Memory Oriented Transfer Learning for Semi-Supervised Image Deraining

Huaibo Huang Aijing Yu Ran He*
National Laboratory of Pattern Recognition, CASIA
Center for Excellence in Brain Science and Intelligence Technology, CAS
Center for Research on Intelligent Perception and Computing, CASIA
School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
{huaibo.huang, aijing.yu}@cripac.ia.ac.cn, rhe@nlpr.ia.ac.cn

Figure 1: Image rain removal in the real world. PReNet [23] and Syn2Real [34] are among the state-of-the-art supervised and semi-supervised methods, respectively. All the methods are trained on DDN-SIRR [29]. The top and bottom rows show that MOSS can remove more rain streaks while preserving better background textures (Note that the missing traffic markings may cause serious accidents in autonomous driving).

Abstract

Deep learning based methods have shown dramatic improvements in image rain removal by using large-scale paired data of synthetic datasets. However, due to the various appearances of real rain streaks that may differ from those in the synthetic training data, it is challenging to directly extend existing methods to the real-world scenes. To address this issue, we propose a memory-oriented semi-supervised (MOSS) method which enables the network to explore and exploit the properties of rain streaks from both synthetic and real data. The key aspect of our method is designing an encoder-decoder neural network that is augmented with a self-supervised memory module, where items in the memory record the prototypical patterns of rain degradations and are updated in a self-supervised way. Consequently, the rainy styles can be comprehensively derived from synthetic or real-world degraded images without the need for clean labels. Furthermore, we present a self-training mechanism that attempts to transfer deraining knowledge from supervised rain removal to unsupervised cases. An additional target network, which is updated with an exponential moving average of the online deraining network, is utilized to produce pseudo-labels for unlabeled rainy images. Meanwhile, the deraining network is optimized with supervised objectives on both synthetic paired data and pseudo-paired noisy data. Extensive experiments show that the proposed method achieves more appealing results not only on limited labeled data but also on unlabeled real-world images than recent state-of-the-art methods.

1. Introduction

Single image deraining (SID), also known as image rain removal, refers to restoring clean and rain-free background scenes from a single rainy image. It is significant for a wide range of outdoor computer-vision tasks, such as au-

*Corresponding Author

tonomous driving and video surveillance, where images captured in rainy days are often heavily degraded in visual quality. SID is a difficult problem since degraded images in the real world may contain rain streaks and accumulation of complex patterns and various appearances.

Recently, deep learning based methods have been introduced into single image deraining and contributed to dramatic improvements. However, most of these CNN based methods [8, 31, 37, 33, 23, 3, 14] rely on paired rain/clean images to train their networks in a fully supervised way. Since the intractability to obtain labeled real rainy images, existing methods are typically trained on synthetic rain datasets [8, 41]. But there are significant gaps between synthetic and real rainy images, where the authentic degradations are much more complex. As a result, the models trained on synthetic datasets may generalize poorly to practical applications in the real world. To address this issue, Wei et al. [29] firstly propose a semi-supervised learning framework to simultaneously utilize supervised and unsupervised knowledge for image deraining. They model the real rain residual through a likelihood term imposed on a Gaussian mixture model and minimize the Kullback-Lerbler divergence between the distributions of synthetic and real rain. Subsequently, Yasarla et al. [34] propose a Gaussian-process (GP) based semi-supervised method that uses GP to model the latent features of rainy images and create pseudo-labels for the unlabeled data. Although existing semi-supervised methods [29, 34] have achieved promising results, it is still challenging to model various appearances of rain and separate complex overlappings of rain and background information for real-world degraded images. For example, as shown in Fig. 1, it is difficult to correctly estimate the rain degradations when background textures (like the traffic markings) have similar appearances with rain streaks. Therefore, real-world rain removal remains an open and challenging problem, leaving much room for improvement.

In this paper, we present a memory-oriented semi-supervised (MOSS) learning framework to fully utilize unlabeled real-world images for better generalization of rain removal. Specifically, we design a memory-oriented encoder-decoder network (MOEDN) to learn the patterns of rain and recover rain-free background images. As illustrated in Fig. 2, MOEDN consists of an encoder to extract rain features from an input image, a memory module to model the appearances of rain degradations, and a decoder to recover rain-free background images. Between the encoder and decoder, a skip-connection following by a subtraction operation is added to ensure that the encoding should focus on rain degradations rather than background information. The memory module is employed to record various appearances of rain, where each item in the memory corresponds to prototypical features of rainy patterns. The encoding of MOEDN is served as a query to retrieve the most relevant items in the memory, and then these items are aggregated based on soft-attentive reading. The memory module is updated in a self-supervised way, i.e., each memory item is updated using an exponential moving average of such query features that the memory item is the nearest one to them. In the training phase, we iteratively optimize the memory module and the rest parts of the network, making it possible to deeply explore the patterns of rain degradations without the need for clean labels.

In addition, we present a self-training mechanism to supervise the unlabeled data by transferring deraining knowledge of rain removal on synthetic datasets. We employ an additional target network, which is updated with an exponential moving average of the online deraining network (i.e., MOEDN) to produce pseudo-labels for unlabeled rainy images. Then, we generate noisy data and their labels by randomly mixing the synthetic/real images as the background and the synthetic or pseudo rain residuals. Finally, the online network is trained by supervised objectives, i.e., the pixel-wise L1 loss, on both synthetic paired data and pseudo-paired noisy data. Furthermore, we adopt a Total-Variance loss function to slightly regularize the smoothness of unsupervised rain removal. The self-training mechanism augments the diversity of rain degradations from both synthetic and real-world rainy images, thus leading to robustness towards complex and volatile rainy scenes in the wild.

The main contributions are summarized as follows:

- A novel memory-oriented transfer learning framework is proposed to conduct semi-supervised image deraining on both labeled synthetic data and unlabeled real-world data.
- A memory-oriented encoder-decoder network is proposed to recover rain-free background images. A self-supervised memory module is presented to adaptively model various appearances of rain degradations.
- A self-training mechanism is proposed to transfer knowledge from supervised deraining to unsupervised cases. The use of noisy data paired with pseudo-labels generated by a target network improves the robustness of image deraining.
- Extensive experiments on different datasets demonstrate that the proposed approach outperforms existing methods both quantitatively and qualitatively.

2. Related Works

2.1. Single Image Deraining

Single image deraining has witnessed significant advance in the past decade. In traditional methods, Luo et

al. [20] present discriminative sparse coding (DSC) to remove rain streaks from the raining part of images and preserve background textures. Li et al. [19] adopt Gaussian mixture models (GMM) to accommodate various types of the rain streaks. Zhu et al. [41] utilize layer-specific priors to judge rainy regions to promote the removing process.

Recently, plenty of deep learning based works have sprung up thanks to the proposal of deep neural network [7, 6, 5]. Fu et al. [8] employ image priori knowledge via paying attention on high-frequency details. Yang et al. [31] present a joint network of rain detection and removal to estimate rain locations and densities. Following [8, 31], many CNN-based methods have been proposed to improve the accuracy of rain removal. According to the research focus, they can be roughly divided into two types, one of which is prior-based and the other is architecture-based (Note many works involve both). The prior-based methods resort to rain-related priors, such as rain density [37], rain mask [36], scene depth [13, 16], confidence maps [33], image segmentations [39], background details [28, 3], and rain layers [27, 25], to guide the separation of rain and background. Besides, many works [22, 28, 40] employ GAN [10] to learn domain specific priors to regularize rain removal. The architecture-based methods attempt to develop advanced network architectures to promote image deraining. Specifically, residual architectures [4, 14], residual sub-networks [24], recurrent frameworks [18, 23, 14], pyramid network [9], and auto-searched architectures [17] are studied to explore multiple-scale features. In addition, researchers set out to add diverse categories of attention modules [22, 15, 18, 13, 26] to proposed networks.

To improve the generalization of deraining in the real world, several previous works [29, 34] based on semi-supervised learning have been proposed. Wei et al. [29] adopt a likelihood term imposed on a Gaussian mixture model and minimize the Kullback-Lerbler divergence between the synthetic and real distributions of rain. Yasarla et al. [34] employ Gaussian-process to model the latent features of rain and generate pseudo-labels to supervise the unlabeled data. Different from them, we utilize a memory module to learn the statistics of rain in a self-supervised way, and a self-training scheme to incorporate the unlabeled data into training the deraining networks.

2.2. Memory Networks

The most classical neural networks with memory are recurrent neural networks (RNNs), including long short-term memory (LSTM) [12] and Gated Recurrent Unit (GRU) [1], which have dominated the domain of processing sequential data through deep learning. To overcome the limitations of RNNs in performing memorization, memory networks were firstly proposed by Weston et al. [30] to reason with an additional memory component for the task of question answering. Then memory-based models have been applied to various tasks, including computer vision ones, such as image captioning [2], image colorizing [35], text-to-image synthesis [42], and video object segmentation [21]. Our work is inspired by these works, but it is the first attempt to augment deraining networks with a memory module that is updated in a self-supervised way, allowing semi-supervised learning for real-world rain removal.

3. Proposed Method

We propose a novel memory-oriented semi-supervised deraining framework for real-world rain removal. Fig. 2 and Fig. 3 illustrate the proposed memory-oriented encoder-decoder network and the self-training mechanism, respectively. In the following, a detailed introduction to each component of our method is given.

3.1. Memory-Oriented Encoder-Decoder Network

3.1.1 Network Architectures

As shown in Fig. 2, the proposed MOEDN consists of an encoder E, a memory module M, and a decoder G. Given an input $x \in \mathbb{R}^{3\times H\times W}$ sampled from a set of rainy images X, the encoder firstly extracts such features $z(x) \in \mathbb{R}^{c\times h\times w}$ that represent the image degradations caused by rain. Then $z(x)$ serves as a query to retrieve the most relevant items in the memory. The memory module $M \in \mathbb{R}^{m\times c}$, where m is the number of memory items, is updated in a self-supervised way to keep each memory item $e_i \in \mathbb{R}^c$ close to such queries that e_i is the most relevant one to them. After updating the memory module, memory-based representations $\hat{z}(x) \in \mathbb{R}^{c\times h\times w}$ are achieved by retrieving again the memory items using $z(x)$ and aggregating them by soft attention. Finally, the decoder predicts rain-free background images from the memory-based representations $\hat{z}(x)$ and the skipped features $s(x)$ from the encoder. During training, the memory module as well as the encoder and decoder are updated iteratively, allowing exploring and recording new patterns of rain degradations.

The encoder consists of a convolution layer that maps the input x into the feature maps $s(x)$, and a stack of residual blocks to extract rain-relevant representations $z(x)$ of size $c \times h \times w$. Symmetrically, the decoder consists of a stack of residual blocks to map the memory-based representations $\hat{z}$ to the rain-residual features $g(x)$ that have the same size with $s(x)$, and a convolution layer followed by a Tanh layer that reconstructs clean background images. An operation of subtraction is injected into the decoder to conduct element-wise subtraction between the skipped features $s(x)$ and the rain-residual features $g(x)$. This ensures that the major components of the encoder, i.e., the residual blocks in E, should focus on extracting rain-relevant features rather than preserving background details that are

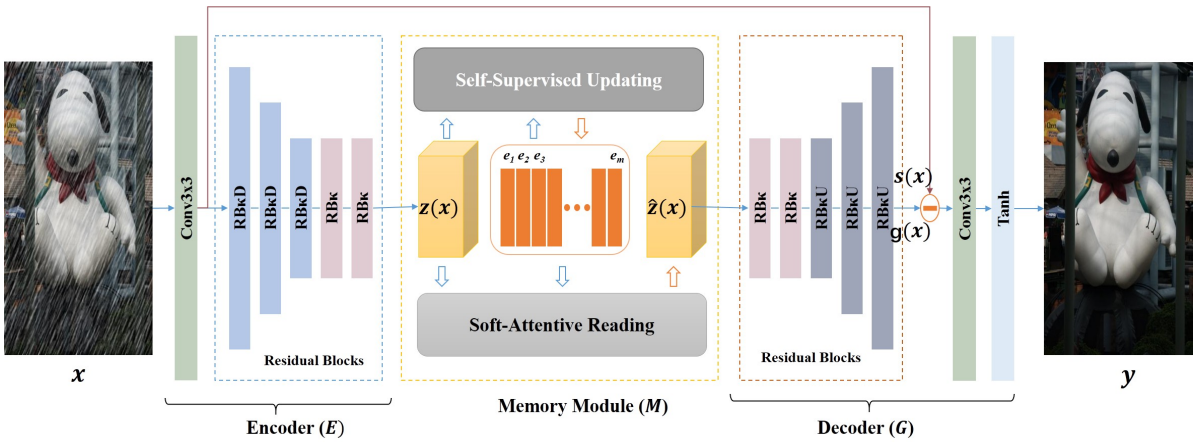


Figure 2: Memory-oriented encoder-decoder network (MOEDN). It consists of an encoder that extracts latent features of rainy degradations, a self-supervised memory module that records various rain degradations, and a decoder that recovers clean background images. $\ominus$ denotes the operation of element-wise subtraction.

irrelevant to rain degradations. Besides, $RB\kappa$ in Fig. 2 denotes a stack of κ basic residual layers (κ is set to be 4 in this paper), and $RB\kappa D$ and $RB\kappa U$ denote those with down-sampling and up-sampling, respectively. More details about the network architectures are in Appendix A.

3.1.2 Self-Supervised Memory Module

As outlined in Fig. 2, the proposed memory $M \in \mathbb{R}^{m\times c}$ consists of m memory items, where the dimension of each item $e_i \in \mathbb{R}^c$ is the same with the channel number of the encoding $z(x) \in \mathbb{R}^{c\times h\times w}$. For simplicity, we reformulate $z(x)$ as $z(x) \in \mathbb{R}^{c\times n} = \{z_1^T(x), ..., z_n^T(x)\}$, where $z_j(x) \in \mathbb{R}^c$ $(j = 1, ..., n)$, and $n = h \times w$. Taking $z_j(x)$ as a query, we retrieve the most relevant memory items and update M in a self-supervised way. Then memory-based representations $\hat{z}(x)$ are achieved through aggregating the memory items based on soft attention.

Self-Supervised Updating. To explore prototypical patterns of rain degradations, the memory M is designed with a self-supervised updating strategy based on the similarity of the query $z(x)$ and the memory items. Firstly, we compute the cosine similarity $s_{ij}(x)$ of the ith memory item e_i of M and the jth column vector $z_j(x)$ of $z(x)$, defined as

$$s_{ij}(x) = \frac{e_i\, z_j^T(x)}{\|e_i\|\|z_j(x)\|}. \tag{1}$$

Then, we retrieve the most relevant memory item $e_{k_{(j)}(x)}$ for $z_j(x)$ using

$$k_{(j)}(x) = \arg\max_i s_{ij}(x). \tag{2}$$

Finally, we update the memory items e_i based on such a query $z_j(x)$ that has the most relevant item $e_{k_{(j)}(x)} = e_i$:

$$e_i \leftarrow \tau\, e_i + (1-\tau)\, \frac{\sum_{x\in X}\sum_{j=1}^{n} \mathbb{1}(k_{(j)}(x) = i)\, z_j(x)}{\sum_{x\in X}\sum_{j=1}^{n} \mathbb{1}(k_{(j)}(x) = i)}, \tag{3}$$

where $\tau \in [0, 1]$ is a decay rate. In practice, we update e_i iteratively with the parameters of the encoder and decoder, where X in Eq. (3) is a batch of rainy images. We term this strategy as self-supervised updating since the moving averages are generated in an unsupervised manner.

Soft-attentive Reading. After updating the memory module, we reconstruct rainy features $\hat{z}(x)$, i.e., memory-based representations, through reading the memory items according to the query $z(x)$. One intuitive manner to attain $\hat{z}(x)$ is based on hard-attention, which directly selects the most similar memory item $e_{k_{(j)}(x)}$ to $z_j(x)$ as the reconstructed feature $\hat{z}_j(x)$, i.e., $\hat{z}_j(x) = e_{k_{(j)}(x)}$, where $k_{(j)}(x)$ is computed by Eq. (2). However, it is intractable for such manner to back-propagate gradients from the decoder to the encoder. To deal with it, we employ a soft-attention based reading strategy to allow gradient back-propagation.

Firstly, we compute again the similarity matrix $S(x) = \{s_{ij}(x)|i = 1, ..., m, j = 1, ..., n\}$ by Eq. (1) with the updated memory items. Then, the attention $A = \{a_{ij}|i = 1, ..., m, j = 1, ..., n\}$ is obtained by a softmax operation:

$$a_{ij} = \frac{\exp(s_{ij})}{\sum_{i=1}^{m}\exp(s_{ij})}. \tag{4}$$

Finally, the memory-based representations are computed by an attention-based aggregation of memory items:

$$\hat{z}_j(x) = \sum_{i}^{m} a_{ij}\, e_i. \tag{5}$$

Note that since the memory items are expected to be up-

dated in a self-supervised manner, they are not updated using back-propagated gradients during memory reading.

3.2. Self-Training Mechanism

To further improve the accuracy of deraining with the help of unlabeled data, we present a self-training mechanism to transfer supervised knowledge of deraining to unsupervised rain removal. As outlined in Fig. 3, the proposed self-training mechanism consists of two processes, one of which is supervised and the other is unsupervised. The total training algorithm is given in Appendix B.

Supervised Deraining. The supervised deraining utilizes labeled data to train the online deraining network f_θ, i.e., MOEDN, where the optimization objective is a pixel-wise L1 loss, defined as

$$L_{SU} = \|f_\theta(x_l) - y_l\|_1, \tag{6}$$

where θ denotes the parameters of the online MOEDN, x_l and y_l are the input and ground-truth image, respectively.

Unsupervised Deraining. Inspired from MoCo [11] that uses a momentum encoder for self-supervised representation learning, we employ an additional target network f_ξ to produce pseudo-labels for unlabeled data. f_ξ is updated with an exponential moving average of the online network f_θ. After each training step, ξ is updated as following:

$$\xi \leftarrow \upsilon\, \xi + (1 - \upsilon)\, \theta, \tag{7}$$

where $\upsilon \in [0, 1]$ is a decay rate.

As outlined in Fig. 3, for every unlabeled image x_u from a rainy image set X_U, we employ the target network f_ξ to produce its corresponding pseudo-label $f_\xi(x_u)$, which makes up the corresponding pseudo-label set Y_P for X_U. Then we obtain a set of rainy residuals

$$R = \{x - y | (x, y) \in (X_L, Y_L) \cup (X_U, Y_P)\}, \tag{8}$$

where (X_L, Y_L) and (X_U, Y_P) are the synthetic and pseudo paired sets, respectively. Finally, we achieve a noisy data set X_N through data augmentation on the image sets, including X_L, Y_L, X_U, and Y_P, together with the rain residual set R. More precisely, we randomly sample an image $\hat{x} \in X_L \cup Y_L \cup X_U \cup Y_P$ with its corresponding label $\hat{y}$ (A clean image's label is itself), and a residual image $r \in R$. The noisy image x_n is computed as following

$$x_n = T(\hat{x} + \alpha\, r), \tag{9}$$

where α is a random value that is sampled from a uniform distribution $U(a, b)$ (specifically, a and b is 0.5 and 1.1 here), and $T(\cdot)$ is a clamp function to ensure x_n has the same range with $\hat{x}$. Hence, we get paired noisy data $(x_n, \hat{y})$. Similar to Eq. (6), we utilize a pixel-wise L1 loss for the augmented data, defined as

$$L_{UN} = \|f_\theta(x_n) - \hat{y}\|_1. \tag{10}$$

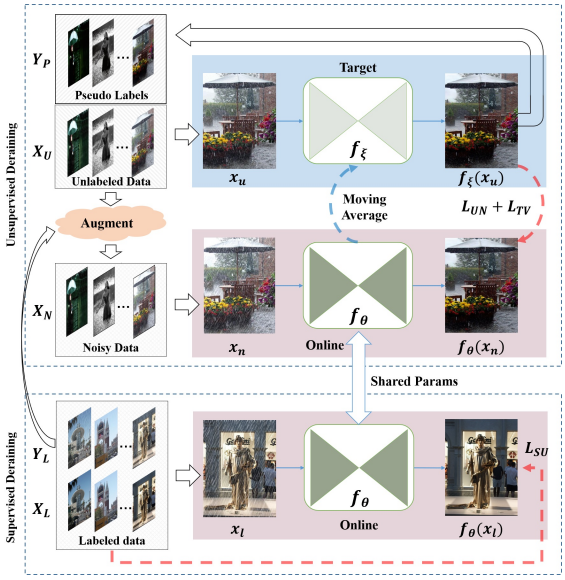


Figure 3: Self-training based deraining. It employs a target network f_ξ to produce pseudo labels for unlabeled data, and then generates noisy data through augmentation from both the synthetic and pseudo paired images. θ of the online network are trained on the synthetic data as well as the augmented noisy data, while ξ of the target network are an exponential moving average of θ.

Note that self-training with augmented noisy data can enrich rain patterns during training and improve robustness towards real-world rain removal.

Total Objective. W adopt a Total Variation regularizer term to smooth the recovered background image $f_\theta(x_n)$:

$$L_{TV} = \|f_\theta(x_n)\|_{TV}. \tag{11}$$

The total objective for the online network f_θ is

$$L_{total} = \lambda_1\, L_{SU} + \lambda_2\, L_{UN} + \lambda_3\, L_{TV}, \tag{12}$$

where $\lambda_{1,2,3}$ are hyper-parameters to balance each item.

4. Experiments

In this section, we evaluate the proposed method against the state-of-the-art methods, including both supervised and semi-supervised ones. We first introduce the datasets and metrics, following by the implementation details. Then we conduct two sets of experiments on semi-supervised deraining. In the first set, we train our network on both labeled synthetic and unlabeled real-world data to evaluate our method in promoting deraining by leveraging real-world rainy images. In the second set, we train our network on synthetic datasets of different percentages of labeled data to evaluate our method on limited labeled data. Finally, we give an analysis of time complexity and an ablation study.

(a) Rainy Image (b) PreNet (c) DRD (d) SIRR (e) Syn2Real (f) Ours (g) Ground Truth

Figure 4: Visual results on DDN-SIRR synthetic test set. Best viewed by zooming in the electronic version.

(a) Rainy Image (b) PreNet (c) DRD (d) SIRR (e) Syn2Real (f) Ours

Figure 5: Visual results on DDN-SIRR real-world test set. Best viewed by zooming in the electronic version.

Table 1: Quantitative results of PSNR on DDN-SIRR synthetic test set. The gain denotes the performance improvement by the use of real-world data $\mathcal{D}_{\mathcal{U}}$.

Dataset	Input	Supervised methods trained on synthetic data $\mathcal{D}_{\mathcal{L}}$							Semi-supervised methods trained on synthetic and real-world data $\mathcal{D}_{\mathcal{L}} + \mathcal{D}_{\mathcal{U}}$								
		JORDER[31] (CVPR '17)	DDN[8] (CVPR '17)	DID-MDN[37] (CVPR '18)	UMRL[33] (CVPR '19)	PReNet [23] (CVPR '19)	MSPFN[14] (CVPR '20)	DRD[3] (CVPR '20)	SIRR [29] (CVPR '19)			Syn2Real [34] (CVPR '20)			Ours		
									$\mathcal{D}_{\mathcal{L}}$	$\mathcal{D}_{\mathcal{L}} + \mathcal{D}_{\mathcal{U}}$	Gain	$\mathcal{D}_{\mathcal{L}}$	$\mathcal{D}_{\mathcal{L}} + \mathcal{D}_{\mathcal{U}}$	Gain	$\mathcal{D}_{\mathcal{L}}$	$\mathcal{D}_{\mathcal{L}} + \mathcal{D}_{\mathcal{U}}$	Gain
Dense	17.95	18.75	19.90	18.60	20.11	20.65	19.54	20.34	20.01	21.60	1.59	20.24	22.36	2.12	20.29	**22.91**	**2.62**
Sparse	24.14	24.22	26.88	25.66	26.94	26.40	26.47	26.04	26.90	26.98	0.08	26.15	27.26	1.11	25.90	**27.78**	**1.88**

4.1. Datasets and Metrics

Datasets. We consider three challenging rain datasets, i.e., the DDN-SIRR dataset created by Wei et al. [29], the Rain200H dataset proposed by Zhu et al. [41], and the Rain800 dataset built by Zhang et al. [38], for semi-supervised deraining experiments. The **DDN-SIRR** dataset [29] is created using both labeled synthetic and unlabeled real-world data for evaluating the performance of semi-supervised deraining. The labeled train set contains $9,100$ image pairs of synthetic rain data, and the unlabeled train set consists of 147 real-world rainy images. The test set contains two types of data: 10 images of dense rain streaks and another 10 of sparse rain streaks. The **Rain200H** dataset [41] contains 1,800 synthetic image pairs in the train set and 200 image pairs in the test set. The **Rain800** dataset [38] comprises 800 synthetic image pairs totally. There are 700 image pairs in the train set and 100 image pairs in the test set.

Metrics. For labeled synthetic data, PSNR and SSIM are calculated on the RGB space using the scikit-image library in Python. For unlabeled real-world images, qualitative comparisons are provided through visual observation. Since no ground-truth labels exist and most of current non-reference metrics for deraining may be not in agreement with visual quality [32], we employ user studies to quantitatively evaluate the visual quality of deraining results.

4.2. Implementation Details

The proposed method is trained on the images of pixel-size 256×256 randomly cropped from the train set and evaluated on the images of arbitrary size in the test set. For the hyper-parameters in Eq. (12), we empirically set λ_1 to be 10, λ_2 to be 1, and λ_3 to be 0.001 to ensure that the network should pay more attention on supervised deraining than unsupervised one. The decay rates in Eq. (3) and Eq. (7) are both set to be a small value, i.e., 0.999, to stabilize the moving average based updating. For each training step with a batch-size of 16, we optimize the memory module and the rest parts of MOEDN iteratively using Eq. (3) and Adam algorithm with a fixed learning rate of 0.0001. For stability, we pre-train the network on labeled data using Eq. (6) during the first 10 epochs. It takes about 100,000 iterations for our network to converge. Code and results will be released.

4.3. Experiments on Real-world Data

We compare our method on the DDN-SIRR dataset [29] against several state-of-the-art methods, including both supervised and semi-supervised ones. For supervised methods, we compare against JORDER [31], DDN [8], DID-MDN [37], UMRL [33], PReNet [23], MSPFN [14], and DRD [3]. They are trained on the labeled train set $\mathcal{D}_{\mathcal{L}}$ of DDN-SIRR. For the semi-supervised methods, we compare against SIRR [29] and Syn2Real [34]. Following the protocols of [29, 34], we train our network on the synthetic train set $\mathcal{D}_{\mathcal{L}}$ and the real-world train set $\mathcal{D}_{\mathcal{U}}$ of DDN-SIRR, and then conduct evaluations on the synthetic test set.

4.3.1 Comparisons on synthetic test set

The quantitative results of PSNR on the DDN-SIRR synthetic test set are reported in Table. 1. The proposed method achieves the best performance compared with the state-of-the-art. Our method performs better than all the supervised methods that merely use labeled synthetic train data $\mathcal{D}_{\mathcal{L}}$. Even though the supervised version of MOSS trained on $\mathcal{D}_{\mathcal{L}}$ achieves only comparable performance against other methods (because our goal is not exploring the most suitable network architectures for deraining), MOSS can improve significantly the accuracy of image deraining through taking advantage of unlabeled real-world data $\mathcal{D}_{\mathcal{U}}$. Besides, our method also achieves better performance than the semi-supervised methods SIRR and Syn2Real. Specifically, the gain value of our method brought by $\mathcal{D}_{\mathcal{U}}$ outperforms SIRR and Syn2Real with significant margins, which implies that our method can utilize real-world data more sufficiently.

We also provide qualitative results on the DDN-SIRR synthetic test set in Fig. 4. It can be observed that our method achieves more promising visual results compared with other methods. Our method and Syn2Real [34] can remove most of rain degradations and recover clean background images, while others preserve more rain streaks. Compared with Syn2Real, our method recovers slightly smoother background scenes (See the background behind the bird in the top row of Fig. 4).

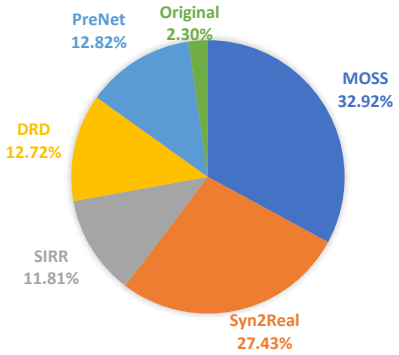


Figure 6: Averaged selection percentage of user study.

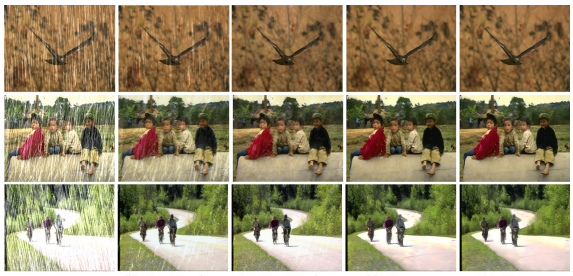
4.3.2 Comparisons on real-world rainy images

Following [29, 34], we also evaluate the proposed method on real-world rainy images. Fig. 1 and Fig. 5 show visual results on the images respectively from Google search and the DDN-SIRR real-world test set. Our method achieves better visual effects as compared to other state-of-the-art methods. PreNet [23] and Syn2Real [34] are among the most promising previous supervised and semi-supervised deraining methods, respectively. Though, our method can remove more rain streaks of various appearances (e.g., thin rain streaks in the top row of Fig. 5) and recover cleaner background scenes (e.g. the bottom row of Fig. 5) while better preserving the structure and details of background. Our method shows superiority in discriminating between rain streaks and background textures in the real world. For example, our method succeed in removing rain streaks while keeping the traffic markings in Fig. 1. This may be attributed to the memory module that records various appearances of rain degradations rather than background details.

Besides, we conduct user studies to evaluate the quantitative performance of real-world rain removal. Fig. 6 shows the averaged selection percentage for each method. As can be observed, our method achieves the best performance for real-world rain removal. For space constraints, more details are provided in Appendix C.

4.4. Experiments on Limited Labeled Data

To demonstrate the effectiveness of the proposed method on limited labeled data, following [34], we conduct experiments on two synthetic datasets, i.e., Rain200H [41] and Rain800 [38], of different percentages of labeled data. Specifically, we run several experiments that train the network using a combination of $\mathcal{D}_{\mathcal{L}}$ and $\mathcal{D}_{\mathcal{U}}$, where $\mathcal{D}_{\mathcal{L}}$ consists of 10%, 20%, 40%, 60%, and 100% paired images,



(a) Rainy Image (b) Syn2Real on D_L (c) Syn2Real on $D_L + D_U$ (d) Ours on D_L (e) Ours on $D_L + D_U$

Figure 7: Results on 10% labeled data from Rain200H.

Table 2: Results on limited labeled data from Rain200H.

$\mathcal{D}_\mathcal{L}$ %	Syn2Real						Ours					
	PSNR			SSIM			PSNR			SSIM		
	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain
10%	22.92	23.64	0.72	0.742	0.767	0.025	25.76	26.06	0.30	0.835	0.841	0.006
20%	23.22	24.00	0.78	0.755	0.776	0.021	26.40	26.87	0.47	0.848	0.858	0.010
40%	23.84	24.75	0.91	0.772	0.794	0.022	26.76	26.97	0.21	0.854	0.859	0.005
60%	24.32	25.26	0.94	0.782	0.808	0.026	26.91	26.99	0.08	0.859	0.861	0.002
100%	25.27	–	–	0.810	–	–	26.99	–	–	0.860	–	–

Table 3: Results on limited labeled data from Rain800.

$\mathcal{D}_\mathcal{L}$ %	Syn2Real						Ours					
	PSNR			SSIM			PSNR			SSIM		
	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain	$\mathcal{D}_\mathcal{L}$	$\mathcal{D}_\mathcal{L}+\mathcal{D}_\mathcal{U}$	Gain
10%	21.31	22.02	0.71	0.729	0.750	0.021	23.28	23.79	0.51	0.785	0.820	0.035
20%	22.28	22.95	0.67	0.752	0.768	0.016	23.96	24.68	0.72	0.810	0.840	0.030
40%	22.61	23.60	0.99	0.761	0.788	0.027	24.50	25.53	1.03	0.814	0.852	0.038
60%	22.96	23.70	0.74	0.775	0.795	0.020	25.39	25.66	0.27	0.844	0.855	0.011
100%	23.74	–	–	0.799	–	–	26.36	–	–	0.848	–	–

and $\mathcal{D}_\mathcal{U}$ consists of the rest rainy images without labels.

It can be observed from Table 2 and Table 3 that our method can improve the performance of deraining by utilizing unlabeled data, which verifies the effectiveness of the proposed semi-supervised deraining framework. Compared with Syn2Real [34], the proposed method achieves better quantitative results in both supervised and unsupervised settings. The reason may be that the patterns of rain in the test set of synthetic datasets may have occurred in the train set $\mathcal{D}_\mathcal{L}$, even when $\mathcal{D}_\mathcal{L}$ is very small. Meanwhile, the proposed method can record the ever-seen patterns, thus leading to better performance on limited labeled data from synthetic datasets. The visual results in Fig. 7 also demonstrate that our method achieves better results on limited labeled data.

4.5. Time Complexity

We compare time complexity of the proposed method against the state-of-the-art models on a single GPU (TITAN RTX). As illustrated in Fig. 8, the proposed method achieves the second best performance on time complexity. It only lags behind DID-MDN [37] with a little margin but achieves more promising results of rain removal (as shown in Table 1 and Fig. 5). Therefore, it can be concluded that the proposed method can achieve pleasing deraining results with a low computation cost.

4.6. Ablation Study

We conduct an ablation study to gain insight into the respective roles of each part of our method in semi-supervised

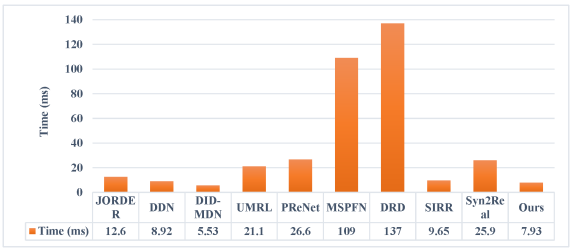


Figure 8: The running time (ms) for a 256×256 image.

Table 4: Results of ablation study on DDN-SIRR.

Dataset	Metrics	Basic	w/o Memory	w/o Self-Training	w/o TV	w/o Skip-Connect	Ours
Dense	PSNR	19.99	22.21	20.99	22.68	22.00	22.91
	SSIM	0.835	0.870	0.860	0.876	0.858	0.883
Sparse	PSNR	25.74	26.82	25.83	27.58	26.80	27.78
	SSIM	0.881	0.906	0.890	0.908	0.900	0.912

image deraining. We utilize the DDN-SSIR dataset [29] to evaluate the performance on rain removal. The proposed network without memory module (i.e., $\hat{z}(x) = z(x)$ in Fig. 2) and self-training is considered as the basic model. We remove individually the memory module, the self-training mechanism, the Total Variation regularizer term L_{TV}, and the skip-connection in MOEDN from MOSS to study the roles of them. As illustrated in Table 4, the performance decreases verify that each component of the proposed method is essential for accurate semi-supervised deraining. Due to space limitations, a detailed discussion of these components as well as the decay rates, i.e., τ in Eq. (3) and υ in Eq. (7), are provided in Appendix D.

5. Conclusion

We proposed a novel memory-oriented semi-supervised method for single image deraining. It attempts to learn rain degradations from both labeled synthetic and unlabeled real-world data. An encoder-decoder network augmented with a self-supervised memory module is developed to recover rain-free background. The memory module can explore and exploit various rain degradations without the need for ground-truth images. Besides, a self-training mechanism is proposed to transfer deraining knowledge from supervised rain removal. Since image deraining is a specific restoration task that shares many similarities with other low-level vision tasks, our method is expected to be extended to various other tasks to boost real-world applications.

Acknowledgments

This work is partially funded by National Key Research and Development Program of China (Grant No. 2020AAA0140001), National Natural Science Foundation of China (Grant No. 62006228), and Youth Innovation Promotion Association CAS (Grant No. Y201929).

References

[1] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734, 2014.

[2] Cesc Chunseong Park, Byeongchang Kim, and Gunhee Kim. Attend to you: Personalized image captioning with context sequence memory networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 895–903, 2017.

[3] Sen Deng, Mingqiang Wei, Jun Wang, Yidan Feng, Luming Liang, Haoran Xie, Fu Lee Wang, and Meng Wang. Detail-recovery image deraining via context aggregation networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14560–14569, 2020.

[4] Zhiwen Fan, Huafeng Wu, Xueyang Fu, Yue Huang, and Xinghao Ding. Residual-guide network for single image deraining. In *ACM International Conference on Multimedia*, pages 1751–1759, 2018.

[5] Chaoyou Fu, Yibo Hu, Xiang Wu, Guoli Wang, Qian Zhang, and Ran He. High-fidelity face manipulation with extreme poses and expressions. *IEEE Transactions on Information Forensics and Security*, 16:2218–2231, 2021.

[6] Chaoyou Fu, Xiang Wu, Yibo Hu, Huaibo Huang, and Ran He. Dual variational generation for low shot heterogeneous face recognition. In *Neural Information Processing Systems*, 2019.

[7] Chaoyou Fu, Xiang Wu, Yibo Hu, Huaibo Huang, and Ran He. Dvg-face: Dual variational generation for heterogeneous face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

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

[9] Xueyang Fu, Borong Liang, Yue Huang, Xinghao Ding, and John Paisley. Lightweight pyramid networks for image deraining. *IEEE Transactions on Neural Networks and Learning Systems*, 2019.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Neural Information Processing Systems*, pages 2672–2680, 2014.

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

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[13] Xiaowei Hu, Chi-Wing Fu, Lei Zhu, and Pheng-Ann Heng. Depth-attentional features for single-image rain removal. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8022–8031, 2019.

[14] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8346–8355, 2020.

[15] Guanbin Li, Xiang He, Wei Zhang, Huiyou Chang, Le Dong, and Liang Lin. Non-locally enhanced encoder-decoder network for single image de-raining. In *ACM International Conference on Multimedia*, pages 1056–1064, 2018.

[16] Ruoteng Li, Loong-Fah Cheong, and Robby T Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1633–1642, 2019.

[17] Ruoteng Li, Robby T Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3175–3185, 2020.

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

[19] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using layer priors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2736–2744, 2016.

[20] Yu Luo, Yong Xu, and Hui Ji. Removing rain from a single image via discriminative sparse coding. In *IEEE International Conference on Computer Vision*, pages 3397–3405, 2015.

[21] Jiaxu Miao, Yunchao Wei, and Yi Yang. Memory aggregation networks for efficient interactive video object segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10366–10375, 2020.

[22] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for raindrop removal from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2482–2491, 2018.

[23] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: a better and simpler baseline. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3937–3946, 2019.

[24] Guoqing Wang, Changming Sun, and Arcot Sowmya. Erl-net: Entangled representation learning for single image deraining. In *IEEE International Conference on Computer Vision*, pages 5644–5652, 2019.

[25] Hong Wang, Qi Xie, Qian Zhao, and Deyu Meng. A model-driven deep neural network for single image rain removal. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3103–3112, 2020.

[26] Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson WH Lau. Spatial attentive single-image deraining with a high quality real rain dataset. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12270–12279, 2019.

[27] Yinglong Wang, Yibing Song, Chao Ma, and Bing Zeng. Rethinking image deraining via rain streaks and vapors. In *European Conference on Computer Vision*, 2020.

[28] Zheng Wang, Jianwu Li, and Ge Song. Dtdn: Dual-task deraining network. In *ACM International Conference on Multimedia*, pages 1833–1841, 2019.

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

[30] Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. In *International Conference on Learning Representations*, 2015.

[31] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1357–1366, 2017.

[32] Wenhan Yang, Robby T Tan, Shiqi Wang, Yuming Fang, and Jiaying Liu. Single image deraining: From model-based to data-driven and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

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

[34] Rajeev Yasarla, Vishwanath A Sindagi, and Vishal M Patel. Syn2real transfer learning for image deraining using gaussian processes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2726–2736, 2020.

[35] Seungjoo Yoo, Hyojin Bahng, Sunghyo Chung, Junsoo Lee, Jaehyuk Chang, and Jaegul Choo. Coloring with limited data: Few-shot colorization via memory augmented networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11283–11292, 2019.

[36] Weijiang Yu, Zhe Huang, Wayne Zhang, Litong Feng, and Nong Xiao. Gradual network for single image de-raining. In *ACM International Conference on Multimedia*, pages 1795–1804, 2019.

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

[38] 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*, 2019.

[39] Kaihao Zhang, Wenhan Luo, Wenqi Ren, Jingwen Wang, Fang Zhao, Lin Ma, and Hongdong Li. Beyond monocular deraining: Stereo image deraining via semantic understanding. In *European Conference on Computer Vision*, 2020.

[40] Ziqiang Zheng, Yang Wu, Xinran Han, and Jianbo Shi. Forkgan: Seeing into the rainy night. In *European Conference on Computer Vision*, 2020.

[41] Lei Zhu, Chi-Wing Fu, Dani Lischinski, and Pheng-Ann Heng. Joint bi-layer optimization for single-image rain streak removal. In *IEEE International Conference on Computer Vision*, pages 2526–2534, 2017.

[42] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5802–5810, 2019.