Deep Variational Network Toward Blind Image Restoration

Zongsheng Yue, Hongwei Yong, Qian Zhao, Member, IEEE, Deyu Meng, Member, IEEE, and Kwan-Yee K. Wong, Senior Member, IEEE

Abstract—Blind image restoration (IR) is a common yet challenging problem in computer vision. Classical model-based methods and recent deep learning (DL)-based methods represent two different methodologies for this problem, each with their own merits and drawbacks. In this paper, we propose a novel blind image restoration method, aiming to integrate both the advantages of them. Specifically, we construct a general Bayesian generative model for the blind IR, which explicitly depicts the degradation process. In this proposed model, a pixel-wise non-i.i.d. Gaussian distribution is employed to fit the image noise. It is with more flexibility than the simple i.i.d. Gaussian or Laplacian distributions as adopted in most of conventional methods, so as to handle more complicated noise types contained in the image degradation. To solve the model, we design a variational inference algorithm where all the expected posteriori distributions are parameterized as deep neural networks to increase their model capability. Notably, such an inference algorithm induces a unified framework to jointly deal with the tasks of degradation estimation and image restoration. Further, the degradation information estimated in the former task is utilized to guide the latter IR process. Experiments on two typical blind IR tasks, namely image denoising and super-resolution, demonstrate that the proposed method achieves superior performance over current state-of-the-arts.

Index Terms—Image restoration, denoising, super-resolution, generative model, variational inference.

I. INTRODUCTION

IMAGE Restoration (IR) is an active research topic in the fields of signal processing and computer vision. It aims at recovering the latent high-quality image z from the observed corrupted counterpart y, i.e.,

\[ y = H z + n, \]

where H is the degradation operator, and n is image noise.

With different degradation settings for \( H \), (1) represents different IR tasks. For example, the classical IR tasks, such as image denoising, deblurring, and super-resolution, can be easily obtained by setting \( H \) as an identity matrix, a blurring operator, and a composition of blurring and downsampling operators, respectively. The difficulties of IR tasks mainly come from \( H \) and \( n \). The former inclines to cause severe information loss in some tasks, like deblurring and super-resolution, and the latter is usually complicated due to the accumulation of noises from multiple sources e.g., capturing instrument, camera pipeline and image transmission [1]. In blind IR tasks, we need to simultaneously solve the problems of degradation estimation and image restoration, which makes it more challenging.

In the past decades, plenty of IR methods have been proposed under the maximum a posteriori (MAP) framework. From the Bayesian perspective, it generally involves a likelihood term and a prior term. More specifically, the likelihood term encodes the image degradation process of (1), while the prior term reflects our subjective knowledge on the latent high-quality image. Most of these methods mainly focused on designing more effective image priors so as to alleviate the ill-posedness of IR tasks. Commonly used image priors include total variation (TV) [2], non-local similarity [3], [4], sparsity [5], [6], [7], low-rankness [8], [9], [10] and so on. In contrast, other works focused on the likelihood term by constructing more flexible noise distributions, e.g., mixture of Gaussian (MoG) [11], mixture of Exponential (MoEP) [12], and Dirichlet Process mixture of Gaussian (DP-MoG) [13], [14], [15]. Even though these model-based methods are with highly intuitive physical meanings and also generalize well in most of scenarios, they still have evident defects. First, these methods are always time-consuming, since they require to re-solve the whole model for any new testing images. Such one-by-one optimized paradigm tends to bring up large computational burden, making them very hard to be applied in real applications. Second, limited by the manually designed likelihood and image priors, which usually cannot faithfully represent the image knowledges, they struggle to handle some complex modeling problems in real cases, such as the blind IR tasks with complicated image degradations.
Different from the aforementioned model-based methods, current deep learning (DL)-based methods represent another research trend. Their core idea is to employ the deep neural networks (DNNs), being with powerful fitting capability, to directly learn the image knowledge from large amount of pre-collected image pairs in an end-to-end training manner. Dong et al. [16] and Zhang et al. [17] first proposed SRCNN and DnCNN that surpassed classical model-based methods in image super-resolution and denoising, respectively. Subsequently, many DL-based methods [18], [19], [20], [21], [22], [23], [24], [25], [26] were proposed and they achieved unprecedented successes in the field of IR. While they have achieved huge boost in performance, most of them ignore the modeling mechanism underlying the image degradation, especially the image noise. To be specific, the $L_2$ or $L_1$ losses commonly-used in current DL-based methods indeed imply that the noise $n$ in (1) obeys the independent and identically distributed (i.i.d.) Gaussian or Laplacian distributions. This, however, always deviates from true noise configurations in real cases. For example, the camera sensor noises in practical image denoising are signal-dependent and spatially variant, and thus evidently non-i.i.d. in statistics. Neglecting such intrinsic noise properties will certainly harm the generalization capability of the model in real scenarios with complicated noises.

As analyzed above, the model-based methods are capable of encoding the image degradation through the likelihood, but hindered by the limited model capacity and the slow inference speed. In contrast, the DL-based methods, equipped with DNNs, are with large model capacity and powerful non-linear fitting capability. What’s more, in the testing phase, these methods are much faster than the model-based method, since they only need one feedforward pass for any newly coming image. This naturally inspires us to develop a new IR method, which is expected to combine both the advantages of the classical model-based methods and recent DL-based methods. In this work, we take one step forward along this research line by proposing a deep variational model for blind IR. It first constructs a traditional probabilistic model for IR, and then embeds the powerful DNNs into its posteriori inference to increase the model capacity. Specifically, the contributions of this work can be summarized from two aspects, namely model construction and algorithm design, as follows:

- On one hand, a Bayesian generative model is built for general IR tasks, and thus naturally inherits the advantages of classical model-based methods. Furthermore, we also consider more complicated degradation process when building our model in this study:
  - Instead of the simple i.i.d. Gaussian or Laplacian noise assumptions in most of the current methods, a pixel-wise non-i.i.d. distribution is adopted in our model to handle more complicated noise types. In essence, such noise model induces a learnable re-weighted loss purely relying on data-self, thus is more flexible.
  - A concise kernel prior is specifically designed for super-resolution, which makes our model able to deal with the task of blind image super-resolution.

- On the other hand, we elaborately design an amortized variational inference (VI) algorithm to solve the proposed generative model. Compared with the classical mean field VI methodology, two-fold substantial modifications are made to better comply with blind IR tasks:
  - Different from the commonly used independent factorization strategy in VI, we factorize the expected posterior, namely the joint distribution of the degradation information and the latent clean image, into a conditional form. Such a formulation derives a unified framework to simultaneously deal with the tasks of degradation estimation and image restoration, in which the degradation information estimated by the former provides sound guidance for the subsequent restoration task.
  - To largely increase the fitting capability of our model, the desirable posteriori distributions are parameterized by DNNs, and then optimized in an amortized manner during training. In the testing phase, the well-trained model is capable of fastly inferring the posteriori distribution of any new testing image in an explicit manner, and thus evidently more efficient than the classical model-based methods.

In summary, this work aims to explore a novel modeling paradigm, which is expected to integrate the merits both of the classical model-based methods and recent DL-based methods, for the IR problem. A preliminary version of this work has been published in NeurIPS 2019 [27] which focuses only on image denoising. This present work makes substantial improvements on model construction, the inference algorithm, and the empirical evaluations over the conference version. Especially, we consider a more general degradation process (i.e., (1)) to build the Bayesian generative model, making it capable of handling more complicated and general IR tasks, such as blind image super-resolution.

The remainder of the paper is organized as follows: Section II introduces the related work. Section III proposes our generative model, and discusses two typical IR tasks. Section IV presents the designed stochastic VI algorithm for solving our model. In Section V, experiments are demonstrated to evaluate the performance of our method. Section VI finally concludes the paper.

II. RELATED WORKS

In this section, we first review model-based and DL-based IR methods. We then briefly summarize recent explorations that attempts to combine both of these two methodologies.

A. Model-Based Methods

Most of the classical model-based methods can be formulated into the MAP framework, which contains a likelihood (fidelity) term and a prior (regularization) term from the Bayesian perspective. Relevant developments thus mainly focused on these two terms.

Prior Modeling Methods: Aiming at alleviating the ill-posed issue of IR, many studies attempted to exploit rational image prior knowledge. Statistical regularities exhibited in images
were first employed, e.g., TV denoising [2] and wavelet cor-
ing [28]. Then, NLM [3] and BM3D [4] were both proposed for
denoising based on the non-local self-similarity prior, meaning
that small image patches in a non-local image area possess
similar configurations. Later, low-rankness [8], [9], [10] and
sparsity [4], [5], [6], [7] priors, which also aim to explore the
characteristics of image patches, became popular and were
widely used in IR tasks. To further increase the model’s capacity
and expression ability, some other methods moved from ana-
litical technologies to data-driven approaches. E.g., Roth and
Black [29] proposed the fields of experts (FoE) to learn image
 priors. Barbu [30] trained a discriminative model for the Markov
random field (MRF) prior, while Sun and Tappen [31] proposed
a non-local range MRF (NLR-MRF) model. More related works
can be found in [32], [33].

Noise Modeling Methods: Different from the prior modeling
methods, noise modeling methods focus on the likelihood (fi-
delity) term of the MAP framework. In fact, the widely used $L_1$
or $L_2$ loss functions implicitly make the i.i.d. Gaussian or Lapla-
cian assumptions on image noise, which often under-estimates
the complexity of real noise. Based on this observation, Meng
et al. [11] proposed the MoG noise modeling method under
the low-rankness framework. Furthermore, Zhu et al. [13], [34]
and Yue et al. [14], [15] both introduced the non-parametric
Dirichlet Process into MoG to increase its flexibility, leading to
the adaptive adjustment for the component number of MoG.

B. DL-Based Methods

DL-based methods represent a data-driven trend for the IR
task. They straightforwardly train an explicit mapping function
parameterized by DNN on large amount of image pairs. The
earliest convolutional neural network (CNN) method can be
traced back to [35], in which a five-layer network was employed.
Some auto-encoder based methods were then proposed in [36],
[37]. Due to the insufficient research of DL technologies, how-
ever, these methods were inferior to the model-based methods
in performance.

The first significant improvement of DL-based methods was
achieved by [38] which obtained comparable performance with
BM3D [4] in denoising task using a plain multilayer perceptron.
Subsequently, Dong et al. [16] proposed the first CNN model
for super-resolution, and it outperformed the classical model-
based methods. With the advances of deep learning, Zhang
et al. [17] trained a deep CNN model named DnCNN and achieved
state-of-the-art performance in denoising, JPEG deblocking,
and super-resolution. Since then, the DL-based methods began
to dominate the research trend in almost all of the IR tasks,
especially in denoising [18], [20], [23], [25], [39] and super-
resolution [40], [41], [42], [43], [44], [45], [46].

Inspired by the development of generative adversarial network
(GAN) [47], some DL-based methods also followed the research
line of noise modeling introduced in Section II-A. Typically,
Chen et al. [48] proposed a noise generator to simulate the
real noise under the adversarial training mechanism, and Kim
et al. [49] further introduced some camera settings (e.g., ISO
level and shutter speed) into the generator as extra guidance.
More recently, Yue et al. [24] proposed a dual adversarial loss to
implement the noise removal and noise generation tasks into
one unique Bayesian framework. Different from these GAN
based implicit noise modeling manners, this study adopts a
more powerful and flexible non-i.i.d. Gaussian distribution to
explicitly model the image noise, which avoids the instability in
training GAN.

C. Some Relevant Explorations

The researches to combine the model-based methods and
the DL-based methods have attracted increasingly attentions in
recent years. Some significant explorations have been attempted
towards this goal.

Deep plug-and-play methods [50], [51], [52], [53], [54]
usually replace the denoising subproblem in model-based meth-
ods with one or multiple pre-trained DNN denoisers, so as to
leverage the abundant image prior knowledge learned by such
denoisers. Due to the lacking of end-to-end training, they
always rely on tedious hyper-parameters tuning to guarantee
stable performance. To alleviate this drawback, deep unfolding
methods [55], [56], [57], [58], [59] take a step forward by em-
bedding the DNNs into traditional optimization algorithms (e.g.,
HQS, ADMM). Attached to the end-to-end training manner,
these deep unfolding method achieve promising performance in
some IR tasks.

Deep image prior (DIP) [60] and its related methods [61],
[62], [63], [64] represent another significant approach along this
research line. These methods aim to seek a corresponding DNN
model that maps the pre-sampled noise to the desirable clean
image for any given corrupted image under the MAP framework.
Similar to the model-based methods, they are mainly limited by
the time-consuming optimization process in the testing phase.

In this study, we develop a general and novel deep IR model
which is evidently different from the above research approaches.
From the model perspective, a Bayesian generative model is con-
structed for general IR tasks, naturally inheriting the capability
of modeling the image degradation from the classical model-
based methods. From the algorithm perspective, we design an
amortized VI algorithm which parameterizes all the involved
posteriori distributions with DNNs. The embedded DNNs in our
algorithm equip it with powerful fitting ability and fast testing
speed as recent DL-based methods.

III. THE PROPOSED METHOD

A. Basic Settings

In this paper, we consider two commonly used settings on
the degradation operator $H$ of (1). In the first case, $H$ is an
identity matrix, corresponding to the task of image denois-
ing. The difficulties of this task are naturally attributed to the
complexity of the image noise, which is often spatially variant
and signal-dependent in real scenarios. Besides, their statistics
(e.g., the noise levels) are always unknown in blind image
denoising. It is thus necessary to devise methods to estimate
the noise distribution and recover the latent high-quality image
simultaneously.

In the second case, we consider a more general IR task,
namely image super-resolution, in which $H$ is a composition
of blurring and downsampling. The downsampling operation leads to serious information lost, especially in the case of large scale factor, making it more challenging compared with denoising. Similarly, blind super-resolution also involves two subtasks, i.e., estimating the degradation information, including both of the blue kernel and noise distribution, and restoring the high-resolution image.

In addition, we briefly introduce some necessary settings on the training and testing data. The training data consists of multiple triplets, i.e., \( D = \{ y^{(j)}, x^{(j)}, H^{(j)} \}_{j=1}^N \), where \( y^{(j)} \) and \( x^{(j)} \) denote the corrupted image and the underlying high-quality image, respectively. \( H^{(j)} \) represents the degradation operator, which is an identity matrix for denoising and a blur kernel for super-resolution. The superscript \( j \) of \( H^{(j)} \) indicates that the degradation model varies from one sample to another in our training data. It should be noted that, in the real-world image denoising task, \( x^{(j)} \) is usually estimated by averaging several noisy images taken under the same camera conditions [65]. In the testing phase, given only one corrupted image, our goal is to first estimate the degradation information, and then recover the high-quality image based on such information.

Next, we formulate a rational Bayesian generative model for the tasks of denoising and super-resolution.

### B. Model Formulation for Denoising

Let \( \{ y, x \} \in D \) denote any noisy/noise-free image pair in the training dataset. For the noisy image \( y \), we assume that it is generated as follows:

\[
y_i \sim \mathcal{N}(y_i | z_i, \sigma_0^2), \quad i = 1, 2, \ldots, d,
\]\n
where \( \mathcal{N}(\cdot | \mu, \sigma^2) \) denotes the Gaussian distribution with mean \( \mu \) and variance \( \sigma^2 \), \( z \) denotes the latent clean image, \( d \) is the number of pixels in the noisy image. Notably, different from the commonly used i.i.d. Gaussian/Laplacian assumption, we model the image noise as pixel-wise non-i.i.d. Gaussian distribution in (2). Such an non-i.i.d. noise assumption largely increases the degrees of freedom of the noise distribution, and is thus expected to better fit complicated real noise as illustrated in Section V-C1.

Next, we introduce some prior knowledges for the latent clean image \( z \) and the noise variance map \( \sigma^2 \). In the real dataset, \( x \) provides an approximate estimation to the underlying clean image \( z \) by some statistical method [65]. Therefore, it can be embedded into the following prior distribution as a constraint for \( z \):

\[
z_i \sim \mathcal{N}(z_i | x_i, \varepsilon_0^2), \quad i = 1, 2, \ldots, d,
\]\n
where \( \varepsilon_0 \) is a hyper-parameter that reflects the closeness between \( z \) and \( x \). In some synthetic experiments where the underlying clean image is accessible, \( x \) is indeed the true clean image \( z \), which can be easily formulated by setting \( \varepsilon_0 \) as a small number close to 0. Under this setting, (3) degenerates to the Dirac distribution centered at \( x \).

As for the variance map \( \sigma^2 \), we construct the following inverse Gamma distribution as its conjugate prior:

\[
\sigma_i^2 \sim \text{IG} \left( \sigma_i^2 | \alpha_0 - 1, \alpha_0 \xi_i \right), \quad i = 1, 2, \ldots, d,
\]\n
where \( \xi_i \) is the prior and it provides an approximate estimation to the underlying clean Dirac distribution of \( \varepsilon \) and \( \sigma \). It is calculated using a Gaussian filter in the neighborhoods of \( z \) instead of \( p = 1 \) denotes the spatial correlation coefficients with size \( \rho \), and \( H \) is usually estimated by averaging several images taken under the same camera conditions [65]. In the testing phase, given only one corrupted image, our goal is to first estimate the degradation information, and then recover the high-quality image based on such information.

Next, we formulate a rational Bayesian generative model for the tasks of denoising and super-resolution.

### C. Extension to Blind Super-Resolution

For the problem of image super-resolution, the degradation model in (1) can be reformulated as

\[
y = (z * k) \downarrow + n,
\]\n
where \( k \) denotes the blur kernel, * is the convolution operator, and \( \downarrow \) is the downampler with scale factor \( s \). Based on this degradation model, we extend the noise assumption of (2) as follows

\[
y_i \sim \mathcal{N}(y_i | [z * k]_i, \sigma_i^2), \quad i = 1, 2, \ldots, d,
\]\n
where \( [z * k]_i \) represents the \( i \)-th pixel of \( z * k \).

To handle blind super-resolution, the most challenging part is how to model the blur kernel. Recently, lots of related literatures [57], [63], [67], [68], [69] have found that the anisotropic Gaussian kernels are sufficient to guarantee pleasing results for image super-resolution. In this study, we follow these related works and adopt the anisotropic Gaussian kernels. Thus, the blur kernel \( k \) with size \( (2r + 1) \times (2r + 1) \) can be defined as

\[
k_{ij} = g(\lambda_1^2, \lambda_2^2, \rho) = \frac{1}{2\pi\lambda_1\lambda_2\sqrt{1 - \rho^2}} \exp \left\{ -\frac{1}{2} \left[ \Sigma \right] \right\},
\]\n
where \( \Sigma = \begin{bmatrix} \lambda_1^2 & \lambda_1\lambda_2\rho & \lambda_2^2 \\ \lambda_1\lambda_2\rho & \lambda_1^2 & \lambda_2^2 \end{bmatrix} \) is the covariance matrix, \( \rho \) is the Pearson correlation coefficient, \( S = [i, j]^T \) denotes the spatial coordinate with \( i, j \in \{-r, \ldots, r\} \). By denoting \( \Lambda = \{ \rho, \lambda_1^2, \lambda_2^2 \} \), one can easily observe that the blur kernel \( k \) is completely determined by \( \Lambda \) when the kernel size is fixed. This inspires us to design a prior distribution for \( \Lambda \) instead of \( k \).

In essence, (8) represents the blur kernel \( k \) through two Gaussian distributions with variance parameters \( \lambda_1^2 \) and \( \lambda_2^2 \) along the horizontal and vertical directions, and their correlations are depicted by \( \rho \). For \( \{ \lambda_1, \lambda_2 \} \) and \( \rho \), we impose the inverse Gamma and Dirac distributions for them as prior constraints, respectively, i.e.,

\[
\Lambda \sim \text{p}(\Lambda) = \text{Dirac}(\rho|\tilde{\rho}) \prod_{l=1}^2 \text{IG}(\lambda_l^2|\kappa_0 - 1, \kappa_0 * \tilde{\lambda}_l^2),
\]\n
1For an inverse Gamma distribution \( \text{IG}(\alpha, \beta) \), we usually call \( \alpha \) and \( \beta \) as the shape parameter and the scale parameter respectively.
where \( \{ \hat{\rho}, \hat{\lambda}_1, \hat{\lambda}_2 \} \) reflect the corresponding true kernel information contained in the training data. Similar to the \( \alpha_0 \) of (4), \( \kappa_0 \) is also a hyper-parameter controlling the shape of the inverse Gamma distributions. For the purpose of easy optimization, we employ a Gaussian distribution with small variance to approximate the Dirac distribution. Thus, based on (8), the blur kernel \( k \) is modelled as

\[
k = g(A),
\]

(10)

\[
A \sim p(A) = \mathcal{N}(\rho|\hat{\rho}, \sigma_0^2) \prod_{l=1}^{2} \text{IG}(\lambda_l^2|\kappa_0 - 1, \kappa_0 * \hat{\lambda}_l^2),
\]

(11)

where the variance \( \sigma_0^2 \) is empirically set as 1e-4 throughout all our experiments.

Combining (3)–(5), (7), and (10)–(11) (or (2)–(5)) together, a full Bayesian model for blind image super-resolution (or image denoising) can be obtained. The goal then turns to infer the posterior of latent variables \( \{ z, \sigma^2, A \} \) (or \( \{ z, \sigma^2 \} \)) from \( y \), namely \( p(z, \sigma^2, A|y) \) (or \( p(z, \sigma^2|y) \)).

IV. STOCHASTIC VARIATIONAL INFERENCE

In this section, a stochastic variational inference algorithm is designed for the proposed generative model. In the following part, we take blind super-resolution problem as an example to present our algorithm, since it can be easily degenerated into the denoising task by setting the blur kernel as Dirac delta function and the scale factor as 1.

A. Form of Variational Posterior

Inspired by the VI techniques [70], we first construct a variational distribution \( q(z, \sigma^2, A|y) \) to approximate the true posterior \( p(z, \sigma^2, A|y) \) led by our generative model. The variational posteriori distribution is then conditionally factorized as

\[
q(z, \sigma^2, A|y) = q(z|y, \sigma^2, A)q(\sigma^2|y)q(A|y).
\]

(12)

Next, we begin to design specific forms for these three factorized posteriori distributions.

The conjugate prior of (4) inspires us to assume \( q(\sigma^2|y) \) as the following inverse Gamma distribution:

\[
q(\sigma^2|y) = \prod_{i} \text{IG}(\sigma_i^2|\alpha_0 - 1, \alpha_0 \beta_i(y; W_S)),
\]

(13)

where \( \beta_i(y; W_S) \) is a mapping function parameterized as a DNN named as sigma network (SNet) with parameters \( W_S \). It aims to predict the scale parameter of \( q(\sigma^2|y) \) directly from the corrupted image \( y \). As for the shape parameter of \( q(\sigma^2|y) \), we simply fix it as the same to the prior distribution, i.e., \( \alpha_0 - 1 \). Different from the strategy of setting them both as learnable parameters in our previous version [27], such a modification largely simplifies the algorithm more stable during training. Similarly, we formulate \( q(A|y) \) as

\[
q(A|y) = \mathcal{N}(\rho|m(y; W_K), \sigma_0^2) \prod_{l=1}^{2} \text{IG}(\lambda_l^2|\kappa_0 - 1, \kappa_0 \eta_l(y; W_K)),
\]

(14)

where \( m(y; W_K) \) and \( \eta_l(y; W_K) \) are jointly parameterized as a DNN with parameters \( W_K \), named as kernel network (KNet). It takes the low-resolution image \( y \) as input and outputs the posteriori parameters for \( q(A|y) \).

As for \( q(z|y, \sigma^2, A) \), we set it as Gaussian distribution

\[
q(z|y, \sigma^2, A) = \prod_{i} \mathcal{N}(z_i|\mu_i(y, \sigma^2, A; W_R), \varepsilon_i^2),
\]

(15)

where \( \mu_i(y, \sigma^2, A; W_R) \) represents the mapping function to evaluate the mean value of the posteriori Gaussian distribution of \( z \). Naturally, it is parameterized as a DNN with parameters \( W_R \), named as restoration network (RNet). For the ease of training, we set the variance parameter of this posteriori distribution as a constant, i.e., \( \varepsilon_i^2 \), being the same with that of the prior distribution in (3).

It is necessary to emphasize that the posteriori distribution \( q(z|y, \sigma^2, A) \) is conditioned on \( \sigma^2 \) and \( A \), which means that \( RNet \) depends on the noise variance map estimated by SNet and the kernel informations predicted by KNet. Generally speaking, the conditional assumption of (12) decomposes the task of blind super-resolution into two cascaded sub-tasks, namely degradation estimation implemented by SNet and KNet, and non-blind image restoration implemented by RNet. The whole inference procedure is shown in Fig. 1.

Remark: In (13), the mode of \( q(\sigma^2|y) \) is just equal to \( \beta(y; W_S) \), which is predicted by SNet. In other words, we leverage SNet to only estimate the core posteriori parameter, namely the mode, instead of the whole posteriori distribution. The reasons underlying this setting are two-fold: on one hand, this strategy inclines to alleviate the learning burden of SNet to some extent. On the other hand, we can directly utilize the output of SNet as an estimated variance map to solve some downstream problems that depends on pre-known noise levels. Similarly, we also employ this partially learning strategy in (14) and (15).

B. Evidence Lower Bound

In this part, we induce a rational objective function to jointly train the networks of SNet, KNet, and RNet. For the convenience of presentation, we simply denote \( \beta_i(y; W_S) \), \( m(y; W_K) \), \( \eta_i(y; W_K) \), and \( \mu_i(y, \sigma^2, A; W_R) \) as \( \beta_i \), \( m \), \( \eta_i \), and \( \mu_i \) respectively. Given any corrupted image \( y \), its logarithm marginal probability can be decomposed as

\[
\log p(y) = \mathcal{L}(z, \sigma, A; y) + KL[q(z, \sigma^2, A|y)||p(z, \sigma^2, A|y)],
\]

(16)

where

\[
\mathcal{L}(z, \sigma, A; y) = E_q[\log p(y|z, \sigma^2, A)p(z)p(\sigma^2)p(A) - \log q(z, \sigma^2, A|y)].
\]

(17)

Here \( E_q[\cdot] \) denotes the expectation w.r.t. the posteriori distribution \( q(z, \sigma^2, A|y) \). The second term of (16) represents the KL
Fig. 1. The inference framework of the proposed generative model for blind image super-resolution. It can be decomposed into two sub-tasks of degradation estimation and image restoration. Given any corrupted image $y$, we first infer the posteriori parameters $\beta$ of $q(\sigma^2|y)$ by SNet and $(m, \eta_1, \eta_2)$ of $q(A|y)$ by KNet in the phase of degradation estimation, and then recover the desirable high-quality image (i.e., the mean value $\mu$ of $q(z|y, \sigma^2, A)$) by RNet under the guidance of the estimated degradation information.

divergence between the variational posterior $q(z, \sigma^2, A|y)$ and the true posterior $p(z, \sigma^2, A|y)$. Due to the non-negativeness of KL divergence, $\mathcal{L}(z, \sigma, A; y)$ constitutes a lower bound of $\log p(y)$, generally called evidence lower bound (ELBO). Therefore, we can naturally approximate the true posterior $p(z, \sigma^2, A|y)$ with $q(z, \sigma^2, A|y)$ through maximizing the ELBO.

Combining the factorized assumption of (12), the ELBO can be rewritten as

$$
\mathcal{L}(z, \sigma, A; y) = E_{q(z, \sigma, A|y)} \left[ \log p(y|z, \sigma^2, A) \right] - E_{q(z, \sigma, A|y)} \left[ KL \left[ q(z|y, \sigma^2, A) || p(z) \right] \right] - KL \left[ q(\sigma^2|y) || p(\sigma^2) \right] - KL \left[ q(A|y) || p(A) \right],
$$
(18)

where $q(\sigma, A|y) = q(\sigma|y)q(A|y)$.

Next, we consider how to calculate each term in (18) step by step. The first term is intractable, mainly because the posterior $q(z, \sigma, A|y)$ is parameterized as complicated forms of DNNs. Fortunately, we can use the reparameterization trick [71] to obtain multiple differentiable samples from the posteriors, and then use them to estimate these two terms by Monte Carlo (MC) like VAE [71]. Concretely, the re-sampling process from $q(z|y, \sigma^2, A)$ can be easily implemented as

$$
\tilde{z} = \mu + \epsilon \sigma, \quad \sigma \sim \mathcal{N}(\mu, \sigma).
$$
(19)

To sample from $q(A|y)$ and $q(\sigma^2|y)$, we adopt the pathwise derivative technology [72] and denote the re-sampled data example as $A$ and $\sigma^2$. Based on $\tilde{z}$, $A$ and $\sigma^2$, the first term of (18) can be approximated as follows:

$$
E_{q(z, \sigma, A|y)} \left[ \log p(y|z, \sigma^2, A) \right] \approx -\frac{1}{2} \sum_{i=1}^{d} \left\{ \log \tilde{\sigma}_i + w_i \left( y_i - \left[ \tilde{z} + k \right]_i \right)^2 \right\},
$$
(20)

where $w_i = \frac{1}{\tilde{\sigma}_i^2}$, $k = g(\tilde{A})$, and $g(\cdot)$ is defined in (10). Note that we have omitted a constant that is independent of the learnable parameters in (20).

As for the last three terms of (18), they can all be analytically calculated as follows:

$$
E_{q(\sigma, A|y)} \left[ KL \left[ q(\sigma^2|y) || p(\sigma^2) \right] \right] = \sum_{i=1}^{d} \alpha_0 \left( \frac{\xi_i}{\beta_i} + \log \frac{\beta_i}{\xi_i} - 1 \right),
$$
(21)

$$
KL \left[ q(A|y) || p(A) \right] = \frac{(m - \rho)^2}{2r_0} + \sum_{l=1}^{2} \kappa_0 \left( \frac{\lambda_l}{\eta_l} + \log \frac{\eta_l}{\lambda_l} - 1 \right).
$$
(22)

Finally we can get the expected objective function, namely the negative ELBO on the entire training dataset, to optimize the network parameters of $W_S$, $W_K$, and $W_R$ as follows:

$$
\min_{W_S, W_K, W_R} \sum_{j=1}^{N} \mathcal{L}(z^{(j)}, \sigma^{(j)}, A^{(j)}; y^{(j)}),
$$
(24)

where $z^{(j)}$, $\sigma^{(j)}$, and $A^{(j)}$ denote the posteriori parameters for the $j$th image pair in training dataset $D$.

With the negative ELBO loss in (24), it is easy to train our model in an end-to-end manner like the DL-based methods. Actually, each term of the ELBO can be intuitively explained: the last three terms of KL divergence in (18) control the discrepancy between the variational posteriors and the priors, and the first term is the likelihood of the observed low-resolution images in the training dataset, which enforces the recovered high-resolution image can be mapped back to the low-resolution one through the estimated degradation model. During training, SNet, KNet, and RNet are refined and guided by each other under the supervision of this loss function.

**Remark:** Most of the current IR methods assume that each element of the data fidelity (i.e., the likelihood) term is with the same importance, i.e., $\sum_i w_i (y_i - \left[ \tilde{z} + k \right]_i)^2$. In this work, we novelty exploit an adaptive manner to re-weight the data fidelity in terms of $l_2$-norm, i.e., $\sum_i w_i (y_i - \left[ \tilde{z} + k \right]_i)^2$ in (20). Each pixel is re-weighted by $w_i = \frac{1}{\sigma_i^2}$, in which $\sigma_i^2$ is sampled from the noise distribution estimated by SNet. This re-weighting strategy...
based on noise variance is generally used in Bayesian statistics, like in [11], [13].

C. Network Structure and Learning

As shown in Fig. 1, SNet takes the corrupted image $y$ as input and outputs the scale-related parameter of $q(\sigma^2(y))$, achieving the goal of noise estimation. In practice, it consists of five convolutional layers, and each is followed with a Leaky ReLU activation except for the first and last layers. As for the KNet, it is designed to predict the posteriori distribution of the kernel parameter $\Lambda$ from the corrupted image $y$. In implementation, we first employ one convolutional layer and eight channel attention blocks (CAB) [23] to extract abundant feature maps, and then fuses them by one convolutional layer followed by an average pooling layer to obtain the posteriori parameters in $q(\Lambda|y)$.

The design of RNet, aiming to infer the conditional posteriori distribution of the desirable high-quality image, plays the most important role in blind IR. We adopt the commonly used ResUNet [54], [56] in low-level vision as our backbone. It replaces the plain convolutional layer in UNet [75] with residual block [76], and thus makes the gradient flow propagate much faster. Furthermore, in purpose of leveraging the estimated noise and kernel informations by SNet and KNet, we concatenate their outputs with the corrupted image $y$ together (see Fig. 1), and then feed them into RNet to recover the high-resolution image. We empirically find that such a simple concatenated operation performs very well and stably in our inference framework.

It should be noted that this work does not aim to design more effective network architectures to surpass current SotA methods, but mainly focus on devising a probabilistic framework based on the deep variational inference to deal with the blind IR task. Therefore, we simply select the commonly used networks in low-level vision as our backbones for SNet, KNet, and RNet, so as to better verify the generality of the proposed model.

V. EXPERIMENTAL RESULTS

In this section, we evaluate the effectiveness of our proposed method on two typical IR tasks, namely image denoising and image super-resolution. We denote our Variational Image Restoration Network as VIRNet for notation convenience in the following presentation.

To optimize the network, we adopted the Adam [77] algorithm with a mini-batch size of 16 and other default settings of PyTorch [78]. The initial learning rate was set as $10^{-4}$ and decayed gradually using the cosine annealing strategy [79]. For computational stability, the gradient clipping strategy was also used during training. In the task of image denoising, we cropped small image patches with a size of $128 \times 128$ for training. The hyper-parameter $\varepsilon_0^2$ of (3) was set to be $10^{-6}$, and the window size $\rho$ of (5) was set to be 7. In the task of image super-resolution, the patch size during training was fixed as 96, 144, and 192 for scale factor 2, 3, and 4, respectively. The hyper-parameter $\varepsilon_0^4$ was set to be $10^{-5}$, while the window size $\rho$ was set as a larger value than that in denoising, since image noise in super-resolution is usually assumed to be i.i.d. Gaussian. As for the shape parameter $\kappa_0$ of the kernel prior distribution in (11), we empirically set it as 50.

A. Image Denoising Experiments

1) Synthetic Non-I.I.D. Gaussian Noise Removal: To verify the effectiveness and robustness of VIRNet under non-i.i.d. noise configurations, we synthesized a large set of noisy/clean image pairs as training data. Similar to [20], a set of high quality source images was first collected as clean ones, including 432 images from BSD500 [80], 400 images from the ImageNet [81] validation set and 4744 images from Waterloo Database [82]. We then randomly generated non-i.i.d. Gaussian noise as

$$n = n^1 \odot M, \ n^1 \sim \mathcal{N}(n^1|0, I),$$

(25)

where $I$ is identity matrix, $M$ is a spatially variant map with the same size as the source image. Finally, the noisy image was obtained by adding the generated noise $n$ to the source image. As for the testing images, two commonly-used datasets, i.e., BSD68 [80] and McMaster [83], were adopted to evaluate the performance of different methods. Note that we totally generated four kinds of $M$s as shown in Fig. 2. The first one (Fig. 2(a)) was used for generating noisy images in training data, and the others (Fig. 2(b1)–(d1)) for three groups of testing data (denoted as Cases 1–3). Under these settings, the noise in training data and testing data are evidently different, which is suitable to verify the generalization capability of VIRNet.

Comparison with the SotAs: We compare VIRNet with several current denoising methods, including two typical model-based methods NLM [3] and CBM3D [4], two deep self-supervised methods S2S [73] and Ne2Ne [74], three supervised learning-based methods, namely DnCNN [17], FFDNet [20], and DRUNet [54]. The PSNR and SSIM results of all comparison methods on three groups of testing data are listed in Table I. We can easily see that: 1) the proposed VIRNet outperforms the other methods in all cases, indicating its superiority on handling these complicated non-i.i.d. noise types; 2) on the whole, DL-based methods (including the self-supervised methods) evidently surpass classical model-based methods NLM and CBM3D, owning to the powerful non-linear fitting capability of DNNs; 3) FFDNet and DRUNet are both non-blind methods that rely on the pre-given noise level as input. In contrast, VIRNet is designed toward blind IR, and thus able to simultaneously infer...
Table I

| Cases | Datasets | Metrics     | NLM [3] | CBM3D [4] | DnCNN [17] | FFDNet* [20] | S2S [73] | Ne2Ne [74] | DRUNet* [50] | VIRNet |
|-------|----------|-------------|---------|-----------|------------|--------------|----------|------------|--------------|--------|
| Case 1 | CBS68   | PSNR        | 24.06   | 26.73     | 28.74      | 28.79        | 28.23    | 27.92      | 29.05        | 29.28  |
|        | SSIM    |             | 0.6190  | 0.7660    | 0.8181     | 0.8181       | 0.7968   | 0.7948     | 0.8349       | 0.8353 |
| Case 2 | McMaster | PSNR        | 25.08   | 27.47     | 29.19      | 30.17        | 29.67    | 29.63      | 29.86        | 31.00  |
|        | SSIM    |             | 0.6910  | 0.7700    | 0.8218     | 0.8594       | 0.8374   | 0.8263     | 0.8545       | 0.8642 |
| Case 3 | CBS68   | PSNR        | 23.42   | 25.42     | 28.13      | 28.42        | 27.87    | 27.55      | 28.64        | 28.93  |
|        | SSIM    |             | 0.5882  | 0.7040    | 0.7989     | 0.8079       | 0.7859   | 0.6870     | 0.8251       | 0.8269 |
|        | McMaster| PSNR        | 23.26   | 25.82     | 28.84      | 29.74        | 29.43    | 27.23      | 30.38        | 30.38  |
|        | SSIM    |             | 0.6126  | 0.7120    | 0.7994     | 0.8315       | 0.8284   | 0.7314     | 0.8545       | 0.8572 |

The best and second best results are highlighted in bold and underline, respectively. Note that * denotes non-blind methods that rely on the pre-given noise-level.

Fig. 3 shows the visual results of different methods under testing cases 1–3 of Table I. Note that we only display the best five DL-based methods due to page limitation. It can be easily seen that the comparison methods are able to remove most of the noises, but also often generate over-smooth and blurry recovery, especially in the heavy-noise areas. This can be explained by the fact that they do not consider the spatial noise variations. To handle such non-i.i.d. noise, the proposed VIRNet elaborately considers the noise configurations and is thus capable of preserving more image details (e.g., edges, structures) than other methods.

Even though our VIRNet is designed and trained on the non-i.i.d. noise settings, it also performs well in additive white Gaussian noise (AWGN) removal tasks. Table II lists the average PSNR and SSIM results of different methods under three noise levels (i.e., $\sigma=15, 25, 50$) of AWGN. In this part, we further add the method RNAN [21] for a more thorough evaluation. It is noteworthy that RNAN is separately trained on some specific noise levels for AWGN, and hence we can only compared with it on the noise level 50. It is easy to see that VIRNet obtains the best (8 out of 12 cases) or at least second best (4 out of 12 cases) performance compared with these comparison methods. Combining the results in Tables I and II, it should be rational to say that the proposed VIRNet is more robust. Specifically, it is hopeful to handle a wider range of noise types, due to its more flexible noise modeling essence.

In Table III, we further list the comparison results on the number of model parameters and FLOPs with four DL-based methods. The FLOPs listed in this table are calculated on images with a size of $512 \times 512$. It should be noted that, for the sake of fair comparisons, the self-supervised methods S2S [73] and Ne2Ne [74] are not reported in Table III. It can be easily observed that VIRNet exhibits a better compromise over current SotA methods RNAN [21] and DRUNet [46] when taking both the model parameters and FLOPs into consideration. The proposed VIRNet is thus expected with better practical applicability in real scenarios.
2) Real-World Noise Removal: In this part, we evaluate the performance of VIRNet on two widely used real-world benchmark datasets, namely DND [84] and SIDD [65]. DND consists of 50 high-resolution images with realistic noise from 50 scenes taken by 4 consumer cameras, but it does not provide any other noisy/clean image pairs as training data. SIDD is another real-world denoising benchmark, containing about 30,000 real noisy images captured by 5 cameras under 10 scenes. Different from DND, each noisy image in SIDD comes with an almost noise-free counterpart as groundtruth, which is estimated by some statistical methods [65]. Further, SIDD also provides a small version dataset containing 320 image pairs, called SIDD-Medium, which is commonly-used as training data in recent works [23], [24], [27]. In order to compare with them fairly, we also train VIRNet only based on the SIDD-Medium dataset. As for the metrics, we adopt PSNR and SSIM calculated on the sRGB space to quantitatively evaluate different methods. We compared VIRNet with several typical real-world denoising methods, including MPRNet [25], CycleISP [88], DANet [24], SADNet [87], VDN [27] and so on (see Table IV). To the best of our knowledge, current SotA method on this two benchmarks is PNGAN [89]. This work, however, mainly focuses on simulating the camera pipeline to generate large amount of image pairs as training data so as to further improve the performance, instead of designing more effective denoising algorithm. Its denoiser architecture and the training strategy completely follow MPRNet. Therefore we mainly compare with MPRNet in this work. In order to comprehensively evaluate all competing methods, Table IV lists the denoising performance, as well as the model profiles, including the number of model parameters, FLOPs, and running time of the denoisers.

The FLOPs and running time are both counted on images with a size of 512 × 512. From the perspective of denoising performance, the proposed VIRNet achieves a slight performance improvements compared with current SotA method MPRNet, indicating its effectiveness. However, VIRNet is with pronounced superieties in terms of model profiles, especially in the comparisons of FLOPs and running time, which more faithfully reflect the relative efficiency of our method. To intuitively compare the denoising results, we visualize two typical real examples in Fig. 4, which are consistent with the quantitative results in Table IV.

B. Image Super-Resolution Experiments

In this section, we apply our proposed VIRNet to blind image super-resolution. Following [69], the DF2K dataset (containing 800 images from DIV2K [92] and 2650 images from Flickr2K [93]) was employed as our training data. When synthesizing the LR images, we followed the settings of current blind SR literatures [57], [69], i.e.,

\[ y = (z \otimes k) j_x^\dagger + n, \]

(26)

where \( y \) and \( z \) denote the low-resolution and high-resolution image respectively, \( \otimes \) is the 2-D convolution, \( j_x^\dagger \) is the direct

\footnote{https://noise.visinf.tu-darmstadt.de}

\footnote{https://www.eecs.yorku.ca/~kamel/sidd/benchmark.php}

\footnote{Extracting the upper-left pixel for each \( p \times p \) patch.}

2) Extracting the upper-left pixel for each \( p \times p \) patch.
and SIDD and the non-blind method
is the \( \sigma \) was generated as

\[
\mathbf{U} = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix}
l_1^2 & 0 \\
0 & l_2^2
\end{bmatrix}, \quad \Sigma = \mathbf{U} \mathbf{A} \mathbf{U}^T. \quad (27)
\]

To be specific, \( l_1, l_2 \), and \( \theta \) are randomly sampled from \([0, s]\), \([0, s]\), and \([0, \pi]\) respectively. For the noise level \( \sigma \), we set its range to be \([0,15]\).

1) Results on Synthetic Data: To be capable of quantitatively evaluating different methods, we first conducted some synthetic experiments on three commonly used datasets, including Set14 [91], CBSD68 [80], and DIV2K100 (the validation set of DIV2K [92]). For the purpose of making a thorough comparison on various degradations, we considered seven representative and diverse kernels as shown in Fig. 5, including three isotropic Gaussian kernels with different kernel widths (i.e., 0.4s, 0.6s, and 0.8s) and four anisotropic Gaussian kernels, where \( s \) is the scale factor. In addition, three noise levels (i.e., 0, 2.55, and 7.65) are considered following [56]. As for the metrics, aside from the commonly used PSNR and SSIM [85], we also adopted LPIPS [95] to measure the perceptual similarity. Note that PSNR and SSIM are calculated on the Y channel of the YCbCr space, while LPIPS is calculated in the sRGB space.

We consider three categories of comparison methods: 1) classical Bicubic interpolation method; 2) five blind super-resolution methods, including HAN [90], IKC [44], DAN [57], DASR [69], and BSRNet [46]; 3) two non-blind methods, i.e., SRMD [68] and USRNet [56], which rely on the pre-given blur kernel and noise level as input. For these non-blind methods, we provided the groundtruth blur kernel and noise level for them, and denoted their results with the format of “GT+X” (e.g., GT+SRMD).

In addition, for the methods of HAN, IKC and DAN, we first denoised the noisy low-resolution image using DnCNN [17] and then super-resolved it in the cases of \( \sigma = \{2.55, 7.65\} \), since these methods do not consider image noise during training.

Table V lists the comparison results of different methods under a scale factor of 4, and more results under scale factors of 2 and 3 are put into the supplementary material. From Table V, it can be seen that the proposed VIRNet achieves the best performance among blind methods in all cases. Especially, compared with the non-blind methods, VIRNet is still able to obtain a slightly better or at least comparable results even though they make use of the groundtruth information of blur kernel and noise level. This indicates the effectiveness of the proposed blind framework which is capable of handling the tasks of degradation estimation and image restoration simultaneously. Further, taking the model profiles into consideration, the superiorities of VIRNet is more evident. Specifically, VIRNet has fewer number of parameters, fewer FLOPs, and faster speed than both the SotA blind method DASR [69] and the non-blind method USRNet [56].

In Fig. 6, we display the denoising results on three typical visual examples of Set14 with a scale factor of 4. Note that we only show the results of blind super-resolution methods for a fair comparison. It can be easily seen that the proposed VIRNet is able to recover more realistic and sharper results, which are evidently closer to the groundtruth high-resolution images than other methods. The results of most comparison methods are relatively blurry and lose some image details. In the second example (the middle row), IKC and DAN lead to a relatively severe corruption on the original image color. That’s possibly caused by the inconsistency of their multiple iterations, since they both adopt the coarse-to-fine manner to gradually adjust the results. Due to the careful considerations on the degradation model, DASR and BSRNet also perform well compared with other methods. However, VIRNet still evidently surpasses them in terms of the quantitative and qualitative results. This further substantiates the effectiveness of the proposed variational framework.

2) Results on Real Data: In this part, we further justify the effectiveness of the proposed VIRNet on the real-world dataset RealSRSet [46]. It contains 20 real images that are commonly

Fig. 4. Denoising results of all competing methods on two typical real-world examples from DND [84] (upper) and SIDD [65] (lower) datasets.

Fig. 5. Seven Gaussian kernels used to generate the LR images in the synthetic super-resolution experiments.
TABLE V
QUANTITATIVE COMPARISONS OF DIFFERENT METHODS UNDER A SCALE FACTOR OF 4

| Methods          | Noise Level | Set14 PSNR | Set14 SSIM | Set14 LPIPS | BSDS68 PSNR | BSDS68 SSIM | BSDS68 LPIPS | DIV2K PSNR | DIV2K SSIM | DIV2K LPIPS | # Param | FLOPs/Times |
|------------------|-------------|------------|------------|-------------|-------------|-------------|-------------|------------|------------|------------|---------|-------------|
| Bicubic          | 0.35        | 24.34      | 0.6352     | 0.5257      | 24.68       | 0.6144      | 0.6044      | 25.50      | 0.6762     | 0.5349     | -       | 0.006       |
| HAN [90]         | 0.35        | 25.36      | 0.6731     | 0.4693      | 25.35       | 0.6494      | 0.5477      | 26.30      | 0.7115     | 0.4775     | 16.07   | 4653        | 4.933    |
| IKC [44]         | 0.35        | 27.24      | 0.7388     | 0.3534      | 26.71       | 0.7105      | 0.4253      | 28.00      | 0.7741     | 0.3541     | 9.05    | 11537       | 7.390    |
| DASR [70]        | 0.35        | 27.49      | 0.7465     | 0.3442      | 26.96       | 0.7204      | 0.4053      | 28.31      | 0.7852     | 0.3354     | 4.33    | 5013        | 2.012    |
| BSRNet [46]      | 0.35        | 27.74      | 0.7512     | 0.3314      | 27.12       | 0.7231      | 0.4026      | 28.32      | 0.7805     | 0.3380     | 7.25    | 839         | 0.525    |
| VIRNet (Ours)    | 0.35        | 26.84      | 0.7192     | 0.3819      | 26.38       | 0.6860      | 0.4540      | 27.49      | 0.7503     | 0.3829     | 16.70   | 4706        | 1.577    |
| GT-SRMD [69]     | 0.35        | 27.89      | 0.7573     | 0.3163      | 27.27       | 0.7330      | 0.3668      | 28.60      | 0.7919     | 0.3165     | 5.72    | 370         | 0.161    |
| GT+USRNet [56]   | 0.35        | 27.83      | 0.7587     | 0.3180      | 27.12       | 0.7263      | 0.3962      | 28.44      | 0.7874     | 0.3250     | 1.55    | 4709        | 0.106    |
| Bicubic          | 0.35        | 27.70      | 0.7747     | 0.3181      | 26.88       | 0.7408      | 0.3998      | 28.64      | 0.8086     | 0.3042     | 17.20   | 38894       | 9.214    |
| DnCNN [17]+HAN [90] | 0.35    | 29.24      | 0.6976     | 0.5213      | 29.23       | 0.6350      | 0.6024      | 26.11      | 0.6925     | 0.5638     | 16.07   | 4653        | 4.933    |
| DnCNN [17]+IKC [44] | 0.35     | 29.40      | 0.6795     | 0.6590      | 25.66       | 0.6546      | 0.5297      | 26.59      | 0.7147     | 0.4702     | 9.05    | 11537       | 7.392    |
| DnCNN [17]+DAN [57] | 0.35   | 29.35      | 0.6685     | 0.4873      | 25.46       | 0.6475      | 0.5524      | 26.32      | 0.7082     | 0.4964     | 4.33    | 5013        | 2.012    |
| DASR [70]        | 0.35        | 27.05      | 0.7197     | 0.3672      | 26.50       | 0.6881      | 0.4479      | 27.58      | 0.7491     | 0.3830     | 7.25    | 839         | 0.525    |
| BSRNet [46]      | 0.35        | 26.70      | 0.7093     | 0.3081      | 26.23       | 0.6797      | 0.4542      | 27.31      | 0.7431     | 0.3588     | 16.70   | 4706        | 1.577    |
| VIRNet (Ours)    | 0.35        | 27.18      | 0.7255     | 0.3524      | 26.63       | 0.6975      | 0.4305      | 27.86      | 0.7597     | 0.3588     | 5.72    | 370         | 0.161    |
| GT-SRMD [69]     | 0.35        | 27.15      | 0.7210     | 0.3049      | 26.59       | 0.6954      | 0.4367      | 27.76      | 0.7578     | 0.3467     | 1.55    | 4709        | 0.106    |
| GT+USRNet [56]   | 0.35        | 27.62      | 0.7461     | 0.3295      | 26.52       | 0.7125      | 0.4249      | 28.27      | 0.7829     | 0.3266     | 17.20   | 38894       | 9.214    |

The PSNR/SSIM/LPIPS values in this table are all averaged over the seven kernels as shown in Fig. 5. The results of the non-blind methods that rely on the pre-processed ground-truth blur kernel and noise level are marked in gray color to denote unfair comparisons. Besides, some commonly considered model profiles, namely, the number of learnable parameters (in M), the FLOPs (in G), and the running time (in seconds), are also listed for more comprehensive comparison. Note that the FLOPs and running time are calculated in the case of super-resolving the low-resolution images with a size of 256×256.

Fig. 6. Visualized super-resolution results of different methods on a synthetic example of Set14 [91] under a scale factor of 4. Specifically, the noise level is 2.55, and the blur kernel is shown on the upper-right corner of the LR image.

used in previous literatures [20], [97], [98], [99] or downloaded from internet. Since the underlying high-resolution images for them are not available, we thus mainly evaluate different methods by visual comparisons. Fig. 7 displays three typical super-resolution examples with a scale factor of 4. In the first (top row) and second (middle row) examples, the LR images both contain some image noises, which makes the super-resolution goal more challenging. The methods of Bicubic, HAN, IKC, and DAN all fail to deal with such cases, and produce some artifacts in the areas with image noises. As for DASR and BSRNet, they unfortunately erase the high frequency image details when removing the image noises. One can easily observe that the proposed VIRNet makes a good trade-off between preserving the image details and removing the image noises. In the third (bottom row) example, the results of IKC and VIRNet are more natural and realistic than others that are all blurry to different extent. These results verify the stable and consistently well performance of VIRNet in the real-world super-resolution task.

In Table VI, we adopt two non-reference metrics (i.e., NRQM [100] and PI [101]) to further quantitatively evaluate different methods. It can be seen that the proposed VIRNet achieves the second best results in terms of both metrics, only slightly worse than IKC, which indicates the effectiveness of our method. Combining its better visual performance as shown...
\textbf{C. Degradation Estimation Experiments}

In this subsection, we empirically verify the effectiveness of our method in the task of degradation estimation, including noise estimation and kernel estimation.

1) \textit{Noise Estimation:} Different from most of the current IR methods, the pixel-wise non-i.i.d. Gaussian assumption is adopted to fit the noise distribution in our method. Next, we analyse the performance of our method with such an assumption under several common noise types in IR tasks:

\textit{I.I.D. Gaussian Noise:} Even though VIRNet is designed on the basis of non-i.i.d. Gaussian noise assumption, it can be generalized well to the i.i.d. Gaussian noise as shown in Table II. To further quantitatively illustrate this point, we take the estimated noise level by VIRNet as the input of FFDNNet, which is a typical non-blind i.i.d. Gaussian denoising method that relies on the pre-known noise level. Table VII lists the PSNR comparison results of FFDNNet taking different noise level settings, in which FFDNNet\textsubscript{VIR} and FFDNNet\textsubscript{GT} denote the results of FFDNNet taking the predicted noise level by VIRNet and the groundtruth noise level as input respectively. We can see that FFDNNet\textsubscript{VIR} is able to achieve the same performance with FFDNNet\textsubscript{GT} when \( \sigma = \{15, 25\} \), or very close performance.
TABLE VIII

| Methods       | Noise Cases | Case 1 | Case 2 | Case 3 |
|---------------|-------------|--------|--------|--------|
| FFDNetGT      | 28.79       | 28.42  | 28.68  |
| FFDNetVR      | 28.75       | 28.39  | 28.63  |

Fig. 9. One typical visual example in SIDD [65] dataset. From left to right: (a) noisy image \(y\); (b) noise-free image \(x\); (c) image noise calculated through \(|y - x|\); (d) variance map predicted by the proposed method.

when \(\sigma = 50\), even though FFDNetGT makes use of the true noise level. This indicates that VIRNet is capable of properly estimating the noise levels of the i.i.d. Gaussian noise.

Non-i.i.d. Gaussian Noise: In Section V-A1, we adopt three groups of noise variance maps (see Fig. 2(b1)–(d1)) to synthesize the testing data, so as to evaluate the performance of VIRNet under the non-i.i.d. Gaussian noise. Correspondingly, Fig. 2(b2)–(d2) further displays the variance maps predicted by VIRNet for easy visualization. It can be seen that these predicted variance maps have very similar spatial variation with the groundtruth ones, which are expected to facilitate the subsequent denoising task or other non-blind denoising methods. To justify this point, we also present these predicted variance maps in FFDNet [20] to test its performance under the non-i.i.d. Gaussian noise, and the quantitative comparisons are listed in Table VIII. One can see that FFDNetVR and FFDNetGT have very similar performance, and the performance difference between them is less than 0.04dB PSNR. This indicates that VIRNet is able to effectively handle such complicated noise distribution.

Signal-dependent Noise: The challenge in real-world image denoising is mainly attributed to the signal-dependentness of the image noise. Fig. 9 shows one typical real-world noisy example coming from SIDD [65] dataset and the corresponding variance maps predicted by VIRNet. Note that the variance map has been enlarged several times for easy visualization. It is easy to see that the estimated noise variance map depicts strong relevance to the pixel illumination, implying that the proposed VIRNet is able to finely approximate the signal-dependent real noise.

2) Kernel Estimation: As is well known, kernel estimation plays an important role in blind image super-resolution [103]. To evaluate the effectiveness of VIRNet in this subtask, we compare three recent kernel estimation methods specifically designed for super-resolution, including KernelGan [96], DIPFKP [63], and BSRDM [64]. Since these three methods are all relatively time-consuming, we randomly select 20 images from the validation set of DIV2K [92] (denoted as DIV2K20) as testing data. The LR images are synthesized using the last four anisotropic Gaussian kernels in Fig. 5 under a scale factor of 4, and the noise level is set as 2.55.

As for the evaluation, we use two ways to compare the performance of different methods. First, the mean square error (MSE) between the estimated kernel and the groundtruth kernel is an intuitive metric that directly reflects the accuracy of the estimate kernel. The detailed comparison results are listed in Table IX. Second, we apply the estimated blur kernels in a non-blind super-resolution method USRNet [56] and then compare the recovered HR image in terms of PSNR, SSIM [85], and LPIPS [95]. The comparison results on these three metrics are listed in Table X. From both tables, one can easily observe that the proposed VIRNet exhibits evident superiority over other competing methods.

D. Analysis on Posteriori Factorization

When designing the variational inference algorithm in Section IV, we factorize the variational distribution \(q(z, \sigma^2, \Lambda | y)\) into a conditional format of (12), which fundamentally induces the cascaded inference framework in Fig. 1. In fact, different factorized assumptions on \(q(z, \sigma^2, \Lambda | y)\) will lead to different designs on the inference framework. For example, the following unconditional factorization

\[
q(z, \sigma^2, \Lambda | y) = q(z | y)q(\sigma^2 | y)q(\Lambda | y),
\]

will induce a parallel inference architecture. Specifically, the three sub-networks, namely \(SNet, KNet,\) and \(RNet,\) will feedforward independently in such a parallel framework, but they are able to interact during back propagation through ELBO. Please refer to our previous conference version [27] for a thorough overview on this point.

To validate the superiority of the conditional form of (12), we consider different posteriori factorizations and empirically compare their performance on the task of image super-resolution, since it involves more general degradations than image denoising. Table XI lists the average comparison results on DIV2K100 with a scale factor of 4. As compared with Baseline1, it can be seen that VIRNet achieves evident performance gain, which indicates that the degradation information (i.e., the noise level and the blue kernel) can facilitate the image restoration task. In
fact, such a conditional factorization in VIRNet is consistent with the classical model-based methods that decompose the blind IR in two subproblems, namely degradation estimation and image restoration. The superiority of Baseline3 over Baseline2 demonstrates that the kernel information can bring up more marginal performance improvement than the noise level, complying with the conclusion in [103]. However, the performance gain of VIRNet over Baseline3 on LPIPS substantiates that conditioning on the noise level can further improve the perceptual quality of the recovered images.

VI. CONCLUSION

In this paper, we have proposed a novel deep variational network for blind IR, which aims to finely integrate the merits of both classical model-based methods and recent DL-based methods. On one hand, we have constructed a Bayesian generative model for blind image denoising and image super-resolution, by carefully considering the image degradation from the perspectives of image noise and blur kernel. On the other hand, a variational inference algorithm has been elaborately designed to solve the proposed model, in which the posterior distribution are all parameterized by DNNs to increase the non-linear fitting capability. Most notably, this variational algorithm induces a unified framework to simultaneously deal with the tasks of degradation estimation and image restoration. Extensive experiments have also been conducted to demonstrate the superiority of our method on image denoising and super-resolution. In the future, we will make further effort to extend our method to deal with more complicated and general image degradations.

REFERENCES

[1] Y. Tsin, V. Ramesh, and T. Kanade, “Statistical calibration of CCD imaging process,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2001, pp. 480–487.
[2] 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.
[3] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2005, pp. 60–65.
[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3-D transform-domain collaborative filtering,” IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
[5] J. Mairal, M. Elad, and G. Sapiro, “Sparse representation for color image restoration,” IEEE Trans. Image Process., vol. 17, no. 1, pp. 53–69, Jan. 2008.
[6] W. Dong, L. Zhang, G. Shi, and X. Li, “Nonlocally centralized sparse representation for image restoration,” IEEE Trans. Image Process., vol. 22, no. 4, pp. 1620–1630, Apr. 2013.
[7] J. Xu, L. Zhang, and D. Zhang, “A trilateral weighted sparse coding scheme for real-world image denoising,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 20–36.
[8] W. Dong, G. Shi, and X. Li, “Nonlocal image restoration with bilateral variance estimation: A low-rank approach,” IEEE Trans. Image Process., vol. 22, no. 2, pp. 700–711, Feb. 2013.
[9] S. Gu, L. Zhang, W. Zuo, and X. Feng, “Weighted nuclear norm minimization with application to image denoising,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2014, pp. 2862–2869.
[10] J. Xu, L. Zhang, D. Zhang, and X. Feng, “Multi-channel weighted nuclear norm minimization for real color image denoising,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2017, pp. 1096–1104.
[11] D. Meng and F. De La Torre, “Robust matrix factorization with unknown noise,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2013, pp. 1337–1344.
[12] X. Cao et al., “Low-rank matrix factorization under general mixture noise distributions,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2015, pp. 1493–1501.
[13] F. Zhu, G. Chen, J. Hao, and P.-A. Heng, “Blind image denoising via dependent dirichlet process tree,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 8, pp. 1518–1531, Aug. 2017.
[14] Z. Yue, D. Meng, Y. Sun, and Q. Zhao, “Hyperspectral image restoration under complex multi-band noises,” Remote Sens., vol. 10, no. 10, pp. 1631–1648, 2018.
[15] Z. Yue, H. Yong, D. Meng, Q. Zhao, Y. Leung, and L. Zhang, “Robust multiview subspace learning with nonindependently and nonidentically distributed complex noise,” IEEE Trans. Neural Netw. Learn. Syst., vol. 31, no. 4, pp. 1070–1083, Apr. 2020.
[16] C. Dong, C. C. Loy, K. He, and X. Tang, “Learning a deep convolutional network for image super-resolution,” in Proc. Eur. Conf. Comput. Vis., Springer, 2014, pp. 184–199.
[17] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising,” IEEE Trans. Image Process., vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
[18] X. Mao, C. Shen, and X.-Y. Yang, “Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections,” in Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 2802–2810.
[19] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, “Image super-resolution using very deep residual channel attention networks,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 294–310.
[20] K. Zhang, W. Zuo, and L. Zhang, “FFDNet: Toward a fast and flexible solution for CNN-based image denoising,” IEEE Trans. Image Process., vol. 27, no. 9, pp. 4608–4622, Sep. 2018.
[21] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, “Residual non-local attention networks for image restoration,” in Proc. Int. Conf. Learn. Representations, 2019.
[22] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, “Toward convolutional blind denoising of real photographs,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 1712–1722.
[23] S. Anwar and N. Barnes, “Real image denoising with feature attention,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 3155–3164.
[24] Z. Yue, Q. Zhao, L. Zhang, and D. Meng, “Dual adversarial network: Toward real-world noise removal and noise generation,” in Proc. Eur. Conf. Comput. Vis., Springer, 2020, pp. 41–58.
[25] S. W. Zamir et al., “Multi-stage progressive image restoration,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 14821–14831.
[26] 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 Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 4096–4105.
[27] Z. Yue, H. Yong, Q. Zhao, D. Meng, and L. Zhang, “Variational denoising network: Toward blind noise modeling and removal,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 1688–1699.
[28] E. P. Simoncelli and E. H. Adelson, “Noise removal via Bayesian wavelet coding,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 1996, pp. 379–382.
[39] Y. Tai, J. Yang, X. Liu, and C. Xu, “MemNet: A persistent memory network for image super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2016, pp. 4801–4809.

[42] P. Qiao, Y. Dou, W. Feng, R. Li, and Y. Chen, “Learning non-local image diffusion for image denoising,” in Proc. ACM Int. Conf. Multimedia, 2017, pp. 1847–1855.

[43] F. Zhu, G. Chen, and P.-A. Heng, “From noise modeling to blind image denoising,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2016, pp. 420–429.

[45] Y.-S. Xu, S.-Y. R. Tseng, Y. Tseng, H.-K. Kuo, and Y.-M. Tsai, “Unified deep neural network prior with application to blind super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 3341–3350.

[46] D. Ren, K. Zhang, Q. Wang, Q. Hu, and W. Zuo, “Neural blind deconvolution using deep priors,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 11026–11035.

[47] J. Liang, K. Zhang, S. Gu, L. Van Gool, and R. Timofte, “Flow-based kernel prior with application to blind super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 10601–10610.

[48] 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 Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 2118–2128.

[49] A. Abdelhamed, S. Lin, and M. S. Brown, “A high-quality denoising dataset for smartphone cameras,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1692–1700.

[50] H. Yong, D. Meng, W. Zuo, and L. Zhang, “Robust online matrix factorization for dynamic background subtraction,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 7, pp. 1726–1740, Jul. 2018.

[51] G. Riegler, S. Schulte, M. Ruther, and H. Bischof, “Conditioned regression models for non-blind single image super-resolution,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2015, pp. 522–530.

[52] K. Zhang, W. Zuo, and L. Zhang, “Learning a single convolutional super-resolution network for multiple degradations,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2018, pp. 3262–3271.

[53] L. Wang et al., “Unsupervised degradation representation learning for blind super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 10581–10590.

[54] C. M. Bishop, Pattern Recognition and Machine Learning. Berlin, Germany: Springer, 2006.

[55] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in Proc. Int. Conf. Learn. Representations, 2014.

[56] M. Jankowiak and F. Obermeyer, “Pathwise derivatives beyond the plug-and-play framework,” in Proc. Int. Conf. Med. Image Anal. Mach. Intell., 2020, pp. 1890–1898.

[57] T. Huang, S. Li, X. Jia, H. Lu, and J. Liu, “Neighbor2Neighbor: Self-supervised denoising from single noisy images,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 14781–14790.

[58] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention, Springer, 2015, pp. 234–241.

[59] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[60] D. P. Kingma and J. Li. Ba, “Adam: A method for stochastic optimization,” in Proc. Int. Conf. Learn. Representations, 2015.

[61] A. Paszke et al., “PyTorch: An imperative style, high-performance deep learning library,” in Proc. Adv. Neural Inf. Process. Syst., 2019, Art. no. 721.

[62] I. Loshchilov and F. Hutter, “SGDR: Stochastic gradient descent with warm restarts,” in Proc. Int. Conf. Learn. Representations, 2017.
