The Visual Computer (2023) 39:1471–1484
https://doi.org/10.1007/s00371-022-02423-3

ORIGINAL ARTICLE

Learning adaptive hyper-guidance via proxy-based bilevel optimization for image enhancement

Jiaxin Gao[1,3] · Xiaokun Liu[1,3] · Risheng Liu[2,3] · Xin Fan[2,3]

Accepted: 23 January 2022 / Published online: 21 February 2022

Abstract

In recent years, image enhancement based on deep network plays a vital role and has become the mainstream research. However, current approaches are generally limited to the manual embedding of auxiliary components (e.g., hyper-parameters, appended modules) to train the network; thus, they can often lack flexibility, adaptability, or even fail to achieve the optimal settings. Moreover, the straightforward learning-based architectures cannot adequately handle the complex latent image distributions in real-world scenarios. To partially address the above issues, in this work, a generic adaptive hyper-training scheme based on bilevel optimization is established. Specifically, we propose a completely new bilevel deep-unfolded strategy to collaboratively optimize the inner-level task-related hyper-guidance and the outer-level image reconstruction. The process can embed the differentiable proxy-based network with parameters to automatically learn the appended control mechanism. Instead of constructing the empirically manual interventions, our strategy can proactively learn to learn self-adaptive auxiliary modules. Extensive experiments demonstrate the superiority of our strategy to address different image enhancement tasks (i.e., image restoration, image rain removal and image haze removal) in terms of flexibility and effectiveness.

Keywords Image enhancement · Bilevel learning · Low-level vision · Hyper-guidance

1 Introduction

Image enhancement aims to reconstruct the original image **x** from its degraded observation **y** as precisely as possible [1–8]. The traditional methodologies generally focus on designing a model-based optimization process to generate the task-specific image enhancement task [1–3], while in recent years, the mainstream image enhancement frameworks are generally established based on training hierarchical Convolutional Neural Network (CNN) [6–8]. These CNN-based methods incorporate sufficient information from the input image and utilize the end-to-end deep networks to directly estimate the clear images from the degraded observations, which completely ignore the domain knowledge in the training process. Inspired by this observation that the physical knowledge can effectively guide the learning process, knowledge-driven based deep unfolding methods have been in full swing [9–11]. They have attracted keen interests from researchers of different low-level tasks due to its practical significance. Both the early fast enhancement networks and the recent knowledge-driven unfolding methods have undoubtedly achieved high performance [4,5]. However, they mainly serve the limited scenarios, such as focusing only on the synthetic data. It can be seen that there exists a common problem for these approaches, that is, some auxiliary hyper-parameters or structures need to be manually adjusted or designed during the training process of deep neural models. In such case, they generally need to fiddly tune the hyper-parameters (e.g., noise level, rain region, depth map, etc.) to train different network models based on diverse data sets. In fact, these empirical manual interventions on the neural network may not make it optimally tuned when dealing with different data sets. It has been verified that this kind of empirical adjustment is challenging and easy to fall into degenerate solutions. Besides, it is inevitable to readjust the corresponding hyper-parameters for the data with different distributions, which is undoubtedly inefficient.

✉ Risheng Liu
rsliu@dlut.edu.cn

1 School of Software Technology, Dalian University of Technology, Dalian, China

2 International School of Information Science and Engineering, Dalian University of Technology, Dalian, China

3 Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, China

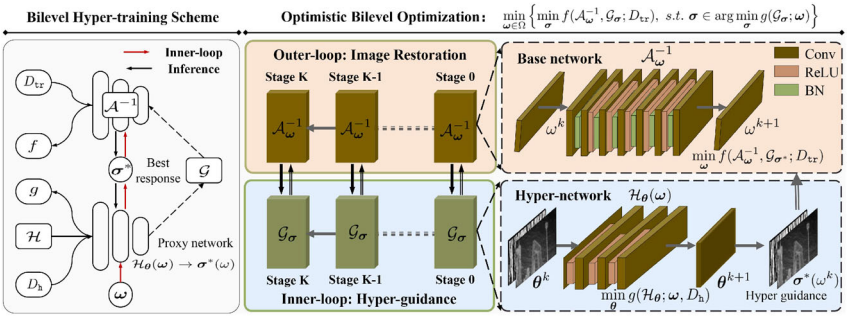


Fig. 1 The pipeline of our adaptive bilevel hyper-training strategy. We illustrate how to optimize inner-level hyper-guidance module with proxy-based network to develop collaborative training. Note that the base network and hyper-network should be set up differently depending on the specific image enhancement task

To partially address the above issues, this paper proposes a completely new bilevel unfolded paradigm to investigate image enhancement tasks with hyper-guidance for real-world low-level vision tasks, as shown in Fig. 1. Specifically, we first introduce a so-called hyper-guidance module to understand the appended propagative control mechanism and formulate task-related guidance for generic image reconstruction. Then we establish a hyper-training scheme to collaboratively train the parameters of image reconstruction and hyper-guidance, and wherein a proxy-based network is especially performed as a parametric function to approximate the optimal solution for the innerloop module. The convergence of our proposed inference framework can be trictly proved only under some mild conditions. To demonstrate the adaptability and flexibility of our work, we first solve a general image enhancement problem with adaptive noise levels and then extend our work for more complex image rain removal and image haze removal applications. Extensive quantitative and qualitative experimental results verify the superiority of our work against other state-of-the-art approaches on all the considered tasks. Overall, the main contributions of this work can be summarized in the following aspects.

- Based on the bilevel optimization programming, we establish a generic hyper-learning framework which introduces the task-related constraint as the hyper-guidance, to collaboratively optimize the inner-level task-related hyper-guidance and outer-level image reconstruction.
- To address the highly complicated bilevel programming, we design an effective strategy by performing a parametric proxy network to automatically learn the appended task-specific control mechanism.
- Extensive experiments on different image enhancement applications verified that our methodology can successfully handle different degradations on both synthetic and real datasets and obtain significantly better adaption ability than existing popular or state-of-the-art hierarchical CNN-based methods.

2 Related works

In this section, we briefly review some related works and the existing appended guidance strategies for image enhancement applications. Correspondingly, we also discuss the fundamental limitations of these approaches for different specific low-level tasks.

2.1 Image enhancement

In the past decades, different techniques have been investigated to design or learn specific low-level vision applications. For example, anisotropic partial differential equations have been introduced to address image denoising task [12,13]. Primitive methods also design energy models to formulate the domain knowledge of the image enhancement tasks and then adopt different optimization techniques to obtain iteration-based image reconstruction [1–3,14]. However, these classical methods are often computationally expensive and heavily dependent on sophisticate priors. With the development of deep learning, a number of CNN-based techniques have been utilized to build image reconstruction for different low-level vision tasks [4,15–20]. As the CNN-based methods depends on the sophisticated heuristic CNN architectures, they neglect specific vision task-related knowledge as a guidance to guide the network prediction. Thus the works in [5,6,21,22] try to combine CNN with optimization-based iterations to generate image reconstruction for different image enhancement applications. Unfortunately, although these methods employ additional knowledge to manipulate image reconstruction, the requirement to manually well-designed knowledge for specific data reduces flexibility and adaptability.

2.2 Appended guidance mechanism

Recent investigations also try to introduce appended components to guide image enhancement. For example, different mathematically deduced guidance have been incorporated into the image reconstruction model to preserve specific image priors or mutual structures [23,24]. As for CNN-based approaches, the work in [8] first introduces noise-level maps as guidance for image denoising task. Then a series of works also consider to train deep network together with different appended architectures for challenging low-level vision applications. For example, the work in [25] introduces another CNN module to learn noise-level maps as the guidance for real-world denoising tasks. The works proposed in [26,27] introduce the rain density map learned through another CNN as the guidance for single image rain removal task. All the above approaches have verified that introducing appended components can incorporate task priors to guide the training process of low-level vision tasks. Unfortunately, these works only use the heuristic trick to manually construct the guidance for reconstruction, and lack strict mathematical modeling and analysis. Moreover, the parameters of appended structures are not properly addressed during their training strategies. As a consequence, it is necessary to provide a new learning paradigm to formulate, understand and properly train image reconstruction with task-driven adaptive guidance for challenging low-level vision problems.

3 The proposed method

In this section, we first introduce the hyper-guidance module into reconstruction and mathematically model the guidance mechanism. Then we propose the bilevel programming model to properly learn the parameters of appended structures. Next by unfolding the bilevel optimization structure, we establish a proxy-based hyper-training scheme to collaboratively train the parameters of image reconstruction and hyper-guidance. Furthermore, we prove that our proposed inference framework is convergent only under some mild conditions.

3.1 Reconstruction with hyper-guidance

We consider the generic low-level vision problems modeled as $\mathbf{y} = \mathcal{A}(\mathbf{x})$, where $\mathbf{y}$ is the observed image, $\mathbf{x}$ denotes the latent clear image, and $\mathcal{A}$ is the task-related degradation process (e.g., blur, noise and rain streaks, just name a few). Then our goal is to establish an inverse model $\mathcal{A}^{-1}$ to obtain the desired solution $\mathbf{x}$, i.e., $\mathbf{x} = \mathcal{A}_{\boldsymbol{\omega}}^{-1}(\mathbf{y})$, where we denote $\boldsymbol{\omega}$ as the parameters of $\mathcal{A}^{-1}$. As discussed above, the most mainstream approaches just directly design/adopt various CNN architectures to build deep reconstruction as $\mathcal{A}^{-1}$ and then learn $\boldsymbol{\omega}$ by solving $\min_{\boldsymbol{\omega}} f(\mathcal{A}_{\boldsymbol{\omega}}^{-1}; D_{\mathtt{tr}})$ on dataset $D_{\mathtt{tr}} = \{\mathbf{y}_i, \mathbf{x}_i\}_{i=1}^N$ with some given training loss f (e.g., ℓ_1 or ℓ_2 errors) [5,17,18,28,29]. However, we observe that building heuristic networks as $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ to directly investigate complex image distributions is indeed a challenging task. More importantly, it is hard to introduce domain knowledge to help these networks to obtain specific task-related solutions. Inspired by this, we consider a more deliberate image reconstruction scheme

$$\mathbf{x} = \mathcal{A}_{\boldsymbol{\omega}}^{-1}(\mathbf{y}; \mathcal{G}_{\boldsymbol{\sigma}}), \tag{1}$$

in which a new module $\mathcal{G}_{\boldsymbol{\sigma}}$ with hyper-parameter $\boldsymbol{\sigma}$ is introduced to guide the reconstruction of $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$. In this way, we actually provide a flexible and interactive manner to formulate different task-requirements for image reconstruction. Hereafter, we call the model in Eq. (1) as Image Reconstruction with Hyper-Guidance (IRHG) and the particular forms of $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ and $\mathcal{G}_{\boldsymbol{\sigma}}$ for specific tasks will be discussed in Sec. 4. Some existing works [7,8] attempt to introduce task-related domain knowledge to help reconstruction, and they can be regarded as special cases of IRHG.

Remark 1 We argue that our hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$ in Eq. (1) is partially analogous to the hyper-parameters in learning scenarios. The hyper-guidance suitable for the specific data frequently demands artificial mine and well design. Inaccurate designed hyper-guidance even has a adverse impact on image reconstruction. Furthermore, the requirement for artificial control design reduces the flexibility of forth putting, especially in real scenes. The conventional training strategies which simultaneously optimize $\boldsymbol{\omega}$ and $\boldsymbol{\sigma}$ often give incorrect solutions as most of them may fail to account for the dependence of $\boldsymbol{\omega}$ on $\boldsymbol{\sigma}$ [30]. In addition, $D_{\mathtt{tr}}$ may not contain domain knowledge. Therefore, we should not naively consider $(\boldsymbol{\omega}, \boldsymbol{\sigma})$ as singleton and perform conventional training strategies on $D_{\mathtt{tr}}$ to obtain the image reconstruction.

3.2 Bilevel hyper-learning strategy

Considering the above limitations, we aim to develop a predominantly hierarchical structure with interactive guided reconstruction (decision-making) modules. In other words, the developed structure should satisfy that each module independently optimize the particular forms (i.e., $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ and $\mathcal{G}_{\boldsymbol{\sigma}}$), but is affected by the actions of the other module through externalities. Specifically, the developed structure should satisfy that the hyper-guidance behavior $\mathcal{G}_{\boldsymbol{\sigma}}$ may be able to influence the image enhancement scheme $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ and can be adaptively obtained. In this subsection, we propose a bilevel optimization learning paradigm which could satisfy

the above features of the form

$$\min_{\boldsymbol{\omega}\in\Omega}\left\{\min_{\boldsymbol{\sigma}} f(\mathcal{A}_{\boldsymbol{\omega}}^{-1},\mathcal{G}_{\boldsymbol{\sigma}};D_{\mathtt{tr}}),\ s.t.\ \boldsymbol{\sigma}\in\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}};\boldsymbol{\omega})\right\}, \tag{2}$$

where $g(\mathcal{G}_{\boldsymbol{\sigma}};\boldsymbol{\omega})$ is convex and denotes the given training loss, Ω is a compact subset of $\mathbb{R}^d$, and $f:\Omega\times\mathbb{R}^m\to\mathbb{R}$ is jointly continuous. Equation (2) is known as optimistic bilevel programming, has drawn increasing attention in bilevel optimization literature [31–34] and also been investigated in learning and vision fields recently [35–37]. The inner-level problem is the constraint on the hyper-parameter, while the outer-level problem is the constraint on reconstruction with the hyper-parameter calculated through the inner-level problem. Traditional machine learning algorithms solve Eq. (2) through gradient-free optimization strategies such as grid search, random search [38] or freeze–thaw Bayesian optimization [39]. As the solution of the inner-level is a high-dimensional optimization problem, it is difficult to compute precisely enough. Furthermore, these methods are computationally expensive, especially in high-dimensional image processing.

Specifically, given the variable $\boldsymbol{\omega}$, we first denote the corresponding best-response mapping as $\boldsymbol{\sigma}^*(\omega)$. Then, we define the inner-level problem as a simple bilevel subproblem[1] with respect to $\boldsymbol{\sigma}$ as $\min_{\boldsymbol{\sigma}} f(\mathcal{A}_{\boldsymbol{\omega}}^{-1},\mathcal{G}_{\boldsymbol{\sigma}};D_{\mathtt{tr}}),\ s.t.\ \boldsymbol{\sigma}\in\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}};\boldsymbol{\omega})$. Next, by defining the solution set of inner simple bilevel problem $\mathcal{S}(\boldsymbol{\omega})$, we could consider any $\boldsymbol{\sigma}^*(\boldsymbol{\omega})\in\mathcal{S}(\boldsymbol{\omega})$ as the best-response mapping, because all points in $\mathcal{S}(\boldsymbol{\omega})$ satisfy the minimum of $f(\mathcal{A}_{\boldsymbol{\omega}}^{-1},\mathcal{G}_{\boldsymbol{\sigma}};D_{\mathtt{tr}})$ in the condition that $\boldsymbol{\sigma}\in\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}};\boldsymbol{\omega})$. Finally, the outer-level problem can be given by the following value-function-based reformulation (a single-level optimization model w.r.t. $\boldsymbol{\omega}$), i.e.,

$$\min_{\boldsymbol{\omega}}\varphi(\boldsymbol{\omega}):=f(\mathcal{A}_{\boldsymbol{\omega}}^{-1},\mathcal{G}_{\boldsymbol{\sigma}^*(\omega)};D_{\mathtt{tr}}). \tag{3}$$

Indeed, by assuming that the minimizer of hyper-guidance subproblem (i.e., $\arg\min_{\boldsymbol{\sigma}} g$) in Eq. (2) is a singleton, we optimize the above bilevel optimization problem in Eq. (2) through two simplified subproblems. Specifically, it can be regarded as an additional introduced proxy-based network $\mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{\omega})$ (with extra parameter $\boldsymbol{\theta}$) to approximately learn the optimal hyper-parameter $\boldsymbol{\sigma}^*(\omega)$, expressed through the formula directly with a parametric function, i.e.,

$$\mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{\omega})\to\boldsymbol{\sigma}^*(\omega):=\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}};\boldsymbol{\omega}). \tag{4}$$

[1] It is known that the simple bi-level optimization is just a specific bilevel optimization problem with only one variable [35] ?,.

Algorithm 1 Bilevel deep-unfolded paradigm

Require: Training dataset $D_{\mathtt{h}}$, $D_{\mathtt{tr}}$ and K.
Ensure: Parameters $\boldsymbol{\omega}^K$ and hyper-parameters $\boldsymbol{\theta}^K$.
1: Initialize parameters $\boldsymbol{\omega}^0$;
2: **for** $k=0,1,2,\cdots,K$ **do**
3: Train $\boldsymbol{\theta}^{k+1}$ on $D_{\mathtt{h}}$ with fixed $\boldsymbol{\omega}^k$ using Eq. (6);
4: Train $\boldsymbol{\omega}^{k+1}$ on $D_{\mathtt{tr}}$ with fixed $\boldsymbol{\theta}^{k+1}$ using Eq. (5).
5: **end for**

Here, we further assume that $\mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{\omega})$ is bounded, and then, the original bilevel optimization model in Eq. (2) can be simplified as the following two subproblems:

$$\begin{cases}\boldsymbol{\sigma}^+=\mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{\omega})=\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}};\boldsymbol{\omega}),\\ \boldsymbol{\omega}^+=\arg\min_{\boldsymbol{\omega}} f(\mathcal{A}_{\boldsymbol{\omega}}^{-1},\mathcal{G}_{\boldsymbol{\sigma}^+};D_{\mathtt{tr}}).\end{cases} \tag{5}$$

Actually, by introducing a proxy-based network, solving the $\boldsymbol{\sigma}$-subproblem is equality to estimate the best parameter $\boldsymbol{\theta}$ with fixed $\boldsymbol{\omega}$. This is because the solution of proxy-based network $\mathcal{H}_{\boldsymbol{\theta}}$ with the best parameters could approximate the optimal hyper-parameter $\boldsymbol{\sigma}$. To obtain the optimal variable $\boldsymbol{\theta}$, we then formulate the training phase of the proxy network as the following formula

$$\min_{\boldsymbol{\theta}} g(\mathcal{H}_{\boldsymbol{\theta}};\boldsymbol{\omega},D_{\mathtt{h}}), \tag{6}$$

where $D_{\mathtt{h}}=\{\mathbf{y}_j,\mathbf{x}_j,\boldsymbol{\sigma}_j\}_{j=1}^M$ are the dataset for hyper-network training. Any optimizer such as SGD [40] or Adam [41] is applied to train the hyper-network without using the particular model to completion. This training process can be regarded as jointly optimizing $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$ through the above two steps. Note that the hyper-network learns the mapping from the input to the hyper-parameter by using the loss function $g(\mathcal{H}_{\boldsymbol{\theta}};\boldsymbol{\omega},D_{\mathtt{h}})$, which is differentiable and cheap to evaluate.

In fact, the naive training strategies which simultaneously optimize $\boldsymbol{\omega}$ and $\boldsymbol{\theta}$ based on the loss function $f+g$ often give incorrect solutions as they fail to account for the dependence of $\boldsymbol{\omega}$ on $\boldsymbol{\theta}$ [30]. In addition, only the knowledge including the dataset $D_{\mathtt{h}}$ can be used in the naive training strategies to optimize $\boldsymbol{\omega}$ and $\boldsymbol{\theta}$ simultaneously. We should not consider $(\boldsymbol{\omega},\boldsymbol{\sigma})$ as a singleton and perform naive training strategies to obtain the image reconstruction. Therefore, we propose the following bilevel deep-unfolded paradigm. For solving $\boldsymbol{\theta}$-subproblem denoted in Eq. (6), we train $\boldsymbol{\theta}$ through the training data $D_{\mathtt{h}}$ and the loss function $g(\mathcal{H}_{\boldsymbol{\theta}};\boldsymbol{\omega},D_{\mathtt{h}})$ based on a fixed $\boldsymbol{\omega}$. For solving the $\boldsymbol{\omega}$-subproblem denoted in Eq. (5), we train $\boldsymbol{\omega}$ through the training data $D_{\mathtt{tr}}$ and the loss function $f(\mathcal{A}_{\boldsymbol{\omega}}^{-1},\mathcal{G}_{\boldsymbol{\sigma}^+};D_{\mathtt{tr}})$ based on the fixed parameter $\boldsymbol{\theta}$ (it implies a fixed $\boldsymbol{\sigma}$). Consequently, the problem reduces to jointly optimize Eqs. (6) and (5). Subsequently, the bilevel deep-unfolded paradigm are summarized in Alg. 1 and the training details will be provided in Sect. 5.

3.3 Theoretical convergence analysis

In this subsection, we analyze the reasonableness and convergence of the developed Alg. 1. The proposed method use $\mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{\omega})$ to learn the hyper-parameter $\boldsymbol{\sigma}$, and then it is necessary to illustrate the validity of the introduced hyper-network. Indeed, the constructed hyper-network can learn continuous best-response functions that meet the solution of problem [i.e., Eq. (4)]. If we further assume $g(\mathcal{H}_{\boldsymbol{\theta}}; \boldsymbol{\omega}, D_{\mathrm{h}})$ is convex with respect to $\boldsymbol{\theta}$, referring to [42], it implies that for the uniform distribution of $\boldsymbol{\omega}$ in Ω, there exists $\boldsymbol{\theta}^*$ satisfying

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} 1/|\Omega| \sum_{\boldsymbol{\omega}\in\Omega} \left[g(\mathcal{H}_{\boldsymbol{\theta}}; \boldsymbol{\omega}, D_{\mathrm{h}})\right].$$

In other words, the hyper-network $\mathcal{H}_{\boldsymbol{\theta}^*}(\boldsymbol{\omega})$ could reach the minimization point of the $\boldsymbol{\sigma}$-subproblem for any given $\boldsymbol{\omega}$. The results are summarized as follows.

Theorem 1 *The sequence $\{(\boldsymbol{\omega}^k, \boldsymbol{\theta}^k)\}$ ($k \in \mathbb{N}$) generated by Alg. 1 jointly converges to the optimal solution of Eq. (5) (denoted as $(\boldsymbol{\omega}^*, \boldsymbol{\theta}^*)$).*

Proof Following the theoretical results in [43], for given sequence $\{(\boldsymbol{\omega}^k, \boldsymbol{\theta}^k)\}$, $k \in \mathbb{N}$, we have that there exist $\boldsymbol{\omega}^* \in \Omega$ and $\boldsymbol{\theta}^* \in \mathbb{R}^d$ such that $(\boldsymbol{\omega}^k, \boldsymbol{\theta}^k) \to (\boldsymbol{\omega}^*, \boldsymbol{\theta}^*)$ satisfying $f(\mathcal{A}_{\boldsymbol{\omega}^k}^{-1}, \mathcal{G}_{\boldsymbol{\theta}^k}; D_{\mathrm{tr}}) \to f(\mathcal{A}_{\boldsymbol{\omega}^*}^{-1}, \mathcal{G}_{\boldsymbol{\theta}^*}; D_{\mathrm{tr}})$. We next to show that the hyper-network could learn a minimizer of $g(\mathcal{G}_{\boldsymbol{\sigma}}; \boldsymbol{\omega})$.

Considering $\boldsymbol{\theta}^*$ as an optimal parameter, we have

$$1/|\Omega|\,\Sigma_{\boldsymbol{\omega}\in\Omega}\, g(\mathcal{H}_{\boldsymbol{\theta}^*}; \boldsymbol{\omega}, D_{\mathrm{h}}) = 1/|\Omega|\,\Sigma_{\boldsymbol{\omega}\in\Omega}[\min_{\boldsymbol{\theta}} g(\mathcal{H}_{\boldsymbol{\theta}}; \boldsymbol{\omega}, D_{\mathrm{h}})].$$

With the convex property of g and the Jensen's inequality, the above equality implies

$$1/|\Omega|\,\Sigma_{\boldsymbol{\omega}\in\Omega}\, g(\mathcal{H}_{\boldsymbol{\theta}^*}; \boldsymbol{\omega}, D_{\mathrm{h}}) \le \min_{\boldsymbol{\theta}} 1/|\Omega|\,\Sigma_{\boldsymbol{\omega}\in\Omega}\, g(\mathcal{H}_{\boldsymbol{\theta}}; \boldsymbol{\omega}, D_{\mathrm{h}}).$$

This means that the hyper-network $\mathcal{H}_{\boldsymbol{\theta}^*}$ could minimize the expected loss $g(\mathcal{H}_{\boldsymbol{\theta}}; \boldsymbol{\omega}, D_{\mathrm{h}})$ for uniform distribution. With the singleton assumption about $\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}}; \boldsymbol{\omega})$, we have $\boldsymbol{\sigma}^* = \mathcal{H}_{\boldsymbol{\theta}^*}(\boldsymbol{\omega})$ where $\boldsymbol{\sigma}^*$ is a minimum point of $g(\mathcal{G}_{\boldsymbol{\sigma}}; \boldsymbol{\omega})$. With the continuous of f and the singleton of $\arg\min_{\boldsymbol{\sigma}} g(\mathcal{G}_{\boldsymbol{\sigma}}; \boldsymbol{\omega})$, we have $f(\mathcal{A}_{\boldsymbol{\omega}^k}^{-1}, \mathcal{G}_{\boldsymbol{\sigma}^*}; D_{\mathrm{tr}}) \to f(\mathcal{A}_{\boldsymbol{\omega}^*}^{-1}, \mathcal{G}_{\boldsymbol{\sigma}^*}; D_{\mathrm{tr}})$. We finally to illustrate that $f(\mathcal{A}_{\boldsymbol{\omega}^*}^{-1}, \mathcal{G}_{\boldsymbol{\sigma}^*}; D_{\mathrm{tr}})$ is the minimization value of $f(\mathcal{A}_{\boldsymbol{\omega}}^{-1}, \mathcal{G}_{\boldsymbol{\sigma}^*}; D_{\mathrm{tr}})$. We optimize $\boldsymbol{\omega}$ by controlling the error $\|\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k\| \to 0$. This implies that $\boldsymbol{\omega}^{k+1} \to \boldsymbol{\omega}^*$ and $\mathbf{0} \in \nabla f(\mathcal{A}_{\boldsymbol{\omega}^*}^{-1}, \mathcal{G}_{\boldsymbol{\sigma}^*}; D_{\mathrm{tr}})$ as $\boldsymbol{\omega}^* \in \arg\min_{\boldsymbol{\omega}} f(\mathcal{A}_{\boldsymbol{\omega}}^{-1}, \mathcal{G}_{\boldsymbol{\sigma}^*}; D_{\mathrm{tr}})$. The proof is over here. □

Remark 2 In general, if the optimization problem is defined by minimizing the expected function for all distributions of $\boldsymbol{\omega}$ with convex support, i.e., $\boldsymbol{\sigma}^* \in \arg\min \mathbb{E}_{p(\boldsymbol{\omega})}[g(\mathcal{G}_{\boldsymbol{\sigma}}; \boldsymbol{\omega})]$, where $p(\boldsymbol{\omega})$ denotes the distribution of $\boldsymbol{\omega} \in \Omega$, the above result still holds. Indeed, the above mentioned uniform distribution is a specific case of $p(\boldsymbol{\omega})$.

4 Applications in image enhancement

In this section, we demonstrate how to apply bilevel unfolded strategy to tackle practical image restoration problems, single image rain removal and single image haze removal. For each task, we first introduce the physical model to generate low-quality images, and then we state task-related domain knowledge used as the hyper-guidance to embed the structure of the reconstruction network.

4.1 Image restoration

Image restoration aims to restore the latent high-quality image $\mathbf{x}$ from corrupted low-quality observation $\mathbf{y}$. The inverse problem leads to a discrete linear system of the form $\mathcal{A}(\mathbf{x}) = \mathbf{x} \otimes \mathbf{k} + \mathbf{n}$, where $\otimes$, $\mathbf{k}$ and $\mathbf{n}$ denote the convolution operation, blur kernel and additive noise, respectively. Specifically, the most commonly used deblurring formulation is the following regularized variational minimization energy function $\mathbf{x} = \mathcal{A}_{\boldsymbol{\omega}}^{-1}(\mathbf{y}) = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{x} \otimes \mathbf{k}\|^2 + \lambda\Phi(\mathbf{x}; \boldsymbol{\omega})$, where $\|\mathbf{y} - \mathbf{x} \otimes \mathbf{k}\|^2$ is the physical model, $\lambda\Phi(\mathbf{x}; \boldsymbol{\omega})$ is the regularization term with trade-off parameter λ.

In general, the CNN-based techniques have been popularly used to deal with this inverse problem and shown great success [5,7,22]. Although using the denoising prior as the regularization term has achieved great success [5], these CNN-based approaches may fail when dealing with large range appended noise and/or spatially variant noise which will be encountered constantly in real-world scenarios. Inspired by the fact that the noise level is a significant parameter to affect the performance of denoising prior, we introduce the noise level map $\boldsymbol{\sigma}$ as the hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$ to understand and control the image reconstruction. Specifically, by employing the noise level map as the hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$, we train the hyper-network $\mathcal{H}_{\boldsymbol{\theta}}$ under loss function $\|\boldsymbol{\sigma} - \boldsymbol{\sigma}_{gt}\|_2^2$ in Eq.(6) to learn the noise level map. The size of the noise level map is the same as the low-quality image, and each point of map represents the noise level of the corresponding pixel to the low-quality image. Then the map is introduced into the regularization term to guide the denoising prior under the given approximate noise level, so as to complete the image restoration task more adaptively. For the hyper-network $\mathcal{H}_{\boldsymbol{\theta}}$, we adopt a four-layer fully convolutional network. Hereafter, with the estimated map, the hyper-guidance module $\mathcal{G}_{\boldsymbol{\sigma}}$ guides the image reconstruction $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ to recover a higher quality-image more adaptively. Further, the image reconstruction structure $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ includes a deconvolution module and

a denoising module which is also composed of a CNN. The input of the denoising module is the concatenate of the low-quality images and the corresponding noise level maps.

Indeed, a few recent works focused on introducing noise level to assist with image restoration. For instance, in [5,7], the noise level was introduced into the model to generate the trade-off parameter or select the specific denoising network. These methods could be considered as a special case of our strategy, in which $\boldsymbol{\sigma}$ is manually introduced as the hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$ to guide $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$. To improve the adaptability and flexibility of the model, we train the hypernetwork under loss function $\|\boldsymbol{\sigma} - \boldsymbol{\sigma}_{gt}\|_2^2$ to learn noise level map and help denoising prior to better achieve denoising tasks. Different from these strategies, our method provided an adaptive mechanism in which the hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$ was learned through the proxy network $\mathcal{H}_{\boldsymbol{\theta}}$ to guide the image reconstruction $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$.

4.2 Single image rain removal

Single image rain removal focuses on removing the sporadic rain steak $\mathbf{n}$ and restoring rain free background scenes $\mathbf{x}$ from the given input rainy image $\mathbf{y}$. The physical model of observation $\mathbf{y}$ can be specified as $\mathcal{A}(\mathbf{x}) = \mathbf{x} + \mathbf{n}$. Most of the existing CNN-based approaches directly learn the reconstruction process given by $\mathcal{A}_{\boldsymbol{\omega}}^{-1}(\mathbf{y})$ from the rainy images to the clear images [44–47]. Almost all of these methods have significant performance only for specific data sets. However, these methods are often limited to be applicable on specific data sets and cannot achieve better performance in the real-world scenario.

To solve the above drawbacks, our study introduced the rain region map $\boldsymbol{\sigma}$ as the hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$ to guide the rain removal. Each point of the rain region map represents whether there are rain streaks to the corresponding pixel to the rainy image. The remaining experimental details including the structure of the hyper-network and the loss function are the same as the non-blind image deconvolution task. Hereafter, with the estimated map, the hyper-guidance module $\mathcal{G}_{\boldsymbol{\sigma}}$ guides the image reconstruction $\mathcal{A}_{\boldsymbol{\omega}}^{-1}$ to recover a clearer background image. It is worth noting that, here, we adopt PReNet proposed in [28] as our base reconstruction network. By concatenating the rainy image with the corresponding rain region binary map as input, the results indicate that the reconstruction network preserves more details while removing more rain streaks.

4.3 Single image haze removal

Single image haze removal aims to remove the interference due to the haze and recover the clear image $\mathbf{x}$ from the given input haze image $\mathbf{y}$. Most of these haze removal methods rely on the physical scattering model [48] formulated by $\mathcal{A}(\mathbf{x}) = \mathbf{tx} + (1 - \mathbf{t})\mathbf{A}$, where $\mathbf{t}$ is the transmission map and $\mathbf{A}$ is the global atmospheric light intensity. Furthermore, the transmission map can be formulated as $\mathbf{t} = \mathbf{e}^{-\mathbf{fid}}$, where $\mathbf{fi}$ and $\mathbf{d}$ are the atmosphere scattering parameter the scene depth, respectively. From this perspective, most physical scattering model based approaches depend on the estimation of the transmission map and the atmospheric light [49–52]. Indeed, these physical-model-free and CNN-based methods directly learn reconstruction $\mathcal{A}_{\boldsymbol{\omega}}^{-1}(\mathbf{y})$ from the hazy image to the clear image [29,53] and have shown excellent performance on the applicability of physical scattering model and the accuracy of feature estimation. However, most of them rely on the comprehensiveness of the training data and are not applicable to the real-world scenario.

To overcome the above shortcomings, we introduce the scene depth map $\boldsymbol{\sigma}$ as the hyper-guidance $\mathcal{G}_{\boldsymbol{\sigma}}$ to guide the image reconstruction. The size of scene depth map is the same as the hazy image, and each point of map represents the depth to the corresponding pixel to the hazy image. The loss function and structure of the hyper-network are the same as the above two applications. We adopt GridDehazeNet proposed in [29] as our reconstruction network. By applying the haze image concatenated with the corresponding scene depth as inputs, the reconstruction network could output more realistic clear images.

5 Experimental results

In this section, we conduct the experiments in three different representative low-level vision tasks. First, as for the image restoration task, we introduce the details of implementation followed by experiments on model evaluation and comparison with State-Of-The-Art (SOTA) methods. Then we illustrate how to apply our strategy to conduct experiments in other two low-level vision tasks, i.e., single image rain removal and single image haze removal tasks, followed by implementation details and comparison with related SOTA methods. All the experiments are conducted on a PC with Intel Core i9 CPU, 32G RAM, and a NVIDIA GeForce GTX 1080Ti GPU.

5.1 Model evaluation and analysis

5.1.1 Training details

We use the gray scale images from the Berkeley segmentation dataset [54] as the ground truth images. We also utilize the kernels20 [55] as the training blur kernels. Specifically, the clear images and kernels are convolved by adding the random Gaussian noise with levels ranging from [1%, 5%]. The size of noise level map is the same as low quality image. All elements in the map are noise level. At last, the images and noise level maps are cropped with the size of 160×160.

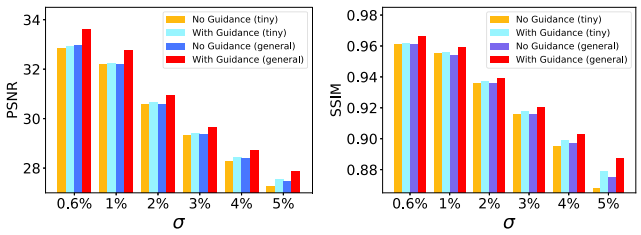


Fig. 2 Ablation experiments under two different setting pairs to illustrate the effectiveness of the hyper-guidance module $\mathcal{G}_{\sigma}$. One of the setting pairs is with (or without) guidance, and the other one is with different depth (i.e., tiny and general) of the propagation network

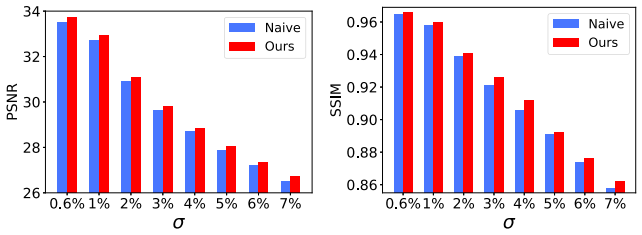


Fig. 3 Ablation analysis of different training strategies. We compare our hyper-training strategy with the joint training strategy (denoted as "Naive") under eight different noise levels

In the training process, we randomly generate 5 low-quality images together with its corresponding kernels as the batch of input. We also use Adam [41] as the optimizer and set β_1, β_2 as 0.9, 0.999, respectively. The learning rate is initialized as 10^{-4} for all the layers. Overall, there are 200,000 iterations performed for the training of each deep enhancement model.

We also use the bilevel hyper-training strategy to train the hyper-network and image reconstruction module. Specifically, we train the hyper-network with the fixed reconstruction network. Similarly, we also train the reconstruction network with a fixed hyper-network. Training behaviors of the guidance loss g in Eq. (6) and reconstruction loss f in Eq. (5) are shown in Fig. 2. It can be shown that the guidance loss converges faster to a smaller value and the hyper-network can guide the convergence of the reconstruction network.

5.1.2 Hyper-guidance verification

We set up two sets of ablation experiments for examining the modeling effectiveness. Firstly, we compare the depth of denoising module in image reconstruction. The tiny denoising module has only six Conv-layers, while the general denoising module has nine Conv-layers. Secondly, whether to introduce the hyper-guidance is exploited by the above two models, respectively. The method without introducing hyper-guidance is equivalent to solving $\mathbf{x} = \mathcal{A}_{\boldsymbol{\omega}}^{-1}(\mathbf{y})$. In other words, no knowledge about noise level is used in image reconstruction and we train the above models separately.

We test the above four models with the interval of [0.6%, 5%] noise levels on the Levin *et al'* benchmark [56], which includes 4 clear images and 8 blurry kernels. We set the size of all the images as 255×255. The experimental results are shown in Fig. 3. The results show that the performance of the method without introducing the hyper-guidance is worse. That is because the denoising module cannot efficaciously remove the larger range noise without the noise level knowledge. Comparing the depth of denoising module without introducing the hyper-guidance, simply improving the network depth already cannot effectively improve the performance. The performance will be significantly improve after introducing the hyper-guidance. The performance of tiny denoising module with the hyper-guidance can achieve or even exceed the general denoising module without the hyper-guidance. For the general reconstruction, it can be inferred that using hyper-guidance is more effective.

5.1.3 Training strategies evaluation

In the following, we compare the impact of training strategy on model performance. There have two loss functions, i.e., f and g described in Eq. (5) and Eq. (6), respectively. f is the constraint on the image propagation to generate the high-quality images and g is the constraint on the hyper-network to generate hyper-guidance. The loss function of naive training strategy is directly adding the above two loss functions together $h = f + g$. The training phase can be formulated as $\min_{\boldsymbol{\omega},\boldsymbol{\theta}} h(\mathcal{H}_{\boldsymbol{\theta}}, \mathcal{A}_{\boldsymbol{\omega}}^{-1}; D_{\mathrm{h}})$.

We train the models using hyper-training strategy and naive training strategy, respectively, and then test the two models with the noise levels interval of [0.6%, 7%] in the Levin *et al'* benchmark [56]. The results are shown in Fig. 4. The model based on our training strategy has better performance in all noise levels, even in the noise levels 0.6%, 6%, 7% that have not been used in the training process. The naive training strategy is to simultaneously optimize image propagation parameters $\boldsymbol{\omega}$ and hyper-network parameters $\boldsymbol{\theta}$, so the two parameters tightly restrict each other. Our training strategy is to first optimize $\boldsymbol{\theta}$ based on the fixed $\boldsymbol{\omega}$ and then optimize $\boldsymbol{\omega}$ based on the fixed $\boldsymbol{\theta}$ alternately. There are still relationships between the two parameters, but better solutions can be reached, respectively.

5.2 State-of-the-art comparisons

5.2.1 General comparison

We compared our method with other SOTA methods on the well-known Sun *et al'* benchmark [57] which includes 80 clear images with size range from 620×1024 to 928×1024. The kernels are the same as Levin *et al'* benchmark [56].

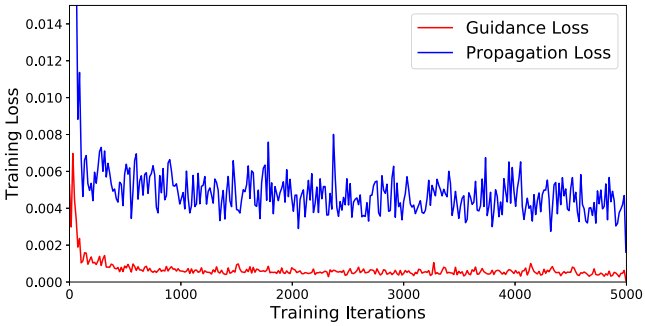


Fig. 4 Illustrating the training behaviors of guidance loss g in Eq. (6) and reconstruction loss f in Eq. (5)

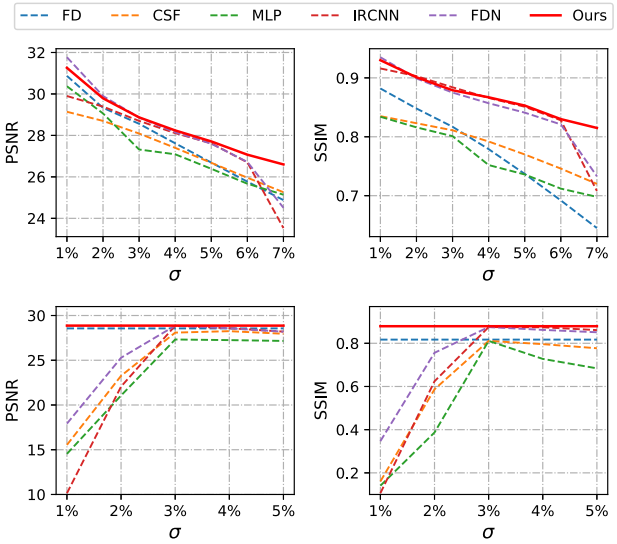


Fig. 5 Averaged quantitative performance on the Sun *et al'* benchmark. *Top* The abscissa is the noise level added to the low-quality image. *Bottom* The low-quality images are generated with 3% noise level. Partial comparison methods require the noise level as the experiment configuration, and the abscissa is the noise level as the deviation configuration of methods

To evaluate the adaptability of each method to the different noise levels, we test the methods in the noise levels interval of [1%, 7%]. Methods for comparison include FD [58], CSF [55], MLP [59], IRCNN [5], and FDN [7]. The recovery qualitative results are shown in top row of Fig. 5. It can be seen that our method achieves good performance, especially in higher noise levels. In the noise level 7% that has not appeared in the training, the decline of our method still retains great stability, but FDN and IRCNN have a sharp decline. The qualitative comparisons are shown in Fig. 6, and our method has restored more details.

5.2.2 Adaptive noise levels

Our method can adaptively obtain noise level through the hyper-network. Most existing methods (e.g., FDN, IRCNN, MLP, and CSF) require to take the noise level as one model configuration. Once the noise level is set as a deviation configuration, it will even have a negative effect on the recovered image. We added 3% noise to low quality images, and used 1%, 2%, 3%, 4% and 5% noise level as the configuration, respectively. Results are shown in the bottom row of Fig. 5. Our method is marked as a red line. Since our method and FD do not need to noise level as the configuration, the quantitative results are straight lines parallel to the X-axis. As for CSF, MLP, FDN and IRCNN, the performance is degraded when the noise level is set as the deviation configuration, especially when it set to smaller than standard.

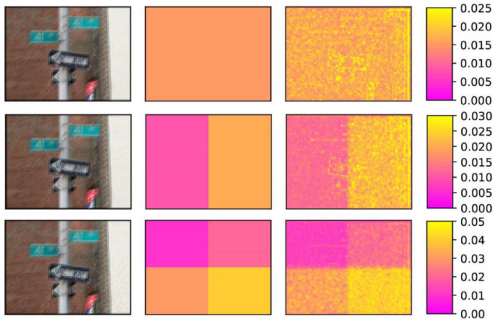


Fig. 6 Visual illustration of noise level maps. *The first column:* Low-quality images. *The second column:* The ground-truth noise level maps used to generate low-quality images. *The third column:* The noise level maps estimated through hyper-network

Another advantage of our method is that it can adaptively handle the spatially variant noise. At present, almost all image restoration methods are based on the fact that the overall noise level of the image is uniform, which is not applicable in real-world scenarios. Figure 7 shows the strong adaptability of the hyper-network in prediction for noise level. The corresponding noise level can be accurately predicted whether it is uniform noise or spatially variant noise. So that is can flexibly and accurately guide the image propagation. Furthermore, we use the Sun *et al'* dataset [57] for evaluation. For simplicity, we adopt a special way to add spatially variant level noises. Low-quality images are evenly divided into four parts: upper left, lower left, upper right, and lower right. Then we add variant levels of noise for different parts of the same image. Qualitative results are shown in Fig. 8. For the methods that use global uniform noise level as the model configuration, local details will be cleared when the local noise is less than the configured value. If the local noise level is less than the configuration, such local details will be erased. If the noise level is greater than the configuration, noise of such local will be retained. Our method can adaptively remove the spatially variant noise through hyper-guidance, balance denoising and details retention.

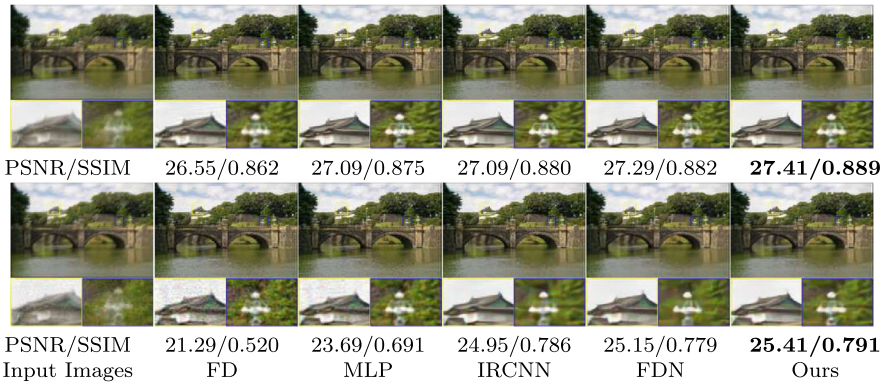


Fig. 7 Image restoration results on one example image with 5% noise level and corresponding results

Fig. 8 Qualitative comparison on challenging images generated with spatially variant noise for image restoration

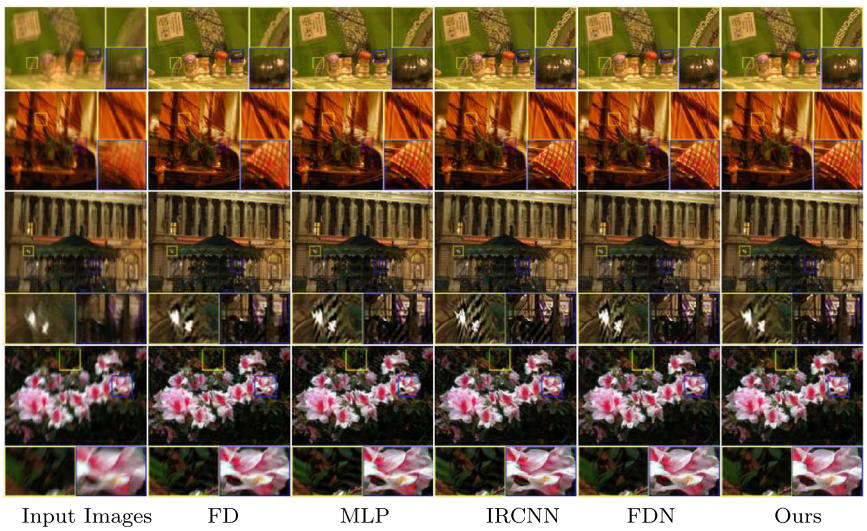


Fig. 9 Qualitative comparison on challenging images for real-world blind image restoration

5.2.3 Blind image restoration

In practical applications, non-blind restoration is part of the blind restoration in which the truth blur kernel is unknown. Since the ground truth image is also unknown, we provide the visual comparison with the SOTA methods. In particular, we adopt the method [60] to estimate the rough blur kernel and then use the estimated blur kernel to recover clear images. As shown in Fig. 9, we illustrate the qualitative comparison of our method and other SOTA methods based on the estimated kernel. It can be seen that the images recovered by our method preserve more details while removing more blur and noise.

5.3 Single image rain removal

We train the reconstruction network using the negative SSIM loss [61] and conduct experiments in two classic data sets: Rain100L with respect to light rainy images and Rain100H [46] with respect to heavy rainy dataset. Rain100L is a dataset about light rainy images, and the training set includes 200 rainy images, clear images and corresponding binary map representing the location of the rain. The testing set includes 100 rainy images and corresponding clear images. As for heavy rainy dataset of Rain100H, the number of training set is 1800 and the number of testing set is

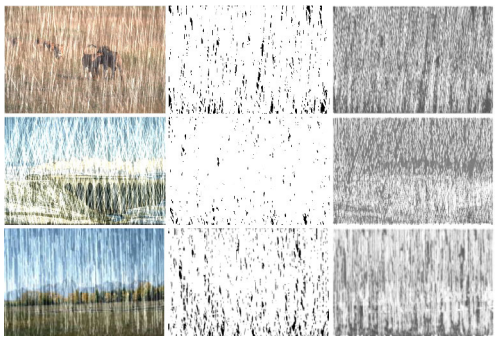


Fig. 10 Visual illustration of rainy images and rain region maps. *Left column:* Rainy images. *Middle column:* Ground truth rain region. *Right column:* Rain region estimated through hyper-network

100. In Fig. 10, we illustrate the rainy image, the ground truth rain region and the rain region estimated through the hyper-network. Although the ground truth rain region is not accurate enough under the heavy rain, the estimated rain region is even relatively accurate. It can be seen that our strategy could precisely comprehend the location of the rain through the estimated rain region, so as to remove the rain streaks more accurately.

Table 1 shows the quantitative results on Rain100L and Rain100H with comparison of a series of SOTA methods (i.e., GMM [23], DDN [44], RGN [45], JORDER [46], RESCAN [47] and PReNet [28]). There is a significant improvement compared with the baseline PReNet, especially in the heavy rain data set Rain100H. Because in the rainy images with a larger rain streak, it is difficult to characterize the relationship of transformations between rainy image to clear image by only the propagation network, and the rain region map can be used to guide the propagation network to better restore background.

Qualitative results are shown in Fig. 11. It can be seen that our method can effectively restore the details when the background is heavily occluded by the rain streak, while others render a darker color. This is mainly because that when the structure of background is similar with rain streak, other methods may mistake them for rain streaks and remove them. We also conducted experiments on real-world rainy images collected from [62], as shown in Fig. 12. It can be seen that our method can remove rain streak and preserve details correctly compared with SOTA methods which fail to remove the rain streaks.

5.4 Single image haze removal

For the single image haze removal task, we train the full network by using combination of the smooth ℓ_1 loss and the perceptual loss. We conduct the experiments in the synthetic dataset RESIDE proposed in [63], which contains two partial data in both indoor and outdoor scenarios. The indoor

Table 1 Averaged quantitative performance on Rain100L and Rain100H

Methods	Metric	[23]	[44]	[45]	[46]	[47]	[28]	*ours*
Rain100L	PSNR	28.66	32.16	33.16	36.61	–	37.48	**37.58**
	SSIM	0.865	0.936	0.963	0.974	–	**0.979**	**0.979**
Rain100H	PSNR	15.05	21.92	25.25	26.54	28.88	29.46	**30.21**
	SSIM	0.425	0.764	0.841	0.835	0.866	0.899	**0.907**

The best results are highlighted in **bold**

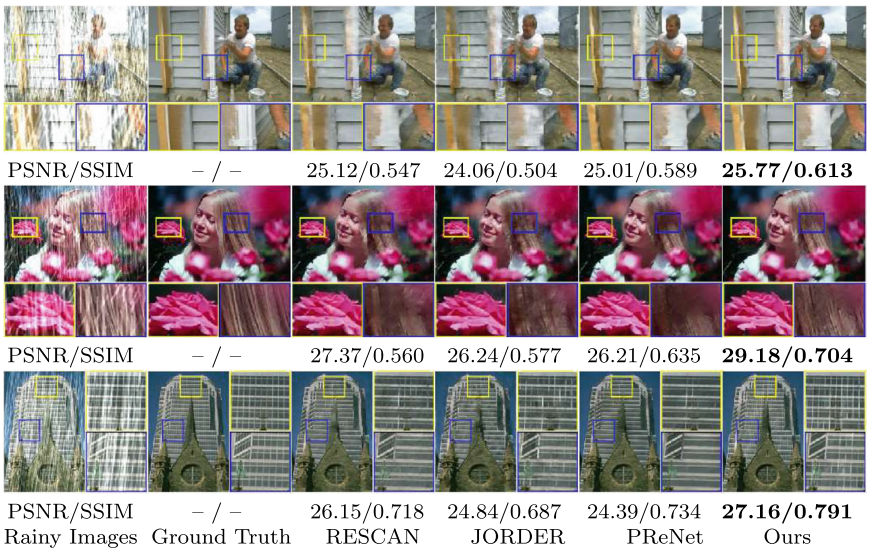


Fig. 11 Qualitative comparison on three example rainy images in Rain100H [46]. The quantitative scores are reported below each image

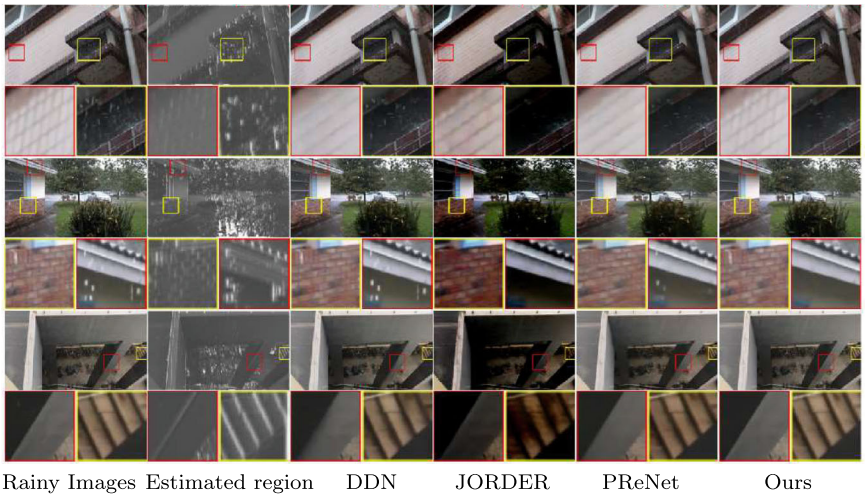


Fig. 12 Qualitative comparison of four challenging images on image rain removal. The input images are real-world rainy images collected from [62]

Fig. 13 Visual illustration of hazy images and depth maps. *Left column:* Hazy images. *Middle column:* Ground truth depth map. *Right column:* Depth map estimated through hyper-network

training set of RESIDE contains a total of 13990 hazy indoor images which are generated from 1399 clear images. The scene depth maps **d** are calculated from the NYU Depth V2 [64] and Middlebury Stereo datasets [65]. We calculate the scene depth maps **d** based on the algorithm proposed by [66]. For testing, the Synthetic Objective Testing Set (SOTS) is adopted, which consists of 500 indoor hazy images and 500 outdoor haze images.

To show the hyper-guidance more intuitively, Fig. 13 illustrates the hazy image, the ground truth depth map and the depth map estimated through the hyper-network. The ground truth scene depth map is coarse-grained, and the estimated depth map is fine-grained which is more suitable for the propagation network. Table 2 shows the quantitative results on the synthetic dataset. Comparison SOTA methods include DCP [49], DehazeNet [50], MSCNN [51], AOD-Net [52], EPDN [53] and GridDehazeNet [29]. Fig. 14 shows the qualitative comparisons on synthetic outdoor images. It can be observed that these methods still remain some haze and suffer from color distortion where the scene is usually darker than ground truth. Our method can remove haze as much as possible and make result closer to ground truth image.

6 Conclusion

In order to effectively design and control the reconstruction behaviors for deep neural network in a principled and interpretable manner, this work establishes a generic adaptive hyper-guidance module to understand and formulate the appended control of training behaviors for low-

Table 2 Averaged quantitative performance on SOTS

Methods	Metric	[49]	[50]	[51]	[52]	[53]	[29]	*Ours*
Indoor	PSNR	16.61	19.82	19.84	20.51	25.06	32.16	**32.19**
	SSIM	0.8546	0.8209	0.8327	0.8162	0.9232	0.9836	**0.9839**
Outdoor	PSNR	19.14	24.75	22.06	24.14	22.57	30.86	**30.90**
	SSIM	0.8605	0.9269	0.9078	0.9198	0.8630	0.9819	**0.9822**

The best results are highlighted in **bold**

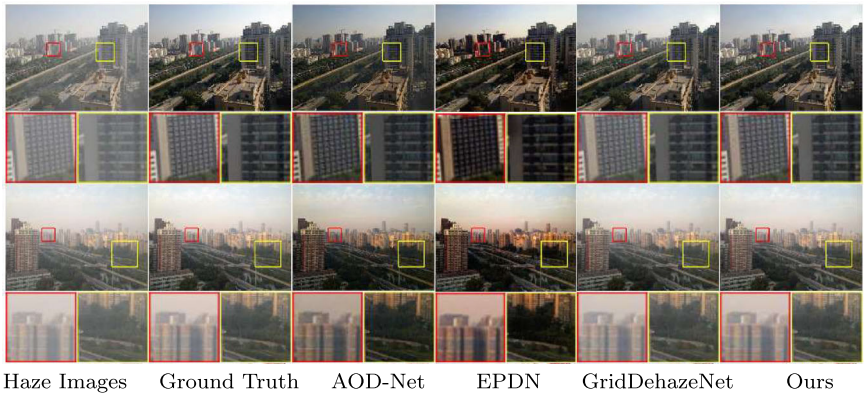


Fig. 14 Qualitative comparison on three example hazy images in SOTS [63]

level vision problems. Then we propose a completely new bilevel unfolded hyper-learning strategy which embeds by a proxy-based network to collaboratively optimize the hyper-guidance and image reconstruction. To demonstrate the adaptability and flexibility of our strategy, we first solve the image restoration problem with adaptive noise levels and then extend for more complex single image rain removal and single image haze removal applications. Extensive quantitative and qualitative experimental results verify the superiority of our method against other state-of-the-art approaches on all the considered tasks.

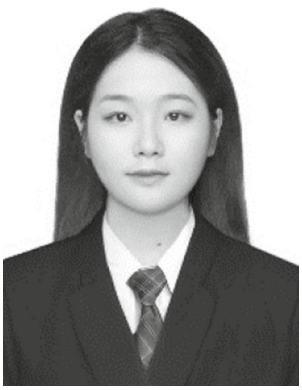
Declarations

Conflict of interest We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, and there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled, "Learning Adaptive Hyper-guidance via Proxy-based Bilevel Optimization for Image Restoration".

References

1. Simoes, M., Almeida, L.B., Bioucas-Dias, J., Chanussot, J.: A framework for fast image deconvolution with incomplete observations. IEEE Trans. Image Process. **25**(11), 5266–5280 (2016)
2. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM J. Imaging Sci. **1**(3), 248–272 (2008)
3. Cheng, J., Gao, Y., Guo, B., Zuo, W.: Image restoration using spatially variant hyper-laplacian prior. Signal Image Video Process. **13**(1), 155–162 (2019)
4. Liu, D., Wen, B., Fan, Y., Loy, C. C., Huang, T. S.: Non-local recurrent network for image restoration. In NeurIPS (2018)
5. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In CVPR, (2017)
6. Tirer, T., Giryes, R.: Image restoration by iterative denoising and backward projections. IEEE Trans. Image Process. **28**(3), 1220–1234 (2019)
7. Kruse, J., Rother, C., Schmidt, U.: Learning to push the limits of efficient fft-based image deconvolution. In ICCV, (2017)
8. Zhang, K., Zuo, W., Zhang, L.: Ffdnet: toward a fast and flexible solution for cnn-based image denoising. IEEE Trans. Image Process. **27**(9), 4608–4622 (2018)
9. Liu, R., Jiang, Z., Fan, X., Luo, Z.: Knowledge-driven deep unrolling for robust image layer separation. IEEE Trans. Neural Netw. Learn. Syst. **31**(5), 1653–1666 (2019)
10. Zhang, K., Gool, L. V., Timofte, R.: Deep unfolding network for image super-resolution. In CVPR (2020)
11. Liu, R., Cheng, S., Ma, L., Fan, X., Luo, Z.: Deep proximal unrolling: algorithmic framework, convergence analysis and applications. IEEE Trans. Image Process. **28**(10), 5013–5026 (2019)
12. Liu, R., Lin, Z., Zhang, W., Su, Z.: Learning pdes for image restoration via optimal control. In ECCV, (2010)
13. Tai, X. C., Lie, K. A., Chan, T. F., Osher, S.: Image processing based on partial differential equations. In Proceedings of the International Conference on PDE-Based Image Processing and Related Inverse Problems. Springer Science and Business Media, Berlin (2006)
14. Liu, R., Cheng, S., He, Y., Fan, X., Lin, Z., Luo, Z.: On the convergence of learning-based iterative methods for nonconvex inverse problems. IEEE Trans. Pattern Anal. Mach. Intell. **42**(12), 3027–3039 (2019)
15. Tao, X., Gao, H., Shen, X., Wang, J., Jia, J.: Scale-recurrent network for deep image deblurring. In CVPR, (2018)
16. Ren, W., Zhang, J., Ma, L., Pan, J., Cao, X., Zuo, W., Liu, W., Yang, M.H.: Deep non-blind econvolution via generalized low-rank approximation. In NeurIPS (2018)
17. Cai, J., Zuo, W., Zhang, L.: Extreme channel prior embedded network for dynamic scene deblurring. arXiv:1903.00763 (2019)
18. Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., Yang, M. H., Shao, L.: Cycleisp: Real image restoration via improved data synthesis. arXiv:2003.07761, (2020)
19. Guo, Y., Chen, J., Wang, J., Chen, Q., Cao, J., Deng, Z., Xu, Y., Tan, M.: Closed-loop matters: Dual regression networks for single image super-resolution. arXiv:2003.07018 (2020)
20. Liu, R., Fan, X., Hou, M., Jiang, Z., Luo, Z., Zhang, L.: Learning aggregated transmission propagation networks for haze removal and beyond. IEEE Trans. Neural Netw. Learn. Syst. **30**(10), 2973–2986 (2018)
21. Liu, R., Ma, L., Wang, Y., Zhang, L.: Learning converged propagations with deep prior ensemble for image enhancement. IEEE Trans. Image Process. **28**(3), 1528–1543 (2018)
22. Liu, R., Cheng, S., Liu, X., Ma, L., Fan, X., Luo, Z.: A bridging framework for model optimization and deep propagation. In NeurIPS (2018)
23. Li, Y., Tan, R. T., Guo, X., Lu, J., Brown, M. S.: Rain streak removal using layer priors. In CVPR (2016)
24. Xiaojie Guo, Yu., Li, J.M., Ling, H.: Mutually guided image filtering. IEEE Trans. Neural Netw. Learn. Syst. **42**(3), 694–707 (2020)

25. Guo, S., Yan, Z., Zhang, K., Zuo, W., Zhang, L.: Toward convolutional blind denoising of real photographs. In CVPR (2019)
26. Zhang, H., Patel, V. M.: Density-aware single image de-raining using a multi-stream dense network. In CVPR (2018)
27. Du, Y., Xu, J., Qiu, Q., Zhen, X., Zhang, L.: Variational image deraining. In WACV (2020)
28. Ren, D., Zuo, W., Hu, Q., Zhu, P., Meng, D.: Progressive image deraining networks: a better and simpler baseline. In CVPR (2019)
29. Liu, X., Ma, Y., Shi, Z., Chen, J.: Griddehazenet: attention-based multi-scale network for image dehazing. In ICCV (2019)
30. MacKay, M., Vicol, P., Lorraine, J., Duvenaud, D., Grosse, R.: Self-tuning networks: Bilevel optimization of hyperparameters using structured best-response functions. arXiv:1903.03088 (2019)
31. Dempe, S.: Foundations of Bilevel Programming. Springer Science and Business Media, Berlin (2002)
32. Dempe, S., Dutta, J., Mordukhovich, B.S.: New necessary optimality conditions in optimistic bilevel programming. Optimization **56**(5–6), 577–604 (2007)
33. Kohli, B.: Optimality conditions for optimistic bilevel programming problem using convexifactors. J. Optim. Theory Appl. **152**(3), 632–651 (2012)
34. Lampariello, L., Sagratella, S.: Numerically tractable optimistic bilevel problems. Comput. Optim. Appl. **76**(2), 277–303 (2020)
35. Liu, R., Mu, P., Yuan, X., Zeng, S., Zhang, J.: A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In ICML (2020)
36. Liu, R., Mu, P., Yuan, X., Zeng, S., Zhang, J.: A generic descent aggregation framework for gradient-based bi-level optimization. In ICML (2021)
37. Liu, R., Liu, X., Yuan, X., Zeng, S., Zhang, J.: A value-function-based interior-point method for non-convex bi-level optimization (2021)
38. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. JMLR **13**(2), 281–305 (2012)
39. Swersky, K., Snoek, J., Adams, R. P.: Freeze-thaw bayesian optimization. arXiv:1406.3896 (2014)
40. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In COMPSTAT (2010)
41. Kingma, D. P., Ba, J.: A method for stochastic optimization. arXiv:1412.6980, (2014)
42. Lorraine, J., Duvenaud, D.: Stochastic hyperparameter optimization through hypernetworks arXiv:1802.09419 (2018)
43. Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., Pontil, M.: Bilevel programming for hyperparameter optimization and meta-learning. In ICML (2018)
44. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.: Removing rain from single images via a deep detail network. In CVPR (2017)
45. Fan, Z., Wu, H., Fu, X., Hunag, Y., Ding, X.: Residual-guide feature fusion network for single image deraining. arXiv:1804.07493 (2018)
46. Yang, W., Tan, R. T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In CVPR (2017)
47. Li, X., Wu, J., Lin, Z., Liu, H., Zha, H.: Recurrent squeeze-and-excitation context aggregation net for single image deraining. In ECCV (2018)
48. Narasimhan, S.G., Nayar, S.K: Chromatic framework for vision in bad weather. In CVPR (2000)
49. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. IEEE Trans. Neural Netw. Learn. Syst. **33**(12), 2341–2353 (2010)
50. Cai, B., Xiangmin, X., Jia, K., Qing, C., Tao, D.: Dehazenet: an end-to-end system for single image haze removal. IEEE Trans. Image Process. **25**(11), 5187–5198 (2016)
51. Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., Yang, M. H.: Single image dehazing via multi-scale convolutional neural networks. In ECCV (2016)
52. Li, B., Peng, X., Wang, Z., Xu, J., Feng, D.: Aod-net: All-in-one dehazing network. In ICCV (2017)
53. Qu, Y., Chen, Y., Huang, J., Xie, Y.: Enhanced pix2pix dehazing network. In CVPR, (2019)
54. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2010)
55. Schmidt, U., Jancsary, J., Nowozin, S., Roth, S., Rother, C.: Cascades of regression tree fields for image restoration. IEEE Trans. Pattern Anal. Mach. Intell. **38**(4), 677–689 (2015)
56. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In CVPR (2009)
57. Sun, L., Cho, S., Wang, J., Hays, J.: Edge-based blur kernel estimation using patch priors. In ICCP (2013)
58. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In NeurIPS (2009)
59. Schuler, C. J., Christopher Burger, H., Harmeling, S., Scholkopf, B.: A machine learning approach for non-blind image deconvolution. In CVPR (2013)
60. Pan, J., Lin, Z., Su, Z., Yang, M. H. Robust kernel estimation with outliers handling for image deblurring. In CVPR (2016)
61. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., et al.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
62. Wang, T., Yang, X., Xu, K., Chen, S., Zhang, Q., Lau, R. W.: Spatial attentive single-image deraining with a high quality real rain dataset. In CVPR (2019)
63. Li, B., Ren, W., Dengpan, F., Tao, D., Feng, D., Zeng, W., Wang, Z.: Benchmarking single-image dehazing and beyond. IEEE Trans. Image Process. **28**(1), 492–505 (2018)
64. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In ECCV, (2012)
65. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In CVPR (2003)
66. Liu, F., Shen, C., Lin, G., Reid, I.: Learning depth from single monocular images using deep convolutional neural fields. IEEE Trans. Pattern Anal. Mach. Intell. **38**(10), 2024–2039 (2015)



Jiaxin Gao received the B.S. degree in Applied Mathematics from Dalian University of Technology, China, in 2018. She is currently pursuing the PhD degree in software engineering at Dalian University of Technology. She is with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, Dalian, China. Her research interests include computer vision, machine learning and optimization.

Xiaokun Liu received the B.S. degree in Software Engineering from Dalian University of Technology, China, in 2018. He also received the M.S. degree in computer science from Dalian University, Dalian, China, in 2021. He was also with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology. His research interests include computer vision, image fusion, and deep learning.

Risheng Liu received the B.S. and Ph.D. degrees both in mathematics from the Dalian University of Technology in 2007 and 2012. He was a visiting scholar in the Robotic Institute of Carnegie Mellon University from 2010 to 2012. He served as Hong Kong Scholar Research Fellow at the Hong Kong Polytechnic University from 2016 to 2017. He is currently a professor with DUT-RU International School of Information Science & Engineering, Dalian University of Technology. He was awarded the "Outstanding Youth Science Foundation" of the National Natural Science Foundation of China. His research interests include machine learning, optimization and computer vision.

Xin Fan (Senior Member, IEEE) was born in 1977. He received the B.E. and Ph.D. degrees in information and communication engineering from Xi'an Jiaotong University, Xi'an, China, in 1998 and 2004, respectively. He was with Oklahoma State University at Stillwater, Stillwater, OK, USA, from 2006 to 2007, as a Postdoctoral Research Fellow. He joined the School of Software, Dalian University of Technology, Dalian, China, in 2009. His current research interests include computational geometry and machine learning, and their applications to low-level image processing and diffusion tensor imaging magnetic resonance image analysis.