ERL-Net: Entangled Representation Learning for Single Image De-Raining

Guoqing Wang[1,2], Changming Sun[2,1] and Arcot Sowmya[1]
[1]University of New South Wales, Australia, [2]CSIRO Data61, Australia
{guoqing.wang,changming.sun}@csiro.au, a.sowmya@unsw.edu.au

Abstract

Despite the significant progress achieved in image de-raining by training an encoder-decoder network within the image-to-image translation formulation, blurry results with missing details indicate the deficiency of the existing models. By interpreting the de-raining encoder-decoder network as a conditional generator, within which the decoder acts as a generator conditioned on the embedding learned by the encoder, the unsatisfactory output can be attributed to the low-quality embedding learned by the encoder. In this paper, we hypothesize that there exists an inherent mapping *from the low-quality embedding to a latent optimal one, with which the generator (decoder) can produce much better results. To improve the de-raining results significantly over existing models, we propose to learn this* mapping *by formulating a residual learning branch, that is capable of adaptively adding residuals to the original low-quality embedding in a* representation entanglement *manner. Using an embedding learned this way, the decoder is able to generate much more satisfactory de-raining results with better detail recovery and rain artefacts removal, providing new state-of-the-art results on four benchmark datasets with considerable improvements (i.e., on the challenging Rain100H data, an improvement of 4.19dB on PSNR and 5% on SSIM is obtained). The entanglement can be easily adopted into any encoder-decoder based image restoration networks. Besides, we propose a series of evaluation metrics to investigate the specific contribution of the proposed entangled representation learning mechanism. Codes are available at* ⟨https://github.com/RobinCSIRO/ERL-Net-for-Single-Image-Deraining⟩.

1. Introduction

To improve the utility of advanced outdoor computer vision systems, such as smart video surveillance and autonomous driving cars, they have to be designed to deal with challenging weather conditions including rain, snow or haze [22, 26, 17, 10]. Recently, benefiting from the invention of convolutional neural networks, especially the design of the pix2pix network architecture [15] and the adversarial training strategy [8], the performance of single image de-raining has experienced significant progress. By training a rainy-to-clean image translation model with synthetic rain streak or raindrop datasets, a rainy image can be well restored removing the artefacts despite the existence of rain streaks or raindrops with different scales, shapes, and densities. However, the blurry results with missing detail (as shown in Fig. 1) resulted from existing network formulations [22, 26, 27] leave room for possible improvement with better formulations or network architectures.

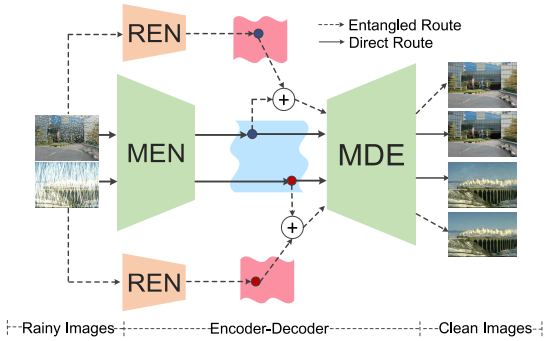

Figure 1: Illustration of the proposed entangled representation learning mechanism. By combining the embeddings from both MEN and REN, a better conditional embedding is formed and used by the MDE to generate a much better de-raining result. 'REN', 'MEN', and 'MDE' denote residual encoder, main encoder, and main decoder respectively as shown in Fig. 2.

Consider this experiment: Given a rainy dataset (*e.g.*, the raindrop data from [22]), and an image-to-image translation network (*e.g.*, U-Net [24]), a comparative experiment was conducted by training the network for two tasks (rainy-to-clean translation and clean-to-clean translation) with the ℓ_1 reconstruction loss. When calculating the PSNR and SSIM of these two tasks, the result of the clean-to-clean translation model (with PSNR=49.51dB and SSIM=0.9948) was unsurprisingly far better than that of the rainy-to-clean translation model (with PSNR=31.67dB and SSIM=0.9195) due to the difference in the input images. By interpreting the image-to-image translation model as a conditional generator with the decoder acting as a generator conditioned on the embedding learned by the encoder, the difference in inputs can be further interpreted as the difference in the learned embeddings. For the conditional generator (decoder) which

is trained to produce the restored output from the learned embedding, high-quality output can be obtained if the conditional embedding can depict the properties (*e.g.*, texture and color) of the desired output image as well as possible. For the encoder taking the clean image as input, the embedding with such properties can be implicitly learned [3]. For the encoder taking the rainy image as input, however, such an embedding cannot be learned well due to the influence from the rainy pixels in the raindrop images or rain streak images. In other words, the conditional embedding learned by the encoder taking the rainy image as input will be biased towards revealing the property of rain-invariance while losing the other essential properties (*e.g.*, texture and color), thus being unable to depict the desired output well. With such an embedding, it is possible to produce an image without the effect of rains but it is difficult to obtain an output recovering sharp details well due to the absence of other properties.

Motivated by this observation, better de-raining results can be obtained by improving the quality of the conditional embedding. Most of the existing de-raining methods can also be implicitly interpreted as improving the conditional embedding by fusing the rain density label with the learned embedding [27], or introducing the rainy region detection map [26] or attention map [22] as auxiliary inputs. However, those formulations are designed to improve the property of rain-invariance, which can be guaranteed or improved by simply designing a better network architecture [19]. Without any improvement in the embedding for depicting other essential properties, the results of all existing models can be further improved with a better formulation.

To this end, we propose an entangled representation learning model (ERL-Net) consisting of a two-branched encoder (as shown in Fig. 2). Specifically, the model is **first** trained to learn a basic embedding (with the main encoder and decoder) depicting the property of rain-invariance. We hypothesize that there exists a smooth connection from the basic embedding to the optimal one in the latent embedding space, and we should be actually able to learn such a connection through a mapping function that adds the residuals as shown in Fig. 1. This argument is supported by the feature equivariance theory [18], which finds that the representation of deep layers in a network depends on the transformations of the input image, and such transformations can be learned by a mapping function from data and the function can be subsequently utilized to manipulate the representation (basic embedding) of an input image to achieve the desired transformation (from basic embedding to the optimal one). To learn such a mapping function, another encoder branch is designed and trained in the **second** stage for adding the residuals to the basic embedding such that the rectified embedding represents more complete properties covering both rain-invariance and other essential factors (*e.g.*, texture and color). In the **third** stage, the overall model (two encoders and one decoder) is fine-tuned to reach better compatibility among all these three modules, and new state-of-the-art de-raining results are consequently obtained.

The contribution of this work is four-fold:

1. An entirely new perspective for analyzing a de-raining network is provided by decomposing the rainy-to-clean image translation model as a combination of an embedding learning net (encoder) and a conditional generator (decoder). Based on this interpretation, an entangled representation learning mechanism is proposed and realized with a simple yet effective network for obtaining better single image de-raining results.
2. The proposed residual learning branch is easy to implement, and can be integrated into any image-to-image translation-based image restoration framework for better performance. It does not alter the original dimensionality of the embedding, thus can be trained end-to-end after being stitched to an existing model.
3. A group of evaluation metrics is proposed for dissecting how ERL-Net improves the de-raining results by the incorporation of the entangled representation learning mechanism. Such simple metrics could potentially be used together with the generic metrics (*e.g.*, PSNR and SSIM) to better analyze the effect of other image de-raining proposals in future.
4. Extensive experiments are conducted on three rain streak datasets and one raindrop dataset, and comparisons against several recent state-of-the-art approaches are carried out to show the significant improvements using the proposed formulation. On the challenging Rain100H dataset for example, we achieve a PSNR of 34.57dB, bringing in a large improvement of 4.19dB over existing state-of-the-art.

2. Related Work

2.1. Single Image De-raining

Recently, benefiting from the incredible learning capability of convolutional neural networks (CNN), the result of the single image de-raining task was significantly improved by training a rainy-to-clean image translation model with synthetic rain streak or raindrop datasets. Fu *et al*. [6] first proposed to synthesize a large-scale rain streak dataset and used it to learn an end-to-end negative residual mapping network for rain streak removal. To improve the de-raining results for more challenging scenarios, Yang *et al*. [26] constructed a more diversified dataset, with which a contextualized network was trained and demonstrated to achieve better results. We argue that the improvement mainly comes from the introduction of the rain streak detection map, serving as prior information guiding the network to focus more on the rain streak regions. Two other similar solutions [22, 27] that introduced rain region related information for better removal were proposed: Zhang *et al*. [27] took advantage of the rain

density label to guide the learning of the rainy-to-clean mapping network. Compared with the utilization of rain density labels which call for extra annotation efforts, Qian *et al*. [22] proposed to directly use the residual between the rainy image and the corresponding clean one to generate the rainy region related map, and used it as ground truth to train a recurrent network for generating attention maps to guide the de-raining network learning. Despite the improvements achieved by these methods [22, 26, 27, 6], we argue that the extra information (*e.g*., density label and attention map) introduced in their formulation can only help improve the rain-invariance property of the learned embedding, which can be obtained instead without using that extra information by constructing a better network architecture [19]. In other words, it is possible to improve the results with the existing formulation by introducing a better latent embedding learning mechanism such as the proposed entangled representation learning network, and the effectiveness of such a network is demonstrated with better de-raining results.

2.2. Representation Learning and Image-to-Image Translation

Learning a mapping function to transform a high-dimensional input image to another high-dimensional output image is a much more challenging task for CNN [15, 30] than learning the transformation from an image to a low-dimensional decision space (*e.g*., class labels). Thanks to the invention of the adversarial training strategy [8] and the pix2pix translation framework [15], image-to-image translation results have improved drastically. Since then, a series of frameworks based on the pix2pix network have been built for learning to perform image-to-image translation tasks including image manipulation [25], text to image translation [28], and image restoration (*e.g*., single image rain streak removal [19] and raindrop removal [22]). However, as evidenced in [30, 7], despite obtaining a plausible output by training a pix2pix network with paired images, the results are usually different from the ground truth. Motivated by the argument from InforGAN [3] that the embeddings naturally decompose into a set of semantically meaningful factors of variation, it is acceptable to attribute the unsatisfactory results to the low-quality semantic embedding if we decompose the image-to-image translation network into a combination of a latent embedding learning subnetwork (encoder) and a conditional generator subnetwork (decoder). Without a good embedding (covering different variation factors of the desired output) as the condition, it is no wonder that the decoder is unable to produce a satisfactory result. For the case of image de-raining, the reason for the unsatisfactory results can be interpreted as the learned embedding can only represent the factor of rain-invariance well but being unable to include most of the other essential factors such as texture or color, thus resulting in an output with missing detail as can be observed in Fig. 4.

3. Entangled Representation Learning Model

To generate satisfactory de-raining results by way of learning a more complete embedding, a simple yet effective approach, termed as entangled representation learning, is proposed, and the formulation and network architecture are presented in this section.

3.1. Problem Formulation

The notion of feature equivariance [18] motivates our method. In [18], the authors proposed to model how a learned representation would change upon transformations of the input image, and they found that such representation changes induced by a transformation of the input image are predictable and can be learned empirically from data.

Formally, a CNN can be interpreted as a complex nonlinear function Φ that maps an image $\mathbf{x} \in \mathcal{X}$ to a semantic feature vector $\Phi(\mathbf{x}) \in \mathbb{R}^d$. The function Φ is proved to be approximately equivariant to a transformation κ of the input image if such a transformation can also be transferred to the representation output [18]. That is, equivariance with κ can be guaranteed when there exists an inherent map $M_\kappa : \mathbb{R}^d \rightarrow \mathbb{R}^d$ that can be learned as follows:

$$\forall \mathbf{x} \in \mathcal{X} : \Phi(\kappa \mathbf{x}) \approx M_\kappa \Phi(\mathbf{x}) \,, \tag{1}$$

Similarly, our goal in this study is to design a light-weight solution to make the learned embedding of the input rainy image to retain more coverage (including the properties of both rain-invariance and other essential factors). In such a case, function Φ corresponds to the encoder, and the embedding obtained with Φ on rainy image $\mathbf{x}$ is $\Phi(\mathbf{x})$. Specifically, let us denote the rainy image and its corresponding clean image as $\mathbf{x}_r$ and $\mathbf{x}_c$ respectively. According to Eq. (1), we wish to obtain a transformed embedding of a rainy image $\mathbf{x}_r$ via a mapping function M_κ, so that $M_\kappa \Phi(\mathbf{x}_r) \approx \Phi(\mathbf{x}_c)$. To facilitate the incorporation of $M_\kappa \Phi(\mathbf{x}_r)$ into a main encoder-decoder network, as shown in Fig. 2, we formulate it as the sum of the original rainy image embedding $\Phi(\mathbf{x}_r)$ with residuals given by a residual function $\mathcal{R}(\mathbf{x}_r)$:

$$M_\kappa \Phi(\mathbf{x}_r) = \Phi(\mathbf{x}_r) + \mathcal{R}(\mathbf{x}_r) \approx \Phi(\mathbf{x}_c) \,, \tag{2}$$

By performing this simple residual guided transformation, a better embedding of the input rainy image is obtained. Such an embedding should be able to cover properties of rain-invariance plus other essential factors, because it can be regarded as an approximate estimation of the latent embedding with the corresponding clean image.

For the network architecture designed for obtaining the entangled embedding, two simple operations as shown in Fig. 2(a) and Fig. 2(b) may be considered in principle. Besides the proposed residual sum strategy, another approach in terms of feature concatenation [24], which is commonly used in disentangled representation recombination-based image-to-image translation task [4, 13], should also work. However, such an operation is not considered because it is said to improve the embedding by way of increasing the dimension

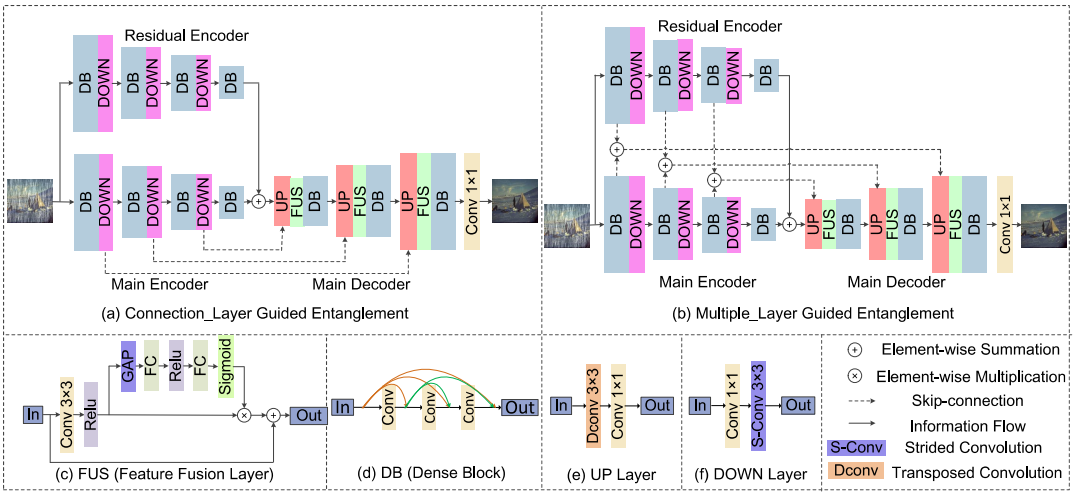

Figure 2: Overview of the proposed entangled representation learning network structure.

of the feature space, which will undoubtedly destroy the original distribution and bring in more uncertainty.

3.2. Network Architectures

To deal with rain streaks or raindrops at different scales, shapes, and densities, many task-specific modules have been designed to build an effective de-raining model, including residual blocks [11], multi-branch dilated convolution [26], multi-stream dense blocks [27], and even recurrent layers [22]. Despite the improved performance obtained by these modules, it is difficult to analyze the specific contributions of different modules and also their combinations.

Our basic network simply stacks the dense block [12] under the U-Net structure [24] because (1) we argue that the proposed entangled representation learning mechanism should be applicable to other image-to-image translation problems, and the construction of a general and simple network architecture with such a mechanism results in an easy-to-use framework for other problems; (2) as claimed in [12], the desired properties of multi-scale, better information reuse, and more stable gradient backpropagation can be fulfilled implicitly by the cross-layer connected dense block.

Feature Fusion Layer. The FUS is proposed to address the issue of feature incompatibility [14] induced by the skip-connections within the U-Net structure [24]. Since the skip-connection was proposed with an encoder-decoder network for better feature reuse [14], it has become a default setup in most image-to-image translation networks [22, 24, 19, 28]. Despite the large performance improvement by the use of skip-connections in our model (as demonstrated in Section 4.2), we argue that such a direct feature map concatenation between layers at different representation levels will bring in feature incompatibility due to the inherent feature hierarchy[1] within a deep CNN. Such an inherent issue results in insufficient exploitation of the skip-connection, leaving room for possible performance improvement with a better skip-connection strategy. To this end, a simple module (as shown in Fig. 2(c)) consisting of feature map recalibration and residual guided layers combination is proposed and incorporated as a post-processing layer immediately after each skip-connection layer in the decoder. Better results by the feature-fusion layer are demonstrated in Section 4.2.

Architecture for Representation Entanglement. To facilitate the proposed residual guided representation entanglement formulation in Eq. (2), a succinct network is constructed as shown in Fig. 2. Built on an encoder-decoder network, an additional encoder branch is introduced for residual learning. It should be noted that the residual branch can be designed as any network structure only if the corresponding dimension for summation is kept consistent. In our implementation, the residual branch is designed to be exactly the same structure with the main branch for simplicity. Besides, due to the existence of skip-connection, two different implementations of feature entanglement are introduced. Specifically, the simpler one (as shown in Fig. 2(a)) is constructed by only focusing on improving the representation in the very middle layer connecting the encoder and decoder. Another approach (as shown in Fig. 2(b)) is realized by introducing the entangled operation into both the middle layer and every skip-connecting layer in the encoder. Such an architecture can implicitly help reduce the representation gaps in a layer-by-layer manner. The performance differences for these two approaches are provided in Section 4.2.

Loss Function. With the proposed entanglement representation learning mechanism, losses in both the pixel level and feature level should be considered. For the loss functioning on the de-rained image, unlike most of the existing de-raining networks [22, 27] that take both the reconstruction loss and other high-level loss (*e.g.*, perceptual loss and adversarial loss) into consideration, only the reconstruction loss is considered in our model because: (1) when combining different losses, it is difficult to find the best configuration

[1]Feature hierarchy means that as the layers become deeper within a CNN, the semantic properties become more abstract [1].

of loss weight to obtain the most appealing results [29]; (2) different setups for perceptual loss (*e.g.*, combination of outputs from different layers in a pre-trained CNN [27, 16]) and the instability in training a discriminator [8, 9] make it tedious to try multiple loss combinations. Thus, the simple reconstruction loss adopted in our work is:

$$L_{\text{pre}} = \|I_{\text{R}} - I_{\text{DR}}\|_1 \,, \tag{3}$$

where L_{pre} indicates the pixel level reconstruction loss, I_{R} is the input rainy image, and I_{DR} is the de-raining result.

Apart from the ℓ_1 loss in Eq. (3), another feature level reconstruction loss is introduced for training the residual branch, and a pre-trained clean-to-clean image translation network is used to generate the ground truth $\Phi(\mathbf{x}_c)$, with which we minimize the following loss:

$$L_{\text{fre}} = \|\Phi(\mathbf{x}_r) + \mathcal{R}(\mathbf{x}_r;\, \theta_{\mathcal{R}}) - \Phi(\mathbf{x}_c)\|_1 \,, \tag{4}$$

where L_{fre} indicates the feature level reconstruction loss and it includes one term for the network in Fig. 2(a) and multiple terms for the network in Fig. 2(b). $\Phi(\mathbf{x}_r)$ is the layer output from the main encoder branch, $\mathcal{R}(\mathbf{x}_r;\, \theta_{\mathcal{R}})$ is the layer output from the residual branch, and $\theta_{\mathcal{R}}$ denotes the parameters of that branch, $\Phi(\mathbf{x}_c)$ is the ground truth as mentioned above.

By combining these two reconstruction losses, the overall network can be trained end-to-end by minimizing the following loss term:

$$L_{\text{ERL-Net}} = L_{\text{pre}} + \lambda_{\text{fre}} L_{\text{fre}} \,, \tag{5}$$

where λ_{fre} is set to 0.01 empirically.

To train the overall entangled representation learning network, these three loss formulations in Eqs. (3)-(5) are used stage-wise according to different training strategies, and we provide the details in the following section.

3.3. Training of ERL-Net

By implementing the representation entanglement mechanism in an efficient and flexible manner, two different strategies (as summarized in Table 1) for training the ERL-Net with the residual branch are designed to find out the best model setup, and comprehensive results on all the strategies will be provided in the experiments section.

Module Assembly (MA). For this strategy, four different approaches are involved with the first stage consistently designed as: a de-raining network consisting of a main encoder (MEN) and a main decoder (MDE) is trained with the L_{pre} loss, and a clean-to-clean image translation network with the same structure is also trained for the L_{fre} loss calculation in the following stages. Based on the pre-trained de-raining network, the residual branch (REN) can be directly incorporated and then trained with four different setups: For the two-staged models (MA_A and MA_B), the $L_{\text{ERL-Net}}$ loss is adopted to update the parameters within REN, while MA_B also considers the update of MDE. Furthermore, based on MA_A, the L_{pre} is adopted to update MDE or the combination of REN and MDE in the third stage, resulting in two three-staged models termed as MA_C and MA_D.

Models		Training branch split and adopted loss functions		
		Stage-I	Stage-II	Stage-III
MA_A	Branch	(MEN, MDE)	REN	—
	Loss	L_{pre}	$L_{\text{ERL-Net}}$	—
MA_B	Branch	(MEN, MDE)	(REN, MDE)	—
	Loss	L_{pre}	$L_{\text{ERL-Net}}$	—
MA_C	Branch	(MEN, MDE)	REN	MDE
	Loss	L_{pre}	$L_{\text{ERL-Net}}$	L_{pre}
MA_D	Branch	(MEN, MDE)	REN	(REN, MDE)
	Loss	L_{pre}	$L_{\text{ERL-Net}}$	L_{pre}
DT		One stage, $L_{\text{ERL-Net}}$ to train the overall network		

Table 1: Different strategies for training the entangled representation learning and de-raining network (REN, MEN, and MDE denote branches of residual encoder, main encoder, and main decoder respectively as illustrated in Fig. 2).

Direct Training (DT). Given a plain encoder-decoder de-raining network without pre-training, the residual branch is incorporated and the overall network can be trained end-to-end with the $L_{\text{ERL-Net}}$ loss. Obviously, the performance improvement will not be guaranteed without the explicit stage-wise representation entanglement mechanism.

4. Experimental Results

4.1. Settings

Datasets and Metrics. For rain streak removal experiments, three challenging benchmark synthetic datasets are considered including the DDN-Data collected by Fu *et al*. [6], the DID-Data synthesized by Zhang *et al*. [27], and the Rain100H dataset provided in [26]. For raindrop removal, a relatively large-scale raindrop dataset (denoted as AGAN-Data [22]) is used. Following the training and testing dataset split described in [22, 26, 27, 6], our model is trained and evaluated quantitatively on all these four datasets with the metrics of PSNR and SSIM on the Y channel (*i.e.*, luminance) of the transformed YCbCr space. Besides, comprehensive ablation study is conducted on the Rain100H and the AGAN-Data for rain streak and raindrop removal analysis respectively. For the qualitative evaluation with real data, the performance is compared by visual observation because there is no rain-free ground truth for real-world images.

Training Details. During training, a 320×320 patch is randomly cropped from each rainy image and used directly for training without any data augmentation. We used a batch size of 1 and trained our ERL-Net for 400 epochs on all datasets on a Tesla P100 GPU. We implemented the models and evaluated all experiments in TensorFlow, and adopted the Adam optimizer with default settings for parameter updates. The learning rate was initialized to 0.0004 and was divided by 10 when the losses plateau. Code is publicly available.

4.2. Network Dissection

Basic Model. Before investigating the effect of the proposed entangled representation learning mechanism, a simple ablation study was conducted to derive a powerful basic model and also to verify the statement (in Section 3.2) on the importance of skip-connection for the de-raining model

design. Removing the residual branch shown in Fig. 2, a simple baseline model termed as MEMD is constructed, on which two variants are implemented by removing the feature fusion layer (denoted as MEMD-) and then removing the skip-connection (denoted as MEMD- -). Comparisons on performance difference of these three models are carried out on the AGAN-Data for raindrop removal and on the Rain100H data for rain streak removal, and the qualitative results are shown in Table 2.

		MEMD- -	MEMD-	MEMD
Skip-Connection		×	√	√
Feature Fusion		×	×	√
AGAN-Data	PSNR	30.4712	31.4305	32.2139
	SSIM	0.8226	0.9186	0.9301
Rain100H	PSNR	24.0743	29.3782	31.6310
	SSIM	0.7315	0.8327	0.8668

Table 2: Quantitative results by different setups on the basic model.

As demonstrated by the improvement with skip-connection and further with the proposed feature fusion block, the MEMD model is designed as the basic structure for the entangled representation learning model.

Effective ERL-Net Setup Determination. As discussed in Section 3.3, the flexibility of the proposed entanglement formulation enables a series of model setups with different training strategies. Besides, as illustrated in Fig. 2(a) and (b), two different entanglement approaches including the CLGD (connection_layer guided entanglement) and MLGD (multiple_layer guided entanglement) are proposed for determining the best formulation. By adding the residual branch to the basic model MEMD, those strategies and entanglement approaches are investigated one by one by training different ERL-Nets, and the corresponding results on both the AGAN-Data and the Rain100H data are listed in Table 3.

As indicated by the results in Table 3, nearly all the setups (except the MA_A and DT strategies for Rain100H) are able to outperform the existing state-of-the-art, fully demonstrating the effectiveness of the proposed formulation and network architecture. Among these powerful models, the combination of MA_D and MLGD achieves the best results on both datasets, thus being used as the baseline setup of ERL-Net[2] for the following analysis and comparisons.

On the other hand, two other general conclusions can be drawn from the results in Table 3: (1) The results with the strategy of DT are worse than that of MS on both datasets regardless of the entanglement approaches (CLGD or MLGD), demonstrating the importance of stage-wise training strategy for the proposed entangled representation learning mechanism; (2) The results of different setups with MLGD are generally better than that of CLGD on both datasets regardless of the specific training strategy, indicating the superiority of the layer-wise feature rectification method which guarantees a more smooth feature property transition in a layer-by-layer manner. Meanwhile, it also reveals the importance of skip-connection implicitly.

Furthermore, by comparing the values in Table 3 and the values of MEMD in Table 2, the results from DT and MEMD are similar and are both worse than that from MS, demonstrating that the performance improvement comes from the representation entanglement mechanism instead of the increase in parameters with the residual encoder branch.

[2]In the following section, ERL-Net specifically corresponds to the model with the setup of combining MA_D and MLGD.

4.3. How ERL-Net Changes De-Raining Result?

With the best-configured ERL-Net obtained in Section 4.2, we elaborately analyze the benefits from the proposed entangled representation learning mechanism on all the four benchmark datasets.

Statistical Distribution of Changed Samples. For the first group of experiments, we propose to analyze the general superiority of the proposed ERL-Net over a basic rainy-to-clean image translation model (MEMD) by counting how many samples are better restored by ERL-Net instead of MEMD. For a specific rainy dataset, we first train both MEMD and ERL-Net on the training set, and then record the PSNR of the testing images. After obtaining the PSNR values from MEMD and ERL-Net for each test image, the proportion of samples with better results from ERL-Net over MEMD are calculated. Following this procedure, such proportion for four different datasets are obtained as in Fig. 3.

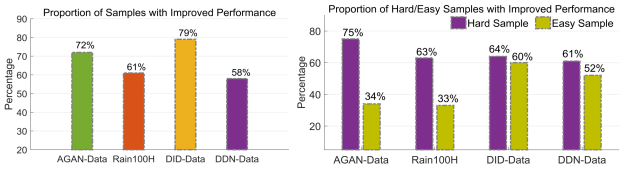

Figure 3: Statistical distribution of improved samples from ERL-Net over the basic MEMD on four benchmark datasets. Left: The proportion of improved samples over the whole dataset; Right: The proportion of improved samples with subsets containing hard samples or easy samples.

Judging from the distribution in Fig. 3, over half the test samples on each dataset, as expected, are better restored with ERL-Net, demonstrating that the proposed formulation is capable of dealing with different raining conditions on different datasets. More specifically, in the second experiment group, we investigate how ERL-Net affects the specific subset containing either hard samples or easy samples. For a specific dataset, the average PSNR is first obtained with the MEMD model, and then a hard sample is defined as the image whose PSNR is lower than the average value. Conversely, the easy sample corresponds to the image with a higher PSNR than the average one. After determining the subsets of hard and easy samples for each dataset, the same calculation as the first group is conducted in each subset to obtain another distribution as illustrated in Fig. 3. The results of all these four datasets show a similar distribution, with more improvements on the hard samples instead of the easy one, demonstrating the characteristics that the proposed

	SOTA	MA_A	MA_B	MA_C	MA_D	DT	
AGAN-Data	31.5712/0.9023	30.9741/0.8803 31.0954/0.8992	**32.5280/0.9285** **32.8603/0.9371**	**32.6427/0.9310** **32.9158/0.9374**	**32.7124/0.9407** **32.9647/0.9458**	**32.2650/0.9279** **32.1982/0.9298**	CLGD MLGD
Rain100H	30.3821/0.8939	31.6521/0.8687 31.6804/0.8692	**32.0452/0.9087** **32.6870/0.9247**	**32.8027/0.9221** **33.2041/0.9364**	**33.0145/0.9294** **34.5724/0.9387**	31.6214/0.8634 31.6287/0.8673	CLGD MLGD

Table 3: De-raining results (on PSNR/SSIM) with different ERL-Net setups (results that outperform the existing state-of-the-art (SOTA) on both PSNR and SSIM are shown in bold).

formulation deals with the heavier rainy conditions better. Some de-raining results generated by MEMD and ERL-Net on both hard and easy samples are shown in Fig. 4.

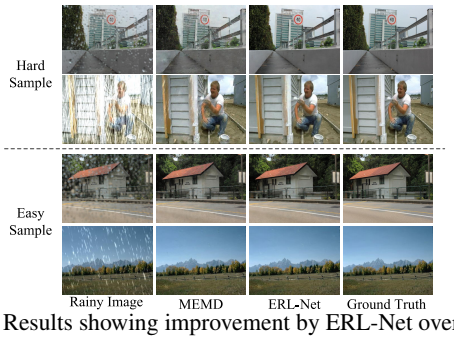

Figure 4: Results showing improvement by ERL-Net over MEMD on both hard samples and easy samples.

As observed from Fig. 5, better de-raining results with ERL-Net can be obtained on both rainy regions and background regions, motivating us to analyze how these two regions are improved by the proposed entangled representation learning mechanism. Inspired by the definition of mIoU (mean intersection over union) for evaluating semantic segmentation performance [2], we propose two metrics in terms of mIoR (mean intersection over rainy regions) and mIoB (mean intersection over background) to evaluate the improvement of ERL-Net on these two regions. Two maps are calculated before evaluation:

RIM (Result Indicator Map): Given a pair of rainy and clean images (I_{clean}) and the corresponding de-rained results by MEMD (I_{MEMD}) and ERL-Net ($I_{\text{ERL-Net}}$), they are first transformed to the YCbCr space, and then the I_{RIM} can be calculated by Eq. (6) on the Y channel:

$$I_{\text{RIM}} = \begin{cases} 0, & |I_{\text{ERL-Net}} - I_{\text{clean}}| \geq |I_{\text{MEMD}} - I_{\text{clean}}| \\ 1, & \text{Otherwise} \end{cases}, \quad (6)$$

Binary Map: We obtain the binary maps for both rainy regions (BMR) and background regions (BMB) by following the thresholding strategy proposed in [22], and we adopt the mean value instead of a fixed one (*e.g.*, 30 in [22]) as the threshold. Then in a BMR/BMB, a pixel belonging to the rainy/background region has a value 1 while a pixel belonging to the other region has a value 0. For the Rain100H dataset [26], we directly use the binary map provided by the authors for evaluation.

With the maps of BMR and BMB, the IoR (intersection over rainy regions) and IoB (intersection over background regions) for an image I with size $M \times N$ are calculated as:

$$I_{\text{IoR}} = \sum_{i=1}^{M}\sum_{j=1}^{N} \frac{I_{i,j}^{\text{BMR}} \cap I_{i,j}^{\text{RIM}}}{I_{i,j}^{\text{BMR}}}, \; I_{\text{IoB}} = \sum_{i=1}^{M}\sum_{j=1}^{N} \frac{I_{i,j}^{\text{BMB}} \cap I_{i,j}^{\text{RIM}}}{I_{i,j}^{\text{BMB}}} \quad (7)$$

where $\cap$ indicates the logic operator 'and'. With the values of I_{IoR} and I_{IoB} for each image, the mIoR and mIoB for a specific rainy dataset with K images can be calculated as:

$$\text{mIoR} = \frac{1}{K}\sum_{k=1}^{K}(I_{\text{IoR}})_k, \; \text{mIoB} = \frac{1}{K}\sum_{k=1}^{K}(I_{\text{IoB}})_k \quad (8)$$

The specific mIoR and mIoB values given by ERL-Net over different datasets are listed in Table 4:

Dataset	AGAN-Data	Rain100H	DID-Data	DDN-Data
mIoR	0.47	0.62	0.43	0.51
mIoB	0.28	0.19	0.30	0.27

Table 4: Results of mIoR and mIoB over four benchmark datasets.

As evidenced by the results in Table 4, the proposed entangled representation mechanism is proved to improve the quality of the de-rained images in both the rainy regions and the background regions, with more focus on the rainy regions. This is because the existence of raindrops or rain streaks makes it difficult to learn accurate representation depicting the property of original pixels behind the rains, while our entangled representation learning mechanism reduces such difficulty by imposing another encoder branch for learning more knowledge about the distribution of the original clean pixels behind the rains (raindrops or rain streaks).

4.4. Comparison with the State-of-the-art

The comparative experiment for rain streak removal is designed following the setup in [19], and the proposed ERL-Net is compared against seven state-of-the-art methods including discriminative sparse coding (DSC) [21], layer prior guided GMM [20], deep detail network (DDN) [6], joint rain detection and removal network (JORDER) [26], density-aware de-raining network (DID) [27], non-locally enhanced de-raining network (NLEDN) [19], and progressive de-raining network (PReNet) [23]. Different from the popularity of rain streak removal, there only exists two recent raindrop removal methods including the 3-layered network proposed by Eigen *et al*. [5] and the latest attentive generative de-raining network (AGAN) [22]. Besides, following the setup in [22], the pix2pix network [15] is also trained and evaluated for the raindrop removal task. For fair comparison, all the deep models for comparison are fine-tuned on the specific training set before evaluation.

Quantitative Results. The result comparisons with PSNR and SSIM for the tasks of rain streak removal and raindrop removal are shown in Table 5 and Table 6 respectively. Notably, as shown in Table 6, our ERL-Net achieves far better results than the state-of-the-arts, outperforming the best [22] by 1.39dB on PSNR and 4.8% on SSIM. On the other hand, for the result of rain streak removal listed in Table 5, ERL-Net achieves the best de-raining results on

Datasets	DSC [21]	GMM [20]	DDN [6]	JORDER [26]	DID [27]	NLEDN [19]	PReNet [23]	ERL-Net
DDN-Data	22.03/0.7985	25.64/0.8361	28.24/0.8654	28.72/0.8741	26.17/0.8409	*29.79/*0.8976	32.60/0.9458	33.92/0.9502
DID-Data	20.89/0.7321	21.37/0.7923	27.33/0.8978	24.32/0.8622	*27.95/*0.9087	*33.16/*0.9192	33.48/0.9229	34.62/0.9403
Rain100H	17.55/0.5379	15.69/0.4181	16.02/0.3579	*23.45/*0.7490	26.35/0.8287	*30.38/*0.8939	29.46/0.8935	34.57/0.9387

Table 5: Average PSNR/SSIM comparisons on synthetic rain streak datasets. Red and blue colors are used to indicate top 1st and 2nd rank respectively. Results with * indicate that the methods use additional labels or adopt data augmentation.

all the three challenging benchmark datasets, outperforming the existing best methods by 1.32db, 1.14db, and 4.19db on PSNR, and by 0.5%, 1.9%, and 5.0% on SSIM. The significant performance gain achieved by the proposed ERL-Net fully demonstrates the necessity of considering the entangled representation learning mechanism for the de-raining network design, and it also shows the effectiveness of the simple network architecture. Furthermore, similar to the observations illustrated in Section 4.3, it can also be concluded from the performance gains in Table 5 that the more challenging the dataset is, the more significant improvement can be achieved by ERL-Net. This is because heavier rainy effect induces more severe information loss in the embedding learned by traditional formulations [22, 26, 27, 6], and the proposed ERL-Net is able to mitigate such information loss with a simple entanglement representation learning mechanism, thus resulting in a superior de-raining model being capable of dealing with more challenging rainy situations.

	Eigen [5]	pix2pix [15]	AGAN [22]	ERL-Net
PSNR	28.59	30.59	31.57	32.96
SSIM	0.6726	0.8075	0.9023	0.9458

Table 6: Quantitative result comparisons for raindrop removal. Red and blue colors are used to indicate top 1st and 2nd rank respectively.

Qualitative Results. As can be observed from both the raindrop removal results shown in Fig. 5 and the rain streak removal results shown in Fig. 6, our model can remove heterogeneously distributed rain streaks or raindrops from different images and furthermore recover the details well. On the contrary, some existing methods fail to handle the extreme cases where the background is contaminated with dense raindrops or rain streaks. Specifically, as observed from the second row in Fig. 5 and the second row in Fig. 6, our network can recover the hidden details of the building walls very well. Consistent improvement over comparative methods is also demonstrated with the de-raining result on the skies in the second and third rows of Fig. 6, where our model produces a much closer recovery to the ground truth while all the others fail to remove the rain streaks thoroughly. Much better results from our ERL-Net can also be found in the real-world de-raining results in Fig. 7 showing that most of the existing methods produce under-deraining results.

5. Conclusion

Starting from an entirely new view that interprets the rainy-to-clean image translation network as a combination of embedding learning network (encoder) and conditional generator (decoder), we attribute the unsatisfactory de-raining results to the *deficiency of the learned embedding* and then present an entangled representation learning formulation to solve such an intrinsic problem. Specifically, given that the decoder is capable of generating a high-quality clean image (demonstrated by the clean-to-clean image translation experiment) with an *optimal embedding*, we bridge the discrepancy from the *deficient* embedding to the *optimal* one by performing an *equivariant mapping* implemented by representation entanglement. Extensive results on three rain streak datasets (DDN-Data, DID-Data, and Rain100H) and one raindrop dataset (AGAN-Data) demonstrate the superiority of the proposed model. Besides, a group of evaluation metrics is established to enable a thorough dissection on the effect of ERL-Net. It is noteworthy that the proposed residual branch induced entangled representation learning formulation should also be applicable to other image restoration problems (*e.g.*, image dehazing and image denoising), which is ongoing work. Furthermore, we believe that the proposed evaluation metrics can be used together with other metrics (*e.g.*, PSNR and SSIM) to provide a better analysis on more image de-raining proposals.

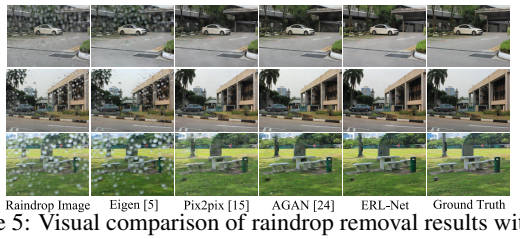

Figure 5: Visual comparison of raindrop removal results with the AGAN-Data (zoom in to see the difference better).

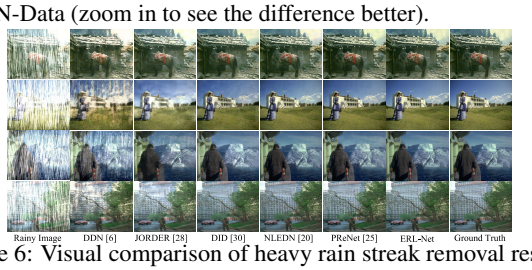

Figure 6: Visual comparison of heavy rain streak removal results with synthetic images (zoom in to see the difference better).

Figure 7: Visual comparison on real-world images. Results from both comparative models and ERL-Net are selected as the best one obtained from models trained on three different synthetic rain streak datasets (zoom in to see the difference better).

Acknowledgments. G. Wang is supported by the UNSW and CSIRO Data61 scholarship.

References

[1] Mathieu Aubry and Bryan C Russell. Understanding deep features with computer-generated imagery. In *ICCV*, 2015.

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *TPAMI*, 39(12):2481–2495, 2017.

[3] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*, 2016.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2018.

[5] David Eigen, Dilip Krishnan, and Rob Fergus. Restoring an image taken through a window covered with dirt or rain. In *ICCV*, 2013.

[6] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *CVPR*, 2017.

[7] Abel Gonzalez Garcia, Joost Van De Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. In *NIPS*, 2018.

[8] Ian Goodfellow, Jean Pouget Abadie, Mehdi Mirza, Bing Xu, David Warde Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. In *NIPS*, 2017.

[10] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *TPAMI*, 33(12):2341–2353, 2011.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

[13] Xun Huang, Mingyu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018.

[14] Nabil Ibtehaz and M Sohel Rahman. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. In *arXiv preprint arXiv:1902.04049*, 2019.

[15] Phillip Isola, Junyan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[16] Johnson Justin, Alahi Alexandre, and Feifei Li. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[17] Jin Hwan Kim, Jae Young Sim, and Chang Su Kim. Video deraining and desnowing using temporal correlation and low-rank matrix completion. *TIP*, 24(9):2658–2670, 2015.

[18] Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *CVPR*, 2015.

[19] 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 Multimedia*, 2018.

[20] Yu Li, Robby T Tan, Xiaojie Guo, Jingbo Lu, and Michael S Brown. Rain streak removal using layer priors. In *CVPR*, 2016.

[21] Yu Luo, Yong Xu, and Hui Ji. Removing rain from a single image via discriminative sparse coding. In *ICCV*, 2015.

[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 *CVPR*, 2018.

[23] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: A better and simpler baseline. In *CVPR*, 2019.

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[25] Ting Chun Wang, Ming Yu Liu, Jun Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *CVPR*, 2018.

[26] 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 *CVPR*, 2017.

[27] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *CVPR*, 2018.

[28] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

[29] Zhifei Zhang, Yang Song, and Hairong Qi. Decoupled learning for conditional adversarial networks. In *WACV*, 2018.

[30] Junyan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.