SINGLE-IMAGE RAIN REMOVAL USING RESIDUAL DEEP LEARNING

Takuro Matsui EEE Dept. Keio University, Japan

Research Background

1. De-raining

Outdoor vision systems

Rain

Blurring effect and haziness

2. Two main approaches

Video-based	Single-image
Advantage: Easy to detect rain streaks	**Advantage**: Low calculation cost
Disadvantage: The redundant temporal information	**Disadvantage**: Less information for detecting rain streaks

We focus on single-image based approach.

Conventional methods

1. Decomposition method

Common approaches decompose a image into high and low frequency domain.

Input rainy image	Y.Luo [1]	DerainNet [2]
	Decomposition + dictionary learning	Decomposition + 3-layer CNN
	Leave rain streaks	Unnatural hue change

2. Problems

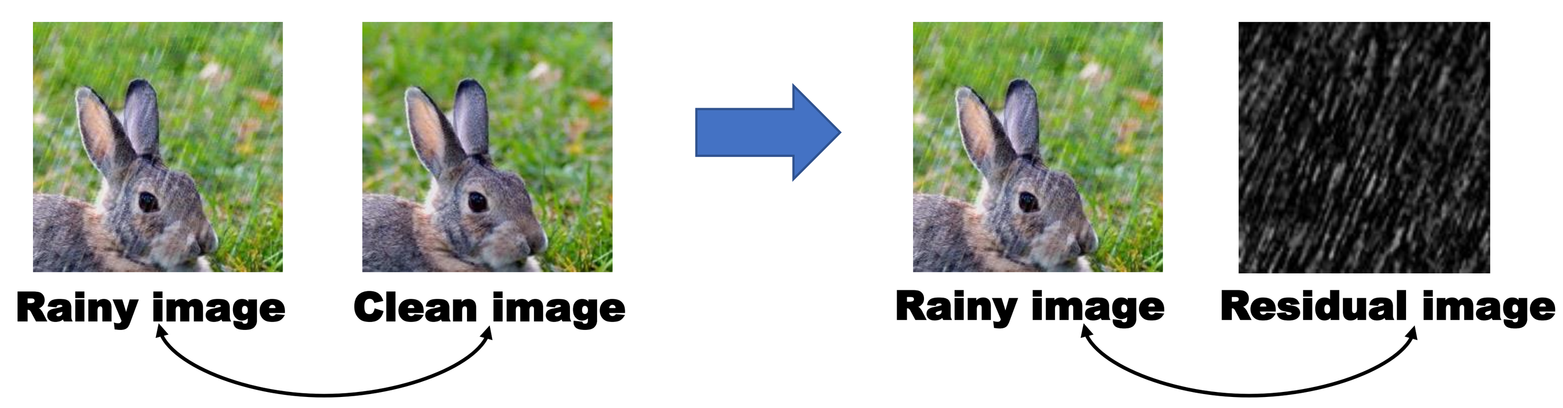
- Hue change
- Over fitting
- Not applicable to various types of real images

References

[1] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in International Conference on ComputerVision (ICCV), 2015.
[2] Fu, Xueyang, et al. "Clearing the Skies: A deep network architecture for single-image rain removal." *IEEE Transactions on Image Processing* 26.6 (2017): 2944-2956.
[3] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, Dec 2011.

Proposed methods

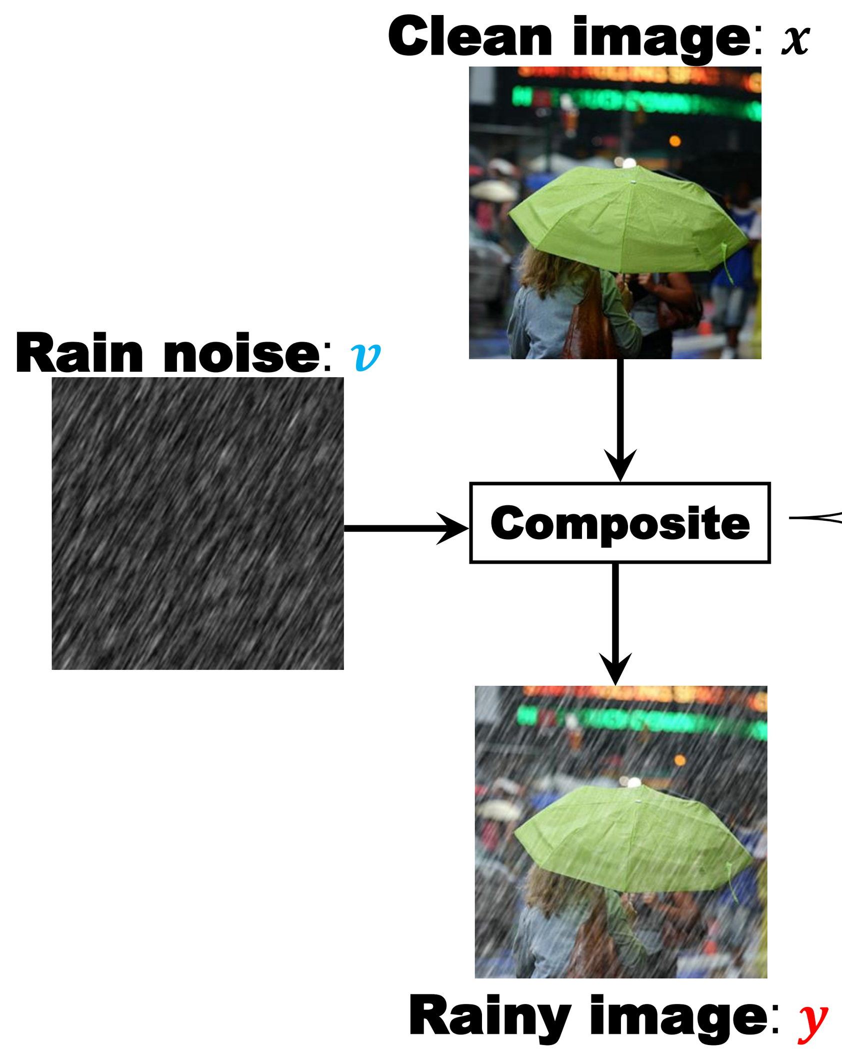
1. Residual Learning



- Speed up training process.
- Solve tradeoff between accuracy and network depth.
- Solve the dependency on image context

2. Composite models

Integration of two composite models → Applicable to a wide variety of real-world images.



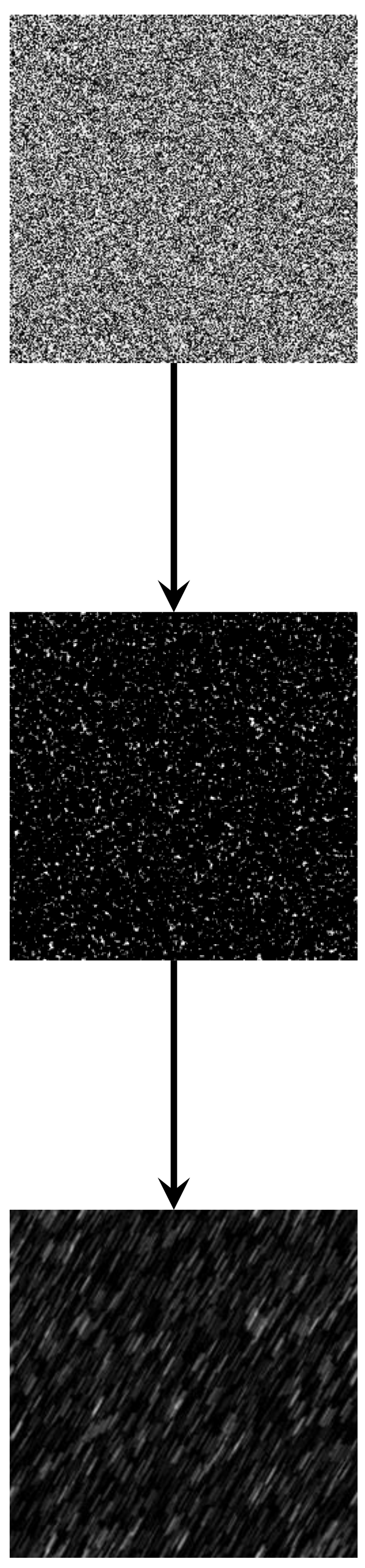
Non-linear screen blend model

$$y = x + v - x \circ v$$

Linear additive composite model

$$y = \min(x + v, 1)$$

3. Generating rain noise



① Generate uniformly distributed random numbers $u \in \mathcal{U}(0,1)$.

② Adjust the noise amount σ_a and crop between 0 and 1.

$$v_i \leftarrow \min(\max(\sigma_a(u_i - \lambda) + \lambda, 0), 1) \quad s.t.\ \lambda = 0.5$$

③ Apply Gaussian filter. $\hat{v} = \mathcal{F}_g v$

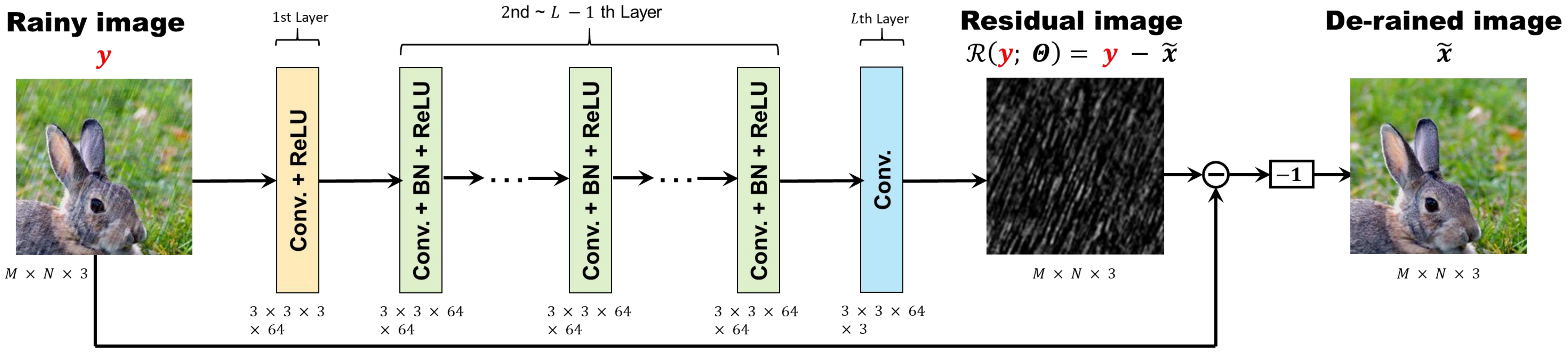
④ The noise is scaled by threshold values.

$$v_i \leftarrow \min\left(\max\left(\frac{\hat{v}_i - \sigma_{T_{\min}}}{\sigma_{T_{\max}} - \sigma_{T_{\min}}}, 0\right), 1\right)$$

⑤ A motion filter and the adjustment of rain scale are applied.

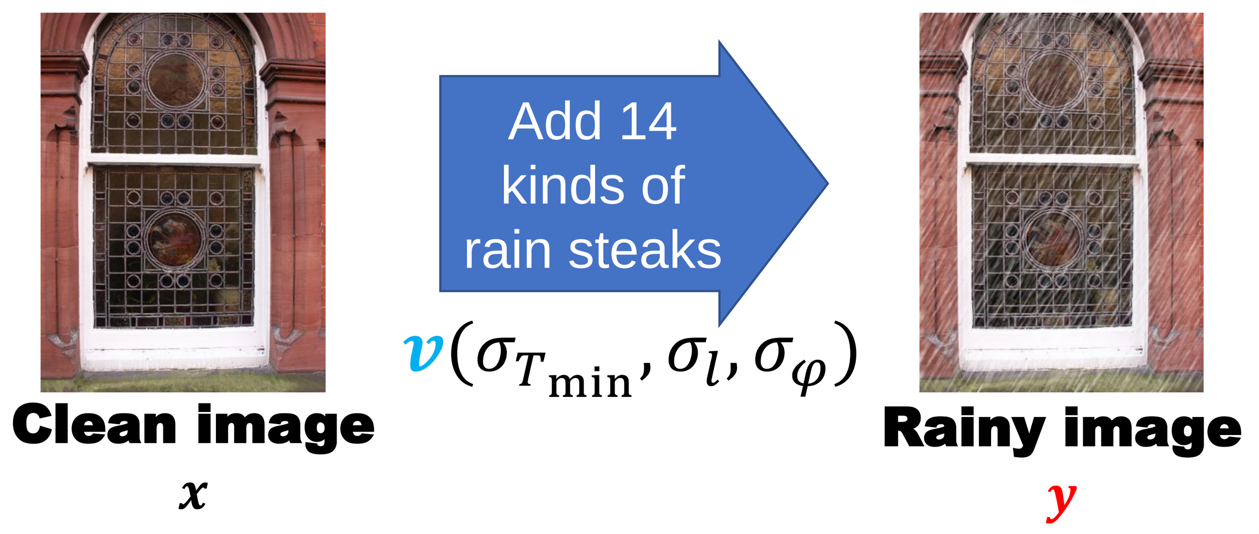
$$v \leftarrow \sigma_s \mathcal{F}_m v$$

Training

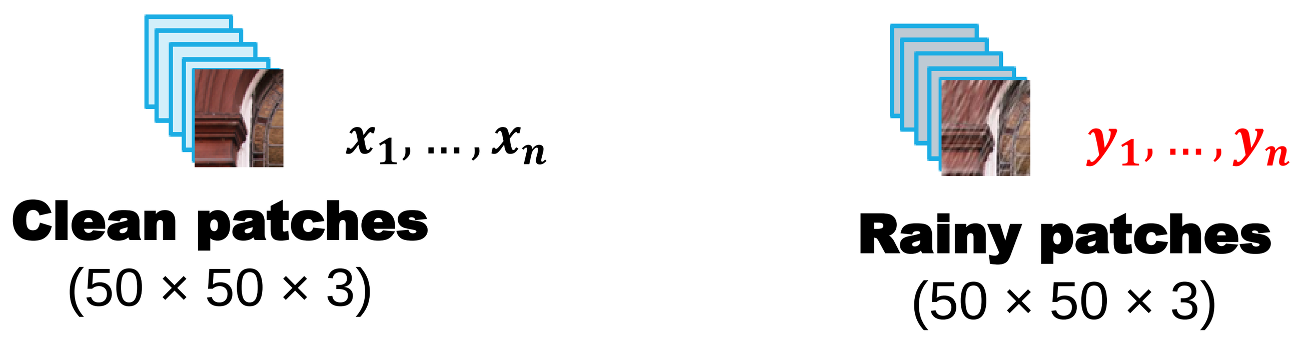


$$z_1 = \phi(\boldsymbol{W_1} * \boldsymbol{y} + \boldsymbol{b_1})$$
$$z_l = \phi\big(\mathrm{BN}(\boldsymbol{W_l} * \boldsymbol{y} + \boldsymbol{b_l})\big),\ l = 2, \dots, L$$
$$\boldsymbol{y} - \tilde{\boldsymbol{x}} = (\boldsymbol{W_L} * z_{L-1} + \boldsymbol{b_L})$$

① Add rain streaks on 900 images.



② 264,000 patches are randomly collected.



③ The depth and breadth of our network are empirically set to $L = 20$ and $n_l = 64$.

④ Caffe software package is used for training. **iters.**: 100,000 (8 hours), **solver**: Adam

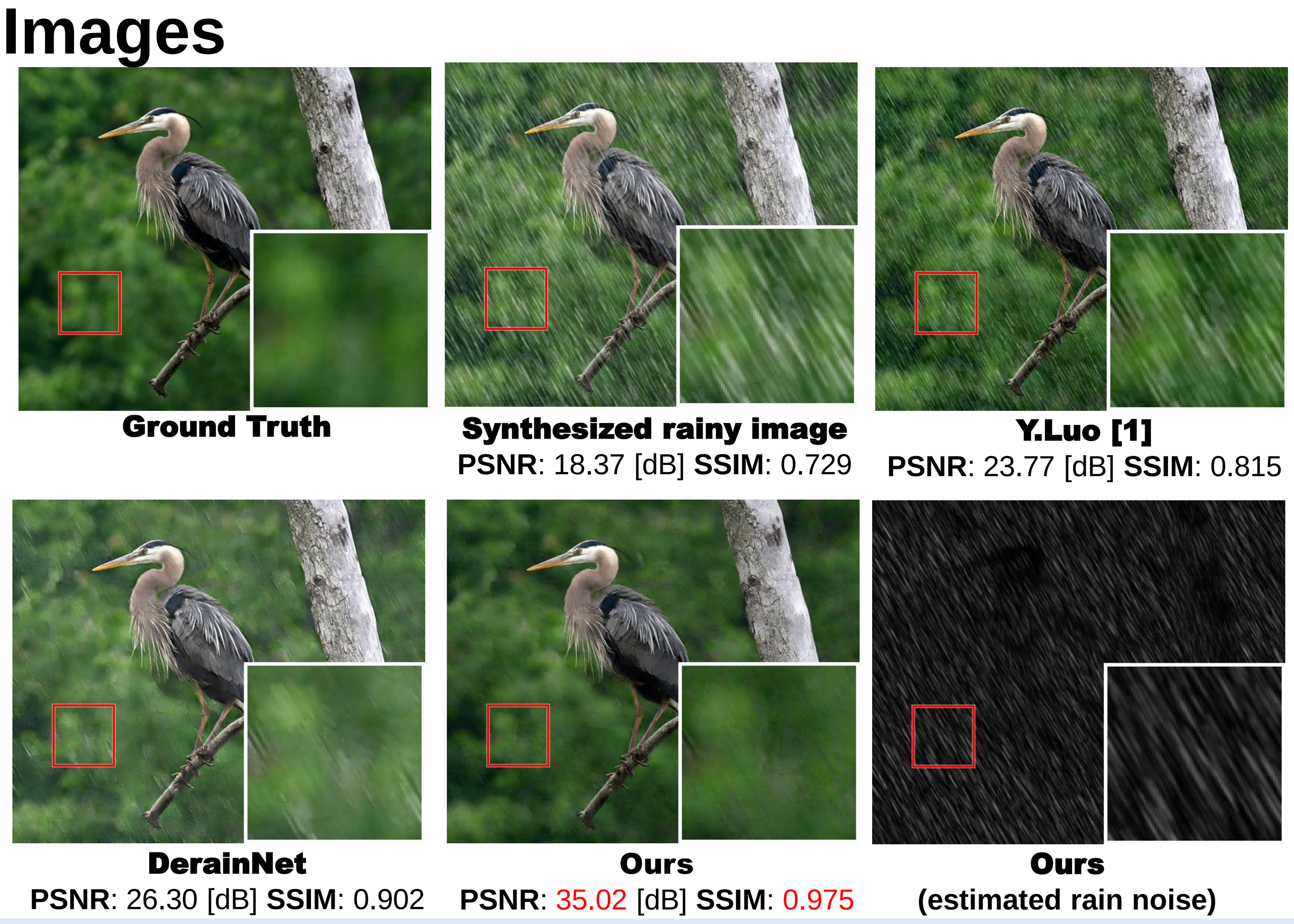
$$E(\Theta) = \frac{1}{2N}\sum_{n=1}^{N} \|(y_n - x_n) - \mathcal{R}(y_n;\,\Theta)\|_2^2,$$
$$s.t.\ \Theta = \{W_1, b_1, \dots, W_L, b_L\},$$

Experimental results

1. results on synthetic data

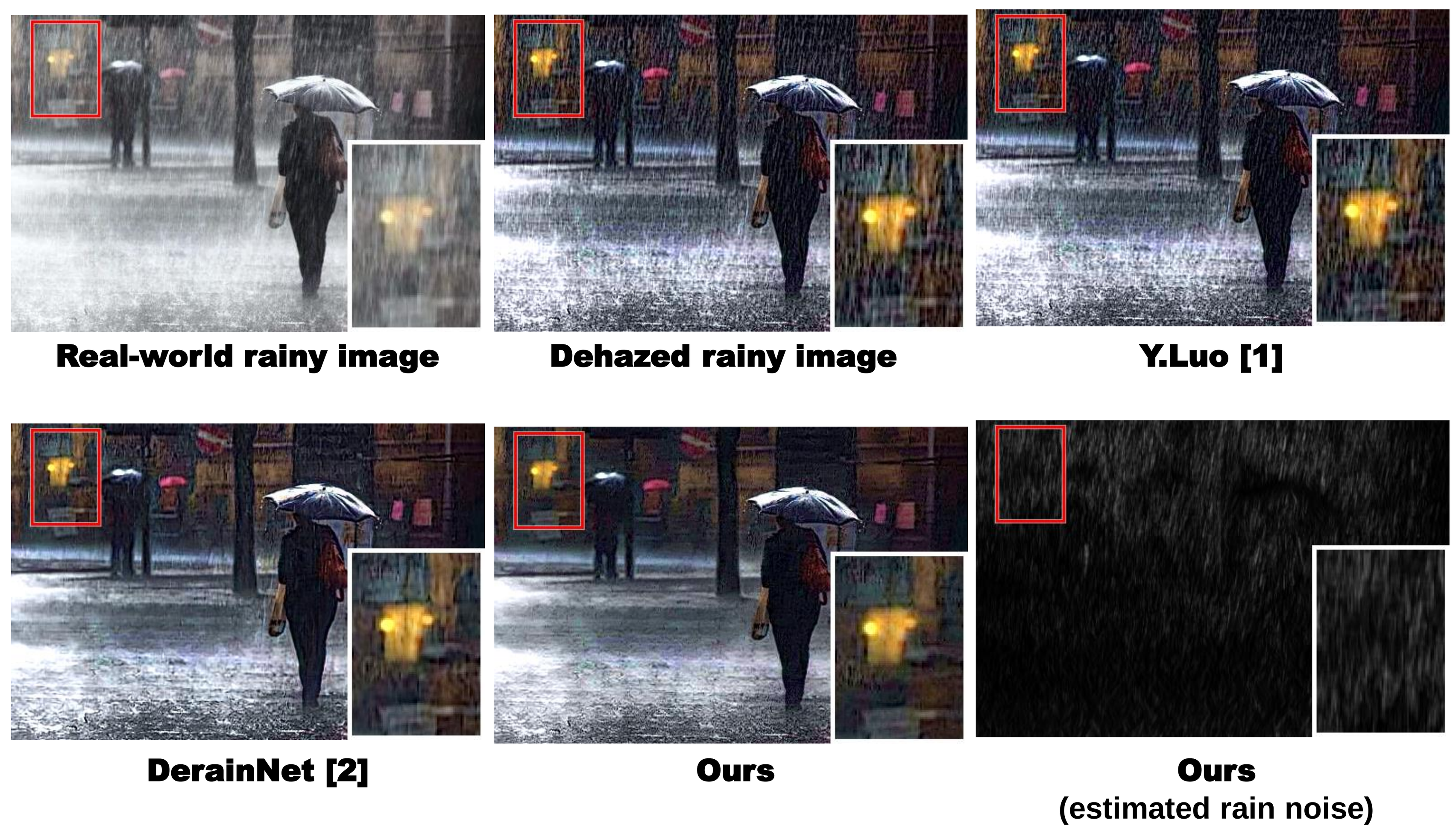
	PSNR				SSIM			
	Rainy image	Y.Luo [1]	DerainNet [2]	Ours	Rainy image	Y.Luo [1]	DerainNet [2]	Ours
umbrella	26.58	31.68	26.30	35.02	0.858	0.910	0.902	0.975
bird	18.37	23.77	19.23	28.83	0.729	0.815	0.847	0.940
BSD100	22.48	26.65	22.89	29.90	0.841	0.878	0.898	0.950
Urban100	22.71	26.21	22.71	29.49	0.888	0.891	0.888	0.949

Images



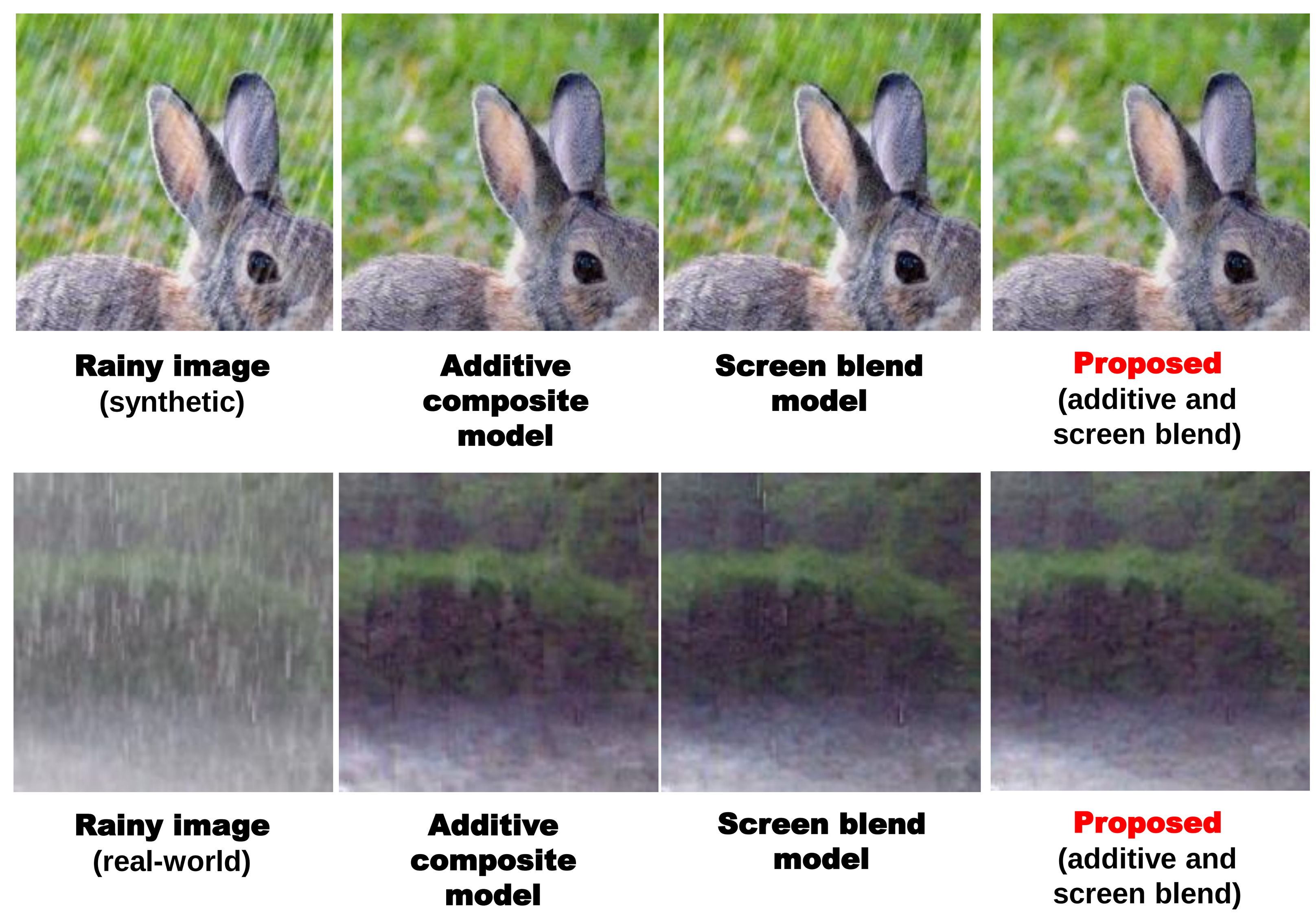
2. Results on real-world data

For clearer appearance, a de-hazing method is applied as a post-processing.



3. Impact of composite models

Mixture of additive and blend model is the best.



Conclusion

- **Residual CNN** for de-raining
- Residual learning and batch normalization achieves favorable performance.
- Our diverse rainy dataset make the model **applicable to real-world images**.
- Proposed method outperforms other state-of-the-art methods quantitatively and qualitatively.