Learning Deep Non-Blind Image Deconvolution Without Ground Truths

Yuhui Quan[1,2⋆], Zhuojie Chen[1⋆⋆], Huan Zheng[2], and Hui Ji[2] ⋆⋆⋆

[1] School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China
[2] Department of Mathematics, National University of Singapore, 119076, Singapore
csyhquan@scut.edu.cn, zhuojie.chen.cs@foxmail.com,
huan_zheng@u.nus.edu, matjh@nus.edu.sg

Abstract. Non-blind image deconvolution (NBID) is about restoring a latent sharp image from a blurred one, given an associated blur kernel. Most existing deep neural networks for NBID are trained over many ground truth (GT) images, which limits their applicability in practical applications such as microscopic imaging and medical imaging. This paper proposes an unsupervised deep learning approach for NBID which avoids accessing GT images. The challenge raised from the absence of GT images is tackled by a self-supervised reconstruction loss that approximates its supervised counterpart well. The possible errors of blur kernels are addressed by a self-supervised prediction loss based on intermediate samples as well as an ensemble inference scheme based on kernel perturbation. The experiments show that the proposed approach provides very competitive performance to existing supervised learning-based methods, no matter under accurate kernels or erroneous kernels.

Keywords: Non-blind Image Deconvolution, Self-Supervised Learning, Unsupervised Deep Learning, Image Deblurring

1 Introduction

Image deconvolution is a challenging problem often encountered in imaging systems and low-level vision. It is about estimating the latent image $\boldsymbol{X}$ from its degraded observation $\boldsymbol{Y}$ generated by

$$\boldsymbol{Y} = \boldsymbol{K}^* \otimes \boldsymbol{X} + \boldsymbol{N}, \tag{1}$$

⋆ Yuhui Quan is also with Pazhou Lab, Guangzhou 510335, China. He would like to thank the support in part by National Natural Science Foundation of China under Grant 61872151 and in part by Natural Science Foundation of Guangdong Province under Grant 2022A1515011755.
⋆⋆ Corresponding author: Zhuojie Chen
⋆⋆⋆ Hui Ji would like to thank the support in part by Singapore MOE AcRF under Grant R-146-000-315-114.

where $\otimes$ denotes the convolution operation, $\boldsymbol{K}^*$ denotes a blur kernel, and $\boldsymbol{N}$ denotes image noise. When the kernel is given as a prior, the problem is called *Non-Blind Image Deconvolution* (NBID); otherwise, it is called blind image deconvolution (BID). The kernel often can be obtained by calibrating an imaging system and capturing the image of Dirac-like dots, or be captured using specific hardware. NBID also serves as a key module for BID, which is often called in the last stage of a BID method after kernel estimation is finished. In this case, the estimated kernel used for NBID usually contains non-ignorable errors.

An image convolved by a blur kernel will have its high-frequency parts significantly attenuated or erased. Together with the unknown image noise and possible kernel errors, it makes NBID a challenging ill-posed inverse problem. In recent years, there have been extensive studies applying deep learning (DL) for NBID, which train a neural network (NN) to predict latent images from the pairs of degraded images and blur kernels; see *e.g.* [21, 50, 51, 19, 3, 38, 12, 29, 14, 35, 10, 11]. While these methods differ in terms of architecture and training scheme, they are all based on supervised learning. That is, their NNs are trained over many pairs of blurred and ground truth (GT) images. Such a prerequisite on GT images limits their applications in certain domains, such as scientific imaging and medical imaging. In these domains, either GT images are very challenging to collect, or have restricted usage for privacy concerns. This issue cannot be effectively addressed by calling GT images from other domains, as it may result in poor generalization performance due to domain shifts and domain gaps.

1.1 Problem Setting and Main Idea

Motivated by the limitation of supervised DL-based NBID in the domains where the access to GT images is very limited, this paper proposes an unsupervised DL approach for NBID whose NN training does not involve any GT image. The unsupervised learning setting we study is as follows.

- **Training data:** Only a set of blurred image $\{\boldsymbol{Y}_j\}_j$ and the associated kernels $\{\boldsymbol{K}_j\}_j$ are provided for training; and the GT images $\{\boldsymbol{X}_j\}_j$ are unavailable.
- **Error sources:** In the training data, there is noise $\boldsymbol{N}_j$ on each image $\boldsymbol{Y}_j$ and possible kernel error defined by $\boldsymbol{\Delta K}_j = \boldsymbol{K}_j - \boldsymbol{K}_j^*$ with GT kernel $\boldsymbol{K}_j^*$.

It is a challenging task to teach an NN to make accurate predictions in such a setting, as there is no GT image to define a loss that can measure the prediction accuracy. In addition, the possible kernel errors may confuse the learning process and lower the accuracy of the learned NBID process.

In the proposed approach, the challenge raised by the absence of GT images is tackled by generalizing the Recorrupted-to-Recorrupted (R2R) training [33] from denoising to NBID. It leads to a noise-resistant self-supervised reconstruction loss which approximates well a supervised loss defined over GT images in the range space induced by the blur kernel. Furthermore, by training with both kernel diversity and cross-image patch recurrence, the loss can also well approximate its supervised counterpart on the full image space.

In comparison to image noise, kernel errors are more challenging to handle. Without GT kernels for training, an NN is hardly aware of the existence of kernel errors, let alone learn to handle it. Our idea to handle kernel errors is viewing a deblurred image output by the NN as a "pseudo" GT and reblurring it with a different kernel to have a paired sample for "pseudo" supervised training. Such data augmentation is effective as long as the original kernel and the reblurring kernel are sufficiently different. In this case, the kernel for reblurring, even erroneous, is viewed as the GT kernel, and we apply a kernel error simulator to have its erroneous version. The rationale comes from that an imperfect blur kernel estimated by some existing method is also a physically valid blur kernel, or at least is very likely to approximate a valid blur kernel well. Based on the above data augmentation scheme, a self-supervised prediction loss is defined for enhancing to the kernel error robustness of the trained NN model.

In addition, an ensemble inference scheme is proposed to further improve the robustness to kernel errors. The scheme is based on the observation that, when restoring the latent image using many instances of the inaccurate kernel perturbed by random noise, the artifacts shown in the corresponding results have certain degrees of statistical independence. Then, the aggregation of these results is likely to attenuate the artifacts.

The proposed approach is applied with a popular off-the-shelf optimization-unrolling-based NN architecture composed of common building blocks. In the experiments on motion deblurring of natural images and microscopic deconvolution, we found that such an NN suffices to yield satisfactory performance, which competes against that of supervised learning-based NBID approaches.

1.2 Main Contributions

This work is one of the few attempts on unsupervised DL for NBID. Our main contributions lie in the design of loss functions and training/test schemes for NBID. See below for the summary of our main contributions.

- A self-supervised reconstruction loss function with good approximation to its supervised counterpart when being applied with kernel diversity and cross-image patch recurrence presented in training data;
- A self-supervised prediction loss function to improve the robustness of the trained model to kernel errors.
- An ensemble inference scheme based on kernel perturbations, which reduces the sensitivity of the NN to kernel errors in the test stage and works for both unsupervised models and supervised models.
- An unsupervised DL approach for NBID, which not only outperforms existing unsupervised methods, but also competes against the supervised ones.

2 Related Works

Non-DL-based NBID Conventional methods for NBID developed handcrafted image priors based on empirical image statistics to regularize the deconvolution process. There is abundant literature on it; see *e.g.* [20, 13, 36].

Supervised DL-based NBID Most supervised DL-based methods focus on NN architecture design. To have an NN working for different kernels, many recent methods unfold an iterative scheme of some regularization-based NBID approach; see *e.g.* [21, 50, 26, 51, 12, 19, 3, 29, 14, 11]. These methods usually decompose each iteration into an inversion process using the image estimate from the previous iteration and a denoising process performed by a convolutional NN (CNN). There are also some studies directly neuralizing existing regularization-based methods via replacing the regularizer by a CNN; see *e.g.* [35, 10].

In comparison to the robustness to image noise, the robustness to kernel errors receives less attention in existing NBID methods, even it is important for real applications. It is known that without a specific mechanism, a NBID method is sensitive to kernel errors; see *e.g.* [18, 37, 42, 28]. There are only a few regularization-based methods considering kernel errors explicitly; see *e.g.* [18, 37]. For DL-based methods, Vasu *et al.* [42] proposed an NN that fuses multiple estimates generated by a hyper-Laplacian regularized inverse with different regularization weights. By unfolding a total least squared estimator, Nan and Ji [28] proposed a dual-path NN with a kernel-error residual estimation module. Dong *et al.* [10] handles kernel errors by performing Wiener deconvolution in feature domain with a multi-scale refinement process.

Unsupervised DL for NBID The study on unsupervised DL-based NBID is scant in the literature. Some studies (*e.g.* [44, 53]) leverage the generative priors of untrained NNs to perform online internal learning on a test image. While avoiding using external training samples (including GTs), their performance is not satisfactory. Chen *et al.* [7] achieved a significant performance improvement by leveraging model uncertainty induced by dropout. However, all these methods need to train different models for different images, whose computational cost can be overwhelming. In contrast, the proposed approach trains a universal model for efficiently processing all test images in an offline manner. There are few works on offline unsupervised DL for NBID. One is Lim *et al.* [25] which uses unpaired blurred images and latent images to train an adversarial generative network with cycle consistency, which is not is GT-free.

There are also a number of unsupervised DL methods for solving linear inverse problems; see *e.g.* [46, 5, 17, 32, 43]. These methods are not specifically designed for NBID, and none of them concern the errors in measurement matrix, *i.e.*, kernel errors in our case.

3 Proposed Approach

Through the paper, the NN for NBID is denoted by $\mathcal{F}(\cdot,\cdot)$, which predicts the latent image from an input pair of degraded image $\boldsymbol{Y}$ and blur kernel $\boldsymbol{K}$.

3.1 Self-Supervised Reconstruction Loss

To make the discussion more accessible, without loss of generality, we temporarily assume the input kernel is accurate in this subsection. Let $\mathcal{N}(\mathbf{0}, \sigma^2\boldsymbol{I})$ denote

the normal distribution with zero mean and diagonal variance matrix $\sigma^2\boldsymbol{I}$. Consider a training sample $(\boldsymbol{Y}, \boldsymbol{K}^*)$, where $\boldsymbol{K}^*$ denotes the GT kernel. Suppose $\boldsymbol{N} \sim \mathcal{N}(0, \sigma^2\boldsymbol{I})$. We first introduce a self-supervised loss to address $\boldsymbol{N}$:

$$\mathcal{L}^{\mathrm{r}} := \mathbb{E}_{\boldsymbol{U}\sim\mathcal{N}(0,\sigma^2\boldsymbol{I})}\|\boldsymbol{Y} - \boldsymbol{U} - \boldsymbol{K}^* \otimes \mathcal{F}(\boldsymbol{Y} + \boldsymbol{U}, \boldsymbol{K}^*)\|_{\mathrm{F}}^2. \tag{2}$$

This loss is motivated by the R2R unsupervised denoising loss [33]. It is resistant to the noise, as shown in Proposition 1.

Proposition 1. *Consider* $\boldsymbol{Y} = \boldsymbol{K}^* \otimes \boldsymbol{X} + \boldsymbol{N}$ *where* $\boldsymbol{N} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{I})$. *Then,*

$$\begin{aligned} &\mathbb{E}_{\boldsymbol{N},\boldsymbol{U}\sim\mathcal{N}(0,\sigma^2\boldsymbol{I})}\|\boldsymbol{Y} - \boldsymbol{U} - \boldsymbol{K}^* \otimes \mathcal{F}(\boldsymbol{Y} + \boldsymbol{U}, \boldsymbol{K}^*)\|_{\mathrm{F}}^2 \\ =&\mathbb{E}_{\boldsymbol{N},\boldsymbol{U}\sim\mathcal{N}(0,\sigma^2\boldsymbol{I})}\|\boldsymbol{K}^* \otimes \big[\boldsymbol{X} - \mathcal{F}(\boldsymbol{Y} + \boldsymbol{U}, \boldsymbol{K}^*)\big]\|_{\mathrm{F}}^2 + \text{const.} \end{aligned} \tag{3}$$

See supplementary materials for the proof. It can be seen that the noise in $\boldsymbol{Y}$ is effectively removed in the loss function even without accessing the GT $\boldsymbol{X}$.

3.2 Approximate Supervision in Image Space with Kernel Diversity and Cross-Image Patch Recurrence

The loss $\mathcal{L}^{\mathrm{r}}$ indeed provides a weak form of the supervised loss $\mathcal{L}^{\mathrm{gt}}$ defined by

$$\mathcal{L}^{\mathrm{gt}} := \|\boldsymbol{X} - \mathcal{F}(\boldsymbol{Y}, \boldsymbol{K}^*)\|_{\mathrm{F}}^2, \tag{4}$$

with training samples in the form of $(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{K}^*)$. To see this, recall that for a kernel $\boldsymbol{K}^*$, the space $\mathbb{R}^{M\times N}$ can be expressed as the direct sum of two orthogonal subspaces: null space $\mathrm{Null}(\boldsymbol{K}^*)$ and range space $\mathrm{Range}(\widetilde{\boldsymbol{K}^*})$, where $\widetilde{\boldsymbol{K}^*}$ is the flipped version of $\boldsymbol{K}^*$ (see [15]), and

$$\mathrm{Null}(\boldsymbol{K}^*) = \{\boldsymbol{E} \in \mathbb{R}^{M\times N} : \boldsymbol{K}^* \otimes \boldsymbol{E} = \boldsymbol{0}\}, \tag{5}$$

$$\mathrm{Range}(\widetilde{\boldsymbol{K}^*}) = \{\widetilde{\boldsymbol{K}^*} \otimes \boldsymbol{X} \,:\, \boldsymbol{X} \in \mathbb{R}^{M\times N}\}. \tag{6}$$

It can be seen that $\mathcal{L}^{\mathrm{r}}$ indeed measures the prediction error in $\mathrm{Range}(\widetilde{\boldsymbol{K}^*})$, not the full space $\mathbb{R}^{M\times N}$. In other words, $\mathcal{L}^{\mathrm{r}}$ is equivalent to some norm defined in $\mathrm{Range}(\widetilde{\boldsymbol{K}^*})$: $\mathbb{E}_{\boldsymbol{U}\sim\mathcal{N}_\sigma}\|\boldsymbol{X} - \mathcal{F}(\boldsymbol{Y} + \boldsymbol{U}, \boldsymbol{K}^*)\|^2_{\mathrm{Range}(\widetilde{\boldsymbol{K}^*})}$.

The remaining task is then how to measure the prediction error in $\mathrm{Null}(\boldsymbol{K}^*)$. While it is challenging with a single image, it is much easier when training the NN on a set of blurred images whose blur kernels have sufficient variations. Note that not every point in $\mathbb{R}^{M\times N}$ is an image. There are certain priors for images and one is patch recurrence [9]. The patch recurrence prior states that image patches are likely to repeat over the image [9] and across different images [47].

Suppose that the dataset we are processing have sufficient variations on blur kernels and have strong patch recurrence across images. Then, the loss function $\mathcal{L}^{\mathrm{r}}$ indeed provides an approximate measure of the prediction accuracy over the full space. Recall that a deep NN can be viewed as processing image patches whose size is limited by its receptive field. Then, for each GT image patch $\boldsymbol{P}$,

suppose we have a set of its blurred correspondences $\{\boldsymbol{Q}_j\}_j$ across many images with different kernels $\{\boldsymbol{K}_j^*\}_j$. Then, with sufficient variations among $\{\boldsymbol{K}_j^*\}_j$, we are likely to have $\cap_j \text{Null}(\boldsymbol{K}_j^*) \approx \{\boldsymbol{0}\}$, or equivalently $\cup_j \text{Range}(\widetilde{\boldsymbol{K}_j^*}) = \mathbb{R}^{M\times N}$. As $\mathcal{L}^{\text{r}}$ measures the prediction error in each $\text{Range}(\widetilde{\boldsymbol{K}_j^*})$, the summation of $\mathcal{L}^{\text{r}}$ over $\{Q_j\}_j$ provides the measure of prediction error on $\boldsymbol{P}$. In other words, $\mathcal{L}^{\text{r}}$ provides an approximation to the supervised loss defined on GT images in this case. The assumption of kernel diversity in the training dataset is reasonable for many domains. For instance, in motion deblurring, when two motion blur kernels have different dominant orientations, the intersection of their null spaces are roughly close to a zero set; see *e.g.* [30, 4, 24].

3.3 Self-Supervised Prediction Loss

For notational simplicity, kernel errors are omitted in the previous discussions. In the presence of kernel errors, the similar conclusion also holds. Suppose the input kernel, denoted by $\boldsymbol{K}$, differs from the GT kernel $\boldsymbol{K}^*$. Then, $\mathcal{L}^{\text{r}}$ will induce some errors related to $\Delta\boldsymbol{K} = \boldsymbol{K} - \boldsymbol{K}^*$, and we have

$$\begin{aligned} &\mathbb{E}_{\boldsymbol{N},\boldsymbol{U}\sim\mathcal{N}_\sigma}\|\boldsymbol{Y} - \boldsymbol{U} - \boldsymbol{K}\otimes\mathcal{F}(\boldsymbol{Y}+\boldsymbol{U},\boldsymbol{K})\|_{\text{F}}^2 \\ =&\mathbb{E}_{\boldsymbol{N},\boldsymbol{U}\sim\mathcal{N}_\sigma}\|\boldsymbol{K}\otimes\boldsymbol{X} - \Delta\boldsymbol{K}\otimes\boldsymbol{X} + \boldsymbol{N} - \boldsymbol{U} - \boldsymbol{K}\otimes\mathcal{F}(\boldsymbol{Y}+\boldsymbol{U},\boldsymbol{K})\|_{\text{F}}^2 \\ =&\mathbb{E}_{\boldsymbol{N},\boldsymbol{U}\sim\mathcal{N}_\sigma}\|\boldsymbol{K}\otimes\big[\boldsymbol{X} - \mathcal{F}(\boldsymbol{Y}+\boldsymbol{U},\boldsymbol{K})\big]\|_{\text{F}}^2 + \delta(\Delta\boldsymbol{K}\otimes\boldsymbol{X}) + \text{const}, \end{aligned} \tag{7}$$

where $\delta(\Delta\boldsymbol{K}\otimes\boldsymbol{X})$ denotes the error term induced by kernel errors, which distorts the measure on prediction error of the NN. In the next, we introduce an additional self-supervised prediction loss to handle such distortion.

The additional loss is defined over the intermediate estimates of the latent image during the training stage, as well as over the re-corrupted versions of the inaccurate kernels in training data. The intermediate estimates are used to simulate the GT images for training. The inaccurate kernels are used to mimic GT kernels which are applied to the simulated GT images to form blurred images as the input images for training. The re-corrupted kernels are used as the input erroneous kernels for training so as to improve the robustness to kernel errors.

Let $\mathbb{K}$ denote the set of kernels in training data. For a sample $(\boldsymbol{Y},\boldsymbol{K})$ where $\boldsymbol{K}$ is erroneous, let $\boldsymbol{Z}_{\boldsymbol{K}}$ denote the output from the NN in some intermediate training stage:

$$\boldsymbol{Z}_{\boldsymbol{K}} = \mathcal{F}(\boldsymbol{Y}+\boldsymbol{U},\boldsymbol{K}),\quad \boldsymbol{U}\sim\mathcal{N}_\sigma, \tag{8}$$

which can be viewed as an approximation of the GT image corresponding to the pair $(\boldsymbol{Y}+\boldsymbol{U},\boldsymbol{K})$, and the noise $\boldsymbol{U}$ comes from the definition of $\mathcal{L}^{\text{r}}$ in (2). Then, we synthesize a set of training samples with the triples: $\big($noisy blurred image $\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}}$, erroneous kernel $\mathcal{S}(\overline{\boldsymbol{K}})$, latent Image $\boldsymbol{Z}_{\boldsymbol{K}}\big)$:

$$\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}} := \overline{\boldsymbol{K}}\otimes\boldsymbol{Z}_{\boldsymbol{K}} + \overline{\boldsymbol{N}}, \tag{9}$$

for $\overline{\boldsymbol{K}}\sim\mathbb{K}/\boldsymbol{K}$, where $\mathcal{S}$ is the error generator proposed in [28] for simulating the erroneous estimate of the GT kernel, and $\overline{\boldsymbol{N}}$ denotes the white Gaussian

(WG) noise with variance randomly drawn from the range of noise variances of training samples. Such a set of triplets $\{(\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}}, \mathcal{S}(\overline{\boldsymbol{K}}), \boldsymbol{Z}_{\boldsymbol{K}})\}_{\overline{\boldsymbol{K}}\sim\mathbb{K}/\boldsymbol{K}}$ are used to supervise the training, using the self-supervised prediction loss defined by

$$\mathcal{L}^{\mathrm{p}} := \mathbb{E}_{\overline{\boldsymbol{N}},\overline{\boldsymbol{K}}\sim\mathbb{K}/\boldsymbol{K},\mathcal{S}\sim\mathbb{S}}\|\boldsymbol{Z}_{\boldsymbol{K}} - \mathcal{F}(\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}}, \mathcal{S}(\overline{\boldsymbol{K}}))\|_1. \tag{10}$$

When provided with accurate kernels, we fix $\mathcal{S}$ to an identity mapping. In this case, $\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}}$ varies with $\boldsymbol{U}$, and thus $\mathcal{L}^{\mathrm{p}}$ still contributes.

An additional benefit from $\mathcal{L}^{\mathrm{p}}$ is that, it can further improve the accuracy of $\mathcal{L}^{\mathrm{r}}$ in measuring prediction errors. Recall that kernel variations are important for $\mathcal{L}^{\mathrm{r}}$ to measure the prediction error in the null space $\mathrm{Null}(\boldsymbol{K}^*)$. The triplets $\{(\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}}, \mathcal{S}(\overline{\boldsymbol{K}}), \boldsymbol{Z}_{\boldsymbol{K}})\}_{\overline{\boldsymbol{K}}\sim\mathbb{K}/\boldsymbol{K}}$ increase kernel diversity due to the generator $\mathcal{S}$, and it also enforces cross-image patch recurrence, as for every $(\overline{\boldsymbol{Z}}_{\overline{\boldsymbol{K}}}, \mathcal{S}(\overline{\boldsymbol{K}}))$, the output of $\mathcal{F}$ is expected to predict the same target.

To conclude, more pairs are generated during the intermediate training stage for NN training. The intermediate image estimates $\boldsymbol{Z}_{\boldsymbol{K}}$ are used as pseudo GT images, and the inaccurate kernels $\boldsymbol{K}$ are used as pseudo GT kernels. The loss $\mathcal{L}^{\mathrm{p}}$ defined on these pairs can improve the robustness to kernel errors, as well as improve the effectiveness of $\mathcal{L}^{\mathrm{r}}$ in the kernel-induced null space.

3.4 Unsupervised Training and Ensemble Inference

Unsupervised training The NN is trained by minimizing the loss:

$$\mathcal{L} := \mathcal{L}^{\mathrm{r}} + \beta\mathcal{L}^{\mathrm{p}}, \tag{11}$$

with $\beta \in \mathbb{R}^+$. In training, β is first set to 0 so as to concentrate on minimizing $\mathrm{Range}(\widetilde{\boldsymbol{K}})$ and then is increased to include $\mathcal{L}^{\mathrm{p}}$. For handling inaccurate kernels, after sufficient training in the first stage, we modify the loss $\mathcal{L}^{\mathrm{r}}$ to be

$$\mathbb{E}_{\boldsymbol{U}\sim\mathcal{N}_\sigma,\mathcal{S}\sim\mathbb{S}}\|\boldsymbol{Y} - \boldsymbol{U} - \mathcal{S}(\boldsymbol{K}) \otimes \mathcal{F}(\boldsymbol{Y} + \boldsymbol{U}, \mathcal{S}(\boldsymbol{K}))\|_{\mathrm{F}}^2, \tag{12}$$

where data augmentation on kernels is done the same as $\mathcal{L}^{\mathrm{p}}$.

Ensemble inference via kernel perturbation For further performance improvement in handling kernel errors of test data, we propose a simple yet effective ensemble inference scheme for a sample $(\boldsymbol{Y}, \boldsymbol{K})$ as follows:

$$\boldsymbol{X}^{\mathrm{est}} = \mathbb{E}_{\boldsymbol{U}\sim\mathcal{N},\boldsymbol{V}\sim\mathbb{V}_{\boldsymbol{K}}}\mathcal{F}(\boldsymbol{Y} + \boldsymbol{U}, \frac{\boldsymbol{K} + \boldsymbol{V}}{\|\boldsymbol{K} + \boldsymbol{V}\|_1}), \tag{13}$$

with a perturbation set $\mathbb{V}_{\boldsymbol{K}}$. We can see that the prediction is done by averaging over the estimates from both perturbed images and perturbed kernels. The motivation of kernel perturbation is that, when using randomly-perturbed kernels to restore the latent image, the artifacts in the recovered images caused by kernel errors tend to show certain degree of independence, which can be attenuated by averaging. The perturbation on input image is to ensure the consistency of noise characteristics between training and test.

The perturbation set $\mathbb{V}_{\boldsymbol{K}}$ is constructed such that it does not cause any shift of the kernel, as the NBID results with different kernel shifts are not aligned and their average cannot cancel the artifacts well. To address this, we first generate a noise map with the same size as $\boldsymbol{K}$, whose elements are sampled from $\mathcal{N}_{0.1\%}$. Then, the Fourier phase spectrum of the noise map is replaced by that of $\boldsymbol{K}$, and the resulting noise map is used as $\boldsymbol{V}$. This ensures the perturbation does not shift the kernel. It is worth mentioning that, experimentally while the ensemble inference with such perturbations brings noticeable improvement for inaccurate, it brings either negligible improvement or little degradation on the performance for accurate kernels.

3.5 NN Architecture

Following recent supervised DL methods for NBID (*e.g.* [50, 21]), the NN used for the experiments is based on the unfolding of the half-quadratic splitting (HQS) scheme for solving a regularization model:

$$\min_{\boldsymbol{X}} \ \|\boldsymbol{Y} - \boldsymbol{K} \otimes \boldsymbol{X}\|_{\mathrm{F}}^2 + \sum\nolimits_p \psi(\boldsymbol{W}_p \otimes \boldsymbol{X}), \tag{14}$$

where $\psi(\cdot)$ is some prior-inducing regularization, $\{\boldsymbol{W}_p\}_p$ is the set of 3×3 wavelet high-pass filters. The iteration scheme from HQS reads: for $t = 1, \cdots, T$,

$$\begin{aligned} &\boldsymbol{X}^{(t)} := \mathrm{argmin}_{\boldsymbol{X}} \ \|\boldsymbol{Y} - \boldsymbol{K} \otimes \boldsymbol{X}\|_{\mathrm{F}}^2 + \sum\nolimits_p \lambda_t \|\boldsymbol{A}_p^{(t)} - \boldsymbol{W}_p \otimes \boldsymbol{X}\|_{\mathrm{F}}^2, \\ &\boldsymbol{A}_p^{(t+1)} := \mathrm{argmin}_{\boldsymbol{A}_p} \ \|\boldsymbol{A}_p - \boldsymbol{W}_p \otimes \boldsymbol{X}^{(t)}\|_{\mathrm{F}}^2 + \psi(\boldsymbol{A}_p), \forall p, \end{aligned} \tag{15}$$

where $\{\lambda_t\}_t$ are hyper-parameters simply set as: $\lambda_0 = 0.5\%$ and $\lambda_t = 10\%$ for $t > 0$. Accordingly, the NN is constructed by stacking two blocks alternatively; see 1 for an illustration of the detailed structure.

(a) Inversion block $\mathcal{G}^{(t)} : \boldsymbol{Y}, \{\boldsymbol{A}_p^{(t)}\}_p, \boldsymbol{K} \to \boldsymbol{X}^{(t)}$. It corresponds to the 1st step in the iteration, which is an unconstrained quadratic problem with an analytic solution. We adopt the FFT-based computation scheme with adaptive boundary padding [21] to calculate the analytic solution.

(b) Denoising block $\mathcal{H}^{(t)} : \{\boldsymbol{X}^{(t)}\}_{t=0}^t \to \{\boldsymbol{A}_p^{(t+1)}\}_p$. It corresponds to the 2nd step of the iteration, replaced by a U-Net to refine $\boldsymbol{W}_p \otimes \boldsymbol{X}^{(t)}$. The U-Net takes all previous estimates as the input for better performance. In the implementation, it first removes noise from $\boldsymbol{X}^{(t)}$ and then applies $\boldsymbol{W}_p$ to obtain $\boldsymbol{A}_p^{(t+1)}$.

We also enforce the loss in the intermediate outputs. Let $\mathcal{F}^{(t)} := \mathcal{G}^{(0)} \to \mathcal{H}^{(1)} \to \cdots \to \mathcal{H}^{(t)} \to \mathcal{G}^{(t)}$, for $t = 1, \cdots, T$. Indeed, $\mathcal{F}^{(t)}$ can be viewed as an NN with different depths. Then, the overall loss is defined on all $\mathcal{F}^{(t)}$s by

$$\mathcal{L}^{\mathrm{total}} := \sum\nolimits_{t=1}^{T-1} \mathcal{L}(\mathcal{F}^{(t)}) + \gamma \mathcal{L}(\mathcal{F}^{(T)}), \tag{16}$$

where γ is fixed to 1.25 in our implementation.

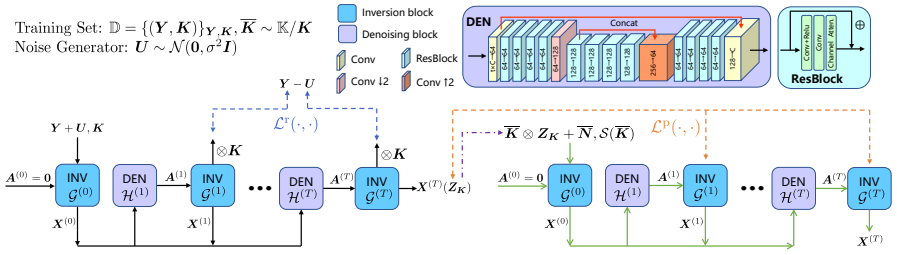

Fig. 1. Diagram of NN architecture used in the proposed approach

4 Performance Evaluation

4.1 Motion Deblurring with Erroneous Kernels

The proposed approach is evaluated on non-blind motion deblurring. To simulate practical scenarios, we follow [28] to prepare the training and test data. For training, random cropping is applied to the images of BSDS500 [2] to have 1500 sharp images of size 256×256, and the 192 motion kernels from [39] are used. The blurred images are generated by first randomly convolving each image with a blur kernel and then adding the WG noise of standard deviation 1%. Afterwards, we apply the BID method [31] to estimate a kernel from each blurred image. The blurred images and the estimated kernels are used as the training samples.

In our training data, each latent image only has one blurred correspondence. This differs from one popular setting in existing supervised NBID methods (*e.g.* [50, 21, 29]) where each image has multiple blurred correspondences with different kernels. Indeed, our setting is more realistic, especially for unsupervised learning. Data augmentation is applied to each pair of a blurred image and the corresponding kernel, using rotation by 90°, 180° or 270°, as well as flipping.

The test is conducted on three datasets: Levin *et al.*'s dataset [23], Sun *et al.*'s dataset, and a subset of Lai *et al.*'s dataset [22]. Several BID methods are called to estimate the kernels which are then used in the NBID process, including [31, 41] for Levin *et al.*'s dataset, [8, 48, 27] for Sun *et al.*'s dataset, and [48, 49, 41, 34] for Lai *et al.*'s dataset. The blurred images are corrupted by the WG noise with standard deviation 1%. In both training and test, the EdgeTapper is used for simulating realistic boundary conditions.

We fix $T = 5$ for our NN. In training, all the model weights are initialized by Kaiming [16] except that the biases are initialized by zeros. The Adam optimizer is used with 300 epochs. In the first 100 epochs, we zero β and set the learning rate to 10^{-4}. Afterwards, we set $\beta = 10^{-2}$ and the learning rate to 10^{-5}. We simply set the random noise $\boldsymbol{U}$ in both training and inference schemes to the WG noise with standard deviation estimated on the input image using [6]. When sampling $\overline{\boldsymbol{K}}$ in $\mathcal{L}^{\mathrm{p}}$, we randomly pick up 10 instances and choose the one with the lowest correlation to $\boldsymbol{K}$. In the test, we perform 10 inferences for averaging.

For experimental comparison, we include *(a)* supervised DL-based methods: IRCNN [51], FCNN [50], FDN [21], VEM [29], DWDN [10] and TLSNN [28]; *(b)*

unsupervised DL-based methods: BPDIP [53], SURE [40] and EI EI [5]; and *(c)* non-DL-based method: DSPSI [45]. Among these methods, TLSNN, DWDN and DSPSI have specific mechanisms to handle kernel errors. The results of FDN, IRCNN, FCNN and TLSNN are quoted from [28]. The model of VEM is retrained on our data for better performance, while that of DWDN is not retrained as no performance gain is observed. The BPDIP is an online unsupervised DL method for NBID, while the SURE and EI are the offline ones for inverse problems but not applied to NBID before. Their loss functions are used to train our NN instead of their original ones for better performance and for a fair comparison.

See Table 1 for the quantitative comparison. The PSNR and SSIM are calculated using the shifting and boundary cut-off scheme of [28] for all the compared methods. For convenience, we name the proposed approach as UNID (Unsupervised Non-blind Image Deconvolution). In all cases, UNID outperformed other unsupervised DL-based methods. The BPDIP cannot leverage external training data and thus yielded much worse performance. Without an effective mechanism to deal with kernel errors, both SURE and EI showed inferior performance to UNID, and the performance of SURE is much worse. In most cases, UNID noticeably outperformed those supervised DL-based methods that do not treat kernel errors specifically. In comparison to the very recent supervised DL-based methods TLSNN and DWDN with specific treatments on kernel errors, UNID still provides comparable performance even it uses neither GT images nor accurate kernels for training. See Fig. 2 for a visual comparison, where UNID can deblur an image with comparable visual quality to that of the supervised NNs.

Table 1. PSNR(dB)/SSIM in motion deblurring with erroneous kernels (bold: highest values among all methods; underlined: highest values among GT-free methods)

	Dataset	Levin *et al.*'s		Sun *et al.*'s			Lai *et al.*'s			
	Kernels	[31]	[41]	[8]	[48]	[27]	[48]	[49]	[41]	[34]
Supervised	IRCNN	30.42/.86	29.56/.83	28.84/.81	29.54/.83	29.23/.82	19.99/.70	19.36/.67	19.46/.67	18.68/.68
	FCNN	31.12/.90	30.27/.88	29.79/.86	30.45/.86	29.84/.84	20.27/.74	19.52/.70	19.80/.70	19.12/.70
	FDN	31.19/**.92**	30.83/.90	29.69/**.87**	30.51/**.88**	29.82/**.86**	N/A	N/A	N/A	N/A
	VEM	31.83/**.92**	31.08/**.91**	30.32/.85	30.61/.86	29.82/.83	22.04/.70	21.95/.70	21.85/.68	21.04/.62
	DWDN	30.87/.91	30.66/.90	29.49/**.87**	30.29/**.88**	29.46/.85	**22.88/.77**	**22.70/.77**	**22.59/.75**	21.21/**.72**
	TLSNN	**31.97/.92**	**31.24/.91**	**30.44/.87**	**30.84**/.87	**30.27/.86**	22.53/.74	22.27/.73	22.31/.72	21.61/.70
w/o GT	DSPSI	29.55/.84	29.10/.82	28.57/.78	29.06/.79	28.74/.78	20.21/.72	19.87/.70	19.91/.69	19.35/.70
	SURE	28.59/.85	27.99/.81	26.91/.69	27.15/.71	27.17/.69	19.24/.65	18.91/.60	19.04/.60	18.39/.55
	BPDIP	28.71/.85	28.12/.84	26.99/.69	27.23/.70	27.14/.69	19.42/.66	19.05/.61	19.11/.60	18.50/.55
	EI	29.37/.85	29.58/.86	28.60/.80	29.05/.83	28.10/.79	20.77/.63	20.24/.63	20.45/.63	19.67/.61
	UNID	31.71/**.92**	30.66/.89	30.24/.86	30.82/.87	30.08/.85	22.43/.73	22.24/.73	22.04/.71	**21.65**/.70

4.2 Motion Deblurring with Accurate Kernels

A majority of existing studies on NBID (*e.g.* [50, 12, 19, 3, 29, 14, 35]) conducted experiments using accurate kernels in both training and test, which simulate the scenarios where kernel errors are sufficiently small and thus negligible. Such a

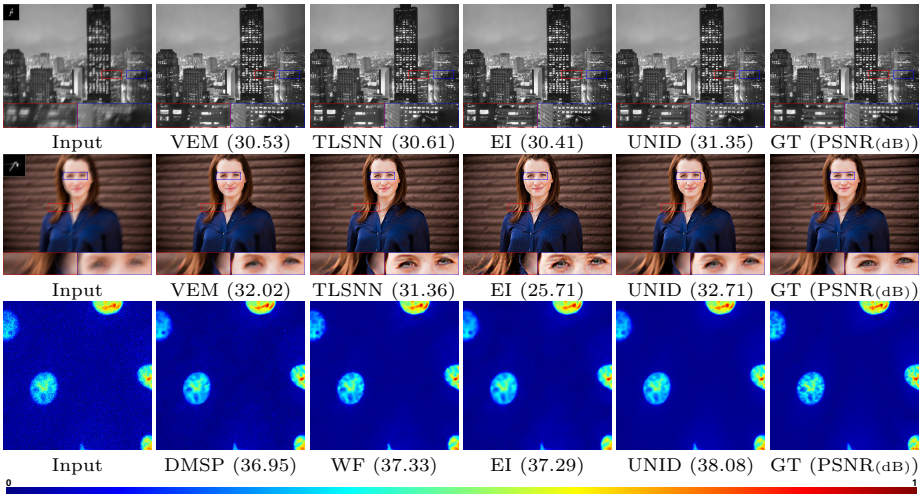

Fig. 2. NBID results of selected methods on motion deblurring with inaccurate kernels (top and middle) and on microscopic deconvolution (bottom)

setting is also used to have a comprehensive evaluation. We follow [21, 50, 29] for the preparation of training and test data, similar to the previous experiment. The 1500 sharp images from random cropping of BSDS500 and the 192 motion kernels from [39] are used to generate the blurred images, with WG noise added whose standard deviation is randomly picked from $[1, 14]/255$. Also, each sharp image only is allowed to form one blurred image to be more realistic. The data augmentation is also applied to have sufficient training samples. An universal model is trained w.r.t. different noise levels. Since there is no kernel error, we lower the contribution of $\mathcal{L}^{\mathrm{p}}$ with $\beta = 10^{-3}$ in the second training stage.

Three datasets are used for test, including Levin *et al.*'s dataset [23], Sun *et al.*'s dataset [41] and Set12 [29]. The blurred images are generated by first convolving each sharp image with the eight motion kernels from [23] respectively and then adding the WG noise with standard deviations of 1%, 3%, 5% respectively. The blurred images and their associated kernels are used for test. The EdgeTapper for boundary simulation is also used in both training and test. Our UNID is compared to the previously-used IRCNN, FDN, VEM, BPDIP, SURE and EI. In addition, two supervised NNs, DMSP [3] and DPDNN [12], are included for comparison. The FDN, FCNN and VEM also use BSD500 for training, DPDNN is re-trained on BSD500, DMSP used with their pre-trained models as they have no training codes provided, and IRCNN is trained on its original noisy data instead of the blurred one. Existing works use different boundary cut-off strategies for evaluation. For a fair comparison, we cut the boundary with half kernel size when calculating the PSNR and SSIM for all methods.

See Table 2 for the quantitative results and comparison. Again, UNID outperformed other unsupervised models in all settings. Surprisingly, while UNID

Table 2. PSNR(dB)/SSIM in motion deblurring with accurate kernels (bold: highest values among all methods; underlined: highest values among GT-free methods)

	Dataset	Sun *et al.*'s			Levin *et al.*'s			Set12		
	STD	1%	3%	5%	1%	3%	5%	1%	3%	5%
Supervised	IRCNN	31.80/.88	28.93/**.80**	27.50/.74	31.11/.89	28.99/.85	27.58/.80	30.37/.87	27.95/.81	26.51/.76
	FDN	32.21/**.89**	28.88/.78	27.61/.73	31.95/.91	28.95/.84	27.45/.80	31.03/.88	27.92/.80	26.47/.76
	DMSP	32.00/.87	28.63/.77	27.48/.74	32.60/.90	29.30/.83	27.84/.81	31.20/.87	27.95/.79	26.43/.76
	DPDNN	31.24/.85	29.09/**.80**	27.90/**.76**	31.15/.89	28.95/.85	27.57/.81	30.11/.86	27.78/.81	26.53/**.78**
	VEM	32.20/**.89**	29.23/**.80**	**27.93/.76**	31.63/.91	29.33/.86	27.79/.81	30.97/.88	28.27/**.82**	26.79/.77
	DWDN	**32.37/.89**	29.19/.80	27.88/.75	**32.92/.92**	**29.85/.87**	28.23/**.83**	**31.37/.89**	**28.36/.82**	**26.89/.78**
w/o GT	SURE	27.32/.73	26.69/.71	25.65/.66	30.40/.88	27.41/.80	25.77/.71	28.69/.83	26.00/.74	24.58/.65
	BPDIP	27.50/.73	26.84/.71	25.71/.66	30.63/.88	27.53/.79	25.86/.72	28.86/.83	26.11/.73	24.65/.66
	EI	31.43/.86	28.80/.77	27.37/.72	31.32/.90	29.05/.85	27.49/.80	29.37/.86	27.50/.80	26.20/.74
	UNID	32.28/**.89**	**29.24/.80**	27.82/.74	32.12/**.92**	29.71/.86	28.16/.82	30.92/.88	28.23/**.82**	26.76/.77

requires no GT images, it also outperformed the supervised models trained with GT images in many settings and it is very competitive to the top performers of the supervised models. All such results demonstrated the effectiveness of UNID.

4.3 Microscopic Deconvolution

Microscopic imaging in science is one domain where GT images are difficult to collect in practice. We follow the same evaluation scheme as [35]. The images of the fluorescence microscopy dataset [52] and the cell segmentation dataset [1] are first scaled to the range of $[0, 1]$ and cropped into patches of size 256×256, among which 975 (230) patches are used for training (test). There are kernels of size 7×7, 9×9, 11×11 or 13×13 provided, among which 25 (5) kernels are used for training (test); see [35] for the details on the kernels. For test, the GT images are first convoluted with one of the five test kernels respectively, and then corrupted by the WG noise with standard deviation $\sigma = 0.1\%, 0.5\%, 1\%, 5\%$ respectively. The training parameters are set the same as previous experiments. In addition, we also simulate erroneous kernels by approximating each GT kernel with a Gaussian kernel that minimizes their MSE. Our model trained by these erroneous kernels is denoted by UNID†.

Similar to previous experiments, for practical scenario simulation, the blurred images for training are generated by convolving each GT image with only one randomly-selected kernel and adding the WG noise with σ randomly picked from $\{0.1\%, 0.5\%, 1\%, 5\%, 10\%\}$. This differs from the supervised learning setting in [35], and the same data augmentation as previous experiments is applied.

In addition to the aforementioned IRCNN, FDN, DMSP, SURE and EI, the WF-KPN-SA [35] (denoted by WF) is also included for comparison, whose results are quoted from [35]. See Table 3 for the quantitative comparison and Fig. 2 for a visual inspection. The boundary cutting scheme of [35] is used to calculate the PSNR and SSIM. The performance of UNID is noticeably better than other unsupervised models and competitive to the supervised ones. Particularly, UNID outperformed all supervised models when $\sigma = 0.5\%, 1\%$. For other noise

Table 3. PSNR(dB)/SSIM results in microscopic NBID (bold: highest values among all methods; underlined: highest values among unsupervised methods)

STD	Supervised				Unsupervised				
	IRCNN	FDN	DMSP	WF	SURE	BPDIP	EI	UNID†	UNID
0.1%	33.33/.86	40.31/**.94**	**40.44/.94**	39.86/**.94**	37.65/.91	37.90/.92	39.41/.93	40.18/**.94**	40.30/**.94**
0.5%	36.88/.90	38.61/.92	39.16/**.93**	38.76/**.93**	37.21/.91	37.37/.91	38.24/.92	38.97/.93	**39.33/.93**
1%	36.80/.90	37.33/.91	37.73/.91	37.81/**.92**	36.60/.90	36.70/.90	36.87/.90	37.59/.90	**38.36/.92**
5%	32.44/.79	33.50/.84	34.31/.86	**34.58/.87**	31.67/.77	31.79/.77	32.27/.79	33.67/.83	34.32/.85

Table 4. PSNR(dB) results using different training loss functions

(a) PSNR(dB) results *w.r.t.* loss

Training Loss	Sun *et al.*'s Dataset			Levin *et al.*'s Dataset	
	[8]	[48]	[27]	[31]	[41]
Only $\mathcal{L}^{\mathrm{p}}$	28.04	28.11	27.83	28.03	27.60
Only $\mathcal{L}^{\mathrm{r}}$	29.92	30.50	29.72	30.96	30.11
Both	30.24	30.82	30.08	31.71	30.66
Fixed $\mathcal{L}^{\mathrm{r}}$	30.13	30.74	30.00	31.44	30.43
$\mathcal{L}^{\mathrm{gt}}$	30.27	30.87	30.13	31.91	30.87

(b) PSNR(dB) results *w.r.t.* kernel perturbations

Kernel Perturb.	Lai *et al.*'s Dataset				Sun *et al.*'s Dataset		
	[48]	[49]	[41]	[34]	[8]	[48]	[27]
Proposed	22.43	22.24	22.04	21.65	30.24	30.82	30.08
Disabled	21.54	20.96	21.13	20.37	29.31	29.49	28.80
Gaussian	22.30	22.15	22.02	20.98	29.48	30.05	29.60

levels, UNID performed a bit worse than the top performers. The UNID† saw a certain performance decrease due to the kernel errors in training data, but its performance is still competitive among all methods and better than other unsupervised models.

4.4 Ablation Studies

Comparison of different loss functions To evaluate how the proposed self-supervised loss functions contribute to the performance, we use the following loss functions to retrain the NN with rigorous tuning-up: *(a)* $\mathcal{L}^{p}$*:* only train with $\mathcal{L}^{\mathrm{p}}$; *(b)* $\mathcal{L}^{r}$*:* only train with $\mathcal{L}^{\mathrm{r}}$; *(c) Fixed* $\mathcal{L}^{r}$*:* do not change $\mathcal{L}^{\mathrm{r}}$ to (12) during training; and *(d)* $\mathcal{L}^{gt}$*:* use (4) and the training data of [29] for supervised training.

See Table 4(a) for the results on two datasets, from which we have the following observations. First, the NN trained by UNID without GT images performs nearly as well as the one trained with GT images. Second, the loss $\mathcal{L}^{\mathrm{r}}$ is critical to the training, and the performance will have a significant decrease when training without it. This is because the restoration on Range($\widetilde{\boldsymbol{K}}$) corresponds to the major part of an image, and the $\mathcal{L}^{\mathrm{r}}$ relies on the success of restoration on Range($\widetilde{\boldsymbol{K}}$) as well. Third, the loss $\mathcal{L}^{\mathrm{p}}$ also has a noticeable contribution to the performance. Lastly, modifying $\mathcal{L}^{\mathrm{r}}$ to (12) brings slight improvement.

Performance gain by ensemble inference Two inference schemes are evaluated as baselines: one is the standard prediction scheme without kernel perturbation (*i.e.* $\mathbb{V}_{\widetilde{\boldsymbol{K}}} = \emptyset$) and the other uses WG noise with standard deviation 0.1% as the perturbation term. See Table 4(b) and Fig. 3 for the comparison.

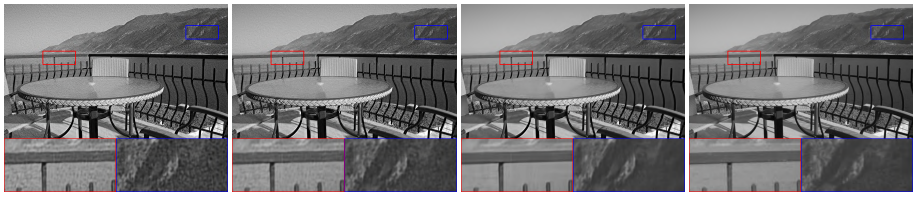

Fig. 3. Effects of kernel perturbations

Table 5. PSNR(dB) results of supervised models with ensemble inference

Dataset	Kernels	FDN	FDN*	VEM	VEM*	DWDN	DWDN*	UNID	UNID*
Sun *et al.*'s	[8]	29.40	30.28	29.69	30.30	29.49	29.98	29.31	30.24
	[48]	30.05	30.99	30.51	30.99	30.29	30.63	29.49	30.82
	[27]	29.12	30.21	29.82	30.07	29.46	29.85	28.80	30.08
Levin *et al.*'s	[31]	30.27	31.77	30.27	31.61	30.87	31.85	30.25	31.71
	[41]	30.11	30.80	30.11	30.76	30.66	31.00	29.80	30.66

Our ensemble inference scheme outperformed the standard one, which shows that averaging predictions from perturbed kernels can improve the robustness to kernel errors. Furthermore, a trivial perturbation using WG noise also brings some improvement but not as significant as ours.

The proposed inference scheme with kernel perturbations is also applied to the supervised models including FDN [21], VEM [29], DWDN [10]. The resulting methods are marked by *. Table 5 lists the results. Note that the reported results of VEM differ from those in Section 4.1 as here we use the pre-trained model instead of the retrained one. It can seen that our ensemble inference scheme can bring noticeable performance gain: ranging from 0.25dB to 1.09dB on Sun *et al.*'s dataset and raning from 0.34dB to 1.5 dB on Levin *et al.*'s dataset. This demonstrates the applicability of ensemble inference to supervised models. Our UNID is also included for comparison. Even using the same inference scheme, it is still competitive to the supervised models.

5 Conclusion

While supervised DL has been the main driving force in the recent development of NBID methods, its applicability to real applications is limited by its requirement on GT images. In this paper, we developed a GT-free unsupervised DL approach to NBID, which provides a complement to existing supervised DL-based methods especially when GT images are of the limited amount or unavailable. Based on a self-supervised training scheme with rigorous mathematical treatment and an ensemble inference scheme based on kernel perturbation, the proposed approach can effectively handle kernels errors and image noise in both training and test data. Extensive experiments on uniform motion deblurring and microscopic deconvolution showed the effectiveness of the proposed approach.

References

1. Al-Kofahi, Y., Zaltsman, A., Graves, R., Marshall, W., Rusu, M.: A deep learning-based algorithm for 2-d cell segmentation in microscopy images. BMC bioinformatics **19**(1), 1–11 (2018)
2. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Transactions on Neural Networks and Learning Systems **33**(5), 898–916 (2010)
3. Bigdeli, S.A., Jin, M., Favaro, P., Zwicker, M.: Deep mean-shift priors for image restoration. In: Advances in Neural Information Processing Systems. pp. 763–772 (2017)
4. Cai, J.F., Ji, H., Liu, C., Shen, Z.: High-quality curvelet-based motion deblurring from an image pair. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1566–1573 (2009)
5. Chen, D., Tachella, J., Davies, M.E.: Equivariant imaging: Learning beyond the range space. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)
6. Chen, G., Zhu, F., Ann Heng, P.: An efficient statistical method for image noise level estimation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 477–485 (2015)
7. Chen, M., Quan, Y., Pang, T., Ji, H.: Nonblind image deconvolution via leveraging model uncertainty in an untrained deep neural network. International Journal of Computer Vision pp. 1–20 (2022)
8. Cho, S., Lee, S.: Fast motion deblurring. In: Proceedings of the ACM SIGGRAPH Asia, pp. 1–8 (2009)
9. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE Transactions on image processing **16**(8), 2080–2095 (2007)
10. Dong, J., Roth, S., Schiele, B.: Deep wiener deconvolution: Wiener meets deep learning for image deblurring. Advances in Neural Information Processing Systems **33** (2020)
11. Dong, J., Roth, S., Schiele, B.: Learning spatially-variant map models for non-blind image deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4886–4895 (2021)
12. Dong, W., Wang, P., Yin, W., Shi, G.: Denoising prior driven deep neural ketwork for image restoration. IEEE Transactions on Neural Networks and Learning Systems **41**(10), 2305–2318 (2019)
13. Dong, W., Zhang, L., Shi, G., Li, X.: Nonlocally centralized sparse representation for image restoration. IEEE Transactions on Image Processing **22**(4), 1620–1630 (2012)
14. Eboli, T., Sun, J., Ponce, J.: End-to-end interpretable learning of non-blind image deblurring. In: Proceedings of the European Conference on Computer Vision (2020)
15. Folberth, J., Becker, S.: Efficient adjoint computation for wavelet and convolution operators [lecture notes]. IEEE Signal Processing Magazine **33**(6), 135–147 (2016)
16. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1026–1034 (2015)
17. Hendriksen, A.A., Pelt, D.M., Batenburg, K.J.: Noise2inverse: Self-supervised deep convolutional denoising for tomography. IEEE Transactions on Computational Imaging **6**, 1320–1335 (2020)

18. Ji, H., Wang, K.: Robust image deblurring with an inaccurate blur kernel. IEEE Transactions on Image Processing **21**(4), 1624–1634 (2011)
19. Jin, M., Roth, S., Favaro, P.: Noise-blind image deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3834–3842 (2017)
20. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. Advances in Neural Information Processing Systems **22**, 1033–1041 (2009)
21. Kruse, J., Rother, C., Schmidt, U.: Learning to push the limits of efficient fft-based image deconvolution. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4586–4594 (2017)
22. Lai, W.S., Huang, J.B., Hu, Z., Ahuja, N., Yang, M.H.: A comparative study for single image blind deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1701–1709 (2016)
23. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Efficient marginal likelihood optimization in blind deconvolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2657–2664. IEEE (2011)
24. Li, W., Zhang, J., Dai, Q.: Exploring aligned complementary image pair for blind motion deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 273–280 (2011)
25. Lim, S., Park, H., Lee, S.E., Chang, S., Sim, B., Ye, J.C.: Cyclegan with a blur kernel for deconvolution microscopy: Optimal transport geometry. IEEE Transactions on Computational Imaging **6**, 1127–1138 (2020)
26. Meinhardt, T., Moller, M., Hazirbas, C., Cremers, D.: Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1781–1790 (2017)
27. Michaeli, T., Irani, M.: Blind deblurring using internal patch recurrence. In: Proceedings of the European Conference on Computer Vision. pp. 783–798. Springer (2014)
28. Nan, Y., Ji, H.: Deep learning for handling kernel/model uncertainty in image deconvolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2388–2397 (2020)
29. Nan, Y., Quan, Y., Ji, H.: Variational-em-based deep learning for noise-blind image deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3626–3635 (2020)
30. Nayar, S.K., Ben-Ezra, M.: Motion-based motion deblurring. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(6), 689–698 (2004)
31. Pan, J., Sun, D., Pfister, H., Yang, M.H.: Blind image deblurring using dark channel prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1628–1636 (2016)
32. Pang, T., Quan, Y., Ji, H.: Self-supervised bayesian deep learning for image recovery with applications to compressive sensing. In: European Conference on Computer Vision. pp. 475–491. Springer (2020)
33. Pang, T., Zheng, H., Quan, Y., Ji, H.: Recorrupted-to-recorrupted: Unsupervised deep learning for image denoising. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2043–2052 (2021)
34. Perrone, D., Favaro, P.: Total variation blind deconvolution: The devil is in the details. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2909–2916 (2014)

35. Pronina, V., Kokkinos, F., Dylov, D.V., Lefkimmiatis, S.: Microscopy image restoration with deep wiener-kolmogorov filters. Proceedings of the European Conference on Computer Vision (2020)
36. Quan, Y., Ji, H., Shen, Z.: Data-driven multi-scale non-local wavelet frame construction and image recovery. Journal of Scientific Computing **63**(2), 307–329 (2015)
37. Ren, D., Zuo, W., Zhang, D., Xu, J., Zhang, L.: Partial deconvolution with inaccurate blur kernel. IEEE Transactions on Image Processing **27**(1), 511–524 (2017)
38. Ren, W., Zhang, J., Ma, L., Pan, J., Cao, X., Zuo, W., Liu, W., Yang, M.H.: Deep non-blind deconvolution via generalized low-rank approximation. Advances in Neural Information Processing Systems **31**, 297–307 (2018)
39. Schmidt, U., Jancsary, J., Nowozin, S., Roth, S., Rother, C.: Cascades of regression tree fields for image restoration. IEEE Transactions on Neural Networks and Learning Systems **38**(4), 677–689 (2015)
40. Soltanayev, S., Chun, S.Y.: Training deep learning based denoisers without ground truth data. Advances in Neural Information Processing Systems (2018)
41. Sun, L., Cho, S., Wang, J., Hays, J.: Edge-based blur kernel estimation using patch priors. In: Proceedings of the IEEE International Conference on Computational Photography. pp. 1–8. IEEE (2013)
42. Vasu, S., Maligireddy, V.R., Rajagopalan, A.: Non-blind deblurring: Handling kernel uncertainty with cnns. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3272–3281 (2018)
43. Wang, W., Li, J., Ji, H.: Self-supervised deep image restoration via adaptive stochastic gradient langevin dynamics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1989–1998 (2022)
44. Wang, Z., Wang, Z., Li, Q., Bilen, H.: Image deconvolution with deep image and kernel priors. Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
45. Whyte, O., Sivic, J., Zisserman, A.: Deblurring shaken and partially saturated images. Proceedings of the IEEE International Conference on Computer Vision **110**(2), 185–201 (2014)
46. Xia, Z., Chakrabarti, A.: Training image estimators without image ground-truth. Advances in Neural Information Processing Systems (2019)
47. Xu, J., Zhang, L., Zuo, W., Zhang, D., Feng, X.: Patch group based nonlocal self-similarity prior learning for image denoising. In: Proceedings of the IEEE international Conference on Computer Vision. pp. 244–252 (2015)
48. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: Proceedings of the European Conference on Computer Vision. pp. 157–170. Springer (2010)
49. Xu, L., Zheng, S., Jia, J.: Unnatural l0 sparse representation for natural image deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1107–1114 (2013)
50. Zhang, J., Pan, J., Lai, W.S., Lau, R., Yang, M.H.: Learning fully convolutional networks for iterative non-blind deconvolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6969–6977 (2017)
51. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. vol. 2, pp. 2808–2817 (2017)
52. Zhang, Y., Zhu, Y., Nichols, E., Wang, Q., Zhang, S., Smith, C., Howard, S.: A poisson-gaussian denoising dataset with real fluorescence microscopy images.

In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11710–11718 (2019)
53. Zukerman, J., Tirer, T., Giryes, R.: Bp-dip: A backprojection based deep image prior. In: Proceedings of the European Signal Processing Conference (2020)