LG-BPN: Local and Global Blind-Patch Network for Self-Supervised Real-World Denoising

Zichun Wang [1], Ying Fu [1*], Ji Liu [2], Yulun Zhang [3]
[1] Beijing Institute of Technology, [2] Baidu Inc., Beijing, China, [3] ETH Zürich
{wangzichun, fuying}@bit.edu.cn, liuji04@baidu.com, yulun100@gmail.com

Abstract

Despite the significant results on synthetic noise under simplified assumptions, most self-supervised denoising methods fail under real noise due to the strong spatial noise correlation, including the advanced self-supervised blind-spot networks (BSNs). For recent methods targeting real-world denoising, they either suffer from ignoring this spatial correlation, or are limited by the destruction of fine textures for under-considering the correlation. In this paper, we present a novel method called LG-BPN for self-supervised real-world denoising, which takes the spatial correlation statistic into our network design for local detail restoration, and also brings the long-range dependencies modeling ability to previously CNN-based BSN methods. First, based on the correlation statistic, we propose a densely-sampled patch-masked convolution module. By taking more neighbor pixels with low noise correlation into account, we enable a denser local receptive field, preserving more useful information for enhanced fine structure recovery. Second, we propose a dilated Transformer block to allow distant context exploitation in BSN. This global perception addresses the intrinsic deficiency of BSN, whose receptive field is constrained by the blind spot requirement, which can not be fully resolved by the previous CNN-based BSNs. These two designs enable LG-BPN to fully exploit both the detailed structure and the global interaction in a blind manner. Extensive results on real-world datasets demonstrate the superior performance of our method. https://github.com/Wang-XIaoDingdd/LGBPN

1. Introduction

Image denoising is a fundamental research topic for low-level vision [7, 36]. Noise can greatly degrade the quality of the captured images, thus bringing adverse impacts on the subsequent downstream tasks [22, 32]. Recently, with the rapid development of neural networks, learning-based methods have shown significant advances compared with traditional model-based algorithms [5, 8, 10, 11].

*Corresponding Author

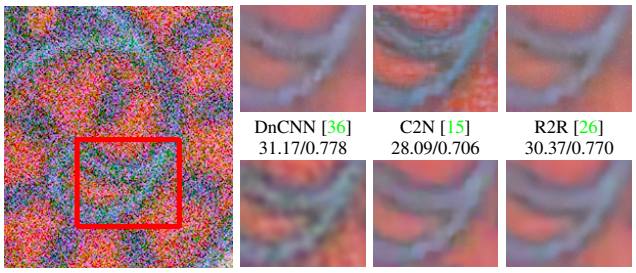


Figure 1. Visual comparison of various methods on the SIDD validation [1] dataset. Compared with DnCNN [36], C2N [15] and R2R [15], LG-BPN can be trained in a self-supervised manner without extra data. CVF-SID [25] still contains noise in the output, and AP-BSN [20] suffers from the loss of details.

Unfortunately, learning-based methods often rely on massive labeled image pairs for training [2, 34, 35]. This can not be simply addressed by synthesizing additive white Gaussian noise (AWGN) pairs, since the gap between AWGN and real noise distribution severely degrades their performance in the real world [2, 12]. To this end, several attempts have been made for collecting real-world datasets [1, 4]. Nonetheless, its application is still hindered by the rigorously-controlled and labor-intensive collection procedure. For instance, capturing ground truth images requires long exposure or multiple shots, which is unavailable in complex situations, *e.g.*, dynamic scenes with motion.

To alleviate the constraint of the large-scale paired dataset, methods without the need for ground truth have attracted increasing attention. The pioneer work Noise2Noise (N2N) [21] uses paired noisy observations for training, which can be applied when clean images are not available. Still, obtaining such noisy pairs under the same scene is less feasible. To make self-supervised methods more practical, researchers seek to learn from one, instead of pairs of observations. Among these methods, blind-spot networks (BSNs) [3, 17, 19, 30] show significant advances to restore clean pixels by utilizing neighbor pixels, with a special blind spot receptive field requirement. Despite their promising results on simple noise such as AWGN, these methods

usually work under simplified assumptions, *e.g.*, the noise is pixel-wise independent. This obviously does not hold for real noise, where the distribution can be extremely complex and present a strong spatial correlation.

Accordingly, a few methods have been proposed for self-supervised real noise removal. Recorrupted-to-Recorrupted (R2R) [26] tries to construct noisy-noisy pairs, while it can not be directly applied without extra information, which is not practical in real situations. CVF-SID [25] disentangles the noise components from noisy images, but it assumes the real noise is spatially invariant and ignores the spatial correlation, which contradicts real noise distribution.

Recently, AP-BSN [20] combines pixel-shuffle downsampling (PD) with the blind spot network (BSN). Though PD can be utilized to meet the noise assumption of BSN, simply combining PD with CNN-based BSN is sub-optimal for dealing with spatially-correlated real noise. It causes damage to local details, thus bringing artifacts to the sub-sampled images, *e.g.*, aliasing artifact, especially for large PD stride factors [20, 38]. Also, though more advanced designs of BSNs have been proposed [18, 19, 31], CNN-based BSNs fail to capture long-range interactions due to their convolution operator, which is further bounded by the limited receptive field under the blind spot requirement.

In this paper, we present a novel method, called LG-BPN, to address these issues on self-supervised real image denoising, including the reliance on extra information, the loss of local structures by noise correlation, and also the lacking of modeling distant pixel interaction. LG-BPN can be directly trained without external information. Furthermore, we ease the destruction of fine textures by carefully considering the spatial correlation in real noise, at the same time injecting long-range interaction by tailoring Transformers to the blind spot network. First, for local information, we introduce a densely-sampled patch-masked convolution (DSPMC) module. Based on the prior statistic of real noise spatial correlation, we take more neighbor pixels into account with a denser receptive field, allowing the network to recover more detailed structures. Second, for global information, we introduce a dilated Transformer block (DTB). Under the special blind spot requirement, this greatly enlarges the receptive field compared with previous CNN-based BSNs, permitting more neighbors to be utilized when predicting the central blind spot pixel. These two designs enable us to fully exploit local and global information, respectively. Extensive studies demonstrate that LG-BPN outperforms other state-of-the-art un-/self-supervised methods on real image denoising, as shown in Figure 1. We summarize our contributions as follows:

- We present a novel self-supervised method called LG-BPN for real-world image denoising, which can effectively encode both the local detailed structure and the capture of global representation.
- Based on the analysis of real noise spatial correlation, we propose DSPMC module, which takes advantage of the higher sampling density on the neighbor pixels, enabling a denser receptive field for improved local texture recovery.
- To establish long-distance dependencies in previous CNN-based BSN methods, we introduce DTB, which aggregates global context while complying with the constraint of blind spot receptive field.

2. Related Work

2.1. Supervised Image Denoising

DnCNN [36] is the first attempt to apply deep learning techniques to the image denoising task, where the training pairs are synthesized by additive white Gaussian noise (AWGN). Following DnCNN, several methods have been proposed for AWGN noise removal. For instance, FFDNet [37] advances it by taking the noise map as additional input. While achieving superior performance on AWGN removal, recent studies [2, 12] reveal the poor generalization ability of these models when applied to real noise, due to the gap between the noise distribution. The primary obstruction of real image denoising lies in the deficiency of real noisy-clean pairs. To this end, some real-world denoising datasets are collected under carefully considered conditions [1, 4]. Based on these datasets, several methods [7, 12] train the network directly on the real image pairs. Despite the decent performance, data collection can be extremely expensive and labor-intensive. Also, it is infeasible to collect clean images under complex scenes containing motion.

2.2. Unsupervised Image Denoising

Another line of research focuses on the situation where paired data is unavailable, including *i)* generating pseudo noisy-clean image pairs, *ii)* generating pseudo noisy-noisy image pairs, and *iii)* training directly on noisy images.

Generating pseudo noisy-clean image pairs. In situations where unpaired noisy-clean data is available, generation-based methods seek to synthesize real noise on clean images for aligned training data, which can be used for supervised methods. Inspired by the generative adversarial network (GAN), GCBD [6] synthesizes realistic noisy images to train the denoising network, while its performance is limited by the inaccurate consideration of noise components. UIDNet [31] takes a step further by combining the distilled knowledge of the self-supervised denoising network and extra information from synthetic pairs. C2N [15] considers various noise components in real-world scenarios for more accurate noise synthesis. However, dealing with the gap between unpaired data is still challenging. The mismatch in scene distribution can result in inaccurate generation, thus degrading the quality of synthesized data.

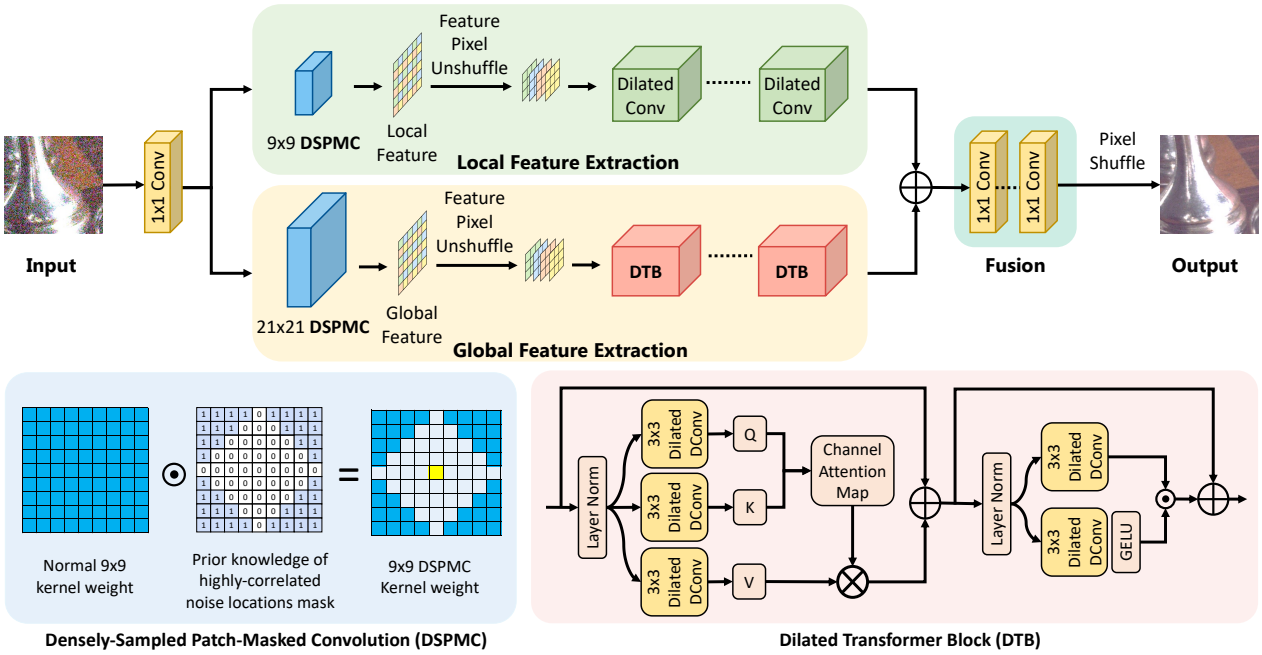


Figure 2. The overall architecture of LG-BPN. Our method is composed of two branches, aiming at extracting local textures and global interactions, respectively. For each branch, the input first goes through a DSPMC module, then further processed for the deep feature. Finally, the output of two branches is fused for the denoised result. [1]

Generating pseudo noisy-noisy image pairs. The semi-supervised method Noise2Noise [21] uses multiple noisy images for training, which can be applied without clean images. However, the acquisition of multiple independent observations under the same scene is still less practical. Therefore, several methods seek to construct noisy-noisy pairs from a single noisy image. Neighbor2Neighbor [14] generates two sub-sampled images under simplified noise assumptions. To handle complex noise distribution in real images, several methods have been proposed, including Noisier2Noise [24], NAC [33] and R2R [26]. Still, these methods either require prior knowledge or are limited by specific constraints, which can be impractical in real situations. Specifically, Noisier2Noise [24] requires noise distribution when synthesizing noisy/noisy pairs. NAC [33] works under the assumption that the noise level is relatively weak. R2R [26] also uses additional information, *e.g.*, noise level function (NLF) and image signal processing (ISP) function.

Training directly on noisy images. Another type of method follows a self-supervised manner, which can be directly trained on the noisy images and free of synthesizing pseudo image pairs. Noise2Void [17] and Noise2Self [3] propose the self-supervised blind-spot strategy by masking the corresponding central pixel. Laine19 [19] and D-BSN [31] are further proposed for advanced BSN designs, while the convolution-based architecture limits their exploitation for long-range dependencies. To ease the information loss by the blind spot, Blind2Unblind [30] introduces a novel re-visible loss term. Unfortunately, the above-mentioned methods work under the assumption that noise is pixel-wise independent, thus inevitably learning identity mapping under spatially-correlated real noise. Towards real image denoising, CVF-SID [25] disentangles the noise components from the clean images, but it assumes the noise is spatially-uncorrelated, which does not match the real noise distribution. Asymmetric pixel shuffle downsampling BSN (AP-BSN) [20] combines the pixel shuffle downsampling (PD) with a CNN-based BSN [31]. While achieving promising results, local structures are damaged by directly applying the PD operation to the image. Sub-sampled images are corrupted by various artifacts, *e.g.*, aliasing artifacts, which are more pronounced under a large PD stride factor [20, 38]. Also, adopting the CNN-based BSN leads to a limited receptive field. Since BSN recovers the central pixels based on its neighbors, fewer available neighbor pixels inevitably lead to performance loss. In summary, this results in the inadequate utilization of information with respect to both local and global contexts. Instead, our method benefits from a denser sampling density for improved local detail extraction, and also enjoys the distant pixel modeling ability for the enlarged receptive field.

3. Method

We first illustrate the overall architecture of LG-BPN in Figure 2, then elaborate on our motivation, and demonstrate our two core designs: DSPMC and DTB.

[1] We use 'global' to differentiate from our 'local' branch. Though a more accurate term is 'non-local', we follow the usage of 'global' as [35].

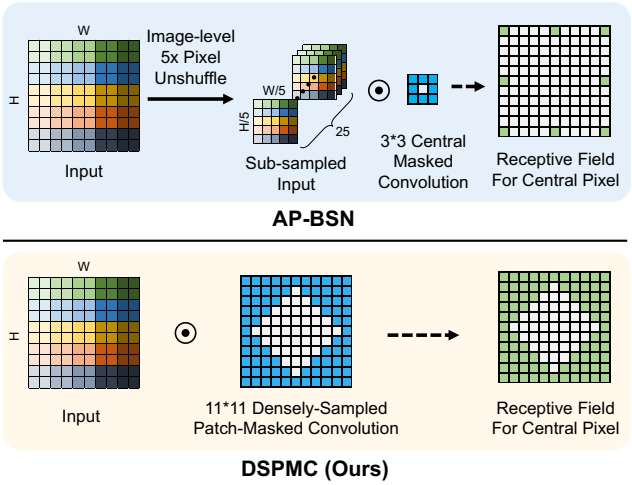


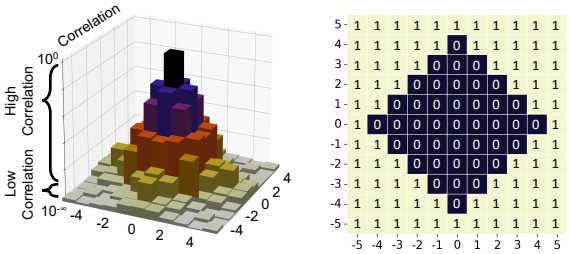
Figure 3. Comparison of receptive field for the central pixel of AP-BSN and our method. Green pixels contribute to the restoration of central pixel. Blue pixels represent convolution kernel. We realize a denser receptive field by utilizing more neighbor pixels. [2]

3.1. Motivation and Modeling

Despite the decent results on simple synthetic noise removal, the performance of self-supervised denoising methods declines significantly when dealing with real noise, due to its strong spatial noise correlation. This easily breaks the assumption on which most state-of-the-art methods are based, *i.e.*, noise is pixel-wise independent. These methods assume the clean signal of the central pixel is dependent on neighbors, while the noise is independent instead. Thus under real scenarios, they inevitably misinterpret the spatially-dependent noise as clean signals, and fail to recover the underlying clean images. Consequently, careful consideration of the spatial correlation is a must for self-supervised real noise removal. While the existing methods either struggle with poor results when completely ignoring this correlation, or suffer details loss from ill consideration. For example, AP-BSN [20] meets the assumption of the powerful BSN by adopting PD on the input image. As shown in Figure 3, though this breaks the spatial correlation, the sampling density is dramatically decreased by the PD. This severely degrades the extraction of fine details based on the Nyquist-Shannon sampling theorem, *i.e.*, the fidelity of the results shows a positive correlation to sampling density.

Besides, though BSN is already adapted for state-of-the-art performance [20], its potential is still heavily hampered by their inherent shortage, *i.e.*, the limit on the receptive field by the masked pixels for avoiding identity mapping. Despite the recent advanced BSN designs [19, 31], CNN-based BSNs are still unable to fully address this issue due to their local convolution operator and fail to model the long-distance dependencies. This adversely affects the performance of BSN, as the number of neighbors around the blind spot used for inferring is dramatically reduced.

[2]We adopt 9×9 DSPMC in the local branch. Here, the 11×11 size is shown for illustration purposes to better compare the receptive field.



(a) Bar plot of correlation by relative position

(b) High-correlation pixels mask

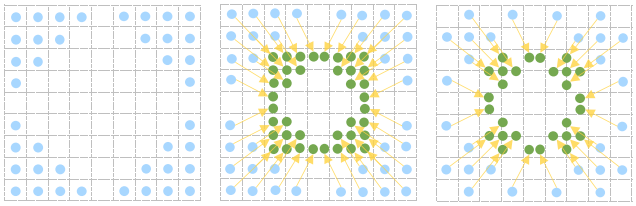
Figure 4. Visualization of spatial correlation in real noise. (a) Bar plot of spatial correlation calculated by the correlation coefficient. Note that we scale the height by log norm for better visualization. The higher bar indicates a stronger correlation. (b) The mask of high-correlation pixels. Locations with 0 mean this pixel is strongly correlated to the central pixel, while 1 means not.

We aim to tackle both of these challenges in our methods. First, we realize better extraction for detailed textures from the perspective of sampling density. As shown in Figure 3, DSPMC enables a denser sampling rate by leveraging more neighbor pixels, which raises the upper limit of reconstruction quality. Second, by tailoring normal Transformers to a blind fashion, DTB is introduced for its powerful global modeling ability to compensate for the limited receptive field of CNN-based BSNs.

Based on these two modules, we now introduce the overall architecture of our method. As shown in Figure 2, LG-BPN is mainly composed of two branches in parallel, aiming at local and global contexts reconstruction respectively. For the local feature extraction branch, we first apply the 9×9 DSPMC module. The densely extracted features are then down-sampled to break the spatial correlation. Then, the feature maps go through dilated convolution with a dilation of 2. For the global branch, the input image first goes through a 21×21 DMPMC module with a larger receptive field, which is then processed by DTBs. Finally, the local and global information from the two branches is fused together for the final output.

3.2. Densely-Sampled Patch-Masked Convolution

Neither adopting a low sample rate nor sampling all neighbor pixels can be an optimal choice for BSN when tacking real noise. In DSPMC, we aim to extract as much local information as possible, at the same time avoiding misinterpretation by these strongly-correlated neighbor pixels. To this end, we start by presenting the relationship between spatial correlation and the relative position, as shown in Figure 4. Following previous works [20, 38], we use Pearson's correlation coefficient to depict the relationship. Specifically, we first obtain the noise map by subtracting the clean images from the noisy images in SIDD medium [1]



(a) Kernel location during training (b) Shifted kernel location during testing (c) Shifted kernel location during testing with dilation

Figure 5. Illustration of shifting kernel strategy and dilation of DSPMC. The blue and green points are the kernel locations during training and testing respectively. Yellow arrows indicate the shift direction. This avoids the strongly correlated pixels when training, sampling pixels closer to the center when testing for more details.

dataset. Then, correlation coefficient can be calculated by:

$$\rho_{N_{cen},N_{nei}} = \frac{\mathrm{cov}(N_{cen}, N_{nei})}{\sigma_{N_{cen}}\sigma_{N_{nei}}}, \tag{1}$$

where N_{cen} and N_{cen} represent the noise of the central pixel and neighbor pixels respectively. In Figure 4(a), we find that there exist more neighbor pixels which can also be leveraged for prediction. These pixels are not strongly correlated to the center pixel, thus bringing useful information, instead of misinterpretation to BSN. Then, the DSPMC kernel can be calculated as:

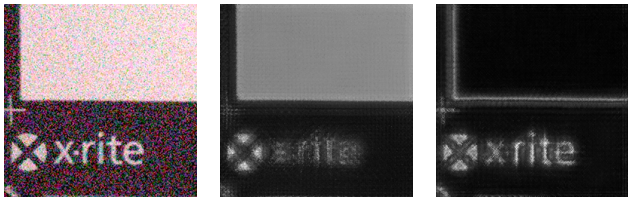
$$\mathbf{K}_{DSPMC} = \mathbf{K}_n \odot \mathbf{Mask}_{cor}, \tag{2}$$

where $\mathbf{K}_{DSPMC}$ is the kernel of DSPMC, $\mathbf{K}_n$ is the kernel of the normal convolution, and $\mathbf{Mask}_{cor}$ is the mask for filtering out highly-correlated pixels shown in Figure 4(b). By integrating this noise distribution prior to sampling locations, our module can effectively take more neighbor pixels while avoiding the strongly correlated pixels. This enables the extracted feature to contain more fine details, and a denser receptive field as well. Also, as shown in Figure 3, since the extracted high-dimension feature already gathers the rich local details, the subsequent feature-level PD can save more useful information compared with the previous image-level PD.

However, directly applying this module is not an optimal choice: *i)* the inference stage requires more details compared with training, so directly using the same architecture can damage high-frequency details [20], *ii)* a large kernel can cause computational inefficiency.

Kernel shift strategy. As shown in Figure 5(b), for the first concern, we need to obtain more details while testing, focusing on local detailed information closer to the center pixel. Inspired by the deformable convolution [9], we apply a set of fixed offsets on the kernel for each location, which enforce that the kernels are more gathered in the center:

$$\begin{aligned} \mathbf{y}(p_0) &= \sum_{k=1}^{\mathbf{K}} w_k \cdot \mathbf{x}\left(p_0 + p_k + \Delta p_k\right), \\ \Delta p_k &= Ratio * (p_k - p_0), \end{aligned} \tag{3}$$



(a) Noisy input (b) AP-BSN [20] (c) LG-BPN (Ours)

Figure 6. The comparison of the feature map visualization of our method and AP-BSN [20] in the training phase. Our feature map shows clearer edges, validating the superiority of local details extraction of DSPMC by imposing denser sampling locations.

where $\mathbf{K}$ is the kernel sampling locations, w_k is the kernel weight, $\mathbf{x}(p)$ and $\mathbf{y}(p)$ denote the features at p in input feature $\mathbf{x}$ and output feature $\mathbf{y}$, and Δp is the applied kernel offset. $Ratio$ is the extent we shift the kernel while testing. By adding offsets to the kernel, we can *shrink* the kernel and capture finer details while testing.

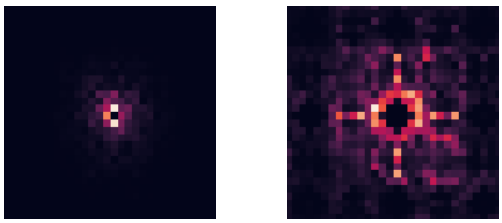
Dilation in DSPMC. As shown in Figure 5(c), for the second concern, we further decrease the computational cost by adding dilation to the convolution kernel. This imposed sparsity makes our DSPMC computationally efficient especially for large kernel size, at the same time maintaining the dense receptive field for capturing detailed structures.

Visualization of the extracted feature map. To validate that our DSPMC module achieves a denser receptive field and is thus better at extracting local high-frequency structures, we present the visualization of the feature map. We select the output feature of the local extraction branch, and the corresponding location in AP-BSN. All channels are averaged and normalized for visualization. As shown in Figure 6, in AP-BSN [20], the local fine texture is damaged due to the insufficient use of neighbor signals. Instead, by leveraging more neighbor pixels, our feature map shows shaper edges and preserves more details.

3.3. Dilated Transformer Block

The receptive field of BSN is restricted by the imposed blind spots, while the local operator in CNN-based BSNs further prevents it from gathering global interaction. However, under the special blind spot constraint on the receptive field, it is non-trivial to directly introduce normal Transformer blocks to the BSN. Inspired by the D-BSN [31], we aim to design the Transformer block without information exchange between spatially-adjacent pixels, which satisfies the blind spot requirement when combined with DSPMC. Under these requirements, we carefully consider the design of two core components in the Transformer: the self-attention calculation and the feed-forward layer.

First, for the self-attention layer, spatial-wise attention enables spatial information exchange and thus does not meet our receptive field requirement. Recently, a grid-like self-attention can meet our requirement [28], while the grid pattern further narrows the receptive field that the blind spot



(a) AP-BSN [20] (b) LG-BPN (Ours)

Figure 7. The comparison of the receptive field between AP-BSN [20] and our method. We calculate the gradient response to the central pixel. The brighter color represents the higher contribution for recovering the central pixel.

has reduced. Instead, we adopt channel-wise attention [35] for its unawareness of spatial location and global perception as well. Furthermore, to enhance the local context while preventing information of adjacent pixels, we introduce dilated depth-wise convolution before computing feature similarity. For the input feature $\mathbf{X}$, the Query ($\mathbf{Q}$), Key ($\mathbf{K}$) and Value ($\mathbf{V}$) matrix is thus calculated by $\mathbf{Q} = g^Q(\mathbf{X})$, $\mathbf{K} = g^K(\mathbf{X})$, $\mathbf{V} = g^V(\mathbf{X})$, where $g^Q(\cdot)$, $g^K(\cdot)$ and $g^V(\cdot)$ denote the dilated 3×3 depth-wise convolution. Given the $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ matrix, the channel interaction can be obtained by the dot-product, where the attention map is of size $\mathbb{R}^{C\times C}$, and C is the number of channels. The overall self-attention layer is represented as:

$$\begin{aligned} \text{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) =& \mathbf{V}\,\text{Softmax}(\mathbf{K}\mathbf{Q}), \\ \hat{\mathbf{X}} =& \text{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) + \mathbf{X}, \end{aligned} \tag{4}$$

where $\mathbf{X}$ and $\hat{\mathbf{X}}$ denote the input and output features.

Second, for the feed-forward layer, adjacent information exchange can be simply avoided by adopting 1×1 convolutions only. Nonetheless, this fails to capture local context, which can be critical for restoring high-frequency details. We address this issue by also introducing dilation into the normal 3×3 convolution in the feed-forward layer. Then, features extracted by dilated depth-wise convolution go through a gating unit for the non-linearity. This gating unit is the element-wise product of two parallel paths, with one of them activated by the GELU unit. The overall process of the feed-forward layer is formulated as:

$$\begin{aligned} \mathbf{G^1} &= g^1(\text{LN}(\mathbf{X})), \\ \mathbf{G^2} &= g^2(\text{LN}(\mathbf{X})), \\ \hat{\mathbf{X}} &= \text{GELU}\left(\mathbf{G^1}\right) \odot \mathbf{G^2} + \mathbf{X}, \end{aligned} \tag{5}$$

where $\odot$ is element-wise multiplication, LN denotes layer normalization, $g^1(\cdot)$ and $g^2(\cdot)$ represent the 3×3 dilated depth-wise convolution. An additional benefit is that, compared to the normal 3×3 convolution, the introduced dilation can also enlarge the receptive field.

To prove the effectiveness of the introduced global dependencies, we also plot the receptive field for recovering the central pixel of our method and the CNN-based BSNs in Figure 7. By injecting the long-range interaction into the blind spot network, more neighbor pixels are activated for restoring the central pixels in our method, offering a broader receptive field compared to the previous CNN-based BSNs.

4. Experiments

4.1. Dataset and Setup Details

We train and evaluate our method on two real-world datasets, *i.e.*, SIDD [1] and DND [27]. Note that for the SIDD benchmark dataset and DND benchmark dataset, we submit the output to the website for online evaluation.

Smartphone Image Denoising Dataset (SIDD) [1] contains paired images for real-world denoising by five smartphone cameras. For training, we use the sRGB images from SIDD-Medium including 320 pairs. For validation and evaluation, we use the sRGB images from the SIDD validation set and benchmark set respectively. Each includes 1280 patches of size 256×256, where the ground truth images are also provided for the validation set.

Darmstadt Noise Dataset (DND) [27] contains 50 noisy images for benchmarking without the ground truth provided, and the results can only be obtained via the online submission system. Therefore, we enjoy a fully self-supervised manner and directly train our method on the test set without extra eternal data.

4.2. Training Details

During training, we keep the same setting as the previous work [20]. Specifically, a batch size of 8 is used in the experiment. We adopt $\mathcal{L}_1$ loss between ground truth and output for training. The learning rate starts with $1e$-4, where Adam optimizer is adopted. The network is trained with 20 epochs until it fully converges. We implement the method in PyTorch 1.8.0, and train our model on the Nvidia RTX 3090. Two metrics are utilized to evaluate the performance of methods, including peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [29]. The larger value of PSNR and SSIM implies better fidelity.

4.3. Evaluation of Real-world Denoising

We validate the effectiveness of our method for real-world image denoising on the commonly-used SIDD benchmark dataset and DND benchmark dataset. Table 1 shows the comparison of various methods on SIDD and DND benchmark datasets. Visualization results of several methods addressed in Table 1 on SIDD and DND datasets can be found in Figure 8 and Figure 9. We achieve better results in quantitive and qualitative metrics than previous un-/self-supervised methods. Compared with unsupervised methods trained on unpaired clean-noisy data, LG-BPN does not rely on extra data for synthesizing training pairs, also avoiding the misalignment between the scene distribution. For the

Type of supervision	Training data	Method	SIDD		DND	
			PSNR	SSIM	PSNR	SSIM
Non-learning based	None	BM3D [8]	25.65	0.685	34.51	0.851
		WNNM [11]	25.78	0.809	34.67	0.865
Supervised	Synthesized Pairs	DnCNN [36]	23.66	0.583	32.43	0.790
		CBDNet [12]	33.28	0.868	38.05	0.942
		AWGN-M [38]	33.99*	0.896*	38.40	0.945
	Real pairs	DnCNN [36]	35.34*	0.885*	37.83*	0.929*
		AINDNet [16]	38.84	0.951	39.34	0.952
		RIDNet [2]	38.70	0.950	39.25	0.952
		DIDN [34]	39.82	0.973	39.62	0.954
Unsupervised	Noisy-clean pairs	GCBD [6]	-	-	35.58	0.922
		UIDNet [13]	32.48	0.897	-	-
		C2N [15] + DIDN [34]	35.35	0.937	36.38	0.887
		D-BSN [31] + MWCNN [23]	-	-	37.93	0.937
	Noisy-noisy pairs	Noise2Self [3]	29.56†	0.808†	-	-
		NAC [33]	-	-	36.20	0.925
		R2R [26]	34.78	0.898	-	-
	Single noisy observation	Noise2Void [17]	27.68†	0.668†	-	-
		CVF-SID [25]	34.71	0.917	36.50	0.924
		AP-BSN [20]	35.97	0.925	38.09	0.937
		LG-BPN (Ours)	**37.28**	**0.936**	**38.43**	**0.942**

Table 1. Quantitative comparison of various methods on SIDD and DND benchmark datasets. Though several supervised methods achieve better results using noisy/clean image pairs, our methods use noisy RGB images only. Results with $\star$ mean these are reproduced and evaluated by ourselves, since they are not evaluated on the dataset we use in their original paper. The results marked with † are reported from R2R [26]. Otherwise, we report the official results from SIDD and DND benchmark websites.

self-supervised methods, NAC [33] works under the weak noise level assumption, while R2R [26] can not be directly applied to sRGB images without extra NLF and ISP functions, both of which harm their performance in real-world situations. In contrast, LG-BPN can be directly applied and not restricted by these assumptions. For methods leveraging single noisy observations, CVF-SID [25] does not consider the strong spatial correlation property in real noise, thus real noise can not be fully removed as shown in Figure 8. AP-BSN [20] suffers from inadequate sample locations with a limited receptive field, so details are blurred as shown in Figure 9. Instead, LG-BPN carefully integrates the spatial correlation into the network design, simultaneously modeling distant context dependencies.

4.4. Analysis of the Proposed Method

Dilation factor in densely-sampled convolution. Directly introducing densely-sample convolution can be computationally expensive. To balance the efficiency and the performance, we further introduce the sparsity to the convolution kernel by adding dilation to the original DSPMC. Figure 5(c) shows the illustration of the dilation.

To explore the better trade-off between performance and efficiency, we provide ablation studies on the dilation rate for our DSPMC in both local and global extraction branches. As shown in Table 2(a), a dilation of 1 for 9×9 DSPMC and 2 for 21×21 DSPMC achieve a better balance. We claim its reason is that the difference in kernel size results in focusing on varied scales of information. This imposes different sensitivity to the dilation, *i.e.*, sampling density. For relatively small 9×9 kernels, it focuses more on the local textures, thus adding dilation can notably lower its ability when reconstructing detailed structures. While for the 21×21 kernel, the larger kernel size makes it aim at the global context more. Thus, the introduced dilation does not severely harm its global extraction ability, at the same time reducing the computational cost.

The exploitation of local and global information. LG-BPN consists of two branches in parallel. Specifically, we combine the small 9×9 DSPMC with dilated convolution and the large 21×21 DSPMC with DTB, focusing on local and global context processing respectively. To validate the reasonableness of our network architecture, we conduct ablation studies on these components. Table 2(b) shows the results of different combinations.

It can be seen that removing the DSPMC of size 9×9 and 21×21 in either branch severely degrades the performance. It is on account of the insufficient sampling density, which limits the reconstruction quality according to the Nyquist-Shannon sampling theorem. In this situation, the sampling density is made severely sparse, which causes insufficient utilization of input signals. This performance drop demonstrates the effectiveness of our DSPMC module.

Furthermore, we validate that the different sizes of

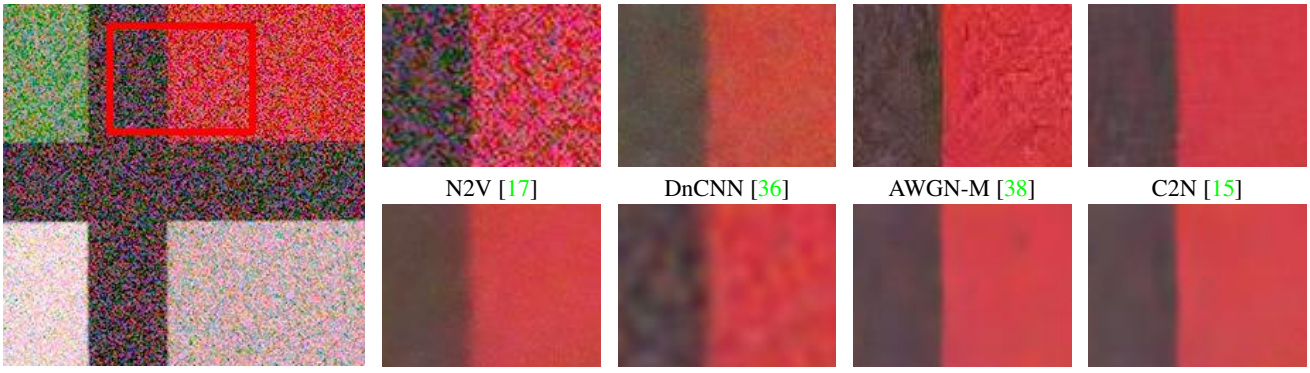


Figure 8. Visual quality comparison on SIDD benchmark dataset. Note that the quantitative results are not available.

9x9 DSPMC dilation	21x21 DSPMC dilation	PSNR	SSIM	FLOPS (G)	Params (M)
1	1	37.23	0.885	88.6	1.35
1	**2**	**37.32**	**0.886**	**29.8**	**0.45**
1	3	37.23	0.883	17.1	0.26
2	1	36.85	0.879	84.6	1.23
2	2	36.84	0.875	25.8	0.39
2	3	36.99	0.873	13.1	0.20

(a) Ablation studies on dilation rates. Different dilation rate combinations are explored on the DSPMC in the local branch and global branch.

Method	PSNR	SSIM
w/o 9×9 DSPMC	35.82	0.855
w/o 21×21 DSPMC	36.21	0.869
Replacing Conv with DTB	36.90	0.880
Replacing DTB with Conv	36.82	0.875
w/o kernel shift	35.64	0.857
LG-BPN (Ours)	**37.32**	**0.886**

(b) Ablation studies on our proposed method. Improvements can be found with our proposed modules and network design.

Table 2. The analysis of our method on the SIDD validation dataset. Experimental results prove the effectiveness of our method design.

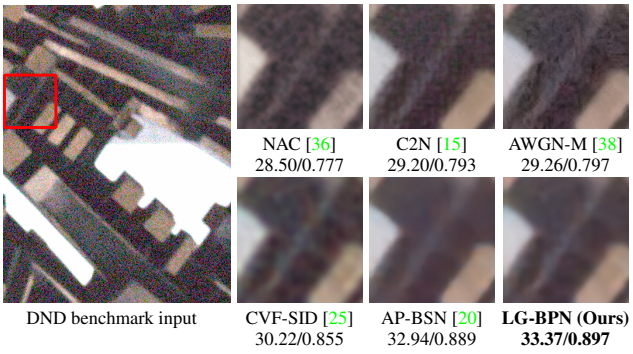


Figure 9. Visual quality comparison on DND benchmark dataset.

DSPMC in local and global feature extraction branches make them focus on various scales of features. Such discrepancy across the scale demands us to treat scale-specific characteristics in a different way. When replacing the dilated convolution in the local branch with our DTB, the lack of local connectivity brings inferior performance. Similarly, when replacing the DTB in the global branch with the dilated convolution, the network is built by convolutions only. It induces a lack of long-range interaction, which greatly limits their recovery quality as well. This proves the superior design of our architecture. By exploiting the locality of convolution with a small DSPMC, also the global dependencies of DTB with a large DSPMC, our architecture enjoys more reasonable exploitation for multi-scale context.

The effect of kernel shift strategy. Since the requirement for pixel-wise independent noise is different in training and testing, directly applying the same kernel of the training phase while testing will lose image details [20]. We proposed a kernel shift strategy, as illustrated in Figure 5(b). In Table 2(b), the lack of kernel shift causes 1.68 dB drops, proving the effectiveness of our shifting paradigm.

5. Conclusion

In this paper, we propose LG-BPN for self-supervised real image denoising, aiming to address the details lost by the coarse consideration for real noise correlation, and the lack of global interaction by the inherent constraint on the receptive field for BSN. First, we propose DSPMC to fully preserve the local structures. Owing to a denser receptive field, we ease the destruction of fine textures and can thus better reconstruct details. Second, we propose DTB, injecting distant interactions into the previously CNN-based blind spot networks. Since blind spot networks rely on neighbor signals for predicting, more clues can be provided by activating more neighbor pixels. Extensive results on real-world datasets reveal the superior performance of LG-BPN.

Acknowledgement This work was supported by the National Natural Science Foundation of China under Grants No. 62171038, No. 61827901, No. 62088101, and No. 62006023.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *CVPR*, pages 1692–1700, 2018. 1, 2, 4, 6

[2] Saeed Anwar and Nick Barnes. Real image denoising with feature attention. In *ICCV*, pages 3155–3164, 2019. 1, 2, 7

[3] Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In *ICML*, pages 524–533, 2019. 1, 3, 7

[4] Benoit Brummer and Christophe De Vleeschouwer. Natural image noise dataset. In *CVPR Workshops*, pages 0–0, 2019. 1, 2

[5] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *CVPR*, pages 60–65, 2005. 1

[6] Jingwen Chen, Jiawei Chen, Hongyang Chao, and Ming Yang. Image blind denoising with generative adversarial network based noise modeling. In *CVPR*, pages 3155–3164, 2018. 2, 7

[7] Shen Cheng, Yuzhi Wang, Haibin Huang, Donghao Liu, Haoqiang Fan, and Shuaicheng Liu. Nbnet: Noise basis learning for image denoising with subspace projection. In *CVPR*, pages 4896–4906, 2021. 1, 2

[8] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE TIP*, 16(8):2080–2095, 2007. 1, 7

[9] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, pages 764–773, 2017. 5

[10] Ying Fu, Zichun Wang, Tao Zhang, and Jun Zhang. Low-light raw video denoising with a high-quality realistic motion dataset. *IEEE TMM*, 2022. Early access. 1

[11] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *CVPR*, pages 2862–2869, 2014. 1, 7

[12] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *CVPR*, pages 1712–1722, 2019. 1, 2, 7

[13] Zhiwei Hong, Xiaocheng Fan, Tao Jiang, and Jianxing Feng. End-to-end unpaired image denoising with conditional adversarial networks. In *AAAI*, pages 4140–4149, 2020. 7

[14] Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2neighbor: Self-supervised denoising from single noisy images. In *CVPR*, pages 14781–14790, 2021. 3

[15] Geonwoon Jang, Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. C2n: Practical generative noise modeling for real-world denoising. In *ICCV*, pages 2350–2359, 2021. 1, 2, 7, 8

[16] Yoonsik Kim, Jae Woong Soh, Gu Yong Park, and Nam Ik Cho. Transfer learning from synthetic to real-noise denoising with adaptive instance normalization. In *CVPR*, pages 3482–3492, 2020. 7

[17] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *CVPR*, pages 2129–2137, 2019. 1, 3, 7, 8

[18] Alexander Krull, Tomáš Vičar, Mangal Prakash, Manan Lalit, and Florian Jug. Probabilistic noise2void: Unsupervised content-aware denoising. *Frontiers in Computer Science*, 2:5, 2020. 2

[19] Samuli Laine, Tero Karras, Jaakko Lehtinen, and Timo Aila. High-quality self-supervised deep image denoising. In *NIPS*, pages 6970–6980, 2019. 1, 2, 3, 4

[20] Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. Ap-bsn: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *CVPR*, pages 17725–17734, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[21] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In *ICML*, pages 2965–2974, 2018. 1, 3

[22] Ding Liu, Bihan Wen, Jianbo Jiao, Xianming Liu, Zhangyang Wang, and Thomas S Huang. Connecting image denoising and high-level vision tasks via deep learning. *IEEE TIP*, 29:3695–3706, 2020. 1

[23] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *CVPR workshops*, pages 773–782, 2018. 7

[24] Nick Moran, Dan Schmidt, Yu Zhong, and Patrick Coady. Noisier2noise: Learning to denoise from unpaired noisy data. In *CVPR*, pages 12064–12072, 2020. 3

[25] Reyhaneh Neshatavar, Mohsen Yavartanoo, Sanghyun Son, and Kyoung Mu Lee. Cvf-sid: Cyclic multi-variate function for self-supervised image denoising by disentangling noise from image. In *CVPR*, pages 17583–17591, 2022. 1, 2, 3, 7, 8

[26] Tongyao Pang, Huan Zheng, Yuhui Quan, and Hui Ji. Recorrupted-to-recorrupted: unsupervised deep learning for image denoising. In *CVPR*, pages 2043–2052, 2021. 1, 2, 3, 7, 8

[27] Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *CVPR*, pages 1586–1595, 2017. 6

[28] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. *ECCV*, 2022. 5

[29] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4):600–612, 2004. 6

[30] Zejin Wang, Jiazheng Liu, Guoqing Li, and Hua Han. Blind2unblind: Self-supervised image denoising with visible blind spots. In *CVPR*, pages 2027–2036, 2022. 1, 3

[31] Xiaohe Wu, Ming Liu, Yue Cao, Dongwei Ren, and Wangmeng Zuo. Unpaired learning of deep image denoising. In *ECCV*, pages 352–368, 2020. 2, 3, 4, 5, 7

[32] Jun Xie, Rogerio Schmidt Feris, Shiaw-Shian Yu, and Ming-Ting Sun. Joint super resolution and denoising from a single depth image. *IEEE TMM*, 17(9):1525–1537, 2015. 1

[33] Jun Xu, Yuan Huang, Ming-Ming Cheng, Li Liu, Fan Zhu, Zhou Xu, and Ling Shao. Noisy-as-clean: Learning self-supervised denoising from corrupted image. *IEEE TIP*, 29:9316–9329, 2020. 3, 7

[34] Songhyun Yu, Bumjun Park, and Jechang Jeong. Deep iterative down-up cnn for image denoising. In *CVPR Workshops*, pages 0–0, 2019. 1, 7

[35] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, pages 5728–5739, 2022. 1, 3, 6

[36] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE TIP*, 26(7):3142–3155, 2017. 1, 2, 7, 8

[37] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE TIP*, 27(9):4608–4622, 2018. 2

[38] Yuqian Zhou, Jianbo Jiao, Haibin Huang, Yang Wang, Jue Wang, Honghui Shi, and Thomas Huang. When awgn-based denoiser meets real noises. In *AAAI*, pages 13074–13081, 2020. 2, 3, 4, 7, 8