances in neural information processing systems*, 2009, pp. 2080–2088. 1, 3

[5] D. Meng and F. De La Torre, "Robust matrix factorization with unknown noise," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1337–1344. 1, 3

[6] X. Cao, Y. Chen, Q. Zhao, D. Meng, Y. Wang, D. Wang, and Z. Xu, "Low-rank matrix factorization under general mixture noise distributions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1493–1501. 1, 3, 4

[7] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1-2, pp. 89–97, 2004. 1

[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," in *SPIE Electronic Imaging*, 2006. 1

[9] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2862–2869. 1

[10] S. Roth and M. J. Black, "Fields of experts," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009. 1, 3, 5

[11] Y. Chen, W. Yu, and T. Pock, "On learning optimized reaction diffusion processes for effective image restoration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5261–5269. 1, 3, 4, 5, 6

[12] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, 2017. 1, 3, 5

[13] S. Lefkimmiatis, "Universal denoising networks: A novel cnn architecture for image denoising," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3204–3213. 1, 3, 5

[14] H. Ji and K. Wang, "Robust image deblurring with an inaccurate blur kernel," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1624–1634, 2012. 2, 4, 8, 9

[15] L. Xu, J. S. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Advances in Neural Information Processing Systems*, 2014, pp. 1790–1798. 2, 8, 9

[16] X. Fu, J. Huang, D. Z. Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 4, 10, 11

[17] A. Goldstein and R. Fattal, "Blur-kernel estimation from spectral irregularities," in *European Conference on Computer Vision*. Springer, 2012, pp. 622–635. 2

[18] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in *European Conference on Computer Vision*. Springer, 2010, pp. 157–170. 2, 4, 8, 9

Fig. 10: Visual quality of deraining results by transferring filters from SFARL models for deconvolution and denoising.

[19] W. Zuo, D. Ren, D. Zhang, S. Gu, and L. Zhang, "Learning iteration-wise generalized shrinkage–thresholding operators for blind deconvolution," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1751–1764, 2016. 2, 4

[20] W. Ren, X. Cao, J. Pan, X. Guo, W. Zuo, and M.-H. Yang, "Image deblurring via enhanced low-rank prior," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3426–3437, 2016. 2

[21] J. Pan, D. Sun, H. Pfister, and M.-H. Yang, "Blind image deblurring using dark channel prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1628–1636. 2

[22] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, "l_0-regularized intensity and gradient prior for deblurring text images and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 342–355, 2017. 2, 4

[23] Y.-L. Chen and C.-T. Hsu, "A generalized low-rank appearance model for spatio-temporally correlated rain streaks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1968–1975. 2, 4, 9, 10

[24] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2736–2744. 2, 4, 9, 10, 11, 12

[25] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3397–3405. 2, 4

[26] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *CVPR*, vol. 1, no. 2, 2017, p. 3. 2

[27] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang, "Deep video deblurring for hand-held cameras," in *CVPR*, vol. 2, 2017, p. 6. 2

[28] J. Pan, W. Ren, Z. Hu, and M.-H. Yang, "Learning to deblur images with exemplars," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 2

[29] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654. 2

[30] P. Getreuer, I. Garcia-Dorado, J. Isidoro, S. Choi, F. Ong, and P. Milanfar, "Blade: Filter learning for general purpose computational photography," in *ICCP*, 2018. 2

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778. 2

[32] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, "Deep joint rain detection and removal from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1357–1366. 2, 4, 10, 11

[33] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 8, 9

[34] S. Vasu, V. R. Maligireddy, and A. Rajagopalan, "Non-blind deblurring: Handling kernel uncertainty with cnns," in *Proceedings of the IEEE*

[35] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017. 2

Conference on Computer Vision and Pattern Recognition, 2018, pp. 3272–3281. 2, 4

[36] A. Eriksson and A. Van Den Hengel, "Efficient computation of robust low-rank matrix approximations in the presence of missing data using the l 1 norm," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 771–778. 3

[37] Q. Zhao, D. Meng, Z. Xu, W. Zuo, and L. Zhang, "Robust principle component analysis with complex noise," in *International Conference on Machine Learning*, 2014, pp. 55–63. 3, 4

[38] X. Chen, Z. Han, Y. Wang, Q. Zhao, D. Meng, and Y. Tang, "Robust tensor factorization with unknown noise," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5213–5221. 3

[39] P. Chen, N. Wang, N. L. Zhang, and D.-Y. Yeung, "Bayesian adaptive matrix factorization with automatic model selection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1284–1292. 3

[40] Z. Gong, Z. Shen, and K.-C. Toh, "Image restoration with mixed or unknown noises," *Multiscale Modeling & Simulation*, vol. 12, no. 2, pp. 458–487, 2014. 3

[41] F. Zhu, G. Chen, and P.-A. Heng, "From noise modeling to blind image denoising," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 420–429. 3

[42] J. Xu, D. Ren, L. Zhang, and D. Zhang, "Patch group based Bayesian learning for blind image denoising," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 79–95. 3

[43] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006. 3

[44] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth, "Discriminative non-blind deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 604–611. 3

[45] U. Schmidt, J. Jancsary, S. Nowozin, S. Roth, and C. Rother, "Cascades of regression tree fields for image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 677–689, 2016. 3

[46] L. Xiao, J. Wang, W. Heidrich, and M. Hirsch, "Learning high-order filters for efficient blind deconvolution of document photographs," in *European Conference on Computer Vision*. Springer, 2016, pp. 734–749. 3, 5

[47] S. Ryan Fanello, C. Keskin, P. Kohli, S. Izadi, J. Shotton, A. Criminisi, U. Pattacini, and T. Paek, "Filter forests for learning data-dependent convolutional kernels," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1709–1716. 3

[48] S. Cho and S. Lee, "Fast motion deblurring," *ACM Transactions on Graphics*, vol. 28, no. 5, p. 145, 2009. 4, 8

[49] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from

a single image," in *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, 2008, p. 73. 4

[50] D. Perrone, A. Ravichandran, R. Vidal, and P. Favaro, "Image priors for image deblurring with uncertain blur." in *The British Machine Vision Conference (BMVC)*, 2012, pp. 1–11. 4

[51] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, "Progressive inter-scale and intra-scale non-blind image deconvolution," in *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, 2008, p. 74. 4

[52] F. Heide, M. Rouf, M. B. Hullin, B. Labitzke, W. Heidrich, and A. Kolb, "High-quality computational imaging through simple lenses," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 5, p. 149, 2013. 4

[53] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004. 5, 6

[54] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1185–1198, 2011. 5, 6

[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015. 5, 8

[56] R. N. Bracewell and R. N. Bracewell, *The Fourier Transform and Its Applications*. McGraw-Hill, New York, 1986. 7

[57] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1964–1971. 8, 9

[58] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2. IEEE, 2001, pp. 416–423. 8

[59] O. Whyte, J. Sivic, and A. Zisserman, "Deblurring shaken and partially saturated images," *International journal of computer vision*, vol. 110, no. 2, pp. 185–201, 2014. 8, 9

[60] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8174–8182. 8, 9

[61] L.-W. Kang, C.-W. Lin, and Y.-H. Fu, "Automatic single-image-based rain streaks removal via image decomposition," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1742–1755, 2012. 9, 10

[62] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 633–640. 9, 10

[63] G. Schaefer and M. Stich, "UCID: An uncompressed color image database," in *SPIE Storage and Retrieval Methods and Applications for Multimedia*, 2004, pp. 472–480. 9

[64] G. Meng, Y. Wang, J. Duan, S. Xiang, and C. Pan, "Efficient image dehazing with boundary constraint and contextual regularization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 617–624. 10, 11
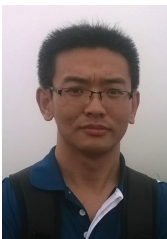
Dongwei Ren received the Ph.D. degrees in computer application technology from Harbin Institute of Technology and The Hong Kong Polytechnic University in 2017 and 2018, respectively. He is currently with the College of Intelligence and Computing, Tianjin University, China. His research interests include low level vision, deep learning and optimization methods.

Wangmeng Zuo (M'09, SM'15) received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. He is currently a Professor in the School of Computer Science and Technology, Harbin Institute of Technology. His current research interests include image enhancement and restoration, object detection, visual tracking, and image classification. He has published over 70 papers in top-tier academic journals and conferences. He has served as a Tutorial Organizer in ECCV 2016, an Associate Editor of the *IET Biometrics* and *Journal of Electronic Imaging*, and the Guest Editor of *Neurocomputing*, *Pattern Recognition*, *IEEE Transactions on Circuits and Systems for Video Technology*, and *IEEE Transactions on Neural Networks and Learning Systems*.

David Zhang graduated in Computer Science from Peking University. He received his MSc in 1982 and his PhD in 1985 in both Computer Science from the Harbin Institute of Technology (HIT), respectively. From 1986 to 1988 he was a Postdoctoral Fellow at Tsinghua University and then an Associate Professor at the Academia Sinica, Beijing. In 1994 he received his second PhD in Electrical and Computer Engineering from the University of Waterloo, Canada. He has been a Chair Professor at the Hong Kong Polytechnic University where he is the Founding Director of Biometrics Research Centre (UGC/CRC) supported by the Hong Kong SAR Government since 2005. Currently he is Presidential Chair Professor in Chinese University of Hong Kong (Shenzhen). He also serves as Visiting Chair Professor in Tsinghua University and HIT, and Adjunct Professor in Shanghai Jiao Tong University, Peking University and the University of Waterloo. He is both Founder and Editor-in-Chief, International Journal of Image & Graphics (IJIG) and Springer International Series on Biometrics (KISB); Organizer, the first International Conference on Biometrics Authentication (ICBA); and Associate Editor of more than ten international journals including IEEE Transactions and so on. Over past 30 years, he has been working on pattern recognition, image processing and biometrics, where many research results have been awarded and some created directions, including palmprint recognition, computerized TCM and facial beauty analysis, are famous in the world. So far, he has published over 20 monographs, 450 international journal papers and 40 patents from USA/Japan/HK/China. He has been continuously listed as a Highly Cited Researchers in Engineering by Clarivate Analytics in 2014, 2015, 2016, 2017 and 2018, respectively. Professor Zhang is a Croucher Senior Research Fellow, Distinguished Speaker of the IEEE Computer Society, and a Fellow of both IEEE and IAPR.

Lei Zhang (M'04, SM'14, F'18) received his B.Sc. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, and M.Sc. and Ph.D degrees in Control Theory and Engineering from Northwestern Polytechnical University, Xi'an, P.R. China, in 1998 and 2001, respectively. From 2001 to 2002, he was a research associate in the Department of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006 he worked as a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Department of Computing, The Hong Kong Polytechnic University, as an Assistant Professor. Since July 2017, he has been a Chair Professor in the same department. His research interests include Computer Vision, Image and Video Analysis, Pattern Recognition, and Biometrics, etc. Prof. Zhang has published more than 200 papers in those areas. As of 2019, his publications have been cited more than 40,000 times in literature. Prof. Zhang is a Senior Associate Editor of IEEE Trans. on Image Processing, and an Associate Editor of SIAM Journal of Imaging Sciences and Image and Vision Computing, etc. He is a "Clarivate Analytics Highly Cited Researcher" from 2015 to 2018. More information can be found in his homepage http://www4.comp.polyu.edu.hk/~cslzhang/.

Ming-Hsuan Yang is a Professor in Electrical Engineering and Computer Science at University of California, Merced. He received his PhD degree in computer science from the University of Illinois at Urbana-Champaign in 2000. Yang served as an associate editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence from 2007 to 2011, and is an associate editor of the International Journal of Computer Vision, Image and Vision Computing, and Journal of Artificial Intelligence Research. He received the Google Faculty Award in 2009 and the NSF CAREER Award in 2012. He is a senior member of the IEEE and the ACM.