

Blind Super-Resolution via Meta-learning and Markov Chain Monte Carlo Simulation

Jingyuan Xia, Zhixiong Yang, Shengxi Li, *Member, IEEE*, Shuanghui Zhang, *Member, IEEE*, Yaowen Fu, Deniz Gündüz, *Fellow, IEEE* and Xiang Li,

Abstract—Learning based approaches have witnessed great successes in blind single image super-resolution (SISR) tasks, however, handcrafted kernel priors and learning based kernel priors are typically required. In this paper, we propose a Meta-learning and Markov Chain Monte Carlo based SISR approach to learn kernel priors from organized randomness. In concrete, a lightweight network is adopted as kernel generator, and is optimized via learning from the MCMC simulation on random Gaussian distributions. This procedure provides an approximation for the rational blur kernel, and introduces a network-level Langevin dynamics into SISR optimization processes, which contributes to preventing bad local optimal solutions for kernel estimation. Meanwhile, a meta-learning based alternating optimization procedure is proposed to optimize the kernel generator and image restorer, respectively. In contrast to the conventional alternating minimization strategy, a meta-learning based framework is applied to learn an adaptive optimization strategy, which is less-greedy and results in better convergence performance. These two procedures are iteratively processed in a plug-and-play fashion, for the first time, realizing a learning-based but plug-and-play blind SISR solution in unsupervised inference. Extensive simulations demonstrate the superior performance and generalization ability of the proposed approach when comparing with state-of-the-arts on synthesis and real-world datasets.

Index Terms—Blind single image super resolution, Markov Chain Monte Carlo simulation, Meta-learning.

1 INTRODUCTION

SINGLE image super-resolution (SISR) plays a crucial role in image processing society. It tends to reconstruct high-resolution (HR) images from the low-resolution (LR) observations. With the fact that the degradation model is typically unknown in real-world scenarios, growing studies begin to predict the blur kernels and the HR images simultaneously, known as the blind SISR problem. A common mechanism to solve the blind SISR problem is underlying an alternating optimization between two sub-problems, *kernel estimation* and *image restoration*, which are iteratively minimized until the HR image is restored. Kernel estimation is a pivotal step in solving blind SISR problems, which determines the HR image reconstruction performance, and thereby becomes the centrality study of this paper.

Most of the existing blind SISR approaches can be categorized as model-based approaches and learning-based approaches. Model-based approaches [1], [2], [3], [4], [5], [6] enjoys better generalization-ability because of the explicit degradation modeling and gradient-based solution scheme. However, these methods suffer from the ill-posedness and

non-convexity of the blind SISR problem, and typically stuck at the bad local mode during optimization. Therefore, learning-based methods [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17] obtain popularity in recent years, which realize significantly better performance via substituting explicit degradation model by network-based learning procedure on a large amount of labelled data, therefore, providing data-driven image and kernel priors for solving the non-convex blind SISR optimization problem. Nevertheless, these methods are still restricted by high data dependency on training samples and dedicated time-consumption for model-training in real-world applications.

In general, we claim that there is a trade-off between the reconstruction performance on individual task and the generalization capacity on different scenarios in blind SISR. On the one hand, stronger priors definitely bring better performance, as the learning-based methods achieve. Meanwhile, the generalized flexibility to real-world applications will be declined since strong priors typically lead to serious overfitting. On the other hand, rational kernel priors and adaptive optimization strategy, instead of exhaustively minimizing each individual sub-problem, are necessary to handle the intrinsic non-convexity and ill-posedness of blind SISR problems. The latest meta-learning based optimization algorithms [18], [19], [20], [21] have substantiate significant advances on solving non-convex optimization problems, in particular, for those are with alternating minimization (AM) framework, just as blind SISR problem does. In light of the meta-learning mechanism that learns to extract mutual knowledge for a “bird’s eye view” on global scope optimization, these algorithms achieve considerably better convergence performance on non-convex geometry. The common idea of meta-learning-based optimization strategies lies on

- Jingyuan Xia, Zhixiong Yang, Shuanghui Zhang, Yaowen Fu and Xiang Li are with College of Electronic Engineering, National University of Defense Technology, Changsha, 410073, China. Shengxi Li is with College of Electronic Engineering, Beihang University, Beijing, 100191, China. Deniz Gündüz is with the Department of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, UK, and the ‘Enzo Ferrari’ Department of Engineering, University of Modena and Reggio Emilia, Italy.
- E-mail: (j.xia16, shengxi.li17, d.gunduz@imperial.ac.uk, (yzx21, zhangshuanghui, fuyaowen, lixiang@nudt.edu.cn.
- Jingyuan Xia and Zhixiong Yang contributed equally to this work (Corresponding authors: Zhixiong Yang and Shuanghui Zhang).

incorporating organized randomness to prevent trapping into bad local optima, and a network-based optimizer is meta-learned across iterations to ensure the incorporated random disturbance following the primary objective of the optimization problem. In this instance, this paper strives to realize a novel blind SISR solution scheme, in which rational kernel priors and adaptive optimization strategy are gained through learning from organized randomness without cumbersome training in advances as well as data dependency.

In this paper, we propose a Meta-learning and Markov Chain Monte Carlo (MLMC) based approach to solve the blind SISR problem. It establishes a two phases SISR optimization, including a *Markov Chain Monte Carlo kernel approximation* (MCKA) phase and a *meta-learning based alternating optimization* (MLAO) phase. In the MCKA phase, a MCMC simulation on random Gaussian distributions is proposed to substitute traditional model-based or learning-based kernel priors, which are either handcrafted or deterministic via pre-training. In contrary, the proposed MCKA aims to learn task-general kernel priors from random Gaussian distributions, and thereby achieves pre-training and labelled data free. A Markov Chains system modeling on optimization trajectory of the blind SISR problem is proposed to provide *organized randomness*, which allows the Monte Carlo simulations to sample Gaussian distributions with a global scope on LR image reconstruction errors, therefore, providing rational kernel priors for better convergence performance. In concrete, a lightweight network is iteratively optimized in an unsupervised manner using MCMC simulations to approximately generate blur kernel, which is trained across random Gaussian distributions. In this way, good generalization properties and flexibility towards arbitrary SISR tasks are ensured, and this organized randomness leads to a relaxation that prevents the kernel estimation trapping into bad local modes. Moreover, the MLAO phase refines blur kernel and restores HR image, alternatively, with respect to the task-specific LR observation. Instead of the exhaustive minimization on each individual sub-problem, the optimizer for kernel estimation is meta-learned by minimizing the accumulated LR image reconstruction errors over MLAO iterations. Then, an adaptive and effective optimization strategy for better convergence performance is obtained by learning the mutual knowledge of solving a sequence of MLAO sub-problems. These two phases are operated alternatively to realize a balance between task-general relaxation via learning from organized randomness and task-specific refinement.

The advantages of the proposed MLMC are listed below: i) *Plug-and-play*. The MCKA phase effectively substitutes the deterministic kernel priors by a plug-and-play learning process through MCMC simulation on Gaussian distributions. This results in negligible time-consumption and circumvents the cumbersome demand of labelled training samples. ii) *Better convergence performance*. Though without training in advances, the MLAO phase achieves an adaptive non-convex optimization strategy that converges to better optimum for the estimated blur kernels. Meanwhile, the MCKA phase can be regarded as a network-level Langevin dynamics [6], [22], [23] towards the kernel estimation in the MLAO phase, which provides a rational and trainable

random disturbance with the kernel estimation via learning from random Gaussian distributions. This also ensures the convergence of kernel estimation. iii) *Stronger generalization capacity and flexibility*. The obtained kernel priors from the MCMC simulation are loose and dynamic, which endows better generalization capability on different degradation scenarios. Besides, the plug-and-play fashion evades the data dependency and re-training requests, therefore, significantly improving the flexibility in practical applications. The main contributions of this paper are listed below:

- An universal statistic framework for network-based degradation model is proposed to elaborate the network-based approach for SISR problem. On the basis of this, an MCMC simulation model on random Gaussian distributions with a glimpse on LR image reconstruction error is established to elaborate a new kernel approximation phase.
- Different to the commonly-applied pre-training or manually designed kernel priors, a random kernel learning scheme replaces the ordinary kernel priors, which realizes a learning-based but plug-and-play kernel prior generation paradigm and contributes to a commonly network-level Langevin dynamics optimization for convergence improvements.
- A meta-learning-based adaptive strategy is constructed to solve the blind SISR problem. It learns to optimize the non-convex and ill-posed blind SISR problem in a less-greedy optimization strategy, and thereof ensures better convergence performance towards ground truth when only depending on the observed LR image.
- To the best of our knowledge, MLMC is the first learning-based but plug-and-play SISR approach that achieves superior performance and can be directly applied to common kernel estimation tasks including isotropic and anisotropic Gaussian, non-Gaussian and motion kernels, with competitive number of parameters, runtime and memory usage comparing to the state-of-the-art, as well as robustness to the noise.

2 RELATED WORK

2.1 Blind SISR methods

The existing blind SISR approaches can be roughly categorised into model-based and learning-based approaches.

Model-based methods. Most of the early model-based blind SISR approaches [1], [24], [25], [26] merely aims to construct specific HR images priors with explicit formulations, such as gradient profile [27], hyper-Laplacian [28], sparsity [29] and total variation (TV) [30], for better reconstruction performance. More recent studies begin to focus on kernel priors designing. For example, Jin *et al.* [31] propose a popular heuristic normalization as kernel prior which realizes better convergence. Yue *et al.* [6] employ an explicit pre-defined Gaussian kernel model, achieving good robustness on noise scenarios. It iteratively optimizes the Gaussian kernel and the input noise via a gradient-based algorithm, which causes limitation towards capturing varying kernel categories. In sum, model-based approaches are typically with good

generalization-ability due to the explicit modelling on task-specific priors. Meanwhile, significant performance decline will appear when the manually designed kernel priors slightly deviate to the degradation model in application. Besides, due to the high ill-posedness and non-convexity of the blind SISR problem, model-based methods can easily get stuck at local optimums via the hand-crafted and rule-fixed gradient descent based optimization algorithms. These urge researchers turn their attentions to learning-based alternatives for better convergence performance and more flexibility.

Learning-based methods. The arise of deep learning has motivated the learning-based blind SISR approaches to learn kernel/image priors via network-based behavior. These models are typically pre-trained on paired LR/HR image samples to obtain image and kernel priors. Typically, end-to-end deep networks are adopted to formulate the degradation model [32], and therewith bring about plenty of deep convolutional neural network (CNN) based SISR optimizers [9], [33], [34], [35], [36], [37], [38], [39], [40], [41]. Specifically, Liang *et al.* [42] propose an end-to-end deep network that estimates kernel for different patches in LR images via residual blocks. Fang *et al.* [39] establish a two-stage framework for solving blind SR task with a statistical modelling on LR image, which learns mean and variance from LR image estimate kernel. To improve the generalization-ability towards diverse degradations of real image, Zhang *et al.* [38] design a practical degradation model with shuffled blur for the synthetic training data. More recently, Xia *et al.* [40] propose a knowledge distillation based implicit degradation estimator for blind SR task, allowing better generalization towards different degradations. Deep image prior (DIP) [2] is one of the most well-known model that is capacity of learning image features in an unsupervised way with superior performance on solving computer vision tasks. However, it suffers from limited generalization-ability on different degradation models when dealing with blind SISR problems. In this instance, recent works [43], [44], [45], [46], [47] introduce meta-learning for better generalization-ability via training across different samples in terms of image sizes, degradation categories and resolution ratios. Nevertheless, these methods demand large amount of training samples and time-consuming dedicated pre-training phase. Meanwhile, other works [11], [42], [48], [49], [50], [51], [52] strive to expand the conventional model-based iterative solutions with learning-based behavior via unfolded deep learning algorithms. Gu *et al.* [53] propose an network-based alternative optimization framework that simultaneously optimizes the blur kernel and HR image via deep models. Zhang *et al.* [50] formulate a fundamental deep unfolding framework for SR task, which unfolds the MAP inference via a half-quadratic splitting algorithm. More recently, Liang *et al.* [13] establish a flow-based kernel prior (FKP) model that realizes an improved performance through pre-training a kernel estimator as non-parametric priors, and incorporates it with the existing SISR approaches. Approaches in [41], [52] formulate deep-unfolding-based model on the basis of half quadratic splitting algorithm and proximal gradient descent algorithm.

In a sum, the latest blind SISR methods illustrate a trend of reducing the dependency on training resource (pre-

training stage, external training data) and strengthening the generalization-ability on varying degradation categories. Although the existing works, such as FKP-DIP [13], in which the necessity of large amount of images is replaced by kernel samples, have achieved less pre-training demand, they still require hours to train their model in advance and the performance is determined by the training samples. To reduce the kernel prior dependency, this work designs a MCMC simulation on random Gaussian distributions to provide an approximation for the blur kernel, while neither pre-training nor labelled data is necessary, and thus improve its generalization-ability towards different degradation categories.

2.2 Meta-learning-based Non-convex Optimization

Differ to the conventional deep learning methods that focus on optimizing each task exhaustively, meta-learning methods [18], [19], [20], [21] prefer to leverage the mutual experience of solving different tasks to obtain good generalization-ability and adaptability for the trained model. For example, Li *et al.* [54] propose a metric-based meta-learning that shares a hierarchical optimization structure to handle the task-specific information and task-across knowledge. Recently, Xia *et al.* [21] introduce a randomness perturbations with gradient descent algorithm principle, which is realized by meta-learning the optimization trajectory during iterations. Inspired by [21], Yang *et al.* [55] further extend the idea of introducing random perturbations for better convergence performance in a plug-and-play fashion on solving another formous non-convex and ill-posed weighted sum rate problem in wireless communication society, and attains state-of-the-art performance with competitive computational complexity. In a nutshell, these approaches aim to balance the to allow better convergence in the non-convex problems. Motivated by this idea, this paper employs a meta-learning scheme for solving SISR problem, hoping to achieve better convergence in the blind SISR problem.

3 NOTATION AND PROBLEM STATEMENT

3.1 Notation

We denote vectors and matrices by bold letters. The superscript represents the index of the iteration, while the subscript indicates practical meaning of variables. We denote the neural network model by $G(\cdot)$ with parameters ϕ . In the rest of this paper, we define variable with $*$ on superscript as the well-optimized results. For the degradation model, let $\mathbf{y}$ denotes the down-sampled LR image, $\mathbf{x}$ denotes the HR image, $\otimes$ indicates the convolution operation, $\downarrow_s$ denotes the down-sampling operation with scale factor s , and $\mathbf{k}$ denotes the blur kernel. We define $\mathbf{n} \sim \mathcal{N}(0, \sigma^2)$ as a zero-mean white Gaussian noise with variance σ^2 . Let $\|\cdot\|_F$ denotes the Frobenius norm.

3.2 Model-based Degradation Model

The fundamental degradation model of image super-resolution [56], [57], [58] is typically formulated as follows

$$\mathbf{y} = (\mathbf{x} \otimes \mathbf{k}) \downarrow_s + \mathbf{n}. \quad (1)$$

Following the maximum a posteriori (MAP) framework, the blind SISR problem can be formulated as a MAP problem:

$$\max_{\mathbf{x}, \mathbf{k}} p(\mathbf{y}|\mathbf{x}, \mathbf{k})p(\mathbf{x})p(\mathbf{k}). \quad (2)$$

In Eq. (2), $p(\mathbf{y}|\mathbf{x}, \mathbf{k})$ denotes the likelihood of the observed LR image, $p(\mathbf{k})$ denotes the kernel prior, and $p(\mathbf{x})$ is the image prior. When optimizing the log-likelihood of (2), the problem can also be expressed as

$$\arg \min_{\mathbf{x}, \mathbf{k}} \|\mathbf{y} - (\mathbf{x} \otimes \mathbf{k})\|_s^2 + \lambda \Phi(\mathbf{x}) + \beta \Omega(\mathbf{k}), \quad (3)$$

where $\Phi(\mathbf{x})$ and $\Omega(\mathbf{k})$ are the prior information functions with weights λ and β , respectively. In the model-based methods, a projected alternation minimization (PAM) [1] based iterative optimization between $\mathbf{x}$ and $\mathbf{k}$ is typically applied to solve problem (3) in the following form

$$\begin{cases} \mathbf{x}^* = \arg \min_{\mathbf{x}} \|\mathbf{y} - (\mathbf{x} \otimes \mathbf{k})\|_s^2 + \lambda \Phi(\mathbf{x}), \\ \mathbf{k}^* = \arg \min_{\mathbf{k}} \|\mathbf{y} - (\mathbf{x} \otimes \mathbf{k})\|_s^2 + \beta \Omega(\mathbf{k}), \end{cases} \quad (4)$$

$$\text{subject to } \|\mathbf{k}\|_1 = 1, \mathbf{k} > 0, \mathbf{x} > 0,$$

where the constraints are satisfied via projected gradient descent at each iteration step, more details of the PAM formulation could be found in [1].

3.3 Network-based Degradation Model

Black-box based Framework Most of network-based degradation models, such as SRCNN [32], RCAN [59] and DASR [12], estimate HR image by an end-to-end deep network $G_{\mathbf{x}}(\cdot)$. Let ϕ denotes the parameters of the deep network, then the optimization problem (3) is converted to

$$\phi^* = \arg \min_{\phi} \sum_{\mathbf{x}_{gt}^j, \mathbf{y}^j \in \mathcal{D}_{\mathbf{x}}} \|\mathbf{x}_{gt}^j - G(\mathbf{y}^j, \phi)\|_F^2, \quad (5)$$

where $\mathbf{x}_{gt}^j$ is the ground truth HR image paired with LR observation $\mathbf{y}^j$ in training dataset $\mathcal{D}_{\mathbf{x}} = \{\mathbf{x}_{gt}^j, \mathbf{y}^j\}_{j=1}^J$.

Double-DIP Framework Following the AM-based framework in (4), Double-DIP framework [4], [60] establishes two networks $G_{\mathbf{x}}(\cdot)$ and $G_{\mathbf{k}}(\cdot)$ with parameters $\phi_{\mathbf{x}}$ and $\phi_{\mathbf{k}}$ to estimate HR image and blur kernel by taking fixed random noises $\mathbf{z}_{\mathbf{x}}$ and $\mathbf{z}_{\mathbf{k}}$ as input in the following form

$$\begin{cases} \phi_{\mathbf{x}}^* = \arg \min_{\phi_{\mathbf{x}}} \|\mathbf{y} - (G_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}}) \otimes \mathbf{k})\|_s^2, \\ \phi_{\mathbf{k}}^* = \arg \min_{\phi_{\mathbf{k}}} \|\mathbf{y} - (\mathbf{x} \otimes G_{\mathbf{k}}(\mathbf{z}_{\mathbf{k}}, \phi_{\mathbf{k}}))\|_s^2. \end{cases} \quad (6)$$

FKP-DIP Framework Differ to Double-DIP framework, FKP-DIP [13] learns kernel prior via pre-training $\phi_{\mathbf{k}}$ on a labelled kernel dataset $\mathcal{D}_{\mathbf{k}}$ as follows,

$$\begin{cases} \phi_{\mathbf{k}}^* = \arg \min_{\phi_{\mathbf{k}}} \sum_{\mathbf{k}_{gt}^j \in \mathcal{D}_{\mathbf{k}}} \|\mathbf{k}_{gt}^j - G(\mathbf{z}_{\mathbf{k}}, \phi_{\mathbf{k}})\|_F^2, \\ \phi_{\mathbf{x}}^* = \arg \min_{\phi_{\mathbf{x}}} \|\mathbf{y} - (G_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}}) \otimes \mathbf{k})\|_s^2, \\ \mathbf{z}_{\mathbf{k}}^* = \arg \min_{\mathbf{z}_{\mathbf{k}}} \|\mathbf{y} - (\mathbf{x} \otimes G_{\mathbf{k}}(\mathbf{z}_{\mathbf{k}}, \phi_{\mathbf{k}}^*))\|_s^2, \end{cases} \quad (7)$$

where $\mathbf{k}_{gt}^j$ is the ground truth blur kernel, and $\phi_{\mathbf{k}}^*$ denotes the pre-trained network parameters with respect to labelled kernel dataset $\mathcal{D}_{\mathbf{k}} = \{\mathbf{k}_{gt}^j\}_{j=1}^J$.

4 THE PROPOSED MLMC APPROACH

In this paper, a statistic formulation for network-based degradation model is introduced, and a MLMC approach is proposed to optimize the blind SISR problem in an alternative framework, containing an MCKA stage for kernel approximation and an MLAO stage alternatives on kernel and HR image estimation.

4.1 Problem Formulation for Network-based Degradation Model

Following the concept of the network-based degradation model that learns to optimize HR image and blur kernel via network-behavior, the optimization of $\mathbf{x}$ and $\mathbf{k}$ are converted into the optimization of parameters $\phi_{\mathbf{x}}$ and $\phi_{\mathbf{k}}$.

Mathematically, according to the Bayes theorem, we have

$$p(\mathbf{x}) = p(\mathbf{x}|\phi_{\mathbf{x}})p(\phi_{\mathbf{x}})/p(\phi_{\mathbf{x}}|\mathbf{x}), \quad (8)$$

$$p(\mathbf{k}) = p(\mathbf{k}|\phi_{\mathbf{k}})p(\phi_{\mathbf{k}})/p(\phi_{\mathbf{k}}|\mathbf{k}). \quad (9)$$

Then, the primary MAP problem (2) can be reformulated as

$$\max_{\phi_{\mathbf{x}}, \phi_{\mathbf{k}}} \frac{p(\mathbf{y}|\mathbf{x}, \mathbf{k})p(\mathbf{x}|\phi_{\mathbf{x}})p(\phi_{\mathbf{x}})p(\mathbf{k}|\phi_{\mathbf{k}})p(\phi_{\mathbf{k}})}{p(\phi_{\mathbf{x}}|\mathbf{x})p(\phi_{\mathbf{k}}|\mathbf{k})}. \quad (10)$$

Given the fact that the obtained HR image $\mathbf{x}$ and $\mathbf{k}$ are determined by the parameters $\phi_{\mathbf{x}}$ and $\phi_{\mathbf{k}}$, respectively, thus $p(\phi_{\mathbf{x}}|\mathbf{x}) = 1$ and $p(\phi_{\mathbf{k}}|\mathbf{k}) = 1$. Then problem (10) can be further reformulated as

$$\max_{\phi_{\mathbf{x}}, \phi_{\mathbf{k}}} p(\mathbf{y}|\mathbf{x}, \mathbf{k})p(\mathbf{x}|\phi_{\mathbf{x}})p(\mathbf{k}|\phi_{\mathbf{k}})p(\phi_{\mathbf{x}})p(\phi_{\mathbf{k}}). \quad (11)$$

Given that $p(\mathbf{y}|\mathbf{x}, \mathbf{k})p(\mathbf{x}|\phi_{\mathbf{x}})p(\mathbf{k}|\phi_{\mathbf{k}}) = p(\mathbf{y}, \mathbf{x}, \mathbf{k}|\phi_{\mathbf{x}}, \phi_{\mathbf{k}})$, then we have

$$\max_{\phi_{\mathbf{x}}, \phi_{\mathbf{k}}} p(\mathbf{y}, \mathbf{x}, \mathbf{k}|\phi_{\mathbf{x}}, \phi_{\mathbf{k}})p(\phi_{\mathbf{x}})p(\phi_{\mathbf{k}}). \quad (12)$$

When rewriting problem (12) in logarithm form, an universal statistical framework for network-based degradation model is given by

$$\max_{\phi_{\mathbf{x}}, \phi_{\mathbf{k}}} \log p(\mathbf{y}, \mathbf{x}, \mathbf{k}|\phi_{\mathbf{x}}, \phi_{\mathbf{k}}) + \log p(\phi_{\mathbf{x}}) + \log p(\phi_{\mathbf{k}}), \quad (13)$$

where the first term denotes a maximum log-likelihood estimation for parameters $\phi_{\mathbf{x}}$ and $\phi_{\mathbf{k}}$ with respect to the joint probability of LR observation $\mathbf{y}$, HR image $\mathbf{x}$ and blur kernel $\mathbf{k}$. The second and third terms in (13) denote the image prior and kernel prior, respectively.

Specifically, the first term is typically demonstrated by a minimization problem of LR image reconstruction error with respect to alternatively optimizing parameters $\phi_{\mathbf{x}}$ and $\phi_{\mathbf{k}}$, as what Double-DIP does in (6). The black-box based end-to-end deep learning solutions, such as SRCNN [32], RCAN [59] and DASR [12], are typically pre-trained on labelled HR image dataset $\mathcal{D}_{\mathbf{x}}$, consequently resulting in a strong image prior. These methods tend to learn an effective image prior from labelled data, referring to $p(\phi_{\mathbf{x}}|\mathcal{D}_{\mathbf{x}})$. Meanwhile, classical image priors are easy to be applied to $p(\phi_{\mathbf{x}})$ in the form of regularization or constraint aligned with LR reconstruction error. In contrast, the latest blind SISR approach, FKP-DIP, tends to learn kernel prior instead of image prior in the form of trainable parameters by pre-trained on kernel dataset $\mathcal{D}_{\mathbf{k}}$, referring to $p(\phi_{\mathbf{k}}|\mathcal{D}_{\mathbf{k}})$.

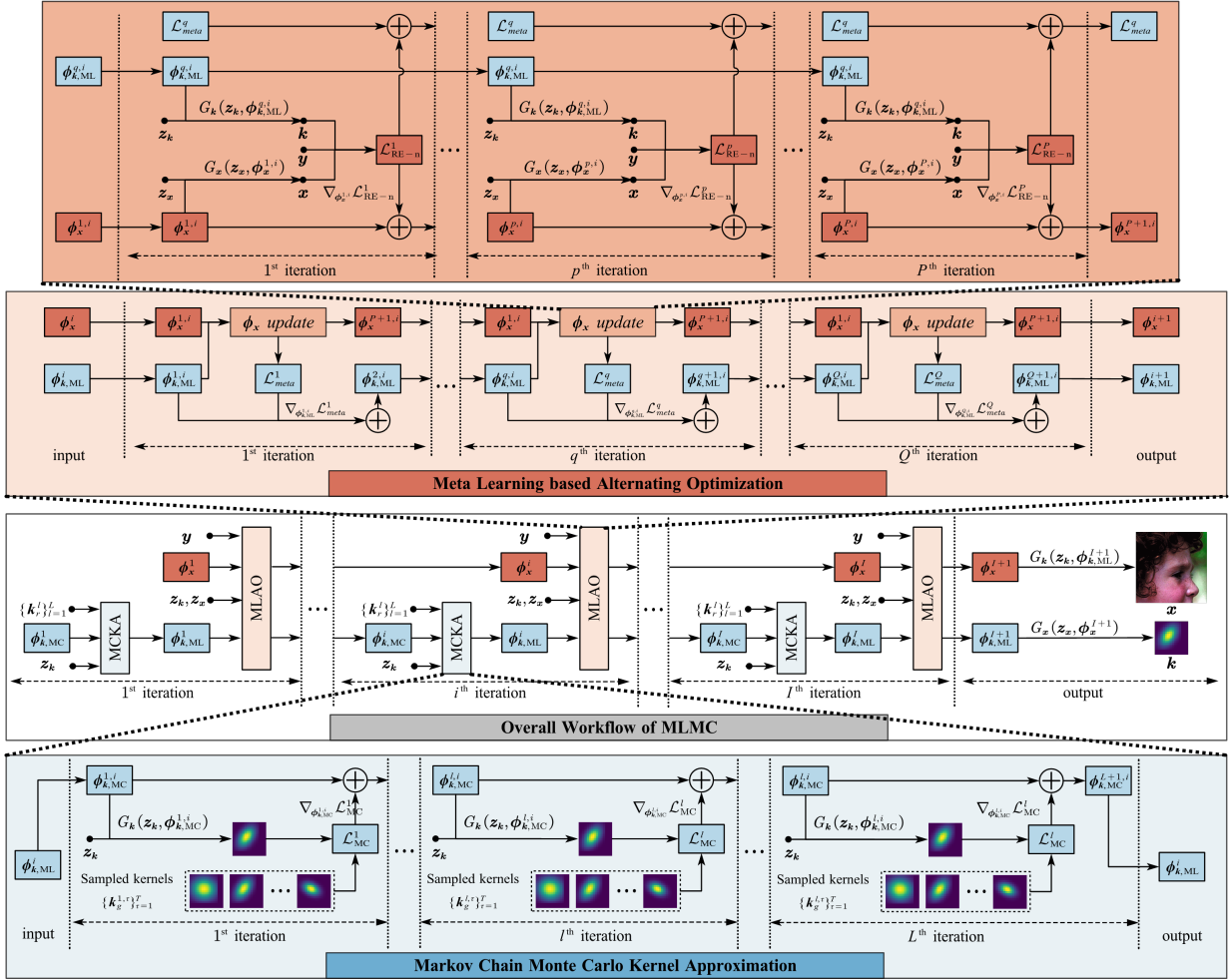


Fig. 1. The overall framework of the proposed MLMC.

On the basis of the problem (13), three developed modules for kernel prior, image prior and maximum likelihood estimation, respectively, are proposed in this paper as follows: i) In Section 4.3, a Markov Chain Monte Carlo simulation on random Gaussian distributions $\mathbf{k}_g$, referring to kernel approximation $p(\phi_k|\mathbf{k}_g)$, is proposed to substitute pre-training based kernel prior $p(\phi_k|\mathcal{D}_k)$; ii) In Section 4.4.2, a hyper-Laplacian prior is adopted to improve the denoising performance, referring to $p(\phi_x|\mathbf{n})$; iii) In Section 4.4, a meta-learning based alternating optimization strategy is incorporated to optimize parameters ϕ_x and ϕ_k to improve the convergence performance of solving the maximum likelihood estimation for $p(\mathbf{y}, \mathbf{x}, \mathbf{k}|\phi_x, \phi_k)$. Detailed demonstrations on each part are given in the following subsections.

4.2 Overall Framework

The overall framework of the proposed MLMC is depicted in Fig. 1. The fundamental paradigm is established on an alternating optimization between MCKA (blue) and MLAO (yellow) phases over the parameters of kernel estimator network ϕ_k and image restorer network ϕ_x . Specifically, ϕ_k are optimized across these two phases referring to $\phi_{k,MC}$ and $\phi_{k,ML}$, while ϕ_x are only optimized within MLAO. The al-

Algorithm 1: The overall framework of the proposed MLMC algorithm.

```

1 Input:  $\mathbf{y}$ .
2 Initialized:  $\phi_x^1, \phi_{k,ML}^1, \phi_{k,MC}^1, \mathbf{z}_x, \mathbf{z}_k$ .
3 for  $i \leftarrow 1, 2, \dots, I$  do
4   % Markov Chain Monte Carlo Kernel
   Approximation
5    $\phi_{k,ML}^i = f_{MCKA}(\{\mathbf{k}_g^{i,\tau}\}_{\tau=1}^T, \mathbf{z}_k, \phi_{k,MC}^i)$ 
6   % Meta-learning based Alternating Optimization
7    $\phi_x^{i+1}, \phi_{k,MC}^{i+1} = f_{MLAO}(\phi_{k,ML}^i, \phi_x^i, \mathbf{y}, \mathbf{z}_x, \mathbf{z}_k)$ 
8 end
9 Output:  $\mathbf{k} = G_k(\mathbf{z}_k, \phi_{k,ML}^I), \mathbf{x} = G_x(\mathbf{z}_x, \phi_x^I)$ 

```

ternative optimization process between MCKA and MLAO is referred as the outer loop with index $i = 1, 2, \dots, I$. At each outer loop step, there are MCKA inner loop with the index $l = 1, 2, \dots, L$ and MLAO inner loop with the index $q = 1, 2, \dots, Q$, respectively. In MCKA inner loop, the parameters of kernel generator $\phi_{k,MC}$ are iteratively updated with respect to L number of MCMC simulations on random Gaussian distributions to endow random kernel priors. In MLAO inner loop, the alternates between kernel

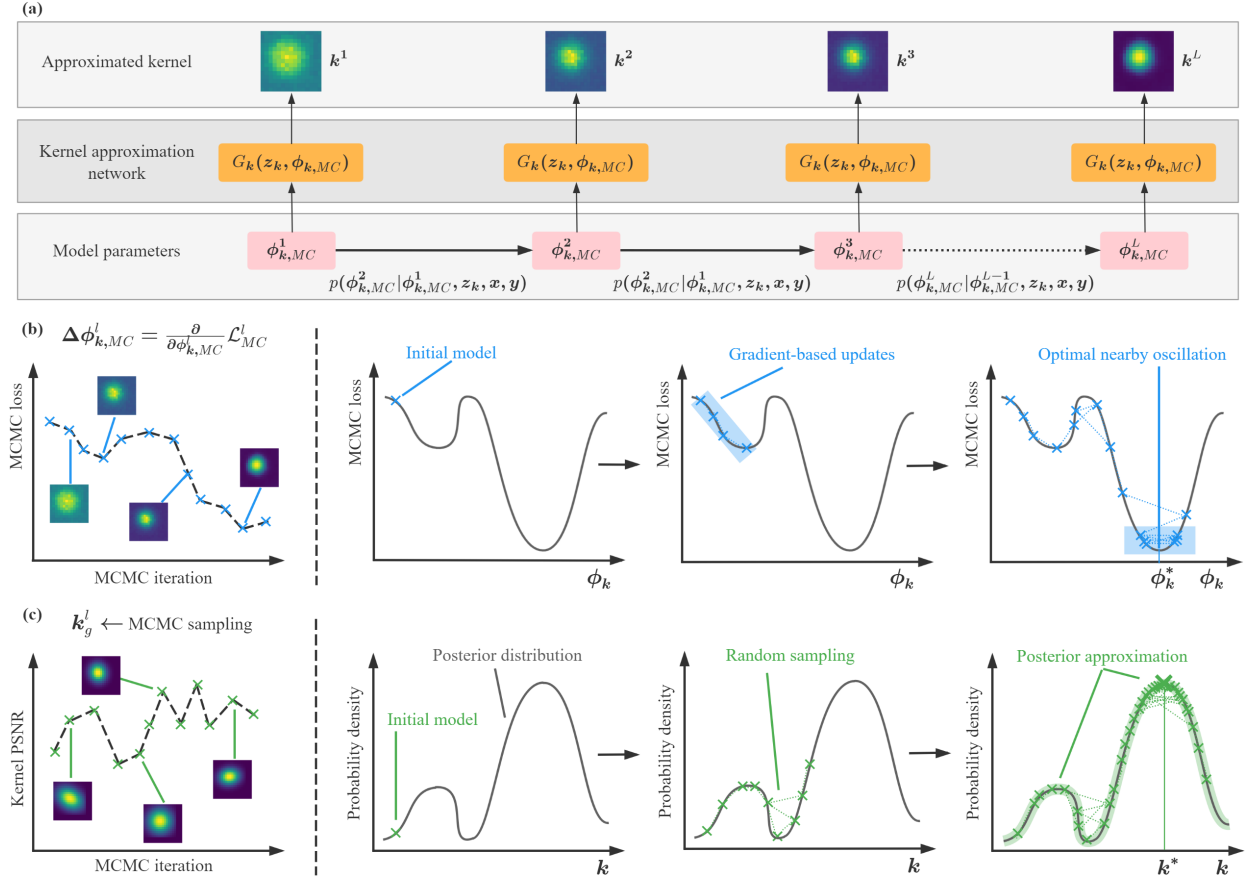


Fig. 2. The overview of the Markov Chain Monte Carlo kernel approximation (MCKA). (a) The approximated blur kernel $k^l = G_k(z_k, \phi_{k,MC}^l)$ is generated from the given noise z_k with fully-connected network (FCN) G_k , the parameters of which are iteratively updated, thereof resulting in Markov chain with possible transition $p(\phi_{k,MC}^{l+1} | \phi_{k,MC}^l, z_k, x, y)$. (b) Left: the network parameters (blue crosses) are optimized by gradient descent-based algorithm iteratively with respect to MCMC loss function in Eq. (25). The obtained approximated kernel present a trend of non-monotonically decreasing on MCMC loss. Right: the three panels show the trajectory of optimization on $\phi_{k,MC}^l$ over the geometry of MCMC loss (blue crosses). From an initial model (obtained from the last MLAO phase), the parameters $\phi_{k,MC}^l$ are updated via gradient descent-based algorithm. (c) Left: the MCMC simulations sample kernels (green crosses) from random Gaussian distributions with respect to a posterior that the sampled kernel should minimize the MCMC loss as well as close to the last generated sample. What stands out in this chart is the fluctuating convergence of the sampled kernel PSNR. This is caused by the posterior distribution formed by the Markov chain possible transition $p(\phi_{k,MC}^{l+1} | \phi_{k,MC}^l, z_k, x, y)$. Right: the three panels illustrate that sufficient MCMC sampling will lead to an approximation of posterior distribution underlying the minimized MCMC loss with respect to k^* . The random samples (green crosses) are distributed on posterior distribution with the guidance of probability density from LR reconstruction loss. Those kernels with less MCMC loss will be sampled with higher probability along with the MCMC process. We note that k^* is the generated blur kernel with respect to ϕ_k^* .

estimation and HR image restoration are processed for Q iterations to ensure the estimation accuracy on the basis of the given LR observation. Mathematically, MCKA phase generates random kernel priors $\{k_g^{l,\tau}\}_{\tau=1}^T$ to optimize parameters $\phi_{k,MC}$ as follows

$$\phi_{k,ML}^i = f_{MCKA}(\{k_g^{l,\tau}\}_{\tau=1}^T, z_k, \phi_{k,MC}^i), \quad (14)$$

where f_{MCKA} denotes the MCKA optimization process, and z_k is a random noise taken as input of the kernel estimator. The obtained $\phi_{k,ML}^i$ will be delivered to MLAO to provide a rational initialization of blur kernel in HR image restoration, and the parameters are optimized with respect to the LR observation:

$$\phi_x^{i+1}, \phi_{k,MC}^{i+1} = f_{MLAO}(\phi_{k,ML}^i, \phi_x^i, y, z_x, z_k), \quad (15)$$

where f_{MLAO} denotes the meta-learning based optimization for HR image restoration, and z_x is a random noise input to the image restorer. The overall framework of the proposed

MLMC is given in Algorithm 1. We will delineate these two phase as follows.

4.3 Markov Chain Monte Carlo Kernel Approximation

In this subsection, a kernel approximation method via MCMC simulations on random Gaussian distributions is proposed to provide rational kernel priors for $p(\phi_k)$. This process is named MCKA and we elaborate it in terms of the MCMC formulation, the random kernel sampling process, the optimization of kernel estimator, and the comparison towards the existing kernel estimators as follows.

4.3.1 The MCMC Formulation for Kernel Estimator

The MCMC method is widely used to systematically generate random samples from distributions, e.g. Gaussian, which allows the algorithms to obtain a sample of the desired distribution by establishing a Markov chain that achieves equilibrium at the desired distribution. In light of

this, we proposed an MCMC simulation on random Gaussian distributions to provide kernel priors, and construct a Markov chain on the parameters of the kernel generator, which are updated iteratively during the MCMC process to approximate blur kernels for blind SISR task. The proposed MCKA contains two major parts, the MCMC simulation on random Gaussian distributions and the optimization on kernel approximation network.

Mathematically, let $l = 1, 2, \dots, L$ denotes the index of MCMC step. At the l^{th} MCMC step, a shallow FCN $G_k(\cdot)$ with parameters $\phi_{k,MC}^l$ is applied to generate blur kernel k^l via taking a fixed random noise z_k as input in the following form:

$$k^l = G_k(z_k, \phi_{k,MC}^l). \quad (16)$$

Meanwhile, let k_g denotes a blur kernel with kernel size $(2d+1) \times (2d+1)$ from random Gaussian distributions,

$$k_g = p(h|\Sigma) = \frac{1}{2\pi|\Sigma|^{-1/2}} \exp\{(h-h_0)^T \Sigma (h-h_0)\}, \quad (17)$$

where $h = [\frac{m}{n}]$ denotes the coordinates, $m, n \in [-d, d]$, $\Sigma = [\begin{smallmatrix} \sigma_1 & \rho \\ \rho & \sigma_2 \end{smallmatrix}]$ denotes the covariance matrix, σ_1 and σ_2 are the horizontal and vertical variances, ρ is the kernel additional random rotation angle, and h_0 denotes the kernel centre coordinate. This indicates that the randomly generated distributions cover different kernel size d , centre h_0 , and categories Σ . At the l^{th} MCMC step, according to Eq. (17), the MCMC sampled random kernel k_g^l is given by

$$k_g^l = \int_{\Sigma} p(h|\Sigma) p(\Sigma|\phi_{k,MC}^l, z_k, x, y) d\Sigma, \quad (18)$$

where the $p(\Sigma|\phi_{k,MC}^l, z_k, x, y)$ denotes the the posterior over the current parameters state $\phi_{k,MC}^l$, the given HR image x and LR image y .

Then, the Markov chain is established on a sequence of generated blur kernels that are determined by the corresponding network parameters $\phi_{k,MC}^1, \phi_{k,MC}^2, \dots, \phi_{k,MC}^L$. The network parameters $\phi_{k,MC}^l$ are iteratively updated with respect to the Markov chain transition possibility in the following form

$$\begin{aligned} & \log p(\phi_{k,MC}^{l+1}, k_g^l | \phi_{k,MC}^l, z_k, x, y) = \\ & \log p(\phi_{k,MC}^{l+1} | k_g^l, \phi_{k,MC}^l, z_k) + \log p(k_g^l | \phi_{k,MC}^l, z_k, x, y), \end{aligned} \quad (19)$$

where $\log p(\phi_{k,MC}^{l+1} | k_g^l, \phi_{k,MC}^l, z_k)$ denotes that the state transition probability is determined by the sampled random kernel k_g^l , $\phi_{k,MC}^l$ and z_k . $\log p(k_g^l | \phi_{k,MC}^l, z_k, x, y)$ denotes the MCMC sampling posterior based on the LR image reconstruction error and the approximated blur kernel. Note that we have omitted the prior terms $p(x)$, $p(y)$ and $p(z_k)$, which are fixed during the MCKA stage. Eq. (19) illustrates that the proposed MCMC process is composed of two major modules: the update on $\phi_{k,MC}^l$, referring to the optimization on the kernel approximation network, and the random sampling for k_g^l , referring to the MCMC simulation on random Gaussian distributions, respectively.

Fig. 2 illustrates the algorithmic principle and objective of the MCKA phase. Specifically, Fig. 2 (a) shows that the approximated kernels are iteratively optimized based on the Markov chain update on $\phi_{k,MC}^l$. It can be seen that the

approximated blur kernels are iteratively optimized to be a rational blur kernel with respect to the updates on $\phi_{k,MC}^l$. In the first column of Fig. 2 (b), it explicitly shows that with the update number, the optimization loss of $\phi_{k,MC}^l$ non-monotonically decreases. This is because that the update is implemented in the way of gradient-based strategy as shown in the third column of Fig. 2 (b). Moreover, thanks to the randomly sampled kernels in MCMC simulations, the optimization on $\phi_{k,MC}^l$ is able to escape from bad local optimum and converges to an equilibrium stationary mode as the forth column of Fig. 2 (b) presented. This is achieved by the parallel MCMC simulations that provide organized randomness to guide the optimization process in Fig. 2 (b).

Specifically, the randomly sampled kernels are shown in Fig. 2 (c). The first column shows the general trend of the sampled kernels with update number. Though the MCMC simulations retain significant fluctuation on kernel PSNR, a converge trend is presented. In the second to the forth columns, the posterior distribution with respect to the LR image reconstruction error loss is presented, where the highest probability density refers to the blur kernel k^* that minimizes the LR image reconstruction error. As shown in the forth column of Fig. 2 (c), the MCMC simulations randomly sample on the posterior distribution, and thereby realize organized randomness that leads to dense distribution around the k^* . The forth columns in Fig. 2 (b) and (c) illustrate that the organized random sampling on posterior distribution allows the optimization on $\phi_{k,MC}^l$ escaping from bad local optimum and achieving equilibrium oscillation around the optima. We conclude that the updates on $\phi_{k,MC}^l$ in Fig. 2 (b) and the random sampling from MCMC simulations in Fig. 2 (c) are iteratively processed in MCKA iterations underlying learning from random Gaussian distributions that are organized by the MCMC based loss, therefore, attaining rational blur kernel as priors and providing better convergence performance.

4.3.2 Random Kernel Sampling Process

A Monte Carlo simulations on random Gaussian distributions is proposed with respect to Markov chain on the state of kernel generator parameters $\phi_{k,MC}^l$. These parameters determine the approximated blur kernels, which are iteratively updated during the MCKA steps on the basis of the randomly sampled kernels and LR reconstruction error.

In practice, the l^{th} sampled kernel k_g^l is nontrivial, thereof, the integral operation is replaced by the summation of sufficient sampling on the whole distribution, as the classic Monte Carlo simulation does as follows

$$k_g^l \approx \sum_{\tau=1}^T p(h|\Sigma^{l,\tau}) p(\Sigma^{l,\tau} | \phi_{k,MC}^l, z_k, x, y) = \sum_{\tau=1}^T \omega^{l,\tau} k_g^{l,\tau}, \quad (20)$$

where $p(h|\Sigma^{l,\tau})$ can be referred to the τ^{th} Gaussian kernel $k_g^{l,\tau}$ dominated by the covariance matrix $\Sigma^{l,\tau}$ with $\sigma_1, \sigma_2 \in (0, \sigma_{max}]$ and $\rho \in [-\pi, \pi]$. σ_{max} denotes the maximum variance of kernel boundary, and T is the total number of random samples in one Monte Carlo simulation. $p(h|\Sigma^{l,\tau}) p(\Sigma^{l,\tau} | \phi_{k,MC}^l, z_k, x, y)$ can be referred to the weight $\omega^{l,\tau}$ of the τ^{th} sampled kernel. Different to the conventional Monte Carlo simulation that uniformly

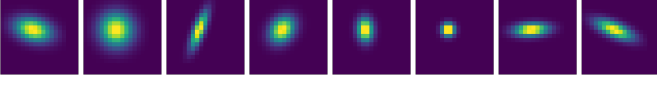


Fig. 3. Blur kernels are randomly sampled by the Monte Carlo simulation from different Gaussian distributions, e.g., different variances (Σ) and rotation angle (ρ).

samples blur kernels from random Gaussian distributions, in this paper, the Markov chain on $\phi_{k,MC}^l$ re-weights all the sampled kernels by kernel weight $\omega^{l,\tau}$ in one Monte Carlo simulation as follows

$$\omega^{l,\tau} = p(k_g^{l,\tau} | \phi_{k,MC}^l, z_k, x, y) \propto \frac{1}{\nu^{l,\tau}} \quad (21)$$

$$\begin{aligned} \nu^{l,\tau} = & \|y - (x \otimes k_g^{l,\tau}) \downarrow_s\|_F^2 + \\ & + \|G_k(z_k, \phi_{k,MC}^l) - k_g^{l,\tau}\|_F^2 + \epsilon, \end{aligned} \quad (22)$$

where ϵ is a small hyper-parameter to prevent $\nu^{l,\tau} = 0$. Eq. (21) and Eq. (22) illustrate that the randomly sampled kernels are weighted by the $\nu^{l,\tau}$ that is composed of the LR image reconstruction term and the MSE of sampled kernel and approximated kernel at the l^{th} iteration. Therefore, we claim that the randomly sampled kernels are organized to be distributed closely to the kernels that minimize the LR image reconstruction error, and the new sampling will prefer to occur in the neighbor of the last sampled kernel. The LR image reconstruction error urges the randomly sampled kernels realizing the posterior distribution as shown in Fig. 2 (c) right, while the MSE brings a balance between steep variation on kernel PSNR and stable convergence on the obtained rational blur kernel as shown in Fig. 2 (c) left.

We note that, instead of exhaustively sampling on the full range of Σ with extremely large sampling number T , as the classical MCMC method does, only a few realizations are randomly sampled in one MCKA stage. In Fig. 3, the visualization of the randomly sampled blur kernels from different Gaussian distributions in one Monte Carlo simulation is presented. It is explicit that the variability of these sampled kernels in terms of coordinate and outline are significantly large. After the Markov chain possible transition reweights the sampled kernels, the obtained integrated kernel k_g^l is applied to optimize the parameters $\phi_{k,MC}^l$. In this way the obtained sampled kernel k_g^l can be regarded as a loose relaxation for the obtained approximated blur kernel k^l , that is dominated by parameters $\phi_{k,MC}^l$, which typically suffers from the overfitting optimization during the MLAO stage even with the proposed meta-learning-based framework. The optimization on approximation network will be demonstrated next.

4.3.3 Optimizing the Kernel.

The parameters of kernel approximation network $\phi_{k,MC}$ are iteratively optimized with respect to the randomly sampled Gaussian distributions at each MCKA iteration. Specifically, at each iteration l , recalling the transition possible in Eq. (19), the optimization on $\phi_{k,MC}^l$ can be expressed as a log-

Algorithm 2: The work flow of the MCKA phase.

```

1 Input:  $y, z_k, \phi_{k,MC}^i$ .
2 % Markov Chain Monte Carlo kernel approximation
3  $\phi_{k,MC}^{1,i} \leftarrow \phi_{k,MC}^i$ 
4 Sample random kernels  $\{k_g^{l,\tau}\}_{\tau=1}^T$  via MC sampling
5 for  $l \leftarrow 1, 2, \dots, L$  do
6   for  $\tau \leftarrow 1, 2, \dots, T$  do
7      $\nu^{l,\tau} = \|y - (x \otimes k_g^{l,\tau}) \downarrow_s\|_F^2 +$ 
8        $\|G_k(z_k, \phi_{k,MC}^l) - k_g^{l,\tau}\|_F^2 + \epsilon$ 
9      $\omega^{l,\tau} = \frac{1}{\nu^{l,\tau}}$ 
10  end
11   $\mathcal{L}_{MC}^l = \sum_{\tau=1}^T \omega^{l,\tau} \|G_k(z_k, \phi_{k,MC}^l) - k_g^{l,\tau}\|_F^2$ 
12   $\phi_{k,MC}^{l+1,i} = \phi_{k,MC}^{l,i} - \gamma_{MC}^l \text{Adam}\left(\frac{\partial}{\partial \phi_{k,MC}^l} \mathcal{L}_{MC}^l\right)$ 
13 end
14  $\phi_{k,ML}^l \leftarrow \phi_{k,MC}^{L,i}$ 
15 Output:  $\phi_{k,ML}^1$ 

```

likelihood maximization problem on transition probability in the following form

$$\begin{aligned} & \max \log p(\phi_{k,MC}^{l+1} | k_g^l, \phi_{k,MC}^l, z_k, x, y) \\ & = \max \log p(\phi_{k,MC}^{l+1} | \{k_g^{l,\tau}\}_{\tau=1}^T, \phi_{k,MC}^l, z_k, x, y), \end{aligned} \quad (23)$$

where each random sampled kernel $k_g^{l,\tau}$ is independent and identically distributed (i.i.d.). However, each k_g^l entails Markov chain based re-weighting recalling to $\omega^{l,\tau}$ in Eq. (22), which bridges the possible transition with the LR reconstruction in terms of x and y . Therefore, the optimization problem (23) can be rewritten as

$$\begin{aligned} & \max \log \prod_{\tau=1}^T p(\phi_{k,MC}^{l+1} | k_g^{l,\tau}) p(k_g^{l,\tau} | \phi_{k,MC}^l, z_k, x, y) \\ & = \min \sum_{\tau=1}^T -\log p(\phi_{k,MC}^{l+1} | k_g^{l,\tau}) p(k_g^{l,\tau} | \phi_{k,MC}^l, z_k, x, y). \end{aligned} \quad (24)$$

As demonstrated in Eq. (21), $\omega^{l,\tau} = p(k_g^{l,\tau} | \phi_{k,MC}^l, z_k, x, y)$. Then, the MCMC loss $\mathcal{L}_{MC}^l$ is given by

$$\min_{\phi_{k,MC}^l} \mathcal{L}_{MC}^l = \sum_{\tau=1}^T \omega^{l,\tau} \|G_k(z_k, \phi_{k,MC}^l) - k_g^{l,\tau}\|_F^2. \quad (25)$$

Eq. (25) elaborates that the parameters $\phi_{k,MC}^l$ are optimized with respect to the MSE between the approximated blur kernels and the randomly sampled kernel, underlying the organized by the MCMC weight $\omega^{l,\tau}$. The parameters are optimized by the Adam [61] optimizer:

$$\phi_{k,MC}^{l+1} = \phi_{k,MC}^l - \gamma_{MC}^l \cdot \text{Adam}\left(\frac{\partial}{\partial \phi_{k,MC}^l} \mathcal{L}_{MC}^l\right), \quad (26)$$

where γ_{MC}^l denotes the learning rate.

We note that the optimization on the kernel approximation network is implemented in a way of plug-and-play fashion, that is “training while solving” the blind SISR task, instead of training in advance. The whole MCKA stage alternatively processes the MCMC simulations for rational blur kernels and optimizes the kernel approximation network via

learning from organized randomness. There are two major contributions of enrolling the proposed MCKA procedure in solving blind SISR problem. On the one hand, the MCMC simulations provide a loose but rational kernel priors for blur kernel estimation, while no pre-training procedure and labelled data are needed. On the other hand, the MCKA phase periodically brings random disturbance to the convergence of the parameters ϕ_k , which are alternatively optimized with respect to the LR image reconstruction error loss in the MLAO phase. Therefore, it prevents the optimization of the parameters ϕ_k converging to bad local modes due to the intrinsic non-convexity and ill-posedness. We note that the randomnesses learned in MCKA will approximately converge to a desire distribution via the MCMC behavior in terms of minimizing the LR image reconstruction error. Thus, the additive disturbance will not lead to a significant deviation that is detrimental to the convergence of solving blind SISR problem.

The overall data flow of the MCKA phase is given in Algorithm 2. It should be also noted that we set L and T to be small for the following two reasons: i) The obtained kernel priors are flexible, and can be easily refined by the LR image reconstruction error; ii) The implementation is less time-consuming and computationally less demanding, ensuring negligible runtime cost and memory usage. Consequently, the MCKA is implemented in a plug-and-play manner and is combined with a meta-learning framework to resolve the blind SISR problem in the next section.

4.3.4 Kernel Estimator Comparisons

As aforementioned in Section 2, the existing kernel estimators can be roughly divided into two categories: designing kernel model in a handcraft way [1], [4], [6] or estimating kernel via networks [13], [42], [60]. We compare the most related existing kernel estimators as follows:

- *Deep-learning methods*: More recent deep learning methods also investigate to design specific modules on the basis of kernel prior knowledge, such as implicit degradation modeling [40], shuffled degradation processes [38], and varying degradations for multiple patches [42]. In a nutshell, these methods tend to estimate kernels via learning on labelled HR images with independent modeling on kernel priors. In contrast, the MLMC strives to learn kernel priors from randomness in an unsupervised manner and follows a plug-and-play fashion.
- *Double-DIP*: Double-DIP adopts a DIP-like network without pre-training for kernel estimation. The adopted network is simply optimized with respect to LR image restoration error and no specific kernel priors available, thus the performance is not satisfactory in most Gaussian and motion kernel cases. Then, the network utilized in MLMC has two differences comparing to Double-DIP: i) the network architecture is much more lightweight; ii) the parameters are optimized on the basis of random kernel priors along with the LR image restoration error.
- *FKP-DIP*: The kernel estimator FKP module is pre-trained in a supervised way with sufficient kernel data. This training process typically requests hours

and is necessary when being applied to different kernel categories. In contrast, the kernel learning module in MLMC is plug-and-play, allowing stronger flexibility for varying kernel categories.

- *BSRDM* [6]: In BSRDM, an explicit two-dimensional Gaussian model is employed to estimate the Gaussian kernel, whose parameters (mean and variance) are optimized via a gradient-based algorithm along with the LR image reconstruction error through the entire solution iterations. In contrast, the kernel estimator in the proposed MLMC only formulates the kernel distribution, allowing a significantly larger scope for kernel prior learning, resulting in more dynamic and flexible kernel priors. Besides, the MLMC kernel estimator is trained on the basis of learned kernel priors and LR observation, ensuring better convergence performance than BSRDM. Moreover, the MLMC can be applied to non-Gaussian kernels, such as the motion kernel. This highlights its better generalization ability and flexibility in practice.

In a nutshell, comparing to the existing kernel estimators, the proposed MLMC combines the merits of model-based and learning-based methods, realizing a training while solving paradigm for kernel estimation without requests on labelled dataset and training before applying. We claim that the random kernel-prior-learning based MCKA is the main technical contribution of the proposed MLMC and the fundamental discrepancy towards the existing kernel estimators.

4.4 Meta-learning based Alternating Optimization

Despite rational kernel priors, referring to $p(\phi_k)$, can be gained from the MCKA phase, the primary blind SISR problem with respect to $p(\mathbf{y}, \mathbf{x}, \mathbf{k} | \phi_x, \phi_k)$ retains high non-convexity and ill-posedness, therefore, being ease of converging to bad local optimums, especially for the kernel estimation. Inspired by [21], [55], a meta-learning based alternating optimization is designed to optimize ϕ_x and ϕ_k from the observed LR image $\mathbf{y}$, the whole process is named MLAO phase and elaborated as follows.

4.4.1 HR Image restoration Optimization

As aforementioned in Section 3.3, the network-based blind SISR problem (6) is typically solved in an AM-based fashion between the kernel estimation and image restoration sub-problems, which alternatively estimate the HR image $\mathbf{x}$ and blur kernel $\mathbf{k}$ via networks $G_x(\mathbf{z}_x, \phi_x)$ and $G_k(\mathbf{z}_k, \phi_k)$, respectively. However, the intermediate solutions from each sub-problem in (6) typically contain significant noise, and the minimization behavior for each sub-problem using only first-order information may not necessarily lead to benign solutions in the view of global convergence. Therefore, a meta-learning scheme is proposed to learn a less-greedy and adaptive updating rule for better convergence performance in terms of kernel estimation.

Mathematically, let $p = 1, 2, \dots, P$ denotes the SISR iteration index, and $q = 1, 2, \dots, Q$ represents the index of the meta-update on the kernel generator. Then, at the p^{th} iteration, the LR image reconstruction error is given by

$$\mathcal{L}_{RE}^p = \|\mathbf{y} - (\mathbf{x}^p \otimes \mathbf{k}^q) \downarrow_s\|_F^2, \quad (27)$$

where $\mathbf{k}^q = \mathbf{G}_{\mathbf{k}}(\mathbf{z}_{\mathbf{k}}, \phi_{\mathbf{k},ML}^q)$ and $\mathbf{x}^p = \mathbf{G}_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}}^p)$. Instead of exhaustively optimize each sub-problem with respect to minimizing $\mathcal{L}_{RE}^p$, the kernel estimation iterations are processed in a meta-learning manner. We tend to minimize the leveraged reconstruction errors over P iterations, denoted by the meta-loss $\mathcal{L}_{ML}^q$, to update the parameters $\phi_{\mathbf{k},ML}^q$ for Q times as follows

$$\begin{cases} \mathcal{L}_{ML}^q = \frac{1}{P} \sum_{p=1}^P \omega^p \mathcal{L}_{RE}^p, \\ \phi_{\mathbf{k},ML}^{q+1} = \phi_{\mathbf{k},ML}^q - \gamma_{ML}^q \cdot \text{Adam} \left(\frac{\partial}{\partial \phi_{\mathbf{k},ML}^q} \mathcal{L}_{ML}^q \right), \end{cases} \quad (28)$$

where γ_{ML}^q denotes the learning rate at the q^{th} meta-update step and ω^p is the weight of $\mathcal{L}_{RE}^p$ at iteration p . In this instance, the parameters $\phi_{\mathbf{k},ML}^q$ are no longer optimized via minimizing each individual reconstruction error but the accumulated losses $\mathcal{L}_{ML}^q$, which results in a less-greedy but more adaptive optimization strategy. As illustrated in [21], $\mathcal{L}_{ML}^q$ essentially leads to a better optima, as it optimizes each individual sub-problem in-exhaustively by globally learning the mutual knowledge of the optimization strategy on partial reconstruction error trajectories across iterations. In this way, the optimization strategy for the blur kernel estimation is endowed with a relaxation, which allows non-optimal solution at the p^{th} sub-problem for $\phi_{\mathbf{k}}$ during the estimation iterations but is capacity of converging to better optimum in the view of P iterations.

Meanwhile, the image restoration sub-problem is optimally solved at each iteration on the basis of the hyper-Laplacian image prior demonstrated in Section 4.4.2, which updates the parameters $\phi_{\mathbf{x}}$ as follows,

$$\phi_{\mathbf{x}}^{p+1} = \phi_{\mathbf{x}}^p - \gamma_{\mathbf{x}}^p \cdot \text{Adam} \left(\frac{\partial}{\partial \phi_{\mathbf{x}}^p} \mathcal{L}_{RE-n}^p \right), \quad (30)$$

where $\gamma_{\mathbf{x}}^p$ denotes the learning rate. We note that the adopted network $\mathbf{G}_{\mathbf{x}}$ is a well-designed image restorer with substantiated performance in the literature [2], [4], [13]. Hence, the vanilla iterative minimization is sufficiently satisfactory to realize good HR image reconstruction when the blur kernel is obtained through the meta-learning based optimization strategy.

In this stage, the overview of the MLAO phase is depicted in the yellow box in Fig. 1 and the work flow is given in Algorithm 3. Explicitly, the whole MLAO stage tends to refine the approximated kernel based on the meta-loss $\mathcal{L}_{ML}^q$, as such $\phi_{\mathbf{k},ML}^q$ are optimized on the basis of the observed LR image $\mathbf{y}$ via a less-greedy and more adaptive optimization strategy. We note that the incorporated meta-learning approach on kernel estimation regards each kernel estimation sub-problem as training sample, and meta-learn across kernel estimation iterations to extract the mutual knowledge of solving a set of sub-problems. In this way, the learned update rule becomes more flexible and non-monotonically, therefore, being able to prevent trapping into bad local optimum.

4.4.2 Image Noise Optimization

In this subsection, we further propose an HR image restoration formulation for the noise scenario. A hyper-Laplacian

Algorithm 3: The work flow of the MLAO phase.

```

1 Input:  $\phi_{\mathbf{k},ML}^i, \phi_{\mathbf{x}}^i, \mathbf{y}, \mathbf{z}_{\mathbf{x}}, \mathbf{z}_{\mathbf{k}}$ .
2 % Meta-learning based alternating optimization
  (MLAO)
3  $\phi_{\mathbf{k},ML}^{1,i} \leftarrow \phi_{\mathbf{k},ML}^i, \phi_{\mathbf{x}}^{1,i} \leftarrow \phi_{\mathbf{x}}^i$ 
4 for  $q \leftarrow 1, 2, \dots, Q$  do
5    $\mathbf{k}^q = \mathbf{G}_{\mathbf{k}}(\mathbf{z}_{\mathbf{k}}, \phi_{\mathbf{k},ML}^{q,i})$ 
6   % Image network update
7   for  $p \leftarrow 1, 2, \dots, P$  do
8      $\mathbf{x}^p = \mathbf{G}_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}}^{p,i})$ 
9      $\sigma^2 = \frac{1}{h \times w} \sum_{j \in N} \{y_j - [(\mathbf{x}^p \otimes \mathbf{k}^q) \downarrow_s]_j\}^2$ 
10     $\mathcal{L}_{RE-n}^p = \frac{1}{\sigma^2} \|\mathbf{y} - (\mathbf{x}^p \otimes \mathbf{k}^q) \downarrow_s\|_F^2 +$ 
11       $\rho \sum_{c=1}^2 (\|f_c \otimes \mathbf{x}^p\|_F^2)^\eta$ 
12     $\phi_{\mathbf{x}}^{p+1,i} \leftarrow \phi_{\mathbf{x}}^{p,i} - \gamma_{\mathbf{x}}^p \cdot \text{Adam} \left( \frac{\partial}{\partial \phi_{\mathbf{x}}^{p,i}} \mathcal{L}_{RE-n}^p \right)$ 
13  end
14   $\phi_{\mathbf{x}}^{1,i} \leftarrow \phi_{\mathbf{x}}^{P,i}$ 
15  % Kernel network update
16   $\mathcal{L}_{ML}^q = \frac{1}{P} \sum_{p=1}^P \omega^p \mathcal{L}_{RE-n}^p$ 
17   $\phi_{\mathbf{k},ML}^{q+1,i} \leftarrow \phi_{\mathbf{k},ML}^{q,i} - \gamma_{ML}^q \cdot \text{Adam} \left( \frac{\partial}{\partial \phi_{\mathbf{k},ML}^{q,i}} \mathcal{L}_{ML}^q \right)$ 
18 end
19  $\phi_{\mathbf{x}}^{i+1} \leftarrow \phi_{\mathbf{x}}^{P,i}, \phi_{\mathbf{k},MC}^{i+1} \leftarrow \phi_{\mathbf{k},ML}^{Q,i}$ 
20 Output:  $\phi_{\mathbf{x}}^{i+1}, \phi_{\mathbf{k},MC}^{i+1}$ 

```

image prior is incorporated with the HR image restoration for $p(\phi_{\mathbf{x}})$. We formulate an independent and identically distributed (i.i.d) additive white Gaussian noise (AWGN) model for the fundamental degradation model as follows,

$$\mathbf{y} \sim \mathcal{N}(\mathbf{y} | (\mathbf{x} \otimes \mathbf{k}) \downarrow_s, \sigma^2), \quad (31)$$

where σ denotes the covariance of the noise distribution. Similar to the previous works [2], [4], [6], [13], a CNN-based network $\mathbf{G}_{\mathbf{x}}(\cdot)$ is established to estimate the HR image as formulated in Eq. (6). As demonstrated in [2], $\mathbf{G}_{\mathbf{x}}(\cdot)$ is typically fragile to the image noise, thus leading to overfitting to bad local optimums. A hyper-Laplacian prior [28] is implemented to constrain the AWGN of the estimated HR image following the proposed statistic framework as follows,

$$(\phi_{\mathbf{x}}, \mathbf{z}_{\mathbf{x}}) \sim p(\phi_{\mathbf{x}}, \mathbf{z}_{\mathbf{x}}) = p(\phi_{\mathbf{x}} | \mathbf{z}_{\mathbf{x}}) p(\mathbf{z}_{\mathbf{x}}), \quad (32)$$

$$p(\phi_{\mathbf{x}} | \mathbf{z}_{\mathbf{x}}) \propto \exp \left(-\rho \sum_{c=1}^2 (\|f_c \otimes \mathbf{G}_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}})\|_F^2)^\eta \right), \quad (33)$$

$$p(\mathbf{z}_{\mathbf{x}}) = \mathcal{N}(0, \sigma^2), \quad (34)$$

where $\{f_c\}_{c=1}^2$ represent the gradient filters along the horizontal and vertical directions, ρ and η denote the hyper-parameters. Equivalently, the parameters of the image estimator $\phi_{\mathbf{x}}$ are optimized by the loss function as follow,

$$\begin{aligned} \min_{\phi_{\mathbf{x}}} \mathcal{L}_{RE-n} &= \frac{1}{\sigma^2} \|\mathbf{y} - (\mathbf{G}_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}}) \otimes \mathbf{k}) \downarrow_s\|_F^2 \\ &+ \rho \sum_{c=1}^2 (\|f_c \otimes \mathbf{G}_{\mathbf{x}}(\mathbf{z}_{\mathbf{x}}, \phi_{\mathbf{x}})\|_F^2)^\eta. \end{aligned} \quad (35)$$

The noise variance σ^2 is typically given by

$$\sigma^2 = \frac{1}{h \times w} \sum_{j \in N} \{y_j - [(G_x(z_x, \phi_x) \otimes \mathbf{k}) \downarrow_s]_j\}^2, \quad (36)$$

where h and w denote the image height and width, respectively, and the number of image pixels $N = h \times w$, j denotes the j^{th} pixel of image.

4.5 Network-level Langevin Dynamics

The vanilla Langevin dynamics [22] is proposed to improve the performance of gradient descent-based optimization algorithms for variable update, which can be typically formulated as follows,

$$\mathbf{z}^{t+1} = \mathbf{z}^t + \frac{\delta^2}{2} \cdot \frac{\partial \log p(\mathbf{z}^t | \mathbf{y})}{\partial \mathbf{z}^t} + \delta \cdot \zeta_n, \quad (37)$$

where t denotes the index of update step, $\mathbf{z}$ denotes the variable to be optimized, $\mathbf{y}$ denotes the observation data, δ denotes the optimization step size. $\log p(\mathbf{z}^t | \mathbf{y})$ demonstrates the standard optimization on the basis of the task-specific data, and ζ_n is a random noise (e.g., zero-mean Gaussian noise). When $t \rightarrow \infty$ and $\delta^2 \rightarrow 0$, Langevin dynamics realizes a sampling from the task-specific data posterior $p(\mathbf{z}^t | \mathbf{y})$ to optimize $\mathbf{z}^{t+1}$. Theoretically, this is achieved by the additive noise ζ_n that correlated with the posterior $p(\mathbf{z}^t | \mathbf{y})$.

Different to the Langevin dynamics in BSRDM, the proposed MLMC tends to resolve an optimization problem of network parameters update. Following the concept of improving the optimization convergence performance via incorporating random fluctuation, the white Gaussian noise should be re-designed to fit the network-based optimization process. A natural idea is to incorporate random samples into the network training process to achieve the similar effect of Langevin dynamics. In light of this concept, in the proposed MLMC, we proposed a novel network-level Langevin dynamics optimization strategy that is designed for network parameters optimization as follows

$$\phi_{\mathbf{k}}^{new} = \phi_{\mathbf{k}}^{old} + \gamma_{ML} \cdot \frac{\partial \log p_{MLAO}(\phi_{\mathbf{k}}^{old} | \mathbf{y})}{\partial \phi_{\mathbf{k}}^{old}} + \gamma_{MC} \cdot \zeta_{\mathbf{k}}, \quad (38)$$

where $\zeta_{\mathbf{k}} = \frac{\partial \log p_{MLAO}(\phi_{\mathbf{k}}^{old} | \mathbf{y})}{\partial \phi_{\mathbf{k}}^{old}}$ denotes the gradients of LR image restoration loss $\mathcal{L}_{RE}^p$ and $\frac{\partial \log p_{MCKA}(\phi_{\mathbf{k}}^{old} | \mathbf{k}_g)}{\partial \phi_{\mathbf{k}}^{old}}$ denotes the gradients of MSE loss $\mathcal{L}_{MC}^l$. At each alternating step, the parameters of the kernel estimator are optimized with respect to the $\mathcal{L}_{RE}^p$ and $\mathcal{L}_{MC}^l$ in MLAO and MCKA phases, respectively. Specifically, the third term essentially plays the role of ζ_n in vanilla Langevin dynamics (37), as the random sampled kernels $\mathbf{k}_g$ can be regarded as a "noise" in the view of network training samples. Besides, the sampling process follows MCMC simulation that also iteratively updates the sampling distribution with respect to the HR image restoration posterior. In this stage, we claim that this is a novel Langevin dynamics framework that is suitable for network-level optimization, as the conventional Langevin dynamics merely works on traditional model-based optimization paradigm. This also provides a theoretical support of rationalizing the incorporation of the MCKA phase.

It is also noteworthy that the proposed network-level Langevin dynamics optimization framework is intrinsically different to the traditional Langevin dynamics, as such applied in BSRDM [6]. In BSRDM, the input noise of DIP network is iteratively optimized via gradient-based algorithm and an additive white Gaussian noise is applied to prevent trapping into local mode. Differently, in the proposed MLMC, we proposed a novel network-level Langevin dynamics optimization strategy that is designed for network parameters optimization. Specifically, random sampled kernels are employed as the additive noises in this network-level Langevin dynamics optimization to play the role of preventing converging to bad local optimum of the kernel estimator. Recalling the depiction in Fig. 2, the MCMC simulations on random Gaussian distributions will converge to a stable equilibrium, therefore, ζ is an organized random sample instead of total randomness. In this way, the proposed MCMC simulations on random Gaussian distributions bring a profitable disturbance towards the optimization process, as well as providing a rational kernel prior.

4.6 Pipeline and Analysis

The overall solution procedure of the proposed MLMC is given in Algorithm 1, and the details of MCKA and MLAO loops are illustrated in Algorithm 2 and 3, respectively.

At this stage, it is clear that the alternating framework between MCKA and MLAO is indispensable. On the one hand, at each iteration in Algorithm 1, the MCKA phase learns a rational kernel prior from MCMC simulations, which provides external knowledge of the sampled kernel distributions. This contributes to a rational initialization of the $\phi_{\mathbf{k},ML}$ in MLAO stage instead of initializing from randomness. On the other hand, in the view of the total optimization on $\phi_{\mathbf{k}}$, the MCKA phase provides an organized randomness with the optimization based on LR observation in MLAO phase, formulating the aforementioned network-level Langevin dynamics.

We note that the whole MLMC is processed in an unsupervised inference and the whole algorithm is applied in a plug-and-play fashion without any training in advance. This significantly improves the flexibility and generalization capacity of the MLMC. In simulations, we show that the proposed MLMC is able to achieve strong generalization ability and flexibility towards out-of-distribution blur kernels, different noise scenarios and non-Gaussian kernels in Section 5.3, as well as realizing superior performance comparing to the state-of-the-arts in Section 5. The hyperparameters involved in the two summations corresponding to the number of sampling times in one MCMC simulation, denoted by T , and the meta-learning interval, denoted by P , will be discussed in Section 5.1.

5 EXPERIMENTAL RESULTS

5.1 Experimental Setup

Data Preparation. Following the widely adopted kernel assumption [6], [13], [44], [62], [63], [64], we conduct most of the experiments on anisotropic Gaussian kernels and a few on non-Gaussian kernels (motion kernels generated by [65]). The kernel sizes are set to $(4s + 3) \times (4s + 3)$, the

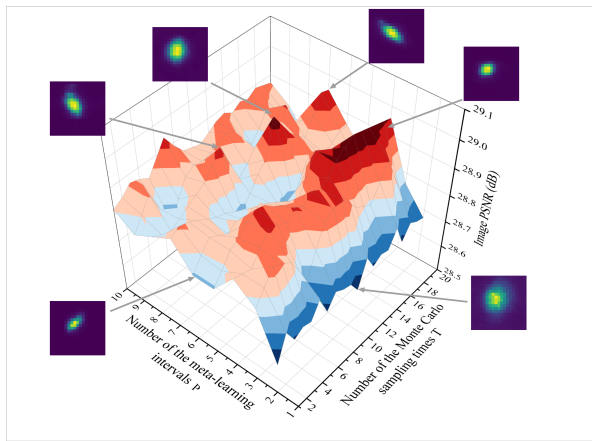


Fig. 4. Image PSNR performance of the proposed MLMC method on Set5 with different hyper-parameter combinations: the number of the Monte Carlo sampling times T and the number of meta-learning optimization intervals P .

Gaussian width ranges are set to $[0.175s, 2.5s]$, and the rotation angle range ρ is set to $[0, \pi]$, for a scale factor $s \in 2, 3, 4$, respectively. We synthesize LR images with random kernels with respect to Eq. (1) for testing data based on four popular public benchmark datasets, including Set5 [66], Set14 [67], BSD100 [68], Urban100 [69] and a real-world dataset, RealSRSet [51]. We compare these kernels in terms of the peak signal to noise ratio (PSNR), and compare HR images in terms of PSNR and structural similarity (SSIM) [70].

Implementation Details. We adopt a two-layer shallow FCN with 1000 nodes at each layer as the kernel generator, and use a CNN-based image generator following the architecture in [2]. The Adam optimizer [61] is applied to optimize the parameters of these two networks with learning rates 0.5 and 0.005 for FCN and CNN, respectively. We compare the proposed MLMC approach with other two kinds of state-of-the-arts, including unsupervised methods (the model is optimized on only one LR image): PAM [1], DIP [2], KernelGAN+ZSSR [64], Double-DIP [4], FKP-DIP [13] and BSRDM [6], and deep-learning-based methods (the model is pre-trained on LR-HR paired dataset): RCAN [59], DASR [12], BSRGAN [38], KDSR [40], KULNet [39], UDKE [41], KXNet [52]. We also use non-blind model USRNet [50] to generate the final SR result based on the kernel estimation from MANet [42] and our MLMC as two deep-learning-based methods: MANet+USRNet and MLMC+USRNet. The parameters of the proposed MLMC approach are randomly initialized from scratch and re-initialized for each test image, ensuring fairness when compared with the deep-learning-based methods. For those methods that are originally applied to bicubic cases (RCAN), we also re-trained their model on the synthesised images with blur kernels before testing.

Hyper-parameter Tuning. As mentioned in Sec. 4 the proposed MLMC has the following hyper-parameters: the MLMC algorithm adopts $I = 100$, $L = 1$, $Q = 5$ for all the simulations. The number of the Monte Carlo sampling times T and the number of meta-learning optimization intervals P . We present the HR image PSNR for different hyper-parameter settings in Fig. 4. It can be seen that the perfor-

mance fluctuates around the equilibrium PSNR when the meta-learning scheme is executed ($P \geq 2$), and the variation on T has little effect on the performance. To balance the efficiency and effectiveness, we thus set $T = 10$ and $P = 5$ for all the experiments.

5.2 Comparisons with state-of-the-arts

Quantitative Results of Unsupervised Methods. Quantitative evaluation results of unsupervised methods (the model is optimized on single LR image) with scale factors from 2 to 4 are presented in Table 1, the best results are highlighted in **bold**. PAM is a classic model-based blind SISR approach, and it is difficult to get satisfactory performance when no prior is available. Specifically, DIP and Double-DIP show less effectiveness due to the poor kernel estimation accuracy. FKP-DIP and BSRDM achieve better results thanks to different pre-designed kernel priors. Meanwhile, the proposed MLMC approach achieves superior performance in all cases. It is worth noting that the proposed MLMC significantly surpasses the counterpart unsupervised method BSRDM, especially for kernel estimation results. This recalls that BSRDM follows a gradient-based kernel estimation via explicit modelling on Gaussian distribution while MLMC formulates a network-level Langevin dynamic to learn from random kernel prior for better convergence performance.

Quantitative Results of Deep-learning-based Methods. Quantitative evaluation results of DL-based methods (the model is pre-trained on LR-HR paired dataset) with scale factors 4 are presented in Table 2. The best and second best results are emphasized with **bold**. Overall, the vanilla MLMC surpasses most of the DL-based supervised methods, showing comparable performance with KULNet and being next to MANet+USRNet. Besides, when the unsupervised MLMC is accompanied with the pre-trained image estimator USRNet, the proposed MLMC+USRnet is able to realize slightly better performance than the supervised MANet+USRNet.

Visual Results. Fig. 5 presents the visually qualitative results from Set14 [67], BSD100 [68] and Urban100 [69], while Fig. 6 further shows visual results from RealSRSet [51]. Apparently, the proposed MLMC obtains the most concise blur kernels as well as restored HR images, almost keeping consistence with the ground truth, while FKP-DIP and BSRDM show different levels of distortion on the estimated kernels, and Double-DIP fails to estimate reasonable blur kernels. Particularly, the real-world image test results demonstrate that all the approaches except MLMC estimate a Gaussian-like blur kernel, whereas MLMC tends to find a non-Gaussian mode. This also verifies that an adaptive and flexible kernel estimation principle is learned from the alternative Monte Carlo simulations, and hence, fitting the real-world application better.

5.3 Generalization to out-of-distribution kernels

To further evaluate the generalization-ability of the proposed MLMC approach, it is compared with Double-DIP, BSRDM, and FKP-DIP, which use different kernel priors in more challenging cases, including out-of-distribution kernels (unseen Gaussian blur kernels with larger width range $[0.35s, 5s]$ than assumption kernel distribution within

TABLE 1

Average PSNR/SSIM of different unsupervised methods (the model is optimized on only one LR image) on public datasets that are synthesized by the random Gaussian kernels with $s = 2, 3, 4$. The best results are emphasized with **bold**.

Method	Scale	Set5	Set14	BSD100	Urban100
Bicubic	$\times 2$	26.58/0.8010	24.85/0.6939	25.19/0.6633	22.35/0.6503
PAM	$\times 2$	17.48/0.3990	17.52/0.3948	17.91/0.3826	17.15/0.3763
DIP	$\times 2$	26.82/0.7518	25.40/0.6868	24.71/0.6508	23.29/0.6749
KernelGAN+ZSSR	$\times 2$	26.45/0.7694	24.64/0.6658	23.96/0.6461	22.02/0.6263
Double-DIP	$\times 2$	24.71/0.6423	22.21/0.5626	23.31/0.5681	21.03/0.5701
FKP-DIP	$\times 2$	30.16/0.8637	27.06/0.7421	26.72/0.7089	24.33/0.7069
BSRDM	$\times 2$	30.56/0.8616	27.15/0.7435	26.69/0.7109	24.52/0.7115
MLMC (Ours)	$\times 2$	31.61/0.8836	28.52/0.7900	28.11/0.7751	25.32/0.7627
Bicubic	$\times 3$	23.38/0.6836	22.47/0.5884	23.17/0.5625	20.37/0.5378
PAM	$\times 3$	15.60/0.3596	16.23/0.3701	16.41/0.3684	15.23/0.3612
DIP	$\times 3$	28.14/0.7687	25.19/0.6581	25.25/0.6408	23.22/0.6512
KernelGAN+ZSSR	$\times 3$	25.57/0.6429	23.51/0.6216	23.08/0.6019	21.98/0.5864
Double-DIP	$\times 3$	23.21/0.6535	20.20/0.5071	20.38/0.4499	19.61/0.4993
FKP-DIP	$\times 3$	28.82/0.8202	26.27/0.6922	25.96/0.6660	23.47/0.6588
BSRDM	$\times 3$	28.84/0.8255	25.63/0.6973	25.88/0.6576	23.68/0.6783
MLMC (Ours)	$\times 3$	30.21/0.8547	27.05/0.7363	26.77/0.7076	23.96/0.7050
Bicubic	$\times 4$	21.70/0.6198	20.86/0.5181	21.95/0.5097	19.13/0.4729
PAM	$\times 4$	15.13/0.3938	15.93/0.3849	16.31/0.3894	15.06/0.3784
DIP	$\times 4$	27.34/0.7465	25.03/0.6371	24.92/0.6030	22.55/0.6128
KernelGAN+ZSSR	$\times 4$	24.46/0.6216	22.65/0.5414	21.49/0.5229	21.04/0.4979
Double-DIP	$\times 4$	20.99/0.5578	18.31/0.6129	18.57/0.3815	18.15/0.4491
FKP-DIP	$\times 4$	27.77/0.7914	24.21/0.6684	25.15/0.6354	22.89/0.6327
BSRDM	$\times 4$	27.81/0.8029	25.35/0.6859	25.61/0.6526	22.36/0.6601
MLMC (Ours)	$\times 4$	28.87/0.8129	26.23/0.6938	25.89/0.6534	23.83/0.6728

TABLE 2

Average PSNR/SSIM of different deep-learning-based methods (the model is pre-trained on LR-HR paired dataset) on public datasets that are synthesized by the random Gaussian kernels with $s = 4$. The best results are emphasized with **bold**.

Method	Scale	Set5	Set14	BSD100	Urban100
BSRGAN	$\times 4$	20.47/0.5958	19.92/0.4938	20.89/0.4814	17.61/0.3585
RCAN	$\times 4$	22.01/0.6210	20.65/0.5031	21.99/0.4986	19.15/0.4503
KDSR	$\times 4$	23.35/0.6380	21.73/0.5102	23.85/0.5643	20.44/0.4879
UDKE	$\times 4$	27.01/0.8038	24.28/0.6923	24.77/0.6695	21.54/0.6515
DASR	$\times 4$	27.37/0.7859	25.43/0.6591	25.11/0.6129	22.88/0.6448
KXNet	$\times 4$	27.15/0.8085	24.86/0.6680	24.42/0.6431	21.78/0.6543
KULNet	$\times 4$	27.89/0.8163	25.43/0.6856	25.03/0.6630	21.98/0.6562
MLMC (Ours)	$\times 4$	28.87/0.8129	26.23/0.6938	25.89/0.6534	23.83/0.6728
MANet+USRNet	$\times 4$	29.87/0.8572	26.62/0.7360	26.06/0.7080	23.93/0.6944
MLMC+USRNet (Ours)	$\times 4$	30.31/0.8607	27.46/0.7518	26.59/0.7121	23.98/0.7093

TABLE 3

Average PSNR/SSIM of image and PSNR of kernels on Set14. The best results are emphasized with **bold**.

Method	Scale	Kernel PSNR	Image PSNR/SSIM
with Out-of-distribution Kernel			
Double-DIP	$\times 2$	46.22	24.69/0.6374
FKP-DIP	$\times 2$	46.49	26.13/0.7065
BSRDM	$\times 2$	42.53	24.35/0.6728
MLMC (Ours)	$\times 2$	52.32	26.63/0.7198
Double-DIP	$\times 4$	50.62	20.51/0.4835
FKP-DIP	$\times 4$	54.46	24.73/0.6344
BSRDM	$\times 4$	45.38	21.57/0.5510
MLMC (Ours)	$\times 4$	58.20	25.09/0.6474

width range $[0.175s, 2.5s]$. As we can see from Table 3, the proposed MLMC significantly outperforms the competitors in the case with out-of-distribution kernel by large margins. We note that FKP-DIP experiences performance drops in

the these challenge cases, due to the high dependency on pre-trained kernel priors, while MLMC still shows superior performance. In this case, we conclude that MLMC has superior generalization-ability to arbitrary kernels and exhibits satisfactory robustness to unseen kernels and non-Gaussian kernels, even without specific kernel priors.

5.4 Generalization to motion kernels

The proposed MLMC can be directly expended on solving motion kernel tasks without re-training. The fundamental technique process keeps consistent with the Gaussian kernel case, since our MLMC only replaces the random kernel generation function follows settings in [65]. We visualize the generated random motion kernels in Fig. 7. Table 5 presents the quantitative evaluation results of different methods in the motion kernel scenario. Compared with the Gaussian kernel scenario, all the methods show different degrees of

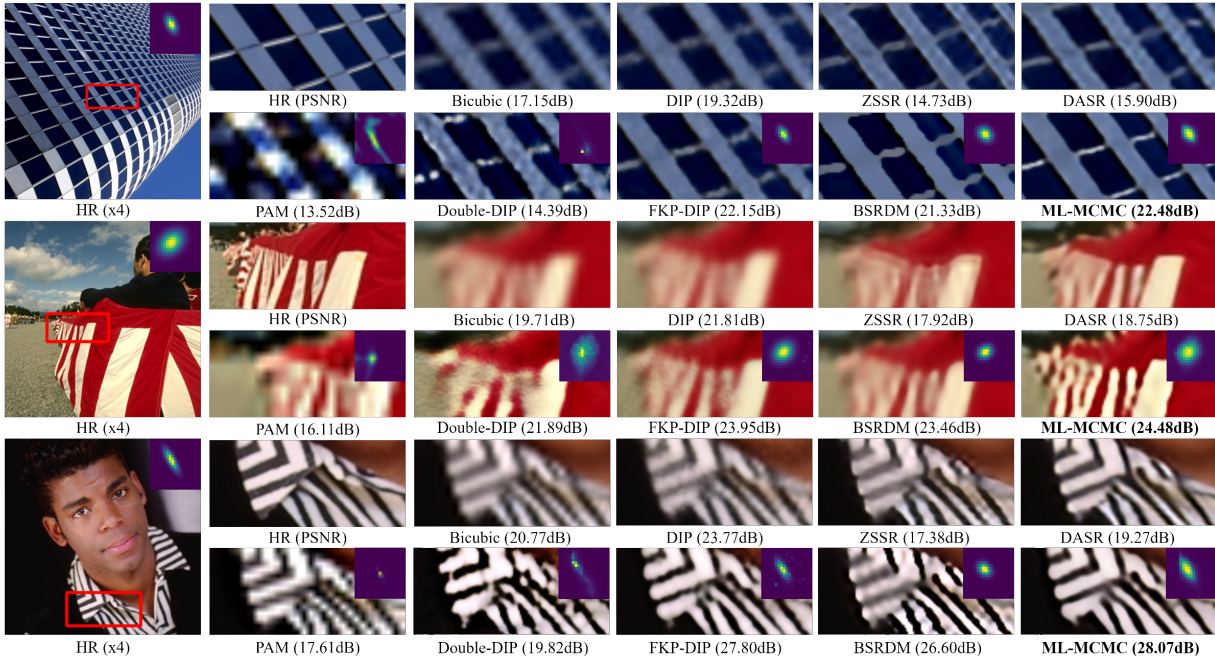


Fig. 5. Visual results of different methods on public datasets for scale factor 4. Estimated/ground-truth kernels are shown on the top right.

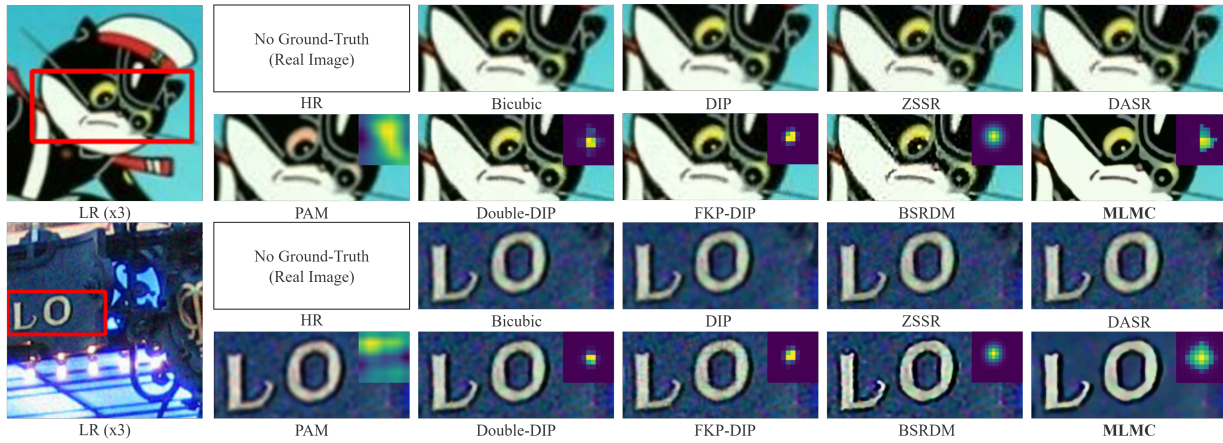


Fig. 6. Visual results of real world images on RealSRSet [51] for scale factor 3. Estimated kernels are shown on the top right.



Fig. 7. The visualization of the random motion blur kernels.

performance drop, while the proposed MLMC shows significantly better performance than other alternatives. This verifies the better generalization ability of our MLMC towards more complex degradation kernels.

5.5 Robustness to image noise

We add image noise (3.92% and 7.84% of the maximum image pixel value) to the LR image after blurring and downsampling. In Table 4, the proposed MLMC produces comparable result in all the cases, and showing good robustness to different levels of image noise. Although MLMC has a modest performance drop when the image is corrupted by noise, it still surpasses all of the other comparative methods.

In this case, we argue that MLMC is able to handle more complicated degradation model with better robustness.

5.6 Ablation Studies

In Fig. 8, intermediate kernel results of three ablation experiments are depicted to highlight the effectiveness of the introduced Monte Carlo kernel approximation and meta-learning SISR stages, respectively. It can be seen that the estimated kernel without Monte Carlo simulations has non-negligible drop in PSNR as well as significant distortion, compared to MLMC. Meanwhile, the absence of the meta-learning scheme leads to a significant visible-deviation during the optimization, indicating a different and worse local optima compared to the one MLMC reached.

In Table 6, we further present the average results of the ablation experiments of the proposed Monte Carlo kernel approximation and meta-learning SISR stages under two different scale factors: 2 and 4, two datasets: Set14 [67] and BSD100 [68]. We can see that the proposed the MLMC

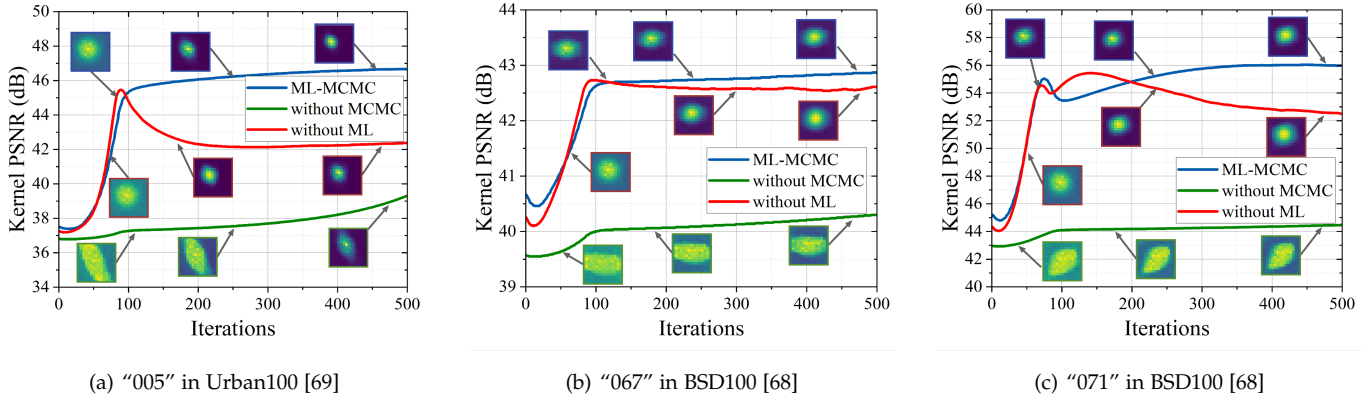


Fig. 8. The intermediate kernel results of MLMC, MLMC without Monte Carlo and MLMC without meta-learning over iterations on three test images.

TABLE 4

Average PSNR/SSIM of image and PSNR of kernels on Set14. The best results are emphasized with **bold**.

Method	Scale	Kernel PSNR	Image PSNR/SSIM
with Image Noise of 3.92%			
Double-DIP	$\times 2$	40.95	23.93/0.6134
FKP-DIP	$\times 2$	44.38	27.14/0.7228
BSRDM	$\times 2$	43.96	24.97/0.7069
MLMC (Ours)	$\times 2$	44.65	27.24/0.7345
Double-DIP	$\times 4$	44.60	18.42/0.3897
FKP-DIP	$\times 4$	47.96	24.68/0.5760
BSRDM	$\times 4$	48.73	20.99/0.5387
MLMC (Ours)	$\times 4$	49.53	24.96/0.5951
with Image Noise of 7.84%			
Double-DIP	$\times 2$	40.23	22.57/0.5376
FKP-DIP	$\times 2$	42.51	26.21/0.6857
BSRDM	$\times 2$	43.02	24.59/0.6781
MLMC (Ours)	$\times 2$	45.84	26.37/0.6913
Double-DIP	$\times 4$	42.88	15.77/0.2628
FKP-DIP	$\times 4$	45.44	24.06/0.5540
BSRDM	$\times 4$	46.87	20.82/0.5219
MLMC (Ours)	$\times 4$	47.20	24.53/0.5834

TABLE 5

Average PSNR/SSIM of image and PSNR of kernels on Set14 and BSD100 in the motion kernel scenario. The best results are emphasized with **bold**.

Method	Scale	Kernel PSNR	Image PSNR/SSIM
with Motion Kernel on Set14			
Double-DIP	$\times 2$	31.07	22.80/0.6449
FKP-DIP	$\times 2$	33.97	25.99/0.8157
BSRDM	$\times 2$	30.31	23.62/0.6447
MLMC (Ours)	$\times 2$	34.44	28.48/0.8327
Double-DIP	$\times 4$	37.30	20.96/0.4817
FKP-DIP	$\times 4$	38.07	22.79/0.6278
BSRDM	$\times 4$	36.27	21.12/0.5491
MLMC (Ours)	$\times 4$	39.44	25.22/0.6776
with Motion Kernel on BSD100			
Double-DIP	$\times 2$	29.94	22.31/0.6388
FKP-DIP	$\times 2$	32.49	25.27/0.7784
BSRDM	$\times 2$	30.63	23.88/0.5999
MLMC (Ours)	$\times 2$	33.93	28.10/0.8087
Double-DIP	$\times 4$	35.45	18.69/0.3646
FKP-DIP	$\times 4$	37.06	23.07/0.6264
BSRDM	$\times 4$	36.65	21.31/0.5543
MLMC (Ours)	$\times 4$	40.08	25.58/0.6554

achieves the best results for all the cases. As we analyze in the Section 4.3 and Section 4.4 that on the one hand, without Monte Carlo simulations, the estimated kernels are not regular, which leads to bad kernel estimation and image restoration performance. On the other hand, without meta-learning SISR scheme, the estimated kernels are easily converged to local optimum, which results in the unsatisfactory image restoration performance. For kernel estimation, we also evaluate the necessity of the MCMC simulation towards the kernel prior learning. Fig. 9 shows the visualization of ablation on real world images from three cases. It is clear that without MCMC simulation, the kernel estimation no longer provide performance improvements on HR image restoration and the estimated kernels are distorted. Meanwhile, we can see that the MLMC tends to estimate non-Gaussian kernels via combinations of Gaussian sampling results to fit the real-world degradations.

In table 7, we validate the effects of different regularizers in Eq. (35). The hyper-Laplacian, Tikhonov, and TV regularizers are compared in two cases with different levels of noise. In both cases, hyper-Laplacian achieves the best

TABLE 6

Ablation results on Set14 and BSD100 with scale factor 2 and 4. The best results are emphasized with **bold**.

Method	Scale	Kernel PSNR	Image PSNR/SSIM
with Gaussian Kernel on Set14			
without Monte Carlo	$\times 2$	41.30	24.23/0.6318
without Meta-learning	$\times 2$	42.54	26.94/0.7353
MLMC (Ours)	$\times 2$	45.01	28.52/0.7900
without Monte Carlo	$\times 4$	44.51	19.83/0.4999
without Meta-learning	$\times 4$	51.39	25.01/0.6869
MLMC (Ours)	$\times 4$	55.98	26.23/0.6938
with Gaussian Kernel on BSD100			
without Monte Carlo	$\times 2$	37.58	22.14/0.5340
without Meta-learning	$\times 2$	40.43	25.76/0.7020
MLMC (Ours)	$\times 2$	47.84	28.11/0.7751
without Monte Carlo	$\times 4$	49.17	20.56/0.4782
without Meta-learning	$\times 4$	51.93	24.85/0.6210
MLMC (Ours)	$\times 4$	53.16	25.89/0.6534

performance, indicating the robustness towards noise level variation.

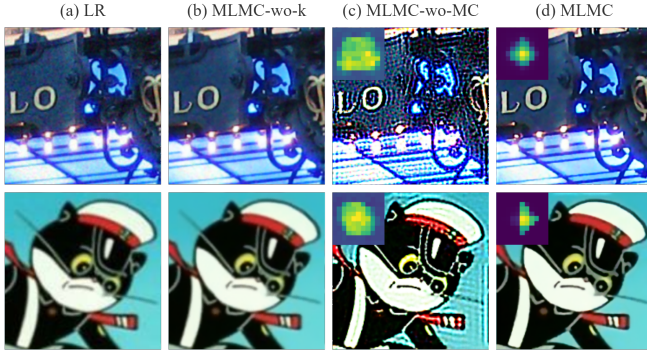


Fig. 9. Three ablation cases: (a) LR image, (b) MLMC-wo-k (MLMC without the kernel estimation process), (c) MLMC-wo-MC (MLMC without the Monte-Carlo sampling process), and (d) original MLMC.

TABLE 7

Ablation results of regularizers in Eq. (35) of our MLMC on Set5 with scale factor 4. R1: Hyper-Laplacian, R2: Tikhonov, R3: TV. The best results are emphasized with **bold**.

R1	R2	R3	Kernel PSNR	Image PSNR/SSIM
with Image Noise of 3.92%, Scale $\times 4$				
$\times$	$\times$	$\times$	49.38	26.25/0.6448
$\checkmark$	$\times$	$\times$	49.40	27.65/0.7707
$\times$	$\checkmark$	$\times$	48.87	26.71/0.6824
$\times$	$\times$	$\checkmark$	49.04	26.82/0.6967
with Image Noise of 7.84%, Scale $\times 4$				
$\times$	$\times$	$\times$	45.57	24.07/0.5163
$\checkmark$	$\times$	$\times$	46.05	25.57/0.6493
$\times$	$\checkmark$	$\times$	45.40	24.59/0.5576
$\times$	$\times$	$\checkmark$	45.53	24.63/0.5598

TABLE 8

Comparison of different methods from tested LR image size of 256×256 with scale factor $s = 2, 3, 4$. Computational requirements are model size (K) and runtime (s).

Method	Scale	Pre-training	Model Size	Time
ZSSR [64]	$\times 2$	$\times$	225K	56s
Double-DIP [4]	$\times 2$	$\times$	2359K + 641K	47s
FKP-DIP [13]	$\times 2$	6 hours	2359K + 195K	45s
BSRDM [6]	$\times 2$	$\times$	766K	37s
MLMC (Ours)	$\times 2$	$\times$	2359K + 562K	41s
ZSSR [64]	$\times 4$	$\times$	225K	235s
Double-DIP [4]	$\times 4$	$\times$	2359K + 641K	239s
FKP-DIP [13]	$\times 4$	6 hours	2359K + 143K	232s
BSRDM [6]	$\times 4$	$\times$	766K	190s
MLMC (Ours)	$\times 4$	$\times$	2359K + 562K	215s

5.7 Model Size, Runtime and Memory Usage

Table 8 compares the results on model size (number of parameters), runtime and pre-training requirements of the four approaches. All simulations are accelerated by GeForce RTX 3090 GPU. The input LR images are with size 256×256 and scale factors $s = 2, 3, 4$. It can be seen that the MLMC approach has similar model size and enjoys competitive runtimes compared to the latest learning-based approaches. Meanwhile, pre-training-based approaches, such as FKP-DIP, which typically requests 5-6 hours for pre-training. The memory usage of our MLMC on a GeForce RTX 3090

GPU for generating a HR image of size 1024×1024 are about 11GB memory, which is close to the Double-DIP (11.2GB) and DIP-FKP (10.6GB). We note that the plug-and-play fashion and the better flexibility towards unknown degradations allow significant merits on dealing with blind SR tasks with real-world scenarios, especially those scenarios without high quality training data and with complex blurring, such as space high-speed targets (e.g., satellites, aircraft) and medical images (e.g., beating heart).

6 CONCLUSION

In this paper, we have proposed a new learning-based blind SISR method, which combines Markov Chain Monte Carlo simulations and meta-learning training to achieve superior kernel estimation. Most strikingly, the proposed approach does not require any supervised pre-training or parametric priors. In future work, we will investigate two main directions for better practicality of the proposed MLMC methods, including i) the expansion to more degradation models, such as compression artifacts, deraining, and shadow-removal, to improve the generalization ability; and ii) the application with more advance pre-trained SR models, for example, USRNet and Diffusion model, to play the role of kernel prior learning module for performance improvements. We believe that the concept introduced here, in particular, learning from randomness to provide priors and the meta-learning-based non-convex optimization algorithm, will lead to a new direction of solving blind image retoration tasks to achieve superior performance with limited computational complexity.

REFERENCES

- [1] D. Perrone and P. Favaro, "A clearer picture of total variation blind deconvolution," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 6, pp. 1041–1055, 2015.
- [2] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9446–9454.
- [3] S. Bell-Kligler, A. Shocher, and M. Irani, "Blind super-resolution kernel estimation using an internal-gan," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [4] D. Ren, K. Zhang, Q. Wang, Q. Hu, and W. Zuo, "Neural blind deconvolution using deep priors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3341–3350.
- [5] M. Emad, M. Peemen, and H. Corporaal, "Dualsr: Zero-shot dual learning for real-world super-resolution," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1630–1639.
- [6] Z. Yue, Q. Zhao, J. Xie, L. Zhang, D. Meng, and K.-Y. K. Wong, "Blind image super-resolution with elaborate degradation modeling on noise and kernel," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2128–2138.
- [7] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Fast and accurate image super-resolution with deep laplacian pyramid networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 11, pp. 2599–2613, 2018.
- [8] S. Anwar and N. Barnes, "Densely residual laplacian super-resolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [9] J. Liu, W. Zhang, Y. Tang, J. Tang, and G. Wu, "Residual feature aggregation network for image super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2359–2368.

- [10] Y. Mei, Y. Fan, Y. Zhou, L. Huang, T. S. Huang, and H. Shi, "Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5690–5699.
- [11] Y. Huang, S. Li, L. Wang, T. Tan *et al.*, "Unfolding the alternating optimization for blind super resolution," *Advances in Neural Information Processing Systems*, vol. 33, pp. 5632–5643, 2020.
- [12] L. Wang, Y. Wang, X. Dong, Q. Xu, J. Yang, W. An, and Y. Guo, "Unsupervised degradation representation learning for blind super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 581–10 590.
- [13] J. Liang, K. Zhang, S. Gu, L. Van Gool, and R. Timofte, "Flow-based kernel prior with application to blind super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 601–10 610.
- [14] Z. Luo, Y. Huang, S. Li, L. Wang, and T. Tan, "Learning the degradation distribution for blind image super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6063–6072.
- [15] X. Kong, X. Liu, J. Gu, Y. Qiao, and C. Dong, "Reflash dropout in image super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6002–6012.
- [16] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [17] Z. Fang, F. Wu, W. Dong, X. Li, J. Wu, and G. Shi, "Self-supervised non-uniform kernel estimation with flow-based motion prior for blind image deblurring," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [18] J. Li and M. Hu, "Continuous model adaptation using online meta-learning for smart grid application," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
- [19] L. Chen, B. Hu, Z.-H. Guan, L. Zhao, and X. Shen, "Multiagent meta-reinforcement learning for adaptive multipath routing optimization," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
- [20] J. Xia and D. Gunduz, "Meta-learning based beamforming design for miso downlink," in *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2021, pp. 2954–2959.
- [21] J.-Y. Xia, S. Li, J.-J. Huang, Z. Yang, I. M. Jaimoukha, and D. Gündüz, "Metalearning-based alternating minimization algorithm for nonconvex optimization," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [22] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient langevin dynamics," in *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 681–688.
- [23] M. Arvinte and J. I. Tamir, "Mimo channel estimation using score-based generative models," *IEEE Transactions on Wireless Communications*, 2022.
- [24] M. F. T. B. C. Russell and W. T. Freeman, "Exploiting the sparse derivative prior for super-resolution and image demosaicing," in *Proceedings of the Third International Workshop Statistical and Computational Theories of Vision*, 2003, pp. 1–28.
- [25] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *2009 IEEE 12th international conference on computer vision*. IEEE, 2009, pp. 349–356.
- [26] T. Michaeli and M. Irani, "Nonparametric blind super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 945–952.
- [27] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.
- [28] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-laplacian priors," *Advances in neural information processing systems*, vol. 22, 2009.
- [29] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 6, pp. 1127–1133, 2010.
- [30] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
- [31] M. Jin, S. Roth, and P. Favaro, "Normalized blind deconvolution," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 668–684.
- [32] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European conference on computer vision*. Springer, 2014, pp. 184–199.
- [33] J. Kim, J. K. Lee, and K. M. 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.
- [34] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3262–3271.
- [35] Y.-S. Xu, S.-Y. R. Tseng, Y. Tseng, H.-K. Kuo, and Y.-M. Tsai, "Unified dynamic convolutional network for super-resolution with variational degradations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 496–12 505.
- [36] Z. Hui, J. Li, X. Wang, and X. Gao, "Learning the non-differentiable optimization for blind super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2093–2102.
- [37] S. Y. Kim, H. Sim, and M. Kim, "Koalernet: Blind super-resolution using kernel-oriented adaptive local adjustment," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 10 611–10 620.
- [38] K. Zhang, J. Liang, L. Van Gool, and R. Timofte, "Designing a practical degradation model for deep blind image super-resolution," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4791–4800.
- [39] Z. Fang, W. Dong, X. Li, J. Wu, L. Li, and G. Shi, "Uncertainty learning in kernel estimation for multi-stage blind image super-resolution," in *European Conference on Computer Vision*. Springer, 2022, pp. 144–161.
- [40] B. Xia, Y. Zhang, Y. Wang, Y. Tian, W. Yang, R. Timofte, and L. Van Gool, "Knowledge distillation based degradation estimation for blind super-resolution," *arXiv preprint arXiv:2211.16928*, 2022.
- [41] H. Zheng, H. Yong, and L. Zhang, "Unfolded deep kernel estimation for blind image super-resolution," in *European Conference on Computer Vision*. Springer, 2022, pp. 502–518.
- [42] J. Liang, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Mutual affine network for spatially variant kernel estimation in blind image super-resolution," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4096–4105.
- [43] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-sr: A magnification-arbitrary network for super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1575–1584.
- [44] J. W. Soh, S. Cho, and N. I. Cho, "Meta-transfer learning for zero-shot super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3516–3525.
- [45] H. Li, Y. Cen, Y. Liu, X. Chen, and Z. Yu, "Different input resolutions and arbitrary output resolution: a meta learning-based deep framework for infrared and visible image fusion," *IEEE Transactions on Image Processing*, vol. 30, pp. 4070–4083, 2021.
- [46] C. Yu, B. Xiao, C. Gao, L. Yuan, L. Zhang, N. Sang, and J. Wang, "Lite-hrnet: A lightweight high-resolution network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 440–10 450.
- [47] B. Xia, Y. Tian, Y. Zhang, Y. Hang, W. Yang, and Q. Liao, "Meta-learning based degradation representation for blind super-resolution," *arXiv preprint arXiv:2207.13963*, 2022.
- [48] V. Cornillere, A. Djelouah, W. Yifan, O. Sorkine-Hornung, and C. Schroers, "Blind image super-resolution with spatially variant degradations," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–13, 2019.
- [49] K. Zhang, W. Zuo, and L. Zhang, "Deep plug-and-play super-resolution for arbitrary blur kernels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1671–1681.
- [50] K. Zhang, L. V. Gool, and R. Timofte, "Deep unfolding network for image super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3217–3226.
- [51] Y. Li, M. Tofighi, J. Geng, V. Monga, and Y. C. Eldar, "Efficient and interpretable deep blind image deblurring via algorithm unrolling," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 666–681, 2020.

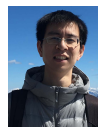
- [52] J. Fu, H. Wang, Q. Xie, Q. Zhao, D. Meng, and Z. Xu, "Kxnet: A model-driven deep neural network for blind super-resolution," in *European Conference on Computer Vision*. Springer, 2022, pp. 235–253.
- [53] J. Gu, H. Lu, W. Zuo, and C. Dong, "Blind super-resolution with iterative kernel correction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1604–1613.
- [54] A. Li, W. Huang, X. Lan, J. Feng, Z. Li, and L. Wang, "Boosting few-shot learning with adaptive margin loss," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12 576–12 584.
- [55] Z. Yang, J.-Y. Xia, J. Luo, S. Zhang, and D. Gündüz, "A learning aided flexible gradient descent approach to miso beamforming," *IEEE Wireless Communications Letters*, pp. 1–1, 2022.
- [56] M. Elad and A. Feuer, "Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images," *IEEE transactions on image processing*, vol. 6, no. 12, pp. 1646–1658, 1997.
- [57] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *International Journal of Imaging Systems and Technology*, vol. 14, no. 2, pp. 47–57, 2004.
- [58] C. Liu and D. Sun, "On bayesian adaptive video super resolution," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 2, pp. 346–360, 2013.
- [59] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 286–301.
- [60] Y. Gandelsman, A. Shocher, and M. Irani, "" double-dip": Unsupervised image decomposition via coupled deep-image-priors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 026–11 035.
- [61] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [62] G. Riegler, S. Schuler, M. Ruther, and H. Bischof, "Conditioned regression models for non-blind single image super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 522–530.
- [63] W.-Z. Shao and M. Elad, "Simple, accurate, and robust nonparametric blind super-resolution," in *Image and Graphics*. Springer, 2015, pp. 333–348.
- [64] A. Shocher, N. Cohen, and M. Irani, ""zero-shot" super-resolution using deep internal learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3118–3126.
- [65] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8183–8192.
- [66] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *British Machine Vision Conference*, 2012, pp. 135–1.
- [67] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*. Springer, 2010, pp. 711–730.
- [68] 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 Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2. IEEE, 2001, pp. 416–423.
- [69] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5197–5206.
- [70] 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.



Jingyuan Xia is currently an Associate Professor with the College of the Electronic Science, the National University of Defense Technology (NUDT). His current research interests include low level image processing, nonconvex optimization, and machine learning for signal processing.



Zhixiong Yang is now pursuing a Ph.D. degree at the College of Electronic Science, NUDT. His research interests include deep learning on signal processing and image processing.



Shengxi Li (Member, IEEE) is a Professor with Beihang University. His research interests include generative models, statistical signal processing, rate-distortion theory, and perceptual video coding.



Shuanghui Zhang (Member, IEEE) is currently a Professor with the College of Electrical Science and Technology, NUDT. His research interests lie in compressive sensing, sparse signal recovery techniques, Bayesian inference, and their applications in radar signal processing.



Yaowen Fu is currently a Professor with the College of Electronic Science and Technology, NUDT. His research interests include information fusion, radar signal processing, and other aspects of research.



Deniz Gündüz is a Fellow of the IEEE, and a Distinguished Lecturer for the IEEE Information Theory Society (2020-22). He is the recipient of a Consolidator Grant of the European Research Council (ERC) in 2022, the IEEE Communications Society - Communication Theory Technical Committee (CTTC) Early Achievement Award in 2017, a Starting Grant of the European Research Council (ERC) in 2016, and several best paper awards. He is currently a Professor in Imperial College London.



Xiang Li was elected as the Academician of the Chinese Academy of Sciences in 2022, and a Professor with NUDT. His research interests include signal processing, automation target recognition, and machine learning.