ERL-Net: Entangled Representation Learning for single image de-raining

1. Response to General Concerns

1.1. More clear explanation on ERL-Net

Motivation and Contributions: Nearly all existing de-raining frameworks are formulated as learning a direct rainy-to-clean image translation network on the synthetic dataset. However, due to the inherent dataset bias (*e.g.*, the imbalance data distribution, limited diversity in the synthetic dataset) or model bias (*e.g.*, inductive bias caused by specific network architecture or the stochastic training procedure), the network usually suffers from model generality which shows as (1) being overfitted to only some specific rain types that are covered by the synthetic training set and (2) being unable to perform well on real-rainy images. Such results are caused by the biased representation learned with the existing formulations. That is, due to existence of rain streaks or raindrops, some important factors regarding the clean images are missed from representation learned with existing formulation, resulting in the biased representation. In this paper, we argue that such bias can be diminished by learning a domain-adaptive representation. To achieve this, a model including a two-branched encoder is designed by enabling the residual encoder capture the missed sample-specific factors from the main encoder. By combining the factors from the main encoder and residual encoder branches via a representation entanglement manner, the latent representation bias can be remedied by modifying the incomplete representation into a more complete one. Benefited from this, the de-raining model will be able to (1) deal with different rainy conditions (*e.g.*, different density, shape, orientation etc.) (2) generalize well to unseen examples. Most importantly, models by the proposed formulation will undoubtedly work well on real rainy images. Besides being used for designing a more generalizable and effective de-raining model, the proposed entangled representation learning formulation can also be (1) used as a simple yet effective framework for solving many other image restoration problems, (2) used as a general framework for solving universal image-to-image translation problems (e.g., style transfer and image manipulation) and (3) also used as a simple formulation for solving the domain adaptation problem.

Another important contribution is the proposal of a more in-depth evaluation metric: A study from the latest ICML paper 'Do ImageNet classifiers generalize to ImageNet' demonstrates an undesired property of neural network: a well-trained model shows inability to perform well on 'slightly' harder images, and such property results demonstrates that even satisfactory top-K accuracy are reported, it cannot be well guaranteed that the model performs equally well on different images in the testing set. This conclusion demonstrates that we should not only care about the average metric results, but also stress more on the results from the hard examples. As suggested by [1], the combination of the metric values on these two sets should provide a more fair and realistic evaluation on the ability of the network. This also holds for de-raining task. Existing evaluation metric for de-raining task only consists of calculating the average PSNR/SSIM value on the overall testing set, without any specification on the hard examples, thus providing unfair conclusion. In Sec 4.3 of our paper, we design a group of new, simple, and more in-depth evaluation metrics, and use them to analyze the effect of our formulation on both hard and easy examples. Also following the common setup in existing de-raining literatures, we provide the results with the overall evaluation metric (PSNR/SSIM) in Sec 4.4. We believe that the combination of these two metrics (as done in our paper) will provide a much better performance evaluation, that is, the proportion of more hard examples being improved, together with a higher overall PSNR/SSIM values indicating a better de-raining model. All the future de-raining papers should consider the combination of the existing PSNR/SSIM and our proposed metrics for providing a much fairer comparison analysis on the de-raining models.

How ERL-Net works: To get insight on what features are learned by both the main branch and residual branch, and also how the simple combination of them obtains better de-raining results, we visualize the features maps in different branches as follows:

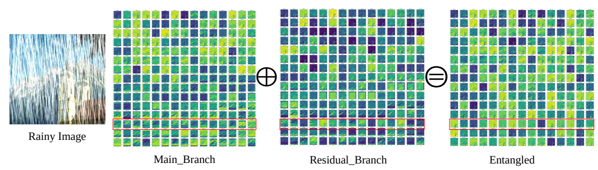

Figure R1: Visualization of activations (before ReLU) for the intermediate layer of ERL-Net (see Fig2 in the paper) in the main branch, residual branch, and their combinations by entanglement (zoom in or click `here` to see higher resolution images).

As can be seen from Fig. R1 and as expected, many background related patterns are learned by the residual branch, which plays a complementary role in providing the background information missed by the main branch (this can be especially observed by the feature maps in the red rectangle regions). By combining the main branch and residual branch via the entanglement manner, more complete encoding of the important background patterns is obtained thus resulting in much better de-raining results with most of the background information being recovered well.

1.2. More results on real rainy images

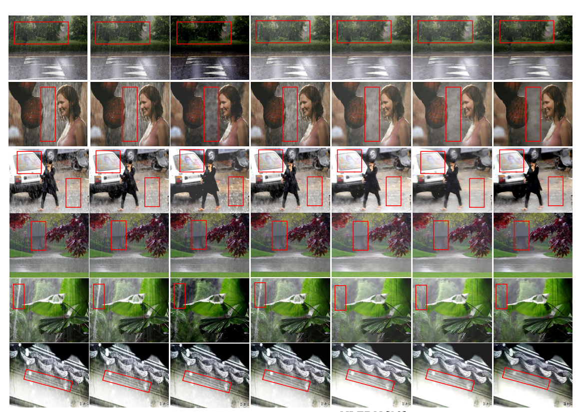

Figure R2: Visual quality comparison on some real rainy images (zoom in or click `here` to see higher resolution images).

As shown in Fig. R2, very promising results on real rainy images are obtained by ERL-Net, which can on the one hand remove the rain streaks thoroughly, and on the other hand recover the detailed structures well with high contrast (*e.g.*, the results in the $2^{nd}/3^{rd}/4^{th}/6^{th}$ rows). Compared with ERL-Net, the other methods usually generate much more blurry results with important details missing. Even for JORDER, which performs the best among the comparative methods, the results are still very blurry (*e.g.*, the results in the $2^{nd}/6^{th}$ rows) and some important structural details of the background cannot be recovered/preserved well (*e.g.*, the result in the 1^{st} row is very dark with low contrast caused by the over-deraining effect, and the structure of the wall in the 4^{th} row is not recovered well).

Failure cases: Due to the lacking of an explicit control on the learning of the representation from the residual branch, some negative results may obtained because: If the original representation from the main branch is good enough, the introduction of the residual representation may play negative effect by destroying the original one, thus resulting in some under-controlled results (*e.g.*, the over-deraining phenomenon as in the 5^{th} row of Fig. R2). In the future, it is possible to add some prior [2] to regularize the learning of residual representation, thus getting rid of unexpected residuals if the original representation from the main branch are good enough to guarantee a high-quality de-raining result.

1.3. Running time analysis

By performing the de-raining task on a computer equipped with a Tesla P100 GPU, the running of time of different models are reported in Table R1:

DDN	JORDER	DID	NLEDN	PReNet	AGAN	ERL-Net
0.26	32.67	1.89	7.65	0.11	0.85	0.46

Table R1: Running time (s) of different models on a 320×320 sized rainy image. Red color indicates the SOTA rain streak removal methods, Cyan color indicates the SOTA raindrop removal method.

As can be seen from the comparison, our model can provide a comparable running time and achieve new SOTA de-raining results for both rain streak and raindrop removal task.

1.4. Generalization ability analysis

To test the cross-dataset performance of the proposed model, we record the results on each dataset with ERL-Net trained on different datasets, results are listed in Table R2:

	Different datasets for training ERL-Net			SOTA results
	DDN	DID	Rain100H	
DDN	33.92/0.9502	32.69/0.9460	32.78/0.9463	32.60/0.9458
DID	34.28/0.9365	34.62/0.9403	34.39/0.9372	33.48/0.9229
Rain100H	34.12/0.9379	34.03/0.9371	34.57/0.9387	30.38/0.8939

Table R2: Average PSNR/SSIM values obtained by ERL-Net trained on different datasets. (Dataset with red color indicates the training set while dataset with blue color indicates the testing set).

As shown in Table R2, even when trained on one dataset (*e.g.*, Rain100H) and tested on another dataset (*e.g.*, DDN), the ERL-Net still achieves better result than the current SOTA, which is obtained by the model trained and tested on the same dataset. Such comparison fully demonstrates the powerful generality of ERL-Net.

2. Response to Specific Comments

Reviewer #1: • As carefully explained in the paper, M_κ can be simply interpreted as an implicit de-raining operator, which can transform the embedding of a rainy image to become similar to the embedding of the corresponding clean image. Experimental results and visualization of feature maps in Fig. R1 demonstrates such a transformation can be achieved by the proposed entangled representation learning mechanism. • The latent code of clean image is used for training the entangled representation learning network, and when testing only the latent code of rainy image is used for decoding the de-raining results. Also see Sec 1.1 on illustrating how our model works.

Reviewer #2: For each model, we use models trained on different dataset (*e.g.*, DID, DDN, and Rain100H in the paper) to obtain the results on each real-world rainy images, and then select the one with best visually effect to show on Fig. R2 in this letter and Fig. 7 in the paper.

Reviewer #3: • Many different network modules are designed in existing de-raining literatures, and the designers only test the effectiveness of each module on specifically-constructed dataset. Thus, there lacks a comprehensive benchmark analysis on the effect of different modules in a fair setting. Consequently, we say it is difficult to analyze the specific contribution of different modules in the literatures. • The model is lightweight because none of existing modules in de-raining literature are adopted due to their complexity, and we only use the general building blocks in the very popular image-to-image translation baselines. Besides, the running time analysis in Table R1 also demonstrates that our model is lightweight with high inference speed. • For Fig. 2 in the paper, the 'Down Layer' should use 'Global Average Pooling', and the 'UP Layer' should use 'Transposed Convolution', we will revise this error in the future version.

References

[1] S. Li, I. Araujo, W. Ren, Z. Wang, E. Tokuda, R. Junior, R. Cesar-Junior, J. Zhang, X. Guo, and X. Cao. Single image deraining: A comprehensive benchmark analysis. In *CVPR*, 2019.

[2] T. Michael, B. Olivier, and L. Mario. Recent advances in autoencoder-based representation learning. In *NeurIPS*, 2018.