DRCNet: Dynamic Image Restoration Contrastive Network

Fei Li[1] Lingfeng Shen[2] Yang Mi[1] Zhenbo Li[1]
[1]College of Information and Electrical Engineering, China Agricultural University
[2]Tecent AI Lab

Abstract

*Image restoration aims to recover images from spatially-varying degradation. Most existing image-restoration models employed static CNN-based models, where the fixed learned filters cannot fit the diverse degradation well. To address this, in this paper, we propose a novel **D**ynamic Image **R**estoration **C**ontrastive **Net**work (DRCNet). The principal block in DRCNet is the **D**ynamic **F**ilter **R**estoration module (DFR), which mainly consists of the spatial filter branch and the energy-based attention branch. Specifically, the spatial filter branch suppresses spatial noise for varying spatial degradation; The energy-based attention branch guides the feature integration for better spatial detail recovery. To make degraded images and clean images more distinctive in the representation space, we develop a novel **Intra**-class **C**ontrastive **R**egularization (Intra-CR) to serve as a constraint in the solution space for DRCNet. Meanwhile, our theoretical derivation proved Intra-CR owns less sensitivity towards hyper-parameter selection than previous CR. DRCNet achieves state-of-the-art results on the ten widely-used benchmarks in image restoration. Besides, we conduct ablation studies to show the effectiveness of the DFR module and Intra-CR, respectively.*

1. Introduction

Image restoration is one of the basic tasks in computer vision, which recovers clean images from degraded version, typically caused by rain [15], noise [19] and blur [30]. It is imperative to restore such degraded images to improve their visual quality. Among the models for image restoration, most of the achieved progress [13, 32, 38] is primarily attributed to static CNN [17,22]. However, the image degradation is spatially varying [43], which is incompatible with static CNN that are in a filter sharing manner across spatial domains [8].

Therefore, static CNN-based approaches perform imperfectly when the input image contains noise pixels, as well as severe intensity distortions in different spatial regions [43]. Conceptually, static CNN-based [20, 23] models have some drawbacks. First, it is relatively inefficient in image restoration tasks [43, 88]. Second, the fixed learned filters can not automatically fit the diverse input degraded images [23, 65]. Considering the limitations mentioned above, we need to design a module to dynamically restore the degraded images since each input image has a variable degree of distortion and specific spatial distribution.

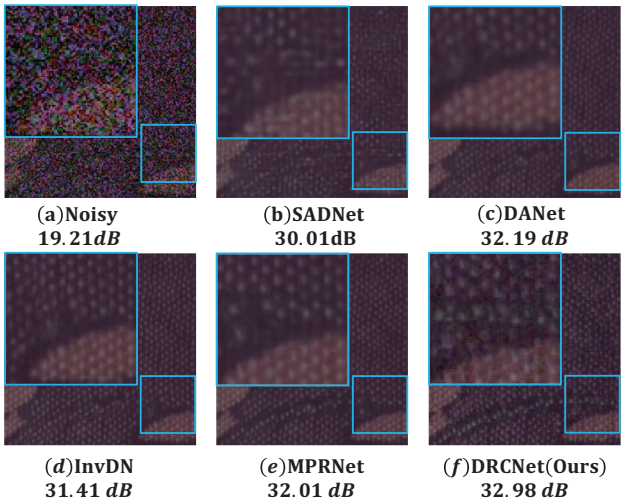

Figure 1. A real noisy image from the SIDD [1] dataset. The competing algorithms, such as SADNet [7], DANet [72], InvDN [34] and MPRNet [74], fail to produce the blotchy texture details and sacrifice the structural contents. In contrast, DRCNet can better preserve the textures with fewer artifacts.

Recently, some efforts [8, 65, 83, 88] have been made to compensate for the drawbacks of static convolution, enabling the model to flexibly adjust the structure and parameters to be suitable for diverse task demands. Few works [43] have employed dynamic convolution for region-level restoration, which may not effectively reconstruct the fine-grained pixels.

To solve this, we propose a new model called **D**ynamic Image **R**estoration **C**ontrastive **Net** (DRCNet) consists of two key components: Dynamic Filter Restoration module (DFR) and Intra-Class Contrastive Regularization (Intra-CR). The core component of DRCNet is DFR, which effectively restore the pixel-level spatial details by using the

dynamic mask to suppress spatial noise and applies feature integration. Specifically, there are two principal designs in DFR. One is a spatial filter branch, which masks the noise pixels and applies adaptive feature normalization. The other one is the energy-based attention branch, which is designed to calibrate features dynamically. Moreover, to make degraded images and clean images more distinctive in the representation space, we propose a new contrastive regularization called Intra-CR, serving as a constraint in the solution space. Specifically, Intra-CR constructs negative samples through mixup [75] while existing CR construct negative samples by random sampling. Its effectiveness is validated through theoretical derivation and empirical studies.

Fig. 1 shows the visualization comparison on the SIDD [1] test dataset in terms of image restoration quality. Our main contributions are summarized as follows:

- We propose a Dynamic Filter Restoration module **(DFR)** that is adaptive in various image restoration scenes. Such a block enables DRCNet to handle spatial-varying image degradation.
- We propose a novel contrastive regularization called **Intra-CR**, which constructs intra-class negative samples through mixup. Empirical studies show its superiority over vanilla contrastive regularization, and our theoretical results show that Intra-CR is less sensitive to hyper-parameter selection.
- Extensive experiments demonstrate the effectiveness of our DRCNet and show that our model achieves state-of-the-art results on 10 synthetic and real-world datasets for across three restoration tasks.

2. Related Work

2.1. Image Restoration

Early image restoration approaches are based on prior-based models [6, 22, 64], sparse models [12, 37], and physical models [5, 41]. Recently, the significant performance improvements in image restoration can be attributed to the architecture of convolution neural networks (CNN) [17, 53]. Most CNN-based methods focus on elaborating architecture designs, such as, multi-stage networks [74, 76], dense connections [86], and neural architecture search (NAS) [18, 77]. Due to the spatial-varying image degradation, static CNN-based models are less capable than desired to handle this issue [43]. In contrast, we propose the DFR module with dynamic spatial filter and energy-based attention, which is more effective than static CNN.

The most relevant work to our work is SPAIR [43]. However, there are several principal differences between DRCNet and SPAIR. Firstly, in degraded pixel suppression, we use a dynamic mask to suppress the degraded pixel, computed at the pixel-level, thus remaining more effective in fine-grained pixels, while SPAIR constructs a region distortion-mask, which may ignore the fine-grained degraded pixels. Second, during feature integration, SPAIR applies a 1x1 sparse convolution while DRCNet constructs an energy-based attention branch to guide the feature integration by measuring the importance score for each feature. Moreover, to make degraded images and clean images more distinctive in the representation space, we propose a novel Intra-class contrastive regularization for DRCNet.

2.2. Dynamic Filter

Compared to standard convolution, the dynamic filters can make dynamic restoration towards different input features. With the key idea of adaptive inference, dynamic filters are applied in various tasks, such as object detection [57, 69, 89], image segmentation [33, 56], super-resolution [46, 62], and style transfer [50]. Dynamic filters can be divided into Scale-adaptive [85] and Spatially-adaptive filters [55]. DFR belongs to spatially-adaptive category, which can adjust filter values to suit for different input features. In particular, dynamic spatially-adaptive filter, such as DRConv [8], DynamicConv [9] and DDF [88], can automatically assign multiple filters to corresponding spatial regions. However, most of dynamic filters are not specifically designed for image restoration, thus result in imperfect performance.

2.3. Contrastive Regularization

Contrastive learning is a self-supervised representation learning paradigm [24, 35] which is based on the assumption that good representation should bring similar images closer while pushing away dissimilar ones. Most existing works often use contrastive learning in high-level vision tasks [10, 21, 31]. Recently, some works [61] have demonstrated that contrastive learning can be used as a regularization to remove the haze. Such contrastive regularization considers all other images in the batch as negative samples, which may lead to sub-optimal performance. In this paper, we propose a novel Intra-CR for image restoration. The essential distinction between Intra-CR and existing contrastive regularization is how the negative examples are constructed. Specifically, we construct negative samples by a mixup [75] operation between the clean image and its degraded version.

3. Methods

In this section, we first provide an overview of DRCNet. Then, we detail the proposed DFR module. Finally, we present our intra-class contrastive regularization.

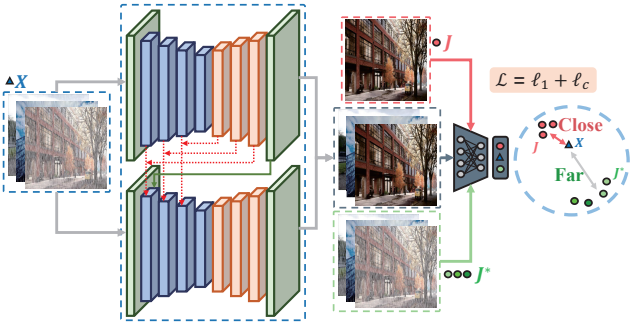

Figure 2. The architecture of the proposed DRCNet. It consists of two sub-networks and employs the encoder-decoder paradigm to restore images. To link the two sub-networks, we utilize the cross-stage feature fusion (CSFF) module and supervised attention module (SAM) from [74] denoted by red dotted lines and green line, separately. We minimize the L1 reconstruction loss (ℓ_1) with contrastive regularization (ℓ_c) to better pull the restored image (i.e. anchor, X) to the clear (i.e. positive, J) image and push the restored image to the degraded (i.e. negative, Ji^*) images.

3.1. Dynamic Filter Restoration Network

The overview of Dynamic Image Restoration Contrastive Net (DRCNet) is shown in Fig. 2, consisting of two encoder-decoder sub-networks with four down-sampling and up-sampling operations. The sub-network first adopts a 3×3 convolution to extract features. Then, the features are processed with four DFR modules for suppressing the degraded pixel in encoders. We employ three ResBlocks [23] in the decoder to generate images with fine spatial details. The restored images are obtained by using a 3×3 convolution to process the decoder output. To link the two sub-networks, we utilize the cross-stage feature fusion (CSFF) module and supervised attention module (SAM) [74] to fuse the features, which are highlighted by the red dotted line and black dotted line as illustrated in Fig. 2. Finally, we propose Intra-Class Contrastive Regularization (Intra-CR), which serves as a regularization to pull away degraded images and clean images in the representation space.

3.2. Dynamic Filter Restoration Module

The structure of DFR is shown in Fig. 3. The DFR module aims to automatically suppress potential degraded pixels and generate better spatial detail recovery with fewer parameters. Generally, it achieves such goals by constructing three different branches for inputs: (1) spatial filter branch, (2) energy-based attention branch, (3) identity branch. In spatial filter branch, we first utilize a 3×3 convolution to refine the input feature $F \in \mathbb{R}^{C_{in} \times H \times W}$ where C_{in}, H, and W denote the input channel, height, width of the feature maps. Inspired by [63], we randomly divided the feature map into two parts; one part utilizes our proposed adaptive feature normalization to mask degraded signals, the other to keep context information, then we concatenate the two parts

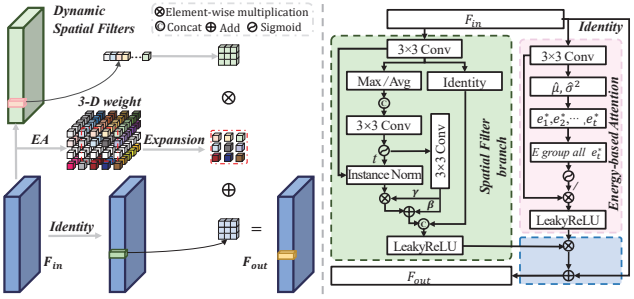

Figure 3. Illustration of the Dynamic Filter Restoration module. The green color region denotes spatial filters branch, and the pink color region denotes energy-based attention. 'Max/Avg' means max-pooling and average pooling. μ_c^i and σ_c^i denotes the mean and variance of feature. '/' means the operator as $1/E$ in Eq. (9).

to output. This operation enables DFR to suppress spatial degradation for adaptability to varying spatial degradation, which may sacrifice texture details. Therefore, we design an energy-based attention (EA) branch that focuses on generating the texture details. Besides, the EA branch guides the feature integration between the EA branch and the spatial filter branch. Finally, the identity branch launches a vanilla transformation with 1×1 convolution to the original inputs, which helps maintain the features from the original inputs. Overall, the whole DFR module can be defined as follow:

$$F'_{(r,i)} = \sum_{j \in \Omega(i)} D_i^{sp}[p_i - p_j] \mathcal{W}_i^{ea}[p_i - p_j] F_{(r,j)} \quad (1)$$

where $F'_{(r,i)}, F_{(r,j)} \in \mathbb{R}$ denotes the output/input feature value at the i^{th}, j^{th} pixel of r^{th} channel. $\Omega(i)$ denotes the k × k convolution window around i^{th} pixel. $D^{sp} \in \mathbb{R}^{h \times w \times k \times k}$ is the spatial dynamic filter with $D_i^{sp} \in \mathbb{R}^{k \times k}$ denoting the filter at i^{th} pixel. $\mathcal{W}^{ea} \in \mathbb{R}^{h \times w \times k \times k}$ is the dynamic attention weights with $\mathcal{W}_i^{ea} \in \mathbb{R}^{k \times k}$ denoting the 3-D attention weights value at i^{th} pixel. To delve into the details of DFR, we detail the two principal modules of DFR: spatial filter branch and energy-based attention branch.

Spatial Filter Branch. We first perform convolution on input feature $F_{in} \in \mathbb{R}^{C \times H \times W}$ to extract initial feature and employ max-pooling and average-pooling $F_{max}, F_{avg} \in \mathbb{R}^{1 \times H \times W}$ along the channel to obtain an efficient feature descriptor [60]. Then, we obtain the spatial response map $\mathcal{M}_{sr}$ by a convolution on F_{max}, F_{avg} with sigmoid function, which represents local representation [70] and can be defined as follows:

$$F_{max}, F_{avg} = Conv(F_{in}) \quad (2)$$

$$\mathcal{M}_{sr} = sigmoid\left(Conv\left([F_{max}, F_{avg}]\right)\right) \quad (3)$$

The threshold t in Fig. 3 aims to detect the degraded pixels with a soft distinction. Then we can obtain the mask $\mathcal{M}_p \in \mathbb{R}^{1 \times H \times W}$, $\mathcal{M}_p$ is 1 when $\mathcal{M}_{sr}$ greater than t, and is

0 otherwise. Specifically, $p \in (h, w)$ represents 2D pixel location. Considering the spatial relationship of $\mathcal{M}_{sr}$, we utilize a convolution layer to obtain a set of learnable parameters expanded along the channel dimension $\gamma_c^i \in \mathbb{R}^{1\times H\times W}$ and bias $\beta_c^i \in \mathbb{R}^{1\times H\times W}$, which enhances the feature representation. The computation for γ_c^i, β_c^i is formulated as follows:

$$\gamma_c^i, \beta_c^i = Conv\left(\mathcal{M}_{sr}\right) \tag{4}$$

The μ_c^i and σ_c^i are the channel-wise mean and variance of the features in i-th layer, which relates to global semantic information and local texture [26]:

$$\mu_c^i = \frac{1}{\sum_p \mathcal{M}_i^p} \sum_p F_{in} \odot \mathcal{M}_p \tag{5}$$

$$\sigma_c^i = \sqrt{\frac{1}{\sum_p \mathcal{M}_p} \sum_p \left(F_{in} \odot \mathcal{M}_p - \mu_c^i + \varepsilon\right)} \tag{6}$$

where $\sum_p \mathcal{M}_p$ indicates the number of masked pixels, $\odot$ represents element-wise product, and ε is a small constant to avoid σ_c^i equal to 0. The final feature output of spatial filter branch is obtained by follows:

$$F_{h,w,c}^i = \gamma_c^i \cdot \frac{F_{in} - \mu_c^i}{\sigma_c^i} + \beta_c^i \tag{7}$$

Energy-based Attention Branch. The spatial filter branch with adaptive feature normalization suppresses degraded pixels, which may impede the restoration in texture areas. Thus, we introduce the energy-based attention (EA) branch, similar to [66], to remedy the deficiency of spatial information by considering the 3-D weights and preserving the detail of the textures in the heavily degraded image. EA can leave the clean pixel features and guide the feature integration by calculating the importance score for each pixel.

Specifically, we first obtain the initial feature from a convolution operation. Then, we calculate the mean $\hat{\mu} = \frac{1}{N}\sum_{i=1}^N x_i$ and variance $\hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^N \left(x_i - \hat{\mu}\right)^2$ over all neurons $(N = H \times W)$ in that channel. $\hat{\mu}$ and $\hat{\sigma}^2$ are used for calculating the energy function for each pixel, which is the same as re-weighting the input feature map. Then we minimize the energy of target neuron t, which is formulated as follows:

$$e_t^* = \frac{4\left(\hat{\sigma}^2 + \delta\right)}{\left(t - \hat{\mu}\right)^2 + 2\hat{\sigma}^2 + 2\delta} \tag{8}$$

where δ is the coefficient hyper-parameter. Then we obtain the refined features $\tilde{X}$ as follows:

$$\tilde{X} = sigmoid\left(\frac{1}{E}\right) \odot X \tag{9}$$

where E is obtained by grouping all e_t^* across the channel and spatial dimensions. As for the identity branch, it generates identity features through a simple 1x1 convolution.

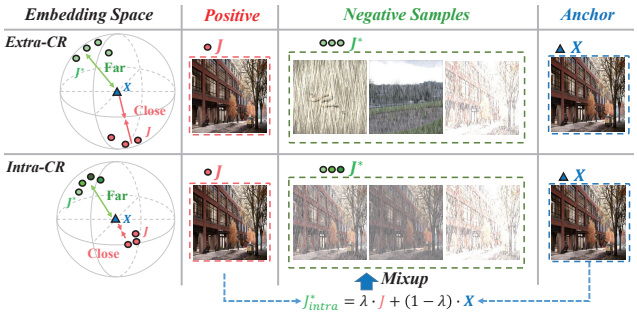

Figure 4. Illustration for differences between Intra-CR and Extra-CR. Extra-CR constructs negative samples by picking the remaining images within a batch. Our Intra-CR designs negative samples through a mixup [75] operation, thus providing specific constraints towards solution space.

The final feature output of DFR is obtained as follows: (1) we multiply the features of spatial branch and EA branch to obtain the intermediate features (2) we add the intermediate features with identity features to obtain the integrated features, as highlighted by the blue region in Fig. 3. The final features output by the encoder will be fed to the decoder for restored image generation.

3.3. Contrastive Regularization

Previous contrastive regularization [10, 61] simply take the other images within the batch as negative samples, namely **Extra**-class **c**ontrastive **r**egularization (Extra-CR), which may result in sub-optimal performance. Thus, we propose a new contrastive regularization method called **Intra**-class **c**ontrastive **r**egularization (Intra-CR). The principal differences between Extra-CR and Intra-CR are shown in Fig. 4, where Intra-CR constructs negative samples through mixup [75] between clean images and degraded images.

In a classical image restoration scenario, a degraded image X is transformed to I to approximate its clean image J. Specifically, we denote $s = (I, J)$ as the pair of I and J, and $s^* = (I, J^*)$ as the pair of I and negative sample J^*. $\ell_1(s, \theta)$ and $\ell_c(s^*, \theta)$ represent L1 reconstruction loss and contrastive regularizer, where θ represents the model's parameters. Then the empirical risk minimizer for model optimization is given by:

$$\theta_{\alpha,s^*} = \text{argmin}_{\theta\in\Theta} \sum_{s\in S} \left[\ell_1\left(s, \theta\right) + \alpha \cdot \ell_c(s^*, \theta)\right] \tag{10}$$

where α is the weight to control the balance the reconstruction loss and contrastive regularization. Besides, Intra-CR constructs the negative samples J^* through a mixup operation between degraded images X and clean images J, defined as follows:

$$J_{intra}^* = \lambda \cdot J + (1 - \lambda) \cdot X \tag{11}$$

where λ is the hyper-parameter in mixup, we choose different λ to construct different negative samples. Then we give the theoretical analysis between Intra-CR and Extra-CR. The idea is to compute the parameter change as the weight α changes.

We define the sensitivity of performance towards α as follows:

$$\mathcal{R}_{sen}(s^*)=\left|\lim_{\alpha\to 0}\frac{d\theta_{\alpha,s^*}}{d\alpha}\right| \tag{12}$$

where H_θ is assumed as a positive definite Hessian matrix. The sensitivity is a metric that measures the change of model's parameters towards α. Then we give our main theorem:

Theorem 1. *Let s^*_{intra} and s^*_{extra} denote the negative pairs in Intra-CR and Extra-CR, then we obtain $\mathcal{R}_{sen}(s^*_{intra}) < \mathcal{R}_{sen}(s^*_{extra})$.*

The detailed proof is deferred to the supplementary materials. The above theorem indicates that Intra-CR is more stable towards hyper-parameter α than Extra-CR, and such a sensitivity can be reflected through performance changes [49, 80], which will be shown in Sec. 4.6. Then the training objective $\mathcal{L}$ in Intra-CR can be formulated as follows:

$$\mathcal{L}=\ell_1\left(s,\theta\right)+\alpha\cdot\frac{\ell_1(G(I),G(J))}{\ell_1(G(I),G(J^*_{intra}))} \tag{13}$$

where G is a fixed pre-trained VGG19 [52]. $G(\cdot)$ aims to extract hidden features of the images, and we leverage $G(\cdot)$ to compare the common intermediate features between a pair of images. Note that our method is different from the perceptual loss [26], which only adds with positive-pair regularization, but Intra-CR also adopts negative pairs. Besides, our Intra-CR is different from the Extra-CR on negative sample construction. Experiments demonstrate that Intra-CR outperforms Extra-CR in image restoration tasks.

4. Experiments and Analysis

In this section, we evaluate our method on three image restoration tasks: image deraining, denoising, and deblurring.

4.1. Benchmarks and Evaluation

We evaluate our method by Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) [58]. As in [74], we report (in parenthesis) the reduce in error for each model relative to the best performing method by RMSE $\left(RMSE\propto\sqrt{10^{-PSNR/10}}\right)$ and DSSIM $(DSSIM=(1-SSIM)/2)$. Meanwhile, qualitative evaluation is shown through visualization on different benchmarks. The benchmarks are listed as follows: **Image Deraining.** Using the same experimental setups of the method [74] on image deraining, we train our model on 13,712 clean-rain image pairs. We perform evaluation on various test, including Test100 [79], Rain100H [67], Rain100L [67], Test2800 [16], Test1200 [78]. **Image Denoising**. We train DRCNet on the SIDD medium version [1] dataset with 320 high-resolution images and directly test on the DND [40] dataset with 50 pairs of real-world noisy images. **Image Deblurring.** We train on the GoPro [38] dataset that contains 2,103 image pairs for training and 1,111 pairs for evaluation and directly apply it on HIDE [51] and RealBlur [47] to demonstrate generalization.

4.2. Implementation Details

Our DRCNet is trained with Adam optimizer [28], and the learning rate is set to 2×10^{-4} by default, and decreased to 1×10^{-7} with cosine annealing strategy [36]. δ in the Eq. (8) is set to 1×10^{-6}, the degraded pixel mask threshold t in Fig. 3 is set to 0.75. Detailed analysis of δ and t will be discussed in our supplementary materials. For Intra-CR, α is set to 0.025, and the number n of negative samples is set to 10. The mixup parameters are selected as $0.80\sim0.98$ with an interval of 0.02. We train our model on 256 × 256 patches with a batch size of 32 for 4×10^6 iterations. Specifically, we apply random rotation, cropping, and flipping to the images to augment the training data.

4.3. Image Deraining Results

For the image deraining task, consistent with prior work [74], Table 1 illustrates that our method significantly advances state-of-the-art by consistently achieving better PSNR/SSIM scores on all derain benchmarks. Compared to the state-of-the-art model SPAIR [43], we obtain significant performance gains of 0.82dB in PSNR and 0.011 in SSIM, and a 9.03% and 2.7% error reduction averaged across all derain benchmarks. Specifically, the improvement on Rain100L can reach 1.3 dB, which well demonstrates that our model can effectively remove rain streaks. Meanwhile, the qualitative results on challenging image derain samples are illustrated in Fig. 5, which demonstrates that DRCNet obtains better visual qualities. In contrast, other approaches compromise structural content and introduce artifacts. Due to the effectiveness of DRF, DRCNet can faithfully restore and preserve the original image contents.

4.4. Image Denoising Results

Table 2 and Fig. 6 show quantitive and qualitative comparisons with recent denoising methods on the SIDD [1] and DND [40] datasets. DRCNet achieves consistently better PSNR, SSIM. The results show that DRCNet outperforms the state-of-the-art denoising approaches, i.e., 0.37 dB and 0.05dB over MPRNet on SIDD and DND. In the SSIM metric, DRCNet also owns a performance rise com-

Figure 5. Visual comparisons of zoomed-in results of competing deraining models on images from the derain test set. Our DRCNet effectively removes rain and generates more natural images, and obtains better visual results.

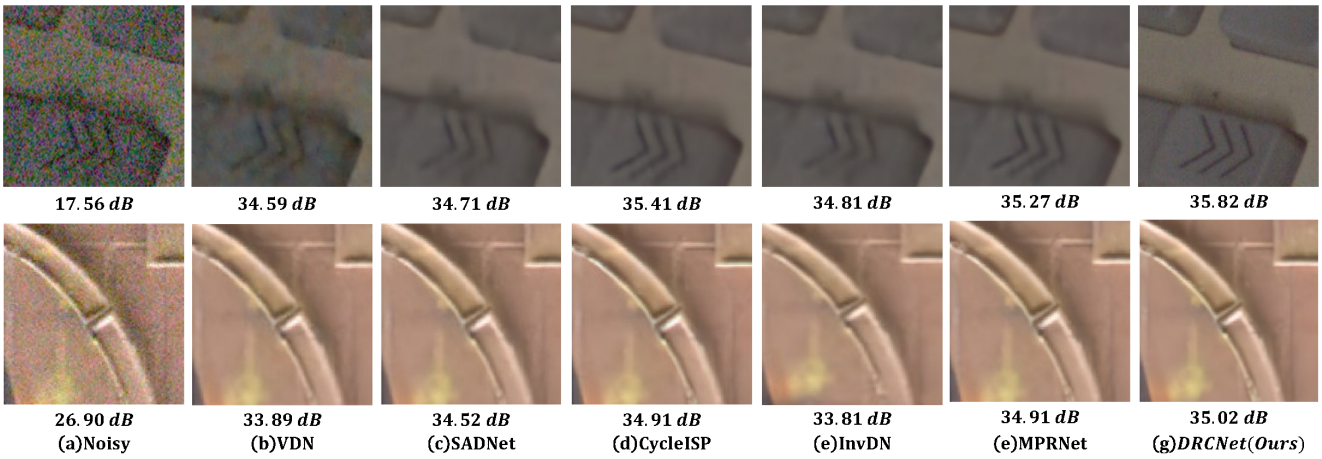

Figure 6. Image denoising comparisons. The first row is from SIDD [1] and the second row is from DND [40]. The proposed DRCNet better preserves fine-grained texture and structural patterns.

pared to MPRNet, boosting from 0.958 to ours 0.972. It means that our model concentrates further on regional textures and local features. Note that the DND dataset does not contain any training images, which indicates the good generalization of DRCNet. Generally, our DRCNet provides better image denoise performance than state-of-the-art methods by effectively removing the noise and artifacts while preserving the main structure and contents.

4.5. Image Deblurring Results

As for the image deblurring task, we report the performance of DRCNet and comparison methods on the synthetic GoPro [38] and HIDE [51] datasets in Table 3 and Fig. 7. Overall, our model slightly outperforms state-of-the-art methods. Our method achieves 32.76 PSNR and 0.961 in SSIM on the GoPro [38] dataset and achieves 30.97 PSNR and 0.941 SSIM on the HIDE dataset. It is worth mentioning that DRCNet is trained only on the GoPro dataset but achieves the state-of-the-art results on the HIDE dataset, thus demonstrating its good generalization. Moreover, we directly evaluate the GoPro trained model on RealBlur, which can further test the generalization of models. Table 3 also shows the experimental results of the DRCNet train and test on the RealBlur-J dataset. Our DRCNet obtains a performance gain of 0.05 dB on the RealBlur-J subset over the existing state-of-the-art methods. Overall, the images deblurred by our model are slightly sharper and have better visual results than competing methods.

4.6. Ablation study

To demonstrate the effectiveness of the proposed DRCNet, we conduct ablation studies to analyze the effective-

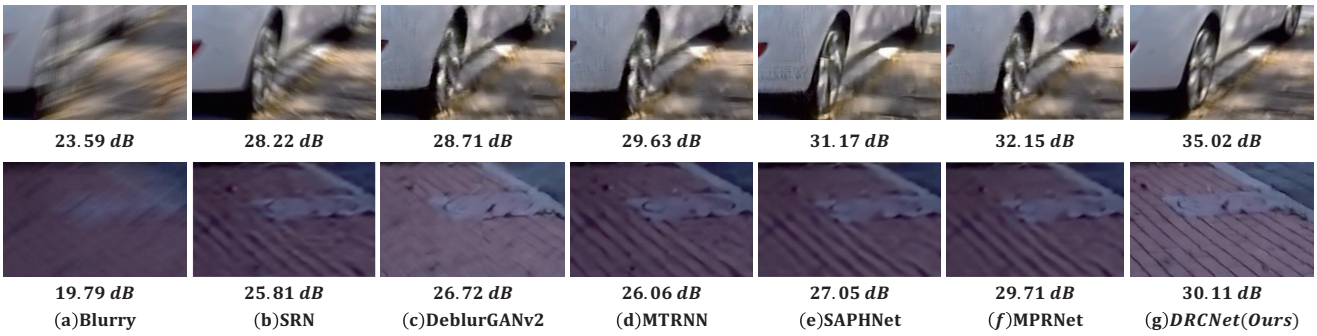

Figure 7. Qualitative comparisons on GoPro [38] test dataset. The deblurred results listed from left to right are from SRN [44], DeblurGANv2 [30], MTRNN [39], SAPHNet [54], MPRNet [74] and ours, respectively.

Table 1. Image deraining results. The best and second best scores are bolden and underlined, respectively. For each method, reduction in error relative to the best-performing algorithm is reported in parenthesis (see Sec. 4.1 for detailed error calculation rules). Our DRCNet achieves substantial improvements in PSNR over the state-of-the-art method SPAIR [43].

Methods	Test100 [79]		Rain100H [67]		Rain100L [67]		Test2800 [16]		Test1200 [78]		Average			
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑		SSIM↑	
DerainNet [15]	22.77	0.810	14.92	0.592	27.03	0.884	24.31	0.861	23.38	0.835	22.48	(72.62%)	0.796	(64.71%)
SEMI [59]	22.35	0.788	16.56	0.486	25.03	0.842	24.43	0.782	26.05	0.822	22.88	(71.33%)	0.744	(71.88%)
DIDMDN [78]	22.56	0.818	17.35	0.524	25.23	0.741	28.13	0.867	29.65	0.901	24.58	(65.13%)	0.770	(68.70%)
UMRL [68]	24.41	0.829	26.01	0.832	29.18	0.923	29.97	0.905	30.55	0.910	28.02	(48.19%)	0.880	(40.00%)
RESCAN [32]	25.00	0.835	26.36	0.786	29.80	0.881	31.29	0.904	30.51	0.882	28.59	(44.68%)	0.857	(49.65%)
PreNet [45]	24.81	0.851	26.77	0.858	32.44	0.950	31.75	0.916	31.36	0.911	29.42	(39.13%)	0.897	(30.10%)
MSPFN [25]	27.50	0.876	28.66	0.860	32.40	0.933	32.82	0.930	32.39	0.916	30.75	(29.06%)	0.903	(25.77%)
MPRNet [74]	30.27	0.897	30.41	0.890	36.40	0.965	33.64	0.938	32.91	0.916	32.73	(10.89%)	0.921	(8.86%)
SPAIR [43]	30.35	0.909	30.95	0.892	36.93	0.969	33.34	0.936	33.04	0.922	32.91	(9.03%)	0.926	(2.70%)
DRCNet(Ours)	**32.18**	**0.917**	**30.96**	**0.895**	**38.23**	**0.976**	**33.89**	**0.946**	**33.40**	**0.934**	**33.73**	(0.00%)	**0.933**	(0.00%)

Table 2. Denoising performance comparisons on SIDD [1] and DND [40] datasets. * denotes the methods that use additional training data, while our DRCNet is only trained on the SIDD train set and directly tested on DND.

Method	SIDD [1]				DND [40]			
	PSNR↑		SSIM↑		PSNR↑		SSIM↑	
DnCNN [84]	23.66	(84.90%)	0.583	(93.29%)	32.43	(57.44%)	0.790	(79.05%)
MLP [4]	24.71	(82.96%)	0.641	(92.20%)	34.23	(47.64%)	0.833	(73.65%)
BM3D [11]	25.65	(81.01%)	0.685	(91.11%)	34.51	(45.93%)	0.851	(70.47%)
CBDNet* [20]	30.78	(65.72%)	0.801	(85.93%)	38.06	(18.62%)	0.942	(24.14%)
RIDNet* [2]	38.71	(14.59%)	0.951	(42.86%)	39.26	(6.57%)	0.953	(6.38%)
AINDNet* [27]	38.95	(12.20%)	0.952	(41.67%)	39.37	(5.38%)	0.951	(10.20%)
VDN [71]	39.28	(8.80%)	0.956	(36.36%)	39.38	(5.27%)	0.952	(8.33%)
SADNet* [7]	39.46	(6.89%)	0.957	(34.88%)	39.59	(2.95%)	0.952	(8.33%)
DANet+* [72]	39.47	(6.78%)	0.957	(34.88%)	39.58	(3.06%)	0.955	(2.22%)
CycleISP* [73]	39.52	(6.25%)	0.957	(34.88%)	39.56	(3.29%)	0.956	(0.00%)
InvDN [34]	39.28	(8.80%)	0.955	(37.78%)	39.57	(3.17%)	0.952	(8.33%)
MPRNet [74]	39.71	(4.17%)	0.958	(33.33%)	39.80	(0.58%)	0.954	(4.35%)
DRCNet(Ours)	**40.08**	(0.00%)	**0.972**	(0.00%)	**39.85**	(0.00%)	**0.956**	(0.00%)

ness of crucial components of DRCNet, including the DFR module and Intra-CR.

Comparison to Other Dynamic Filters. Since there are other proposed dynamic convolution filters, we conduct experiments to compare them in image restoration tasks, as shown in Table 4. We replace the DFR module with three dynamic filters: SACT [14], CondConv [65], and DDF [88].

Table 4 compares the performance and the parameters of the whole network in various restoration tasks. The experimental results show that models with other dynamic filters obtain significantly worse performance and have more parameters than DFR in the image restoration tasks. Such results validate that DFR is suitable for image restoration tasks.

Table 3. Deblurring results. Our method is trained only on the GoPro dataset [38] and directly tested to the HIDE dataset [51] and RealBlur-J [47] datasets. The scores in the PSNR ‡ column were obtained after training and testing on RealBlur-J dataset.

Method	GoPro [38]		HIDE [51]		RealBlur-J [47]		
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR‡
Xu *et al.* [64]	21.00	0.741	-	-	27.14	0.830	
DeblurGAN [29]	28.70	0.858	24.51	0.871	27.97	0.834	
Nah et al. [38]	29.08	0.914	25.73	0.874	27.87	0.827	
Zhang et al. [81]	29.19	0.913	-	-	27.80	0.847	
DeblurGAN-v2 [30]	29.55	0.934	26.61	0.875	28.70	0.866	29.69
SRN [44]	30.26	0.934	28.36	0.915	28.56	0.867	31.38
Shen *et al.* [51]	30.26	0.940	28.89	0.930	-	-	
DBGAN [82]	31.10	0.942	28.94	0.915	-	-	
MT-RN [39]	31.15	0.945	29.15	0.918	-	-	
DMPHN [76]	31.20	0.940	29.09	0.924	28.42	0.860	
RADN [42]	31.76	0.952	29.68	0.927	-	-	
SAPHNet [54]	31.85	0.948	29.98	0.930	-	-	
SPAIR [43]	32.06	0.953	30.29	0.931	28.81	0.875	31.82
MPRNet [74]	32.66	0.959	30.96	0.939	28.70	0.873	31.76
DRCNet(Ours)	**32.76**	**0.961**	**30.97**	**0.941**	**28.86**	**0.883**	**31.85**

Effectivenes of DFR. We first constuct our base net-

Table 4. Comparison of the parameter number and PSNR (dB). 'Params' means the number of parameters (Millions).

	Filter	SACT [14]	CondConv [65]	DDF [88]	**DFR**
	Params	103.1M	165.7M	87.4M	**18.9M**
PSNR	Derain [74]	19.28	23.53	25.12	**32.39**
	SIDD [1]	27.28	39.43	39.04	**39.92**
	GoPro [38]	19.97	23.09	29.01	**32.21**

Table 5. Ablation studies on DRCNet on SIDD benchmark.

Model	CR	PSNR	SSIM
base	-	33.25	0.812
base+Spatial Filter	-	37.76	0.933
base+Energy-based Attention	-	37.12	0.945
base+Spatial Filter+identity branch	-	38.79	0.925
base+Spatial Filter+Energy-based Attention	-	38.89	0.965
base+DFR	-	40.01	0.976
base+DFR	Extra-CR	39.95	0.956
base+DFR	Intra-CR	**40.08**	**0.972**

work as baseline, which mainly consists of normal UNet [48] with Resblock [23] in encoding and decoding phrases with SAM and CSFF [74]. Subsequently, we replace our DFR module and add Intra-CR scheme into base network as: (1) **base+spatial filter**: only add spatial filter branch into baseline. (2) **base+Energy-based Atention**: Only add energy-based module into base. (3) **base+Spatial Filter+identity branch**:add spatial filter branch and identity branch. (4) **base+Spatial Filter+Energy-based Atention**:add Spatial Filter+Energy-based Atention. (5) **base+DFR**:Add combination of three branch as DFR module. (6) **base+DFR+CR**:Add DFR module and extra-constrastive. (7) **DRCNet**: The combination of DFR module and intra-constrastive for train. The performance of these model are summarized in Table 5.

As shown in Table 5, the spatial filter branch can strengthen DRCNet with more representation power than the base model. Besides, the Energy-based Attention also improves the model's restoration capacity by dynamically guiding feature integration. Besides, Table 5 shows a significant performance drop in PSNR from 40.01 dB to 33.25 dB by removing the whole DFR, which shows that DFR is a successful and crucial module in DRCNet. Specifically, we show the visualization of intermediate features produced by different branches of DFR. As shown in Fig. 8 denotes that spatial filter can effectively reduce the noise pixel, the Energy-based Attention branch focuses on the textures and sharpness in terms of SSIM. Besides, the identity branch can further enhance the feature integration. Overall, the combination of the three branches achieves the best results.

Effect of Contrastive Regularization. This section illustrates the effectiveness of our Intra-CR. Specifically, we apply Intra-CR and Extra-CR on DRCNet, respectively. Moreover, we vary the value of weight α in Eq. (13) and observe the tendency of Intra-CR and Extra-CR. As shown in Fig. 9, Intra-CR outperforms Extra-CR as α varies from 0 to 0.14. Moreover, Intra-CR achieves more stable results towards α, which matches our theoretical analysis that Intra-CR is less sensitive to α. Overall, the results validate the superiority of our Intra-CR.

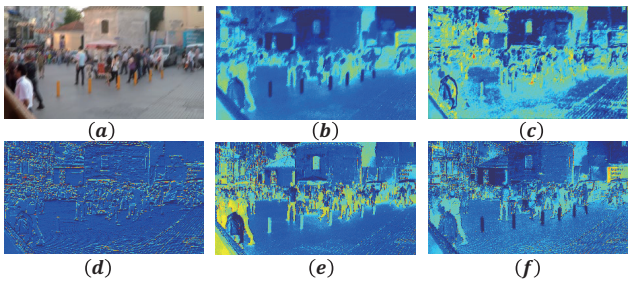

Figure 8. Visualization of intermediate features on images from the GoPro test set [38]. (a-b) Input blurred image and feature map. (c-e) Comparisons among the feature map by using Spatial filter,Energy-based attention and DFR module, respectively. (f) ground truth feature map.

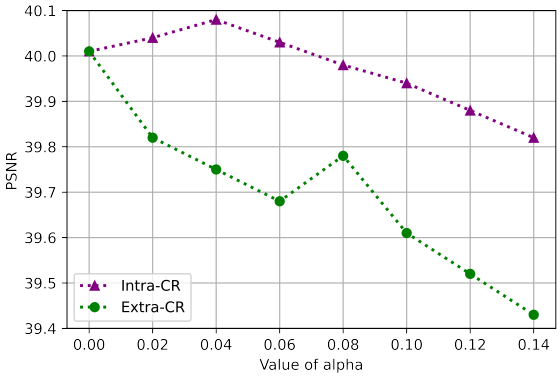

Figure 9. Comparison between Intra-CR and Extra-CR on SIDD benchmark.

5. Conclusion

In this paper, we propose a dynamic restoration contrastive network (DRCNet) for image restoration with two principal components: dynamic filter restoration module (DFR) and intra-class contrastive regularization (Intra-CR). DFR module based on spatial filter branch and energy-based attention branch benefits from being dynamically adaptive towards the spatial-varying image degradation. The key insight of Intra-CR is to construct intra-class negative samples, which is accomplished through mixup operations. Through comprehensive evaluation of the performance of DRCNet on various benchmarks, we validate that the DRCNet achieves state-of-the-art results on ten datasets across various restoration tasks.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1700, 2018. 1, 2, 5, 6, 7, 8, 14, 16

[2] Saeed Anwar and Nick Barnes. Real image denoising with feature attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3155–3164, 2019. 7

[3] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 13

[4] Harold C Burger, Christian J Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2392–2399. IEEE, 2012. 7

[5] Xiangyong Cao, Yang Chen, Qian Zhao, Deyu Meng, Yao Wang, Dong Wang, and Zongben Xu. Low-rank matrix factorization under general mixture noise distributions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1493–1501, 2015. 2

[6] Tony F Chan and Chiu-Kwong Wong. Total variation blind deconvolution. *IEEE Transactions on Image Processing*, 7(3):370–375, 1998. 2

[7] Meng Chang, Qi Li, Huajun Feng, and Zhihai Xu. Spatial-adaptive network for single image denoising. In *European Conference on Computer Vision*, pages 171–187. Springer, 2020. 1, 7

[8] Jin Chen, Xijun Wang, Zichao Guo, Xiangyu Zhang, and Jian Sun. Dynamic region-aware convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8064–8073, June 2021. 1, 2

[9] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11030–11039, 2020. 2

[10] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 715–724, 2021. 2, 4

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

[12] Weisheng Dong, Lei Zhang, Guangming Shi, and Xin Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2012. 2

[13] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3575–3584, 2019. 1

[14] Michael Figurnov, Maxwell Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov. Spatially adaptive computation time for residual networks. pages 1790–1799, 07 2017. 7, 8

[15] Xueyang Fu, Jiabin Huang, Xinghao Ding, Yinghao Liao, and John Paisley. Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing*, 26(6):2944–2956, 2017. 1, 7

[16] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3855–3863, 2017. 5, 7

[17] Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3848–3856, 2019. 1, 2

[18] Yuanbiao Gou, Boyun Li, Zitao Liu, Songfan Yang, and Xi Peng. Clearer: Multi-scale neural architecture search for image restoration. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[19] Shuhang Gu, Yawei Li, Luc Van Gool, and Radu Timofte. Self-guided network for fast image denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2511–2520, 2019. 1

[20] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1712–1722, 2019. 1, 7

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 2

[22] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2341–2353, 2010. 1, 2

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2016. 1, 3, 8

[24] Jongheon Jeong and Jinwoo Shin. Training GANs with stronger augmentations via contrastive discriminator. In *International Conference on Learning Representations*, 2021. 2

[25] Kui Jiang, Zhongyuan Wang, Peng Yi, Chen Chen, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun Jiang. Multi-scale progressive fusion network for single image deraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8346–8355, 2020. 7

[26] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. volume 9906, pages 694–711, 10 2016. 4, 5

[27] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3482–3492, 2020. 7

[28] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014. 5

[29] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8183–8192, 2018. 7

[30] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *The IEEE International Conference on Computer Vision*, Oct 2019. 1, 7

[31] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*, 2020. 2

[32] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha. Recurrent squeeze-and-excitation context aggregation net for single image deraining. In *Proceedings of the European Conference on Computer Vision*, pages 254–269, 2018. 1, 7

[33] Yanwei Li, Lin Song, Yukang Chen, Zeming Li, Xiangyu Zhang, Xingang Wang, and Jian Sun. Learning dynamic routing for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8553–8562, 2020. 2

[34] Yang Liu, Zhenyue Qin, Saeed Anwar, Pan Ji, Dongwoo Kim, Sabrina Caldwell, and Tom Gedeon. Invertible denoising network: A light solution for real noise removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13365–13374, 2021. 1, 7

[35] Yi-Chen Lo, Chia-Che Chang, Hsuan-Chao Chiu, Yu-Hao Huang, Chia-Ping Chen, Yu-Lin Chang, and Kevin Jou. Clcc: Contrastive learning for color constancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8053–8063, June 2021. 2

[36] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. 08 2016. 5

[37] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2272–2279. IEEE, 2009. 2

[38] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3883–3891, 2017. 1, 5, 6, 7, 8, 14, 17

[39] Dongwon Park, Dong Un Kang, Jisoo Kim, and Se Young Chun. Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In *European Conference on Computer Vision*, pages 327–343. Springer, 2020. 7

[40] Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1586–1595, 2017. 5, 6, 7

[41] Javier Portilla, Vasily Strela, Martin J Wainwright, and Eero P Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image processing*, 12(11):1338–1351, 2003. 2

[42] Kuldeep Purohit and AN Rajagopalan. Region-adaptive dense network for efficient motion deblurring. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11882–11889, 2020. 7

[43] Kuldeep Purohit, Maitreya Suin, A. N. Rajagopalan, and Vishnu Naresh Boddeti. Spatially-adaptive image restoration using distortion-guided networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2309–2319, October 2021. 1, 2, 5, 7

[44] Dongwei Ren, Wei Shang, Pengfei Zhu, Qinghua Hu, Deyu Meng, and Wangmeng Zuo. Single image deraining using bilateral recurrent network. *IEEE Transactions on Image Processing*, 29:6852–6863, 2020. 7

[45] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3937–3946, 2019. 7

[46] Gernot Riegler, Samuel Schulter, Matthias Ruther, and Horst Bischof. Conditioned regression models for non-blind single image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 522–530, 2015. 2

[47] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *European Conference on Computer Vision*, pages 184–201. Springer, 2020. 5, 7, 14, 17

[48] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015. 8

[49] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014. 5, 13

[50] Falong Shen, Shuicheng Yan, and Gang Zeng. Neural style transfer via meta networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8061–8069, 2018. 2

[51] Ziyi Shen, Wenguan Wang, Xiankai Lu, Jianbing Shen, Haibin Ling, Tingfa Xu, and Ling Shao. Human-aware motion deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5572–5581, 2019. 5, 6, 7

[52] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[53] Masanori Suganuma, Mete Ozay, and Takayuki Okatani. Exploiting the potential of standard convolutional autoencoders for image restoration by evolutionary search. In *International Conference on Machine Learning*, pages 4771–4780. PMLR, 2018. 2

[54] Maitreya Suin, Kuldeep Purohit, and AN Rajagopalan. Spatially-attentive patch-hierarchical network for adaptive

motion deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3606–3615, 2020. 7

[55] Domen Tabernik, Matej Kristan, and Aleš Leonardis. Spatially-adaptive filter units for compact and efficient deep neural networks. *International Journal of Computer Vision*, 128(8):2049–2067, 2020. 2

[56] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6411–6420, 2019. 2

[57] Paul Viola and Michael J Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004. 2

[58] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 5

[59] Wei Wei, Deyu Meng, Qian Zhao, Zongben Xu, and Ying Wu. Semi-supervised transfer learning for image rain removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3877–3886, 2019. 7

[60] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision*, September 2018. 3

[61] Haiyan Wu, Yanyun Qu, Shaohui Lin, Jian Zhou, Ruizhi Qiao, Zhizhong Zhang, Yuan Xie, and Lizhuang Ma. Contrastive learning for compact single image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10551–10560, 2021. 2, 4

[62] Jialin Wu, Dai Li, Yu Yang, Chandrajit Bajaj, and Xiangyang Ji. Dynamic filtering with large sampling field for convnets. In *Proceedings of the European Conference on Computer Vision*, pages 185–200, 2018. 2

[63] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L. Yuille, and Quoc V. Le. Adversarial examples improve image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[64] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1107–1114, 2013. 2, 7

[65] Brandon Yang, Gabriel Bender, Quoc V Le, and Jiquan Ngiam. Condconv: Conditionally parameterized convolutions for efficient inference. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 1307–1318. Curran Associates, Inc., 2019. 1, 7, 8

[66] Lingxiao Yang, Ru-Yuan Zhang, Lida Li, and Xiaohua Xie. Simam: A simple, parameter-free attention module for convolutional neural networks. In *International Conference on Machine Learning*, pages 11863–11874. PMLR, 2021. 4

[67] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1357–1366, 2017. 5, 7, 14, 16

[68] Rajeev Yasarla and Vishal M Patel. Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8405–8414, 2019. 7

[69] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2678–2687, 2016. 2

[70] Tao Yu, Zongyu Guo, Xin Jin, Shilin Wu, Zhibo Chen, Weiping Li, Zhizheng Zhang, and Sen Liu. Region normalization for image inpainting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12733–12740, 2020. 3

[71] Zongsheng Yue, Hongwei Yong, Qian Zhao, Deyu Meng, and Lei Zhang. Variational denoising network: Toward blind noise modeling and removal. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 1690–1701. Curran Associates, Inc., 2019. 7

[72] Zongsheng Yue, Qian Zhao, Lei Zhang, and Deyu Meng. Dual adversarial network: Toward real-world noise removal and noise generation. In *Proceedings of the European Conference on Computer Vision*. August 2020. 1, 7

[73] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Cycleisp: Real image restoration via improved data synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2696–2705, 2020. 7

[74] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14821–14831, 2021. 1, 2, 3, 5, 7, 8, 14

[75] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. Mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 2, 4

[76] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5978–5986, 2019. 2, 7

[77] Haokui Zhang, Ying Li, Hao Chen, and Chunhua Shen. Memory-efficient hierarchical neural architecture search for image denoising. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3654–3663, 2020. 2

[78] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 695–704, 2018. 5, 7, 14, 15

[79] He Zhang, Vishwanath Sindagi, and Vishal M Patel. Image de-raining using a conditional generative adversarial network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11):3943–3956, 2019. 5, 7, 14, 15

[80] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pages 7472–7482. PMLR, 2019. 5

[81] Jiawei Zhang, Jinshan Pan, Jimmy Ren, Yibing Song, Linchao Bao, Rynson WH Lau, and Ming-Hsuan Yang. Dynamic scene deblurring using spatially variant recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2521–2529, 2018. 7

[82] Kaihao Zhang, Wenhan Luo, Yiran Zhong, Lin Ma, Bjorn Stenger, Wei Liu, and Hongdong Li. Deblurring by realistic blurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2737–2746, 2020. 7

[83] Kane Zhang, Jian Zhang, Qiang Wang, and Zhao Zhong. Dynet: Dynamic convolution for accelerating convolution neural networks, 2020. 1

[84] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 7

[85] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Scale-adaptive convolutions for scene parsing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2031–2039, 2017. 2

[86] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vsion and Pattern Recognition*, pages 2472–2481, 2018. 2

[87] Yaowei Zheng, Richong Zhang, and Yongyi Mao. Regularizing neural networks via adversarial model perturbation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8156–8165, 2021. 13

[88] Jingkai Zhou, Varun Jampani, Zhixiong Pi, Qiong Liu, and Ming-Hsuan Yang. Decoupled dynamic filter networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6647–6656, 2021. 1, 2, 7, 8

[89] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019. 2

This supplementary material is organized as follows. Section 6 contains comprehensive proof of **Theorem 1**. Section 7 and 8 describe the results under different threshold t and the coefficient hyper-parameter δ. Section 9, 10, and 11 contain additional visualization results.

6. Proof of Theorem 1

Firstly, we provide a standard derivation of $\mathcal{R}_{\text{sen}}$ based on asymptotic attributes in statistics. Recall the definition of $\ell(s,\theta)$ and $\mathcal{R}_{\text{sen}}(s^*)$, we give the assumption for $\ell_1(s,\theta)$. $\ell_1(s,\theta)$ and $\ell_c(s^*,\theta)$ are twice differentiable and strongly convex in θ. Based on Assumption 6, we give the lemma below: $\mathcal{R}_{\text{sen}}(s^*) = \left|H_\theta^{-1}\sum_{s\in S}\nabla_\theta \ell_c(s^*,\theta)\right|$ The empirical risk $R(\theta)$ is formulated as follows:

$$R(\theta) = \sum_{s\in S}\left[\ell_1(s,\theta) + \alpha\cdot\ell_c(s^*,\theta)\right] \tag{14}$$

Then we make derivation on $R(\theta)$, the Hessian matrix of $R(\theta)$ can be obtained based on Assumption 6:

$$H_\theta = \nabla^2 R(\theta) = \sum_{s\in S}\left[\nabla_\theta^2\ell_1(s,\theta) + \alpha\cdot\nabla_\theta^2\ell_c(s^*,\theta)\right] \tag{15}$$

Assumption 6 guarantees that H_θ exists and is positive definite and guarantees the existence of H_θ^{-1}, which will be used in the following derivation.

Recall the definition of the minimizer θ_{α,s^*} of $\mathcal{R}_{\text{sen}}$:

$$\theta_{\alpha,s^*} = \text{argmin}_{\theta\in\Theta}\sum_{s\in S}\left[\ell_1(s,\theta) + \alpha\cdot\ell_c(s^*,\theta)\right] \tag{16}$$

Define the parameter change $\Delta_{\theta_{\alpha,s^*}} = \theta_{\alpha,s^*} - \theta_{0,s^*}$ when α is close to 0, then the change of θ towards α can be defined as follows:

$$\frac{d\theta_{\alpha,s^*}}{d\alpha} = \frac{\Delta_{\theta_{\alpha,s^*}}}{d\alpha} \tag{17}$$

Besides, let $L_1(s,\theta)$ and $L_c(s^*,\theta)$ be as follows for brevity:

$$L_1(s,\theta) = \sum_{s\in S}\ell_1(s,\theta)$$

$$L_c(s^*,\theta) = \sum_{s\in S}\ell_c(s^*,\theta)$$

Considering that θ_{α,s^*} is a minimizer of $R(\theta_{\alpha,s^*})$, then we can obtain the following property:

$$\nabla_{\theta_{\alpha,s^*}} L_1(s,\theta_{\alpha,s^*}) + \alpha\cdot\nabla_{\theta_{\alpha,s^*}} L_c(s^*,\theta_{\alpha,s^*}) \approx 0 \tag{18}$$

$\theta_{\alpha,s^*} \to \theta_{0,s^*}$ when $\alpha\to 0$ Then, we have a further derivation of Eq.(18) based on Assumption 6 and Taylor expansion, which is listed as follows:

$$\nabla_{\theta_{\alpha,s^*}} L_1(s,\theta_{\alpha,s^*}) + \alpha\cdot\nabla_{\theta_{\alpha,s^*}} L_c(s^*,\theta_{\alpha,s^*}) +$$
$$[\nabla^2_{\theta_{\alpha,s^*}} L_1(s,\theta_{\alpha,s^*}) + \alpha\cdot\nabla^2_{\theta_{\alpha,s^*}} L_c(s^*,\theta_{\alpha,s^*})]\Delta_{\theta_{\alpha,s^*}} \approx 0$$

where we have dropped high-order terms. Naturally, we can get the formulation of $\Delta_{\theta_{\alpha,s^*}}$:

$$\Delta_{\theta_{\alpha,s^*}} \approx -[\nabla^2_{\theta_{\alpha,s^*}} L_1(s,\theta_{\alpha,s^*}) + \alpha\cdot\nabla^2_{\theta_{\alpha,s^*}} L_c(s^*,\theta_{\alpha,s^*})]^{-1}$$
$$[\nabla_{\theta_{\alpha,s^*}} L_1(s,\theta_{\alpha,s^*}) + \alpha\cdot\nabla_{\theta_{\alpha,s^*}} L_c(s^*,\theta_{\alpha,s^*})]$$

Then we give the assumption for θ_{α,s^*} as follows: $\nabla_{\theta_{\alpha,s^*}} L_1(s, \theta_{\alpha,s^*}) \approx 0$ Such an assumption is natural since empirical risk minimization (ERM) process still optimizes $L(\cdot)$ well when the weight α of regularizer is small [49].

Then, based on Assumption 6, we have $\nabla L(s, \theta_{\alpha,s^*}) \approx 0$. After abandoning the $o(\alpha)$ terms, we have

$$\Delta_{\theta_{\alpha,s^*}} \approx -\alpha[\nabla^2_{\theta_{\alpha,s^*}} L(s, \theta_{\alpha,s^*})]^{-1} \nabla_{\theta_{\alpha,s^*}} L_c\left(s^*, \theta_{\alpha,s^*}\right)$$

Combined with Eq.(15) and Eq.(17), we have

$$\left.\frac{d\theta_{\alpha,s^*}}{d\alpha}\right|_{\alpha\to 0} = -H^{-1}_{\theta_{\alpha,s^*}} \nabla L_c(s^*, \theta_{\alpha,s^*}) \tag{19}$$

Since $\mathcal{R}_{sen}(s^*) = \left|\lim_{\alpha\to 0} \frac{d\theta_{\alpha,s^*}}{d\alpha}\right|$, we have that:

$$\mathcal{R}_{\text{sen}}\left(s^*\right) = \left|H^{-1}_{\theta} \nabla_{\theta} L_c\left(s^*, \theta\right)\right| \tag{20}$$

which completes the proof of Lemma 6.

Then we provide deeper derivation and present the Lemma 6. Note that we replace ∇_θ with ∇ for brevity, and the omitted θ is aligned to its following θ which is the same as the Proof for Lemma 6. For example, $\nabla L_c(s_1^*, \theta_{\alpha,s_1^*})$ means $\nabla_{\theta_{\alpha,s_1^*}} L_c(s_1^*, \theta_{\alpha,s_1^*})$ Let s^*_{intra} (s_1^*) and s^*_{extra} (s_2^*) be the negative pairs constructed by Intra-CR and Extra-CR, we have

$$|H^{-1}_{\theta_{\alpha,s_1^*}} \nabla L_c(s_1^*, \theta_{\alpha,s_1^*})| < |H^{-1}_{\theta_{\alpha,s_2^*}} \nabla L_c(s_2^*, \theta_{\alpha,s_2^*})|$$

$H_{\theta_{\alpha,s_1^*}} \approx H_{\theta_{\alpha,s_2^*}}$

$\theta_{\alpha,s_1^*} \approx \theta_{\alpha,s_2^*} = \theta_\alpha$ Since the objective can be decomposed into the dot product of two vectors:

$$\begin{aligned} &|H^{-1}_{\theta_{\alpha,s_1^*}} \nabla L_c(s_1^*, \theta_{\alpha,s_1^*})| \\ &= |H^{-1}_{\theta_{\alpha,s_1^*}}| \cdot |\nabla L_c(s_1^*, \theta_{\alpha,s_1^*})| \cdot |cos(\gamma)| \end{aligned} \tag{21}$$

where $H^{-1}_{\theta_{\alpha,s_1^*}} \in R^{\Theta\times 1}$, $\nabla L_c(s_1^*, \theta_{\alpha,s_1^*}) \in R^{1\times\Theta}$, and Θ is the total number of model's parameters. γ is the angle between such two vectors, which we assume the same.

Then, based on Assumption 6 and Assumption 6, we only need to investigate the relation between $|\nabla L_c(s_1^*, \theta_{\alpha,s_1^*})|$ and $|\nabla L_c(s_2^*, \theta_{\alpha,s_2^*})|$.

Then we directly compare $|\nabla L_c(s_1^*, \theta_{\alpha,s_1^*})|$ and $|\nabla L_c(s_2^*, \theta_{\alpha,s_2^*})|$. Firstly, we rewrite the formulation of $\nabla L_c(s_1^*, \theta_{\alpha,s_1^*})$ as follows:

$$\nabla L_c(s_1^*, \theta_\alpha) = \frac{L_c(s_1^*, \theta_\alpha + \epsilon) - L_c(s_1^*, \theta_\alpha)}{d\epsilon} \tag{22}$$

Similarily, we also have:

$$\nabla L_c(s_2^*, \theta_\alpha) = \frac{L_c(s_2^*, \theta_\alpha + \epsilon) - L_c(s_2^*, \theta_\alpha)}{d\epsilon} \tag{23}$$

where $\epsilon \to 0$ is a perturbation. Since θ_α is the minimizer for $L_c(s_1^*, \theta_\alpha)$ and $L_c(s_2^*, \theta_\alpha)$, respectively, which indicates that any small perturbation ϵ on θ will increase the empirical risk. Therefore, $L_c(s_2^*, \theta_\alpha + \epsilon) - L_c(s_2^*, \theta_\alpha) > 0$ for any small perturbation ϵ.

Based on this, we present our subsequent derivation for $\nabla L_c(s_1^*, \theta_\alpha)$ and $\nabla L_c(s_2^*, \theta_\alpha)$. First of all, $\nabla L_c(s_2^*, \theta_\alpha)$ and $\nabla L_c(s_2^*, \theta_\alpha)$ are positive due to the property of θ_{α,s_1^*} and θ_{α,s_2^*}. Therefore, we have:

$$\begin{aligned} &|\nabla L_c(s_2^*, \theta_\alpha)| - |\nabla L_c(s_1^*, \theta_\alpha)| \\ &= |\frac{L_c(s_2^*, \theta_\alpha + \epsilon) - L_c(s_2^*, \theta_\alpha)}{d\epsilon}| \\ &- |\frac{L_c(s_1^*, \theta_\alpha + \epsilon) - L_c(s_1^*, \theta_\alpha)}{d\epsilon}| \\ &= (\frac{L_c(s_2^*, \theta_\alpha) - L_c(s_1^*, \theta_\alpha)}{d\epsilon}) \\ &- (\frac{L_c(s_2^*, \theta_\alpha + \epsilon) - L_c(s_1^*, \theta_\alpha + \epsilon)}{d\epsilon}) \end{aligned} \tag{24}$$

Recall the Assumption 6 and the fact that θ is the local minimizer of empirical risk of L_c, such conditions indicate that the norm of gradient on θ is smaller than the one on $\theta + \epsilon$. Based on the theoretical justification on empirical risk [87] and convexity of loss function [3], the change of input s^* causes higher risk change at $\theta + \epsilon$ than θ, which means:

$$\begin{aligned} &|\frac{L_c(s_1^*, \theta_\alpha) - L_c(s_2^*, \theta_\alpha)}{d\epsilon}| \\ &< |\frac{L_c(s_1^*, \theta_\alpha + \epsilon) - L_c(s_2^*, \theta_\alpha + \epsilon)}{d\epsilon}| \end{aligned} \tag{25}$$

Then recall the definition of ℓ_c and construction of s_1^* and s_2^*:

$$\ell_c(s^*) = \frac{\ell_1(G(I), G(J))}{\ell_1(G(I), G(J^*))}$$

Since s_1^* and s_2^* are intra-class and extra-class image pairs, the distance between $G(I), G(J^*_{intra})$ is lower than $G(I), G(J^*_{extra})$, thus $\ell_c(s^*_{intra}) > \ell_c(s^*_{extra})$. Therefore, we have follows:

$$L_c(s_2^*, \theta_\alpha) - L_c(s_1^*, \theta_\alpha) < 0 \tag{26}$$

Then we give further derivation on Eq.(24):

$$\begin{aligned} &(\frac{L_c(s_2^*, \theta_\alpha) - L_c(s_1^*, \theta_\alpha)}{d\epsilon}) \\ &- (\frac{L_c(s_2^*, \theta_\alpha + \epsilon) - L_c(s_1^*, \theta_\alpha + \epsilon)}{d\epsilon}) \\ &= -|\frac{L_c(s_2^*, \theta_\alpha) - L_c(s_1^*, \theta_\alpha)}{d\epsilon}| \\ &+ |\frac{L_c(s_2^*, \theta_\alpha + \epsilon) - L_c(s_1^*, \theta_\alpha + \epsilon)}{d\epsilon}| > 0 \end{aligned} \tag{27}$$

The equation in Eq.(27) holds due to Eq.(26).

The result completes the Lemma 6. Combined Lemma 6 and Lemma 6, we complete the proof of the theorem .

Table 6. Comparison of the different parameter value of threshold t in three main restoration datasets in term of PSNR (dB).

	t	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95
PSNR	Derain [74]	26.48	26.73	26.82	27.10	27.22	**27.39**	27.31	27.23	27.21	27.13
	SIDD [1]	36.92	37.09	37.11	37.37	37.72	**37.76**	37.74	37.61	36.99	36.92
	GoPro [38]	26.05	29.12	29.17	29.71	30.12	**30.21**	30.19	28.17	28.15	27.10

Table 7. Comparison of the different parameter value of threshold δ in three main restoration dataset in term of PSNR (dB).

	δ	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}
PSNR	Derain [74]	25.08	25.13	25.42	25.56	25.57	**25.59**	25.54
	SIDD [1]	36.97	37.00	37.01	37.07	37.09	**37.12**	37.04
	GoPro [38]	31.51	31.58	31.68	31.69	31.69	**31.71**	31.60

7. Threshold of t

Threshold t is set in the spatial filter branch to detect the degraded pixels with a soft distinction. It could affect the masking progress of the degraded pixels and further affect the quality of restoration results. We conduct a set of experiments to explore the most suitable value of three t restoration tasks. The PSNR results on Derain [74], SIDD [1] and GoPro [38] show that the proposed method only using the spatial filter-branch achieves the best results when threshold t equals to 0.75, as shown in Table 6. A range of t varies from 0.5 to 0.95 with an interval of 0.05, and we test the model three times for each value of t, and select the mean as the result. For each time, the training progress takes less than 400,000 epochs.

8. Analysis of δ

In order to explore the best value of the coefficient hyper-parameter δ in the energy-based attention branch selected as it provides a good trade-off between the re-calibration feature effect with texture detail and PSNR. The value of δ varies from $10^{-1} \sim 10^{-7}$. The performance on the Derain [74], SIDD [1] and GoPro [38] show that the energy-based attention branch achieves the best results when δ equals to 10^{-6}, which provides a good performance of three datasets, as shown in Table 7.

9. Image Deraining Results

Fig. 10, 11, and 12 show deraining results of our DRCNet and those of the state-of-the-art methods on several challenging images from different datasets [67, 78, 79].

10. Image Denoising Results

We provide additional denoising comparisons of our method with the state-of-the-art methods. Results on the SIDD dataset [1] are shown in Fig 13.

11. Image Deblurring Results

For the case of deblurring datasets, the visual results are shown in Fig. 14 and Fig. 15 on the GoPro [38] and RealBlur-J [47] datasets.

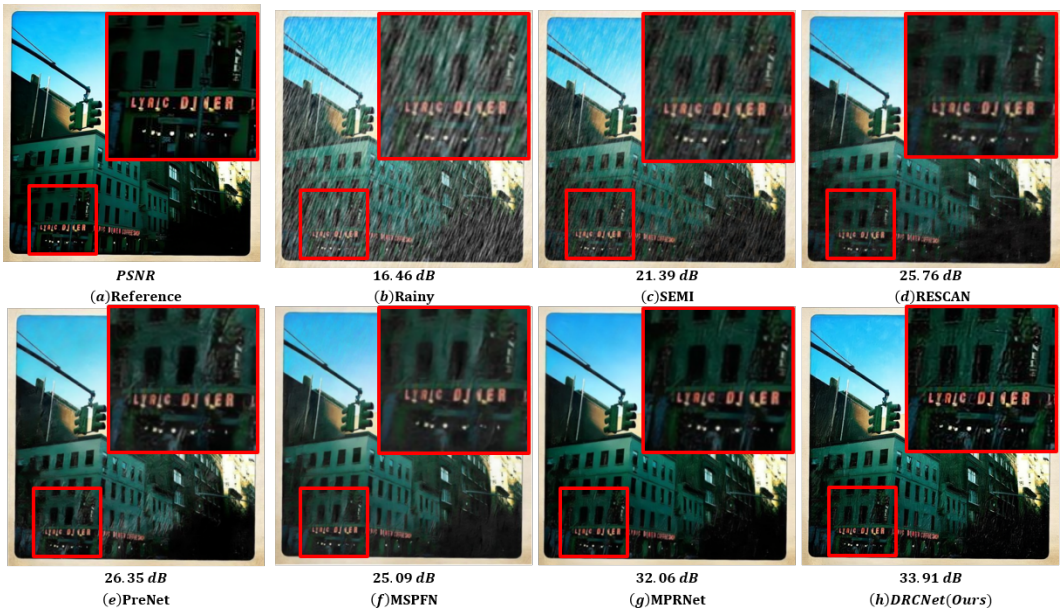

Figure 10. Image deraining on the Test1200 dataset [78].

Figure 11. Image deraining on the Test100 dataset [79].

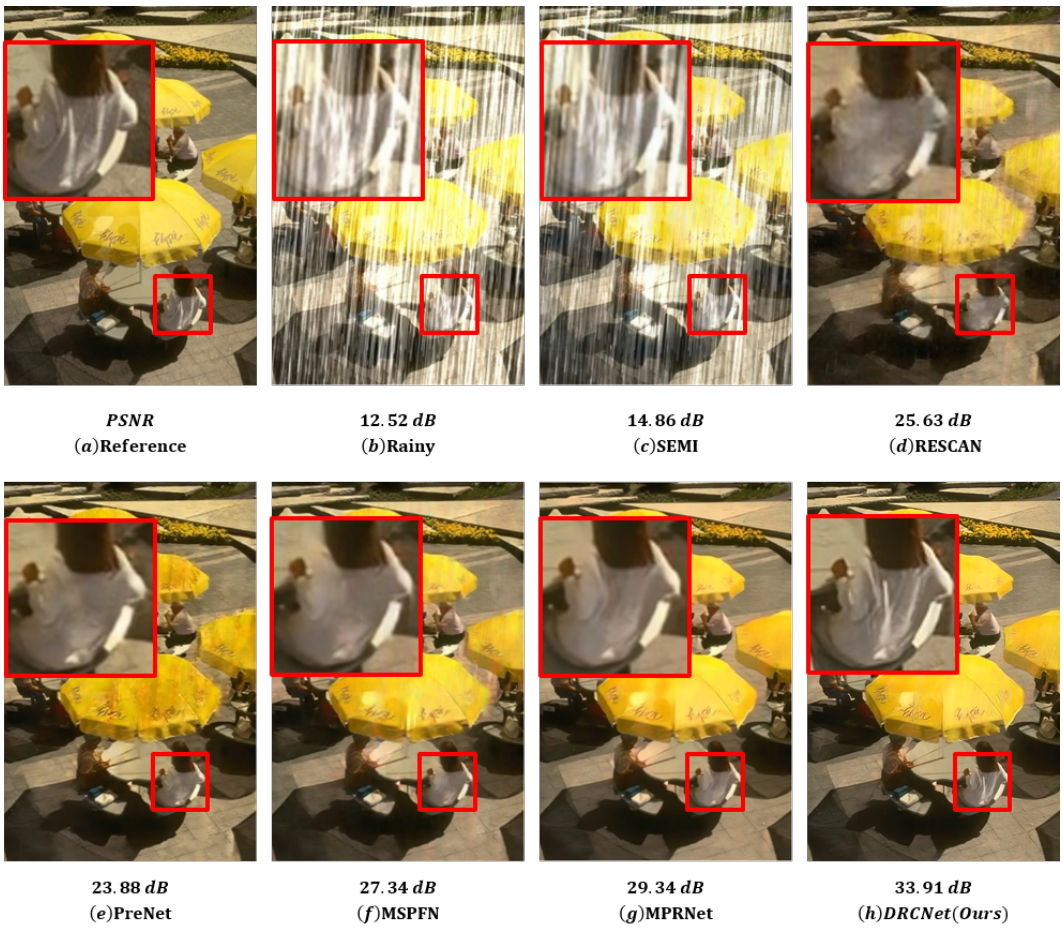

Figure 12. Image deraining on the Rain100H dataset [67].

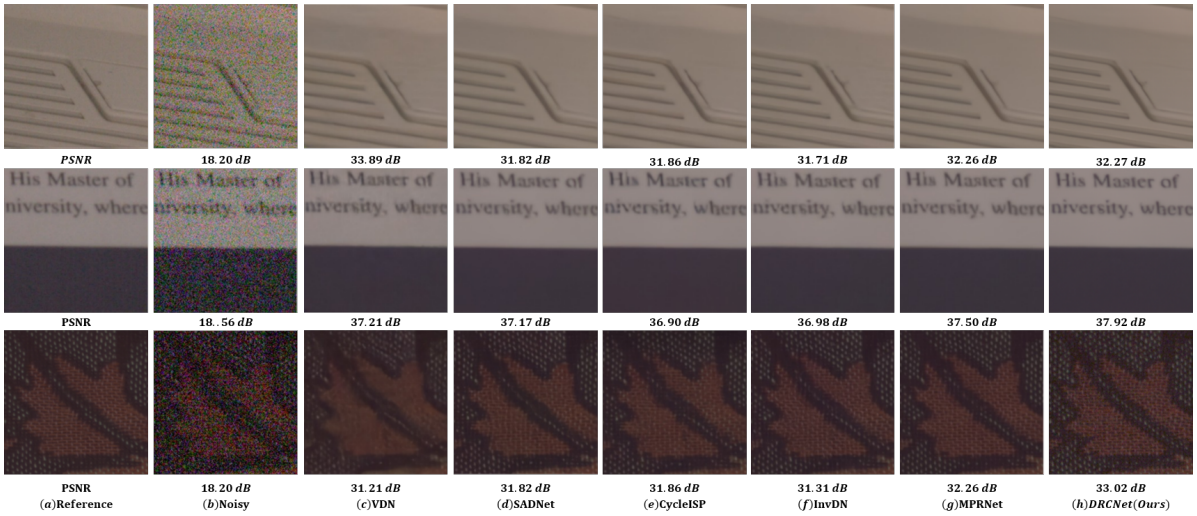

Figure 13. Image denosing on the SIDD dataset [1]

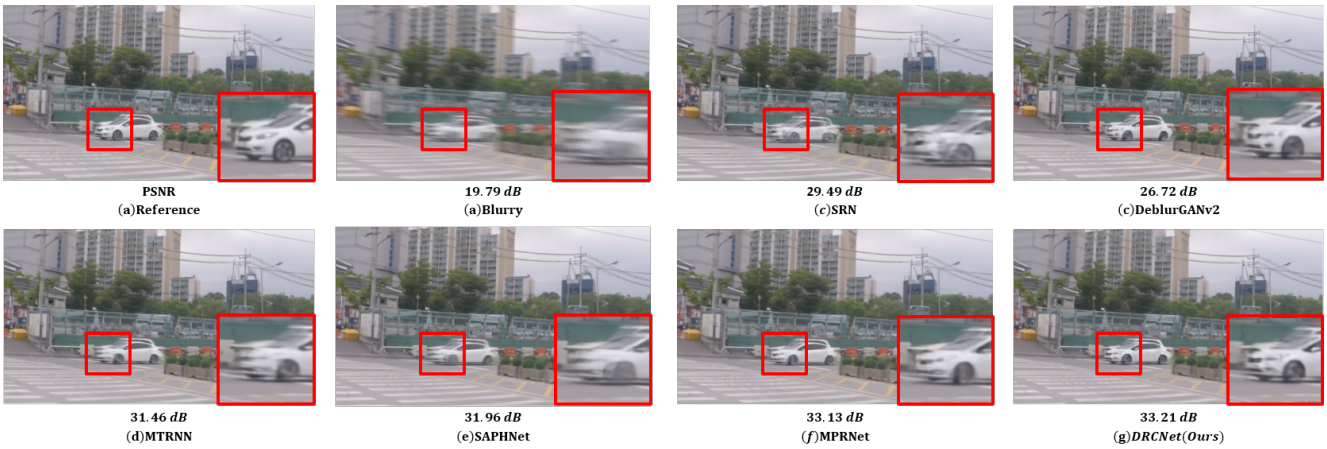

Figure 14. Image deblurring on the GoPro dataset [38]

Figure 15. Image deblurring on the RealBlur-J dataset [47]