Learning From Synthetic Photorealistic Raindrop for Single Image Raindrop Removal

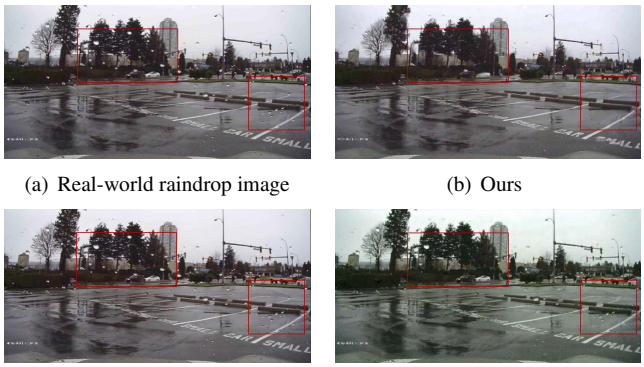
Zhixiang Hao[1] Shaodi You[3] Yu Li[4] Kunming Li[5] Feng Lu[1,2,*]
[1]State Key Laboratory of VR Technology and Systems, Beihang University, Beijing, China
[2]Peng Cheng Laboratory, Shenzhen, China, [3]Data61-CSIRO, [4]Tencent, [5]Australian National University
{haozx, lufeng}@buaa.edu.cn, youshaodi@gmail.com, yul@illinois.edu, u5580030@alumni.anu.edu.au

Abstract

Raindrops adhered to camera lens or windshield are inevitable in rainy scenes and can become an issue for many computer vision systems such as autonomous driving. Because raindrop appearance is affected by too many parameters, it is unlikely to find an effective model based solution. Learning based methods are also problematic, because traditional learning method cannot properly model the complex appearance. Whereas deep learning method lacks sufficiently large and realistic training data. To solve it, in our work, we propose the first photo-realistic dataset of synthetic adherent raindrops with pixel-level mask for training. The rendering is physics based with consideration of the water dynamic, geometric and photometry. The dataset contains various types of rainy scenes and particularly the rainy driving scenes. Based on the modeling of raindrop imagery, we introduce a detection network which has the awareness of the raindrop refraction as well as its blurring. Based on that, we propose the removal network that can well recover the image structure. Rigorous experiments demonstrate the state-of-the-art performance of our proposed framework.

1. Introduction

Most computer vision studies assume that the input image is of good visibility and clean content. However, rainy weather causes several different types of degradation to the image captured. It is common that the raindrops hit and flow on a camera lens or a windscreen of the vehicle. These adherent raindrops can obstruct, deform, and/or blur part of the area in the imagery of the background scenes, and then significantly degrade the performances of many vision algorithms *e.g.* feature detection [26, 12, 25], tracking [34, 5, 31], stereo correspondence [29, 30, 9], *etc.* A method to automatically remove the raindrop and recover the clear scene is, therefore, desired.



(a) Real-world raindrop image (b) Ours (c) Qian [24] (d) Pix2Pix [17]

Figure 1. Visual comparison of raindrop removal in real rainy scenes. Our method removes most of raindrops although the raindrops have large variety.

Unlike the rain streaks [36, 21], that mostly are thin and vertical stripes, adherent raindrops have more varieties in shape, position, and size, as can be seem in Fig. 1a. You *et al.* [39, 38, 37], Roser *et al.* [27], Eigen *et al.* [6] and Qian *et al.* [24] are a few example approaches focusing on detecting or removing the adherent raindrops. However, the method in [39] requires the rich temporal information, whereas the required video sequence cannot be applied to single image. Roser *et al.*'s method [27] can detect raindrop from a single image, but the model is over simplified and far from the real cases. Rather than model based methods, Eigen *et al.* [6] first adopt deep neural network, but the network only contains three layers and cannot properly learn the appearance of real raindrops. Qian *et al.* [24] integrate attention mechanism into GAN based CNNs, but their method is only tested on a small dataset. Although their dataset uses real raindrop, but the scene is in sunny day, which is not realistic. And therefore their method cannot fully handle real

*Corresponding Author: Feng Lu
This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 61972012 and Grant 61732016.

rainy scenes (Fig. 1c).

In this paper, we propose a physics driven as well as data driven method to detect and remove the adherent raindrops jointly. We utilize the realistic adherent raindrops imagery model proposed by Roser *et al*. [28] and You *et al*. [39]. Based on understanding the physics, we design a novel deep learning multi-tasks network which does both end to end detection and removal adherent raindrops from a single image. Unlike existing networks, the proposed network directly reflects the appearance of raindrop such that it is partially blended into the image and is a reflection of the background image. In brief, we separate the difficult task of restoring image into three sub-problems: (*i*) detect raindrop locations and shapes via a deeply supervised sub-network, and then (*ii*) restore adherent raindrop regions through deep learning network, subsequently (*iii*) a small CNN network is employed to smooth the blended image.

To enable proper training of the network, a new dataset is introduced which consisting photo-realistic rendering of the rainy scenes and clear scenes. The dataset uses Cityscapes dataset [4] as background image, which contains representative outdoor scenes. The dataset contains about $30K$ images. Each image has 50 to 70 raindrops with size varying from 0.8 to 1.5 centimeters, and the blurring level varying from 7 to 20 pixel.

This paper makes the following contributions:

- We propose a physics aware end-to-end neural network for joint raindrop detection and removal. The architecture is designed in cope with the physics of raindrop imagery.
- We develop a practical dataset of realistically rendered adherent raindrop images, which contains the pixel-level raindrop binary masks.
- The proposed method significantly out performances existing methods on all existing dataset and real-world rainy images.

2. Related Work

Removing raindrops from a single image is an ill-posed problem and would be beneficial to outdoor computer vision systems which work in bad weather, particularly surveillance systems and intelligent vehicle systems. Although there are many papers focus on removing haze [13, 2] or rain streaks [22, 21, 42], the researches on raindrop removal from a single image are relatively insufficient.

2.1. Adherent Raindrop Modeling

Halimeh *et al*. [11] introduce a raindrop modeling method based on ray-tracking. They propose an algorithm which models the geometric shape of a raindrop by utilizing its photometric properties. Roser *et al*. [28] mainly focus on modeling the raindrop geometric shape. They leverage the Bézier curves to represent a raindrop surface in low dimensions which is physically interpretable. Von Bernuth *et al*. [32] propose a novel method to render these raindrops using Continuous Nearest Neighbor search leveraging the benefits of R-trees. They use the synthetic raindrops for robustness verification of camera-based object recognition.

Recently, You *et al*. [40] model raindrops by considering both liquid dynamics and optics. They reconstruct the 3D geometry of a raindrop by minimizing surface energy constraints and total reflection constraint. The accurate raindrop model proposed by You *et al*. can be used in applications such as depth estimation and image refocusing. Later, You *et al*. [39] model adherent raindrops by taking consideration of physical properties such as gravity, water-water surface tensor and water-adhering-surface tensor.

2.2. Raindrop Removal

Most existing methods for detecting or removing raindrops are stereo or video based and therefore not applicable to a single image. Roser and Geiger [27] propose a method which detects raindrop in a single image based on a photometric raindrop model. The raindrop detection can improve image registration accuracy, then removing raindrops by fusing multiple views into one frame. You *et al*. [39] combine video completion technique with temporal intensity derivative to remove raindrops in video after detecting the locations of raindrops.

Due to the lack of temporal information, raindrop removal from a single image is more challenging. Eigen *et al*.'s work [6] is the first one to remove raindrops from a single image. They propose a 3-layer CNN network trained on rainy/clear pairs, the network can remove relatively sparse and small raindrops as well as dirt. However, the method suffers from blurred outputs and cannot remove dense raindrops. Recently, Qian *et al*. [24] propose a method based on GAN [10]. They create an aligned dataset by using a piece of glass sprayed with water to get images containing raindrops. With this dataset, they propose a GAN based network which integrates attention mechanism both in generator and discriminator. The method can produce sharp and clear image on their test set.

There are also some general Image-to-Image translation methods such as Pix2Pix [17] can tackle this problem, but they are not specifically designed for raindrop removal from a single image.

3. Raindrop Imagery Model and Photorealistic Dataset

As preliminary, we briefly introduce the raindrop imagery model developed by Roser *et al*. [28] and extended by You *et al*. [39] and the implementation detail on our photorealistic dataset generated from such model. It will later

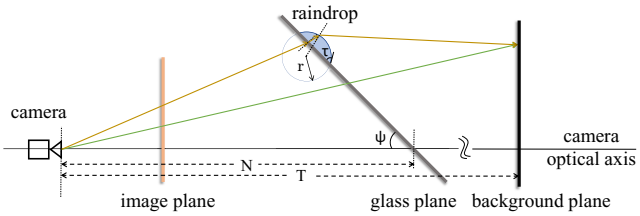


Figure 2. Refraction model. The light ray colored in green does not go through any raindrops. The light ray colored in yellow goes through a raindrop and is refracted twice.

drive us to design the network structure in Sec. 4. Also, we introduce the detail of the new photo realistic dataset.

Motivation: Data driven methods, particularly deep neural networks, need a large training data with ground truth. In particular, we need images with raindrops and the corresponding clear images to perform supervised learning in the context of raindrop removal from a single image. However, it is difficult and expensive to get strictly aligned rainy/clear image pairs of the exact same scene. Qian *et al*. [24] create a dataset contains 1119 pairs in total. The dataset is the only one for adherent raindrops, but it is relatively small and lacks the pixel-level masks of raindrops. In order to train our network, we create the first photo-realistic adherent raindrop dataset with pixel-level mask in autonomous driving settings based on Cityscapes dataset [4]. Inspired by [11] and [28], we synthesize adherent raindrop appearance on a clear background image by tracking the ray from camera to environment through the raindrops.

Dataset Generation: Geometric Rendering and Ray-tracing. As shown in Fig. 2, in order to get the synthetic adherent raindrop images, we set a scene with a camera at the origin, a glass plane at N centimeters ahead the camera, and a background plane at T centimeters ahead the camera. The angle between the glass plane and the ground is ψ. On the glass plane, we randomly sprinkle raindrops and ignore the refraction introduced by glass. A raindrop is modeled by spherical cap where the radius of the sphere is r and the angle between tangent and glass plane is τ. These two parameters determine the volume of the raindrop in glass. If a light ray determined by origin and the location of a pixel in image plane does not go through any raindrops, we set the pixel value unchanged as the background pixel. On the contrary, if a light ray goes through a raindrop in glass plane, we track the light ray by considering the refraction introduced by the raindrop, and set the pixel to the crossover point of light ray and the background plane. In Fig. 2, the light ray represented by the green line does not go through any raindrop, so we keep the corresponding pixel in image plane unchanged. The light ray represented by yellow line is refracted twice and reaches the same point in the background plane as the green line, so we set the corresponding pixel in image plane same as the green line. If total reflection happened when the light ray propagates from the inside of a raindrop to the air, we set the corresponding pixel in image plane to black. This phenomenon is quite common at real world raindrop's boundary which called dark bands [40].

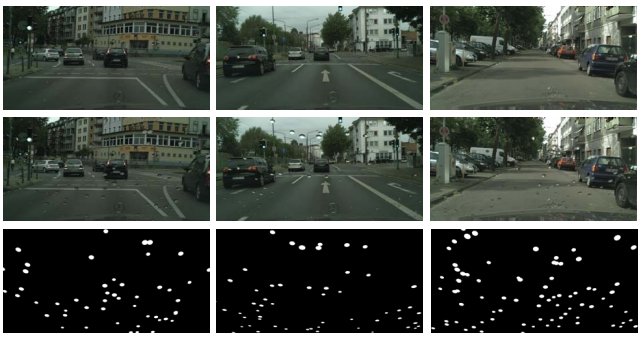


Figure 3. Samples of our synthetic raindrop images. Top: The ground truth clear image in Cityscapes dataset [4]. Middle: The synthetic raindrop image produced by our refraction model. Bottom: The ground truth binary mask of the raindrops.

Dataset Generation: Blurring and Blending. In the real world, raindrops will be blurred when a camera focuses on the environment scene. We use a disk blur kernel to blur the areas occupied by raindrops in synthetic image. As we observed, it is more realistic for the scene in our dataset to set the diameter of the disk blur kernel to be $7 \sim 20$ pixels. Since we already know the locations of raindrops on glass, it is also very convenient to get the ground truth pixel-level binary mask of raindrop image.

Dataset Generation: Environment Realness. We use images in Cityscapes [4] as the background images. Unlike the dataset created by Qian *et al*. [24] which is based on campus scenes, the scenes in Cityscapes are mainly focus on urban street where most outdoor vision systems work. And there are many data recorded in cloudy weather in Cityscapes, while the data in Qian's dataset is recorded in fine weather. So our dataset is more suitable for raindrop removal in outdoor vision systems especially autonomous driving.

Summary of the dataset: In order to make the raindrop appearance close to real ones, we set $N \in [20, 40]$, $T \in [800, 1500]$, $r \in [0.8, 1.5]$, $\psi \in [30°, 45°]$ and $\tau \in [30°, 45°]$. For each background image, we generate 50 to 70 raindrops. Finally, we make a dataset containing about 30000 images based on the training set of Cityscapes for training and 1525 images based on the test set of Cityscapes for testing.

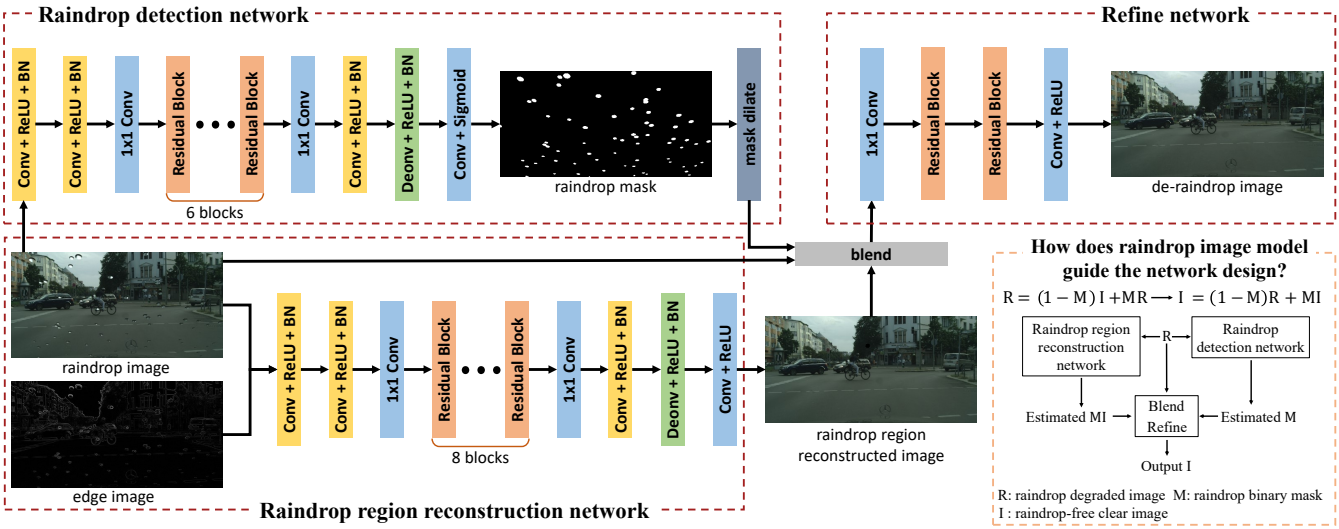


Figure 4. Network architecture of our proposed method. The whole architecture consists of three sub-networks for raindrop detection, raindrop region reconstruction and refining respectively.

4. End-to-End Raindrop Detection and Removal Network

We devise an end-to-end multi-task network which explicitly incorporate the raindrop imagery model. The observed raindrop degraded image $\mathbf{O}$ can be modeled as $\mathbf{O} = (1 - \mathbf{M})\mathbf{B} + \mathbf{R}$. Where $\mathbf{M}$ is raindrop binary mask, $\mathbf{B}$ is the clear background image and $\mathbf{R}$ is the raindrop layer. Based on this model, it is intuitive to separate the difficult task into three sub-problems: the first sub-network of our proposed method is designed to detect the raindrop binary mask $\mathbf{M}$, the second is designed to restore the regions occupied by raindrops, the third is designed to smooth and refine the blended image. As shown in Fig. 4, our proposed raindrop removal network consists of three sub-networks to address the three sub-problems respectively. In this section, we first introduce these sub-networks in detail. Then, by combining all sub-networks, we describe the whole architecture of our proposed network and some implementation details.

4.1. Raindrop Detection Network

The purpose of our raindrop detection network is to detect the areas of raindrops from input image. The network outputs a pixel-level binary mask in which the pixels of raindrops are marked as ones and the pixels of raindrop-free background are marked as zeros. We can separate the raindrops layer from the background layer by leveraging this binary mask.

Our raindrop detection network is inspired by I-CNN [7] and contains stacked residual blocks [14, 15]. Different from the general semantic segmentation or detection networks [41, 12, 3], the binary mask of raindrops has little semantic information. Hence, we just downsample the internal feature maps to half size in order to enlarge the receptive field. It makes the feature maps denser and keeps more accurate location information.

As shown in Fig. 4, the proposed raindrop detection network has 5 convolution layers and 6 residual blocks. In the second convolution layer which with stride 2, the resolution of feature maps is reduced to the half of input image. There is a 1×1 convolution layer in which the channels of feature maps increase from 64 to 256. In order to reduce the training time and memory usage, we use residual block in bottleneck fashion. The residual block consists of two 1×1 and one 3×3 convolution layers, where the 1×1 layers will reduce/increase the channels of internal feature maps to 64/256 respectively, and the middle 3×3 layer has 64-dimensional feature maps in both input and output. All convolution layers in our proposed network are followed by batch normalization (BN) [16] and ReLU [23]. We use the binary cross-entropy as loss function of the raindrop detection network, and the loss defined as:

$$\mathcal{L}_{det}(M,\hat{M}) = -\frac{1}{n}\sum_{i}^{n}\Big[M_i \log(\hat{M}_i) + (1-M_i)\log(1-\hat{M}_i)\Big], \quad (1)$$

where M is the ground truth binary mask, $\hat{M}$ is probability mask predicted by our network, n is the number of pixels in mask, and i is pixel index.

4.2. Raindrop Region Reconstruction Network

The raindrop region reconstruction network is designed to recover the areas occupied by blurred raindrops according to the contextual information, and it shares the similar CNN architecture with the proposed raindrop detection net-

work. Different from the raindrop detection network, we increase the number of residual blocks from 6 to 8. We combine the input image and the edge of input image to a 4-channel tensor as the input. The edge cues can help tasks like reflection removal and image smoothing according to [19, 20, 35]. We compute the edge image E of a raindrop image R by the equation defined as:

$$E_{x,y} = \frac{1}{4}\sum_{c}(|R_{x,y,c} - R_{x+1,y,c}| + |R_{x,y,c} - R_{x-1,y,c}| + |R_{x,y,c} - R_{x,y+1,c}| + |R_{x,y,c} - R_{x,y-1,c}|), \quad (2)$$

where x, y are pixel coordinates, and c is the color channels in RGB image.

The loss function of raindrop region reconstruction is defined as:

$$\mathcal{L}_{recons}(I, \hat{I}) = \frac{1}{n}\sum_{i}^{n}\lambda_i |I_i - \hat{I}_i|, \quad (3)$$

where I is the ground truth clear image and $\hat{I}$ is the image predicted by our network. The λ_i is a weight which is set to 20 when pixel i belongs to a raindrop, otherwise to 1. By introducing λ, our network will pay more attention to reconstruct the raindrop region.

4.3. Refine network

Combing the two sub-networks described above, we propose the refine network. The blended input image B of refine network is defined as:

$$B = \hat{M}\hat{I} + (\mathbf{1} - \hat{M})R, \quad (4)$$

where R is raindrop image, $\hat{M}$ is binary mask produced by raindrop detection sub-network, and $\hat{I}$ is the output of raindrop region reconstruction sub-network. B consists of background pixels in R and reconstructed pixels in $\hat{I}$. The architecture of refine network is relatively simple, it contains two convolution layers and two residual blocks. To train the refine network by considering both image structure similarity and color similarity [43], we use loss function mixed with SSIM [33] loss and ℓ_1 loss.

$$\mathcal{L}_{ref}(I, \tilde{I}) = \alpha(1 - \mathcal{L}_{ssim}(I, \tilde{I})) + (1 - \alpha)\mathcal{L}_{\ell_1}(I, \tilde{I}), \quad (5)$$

where $\tilde{I}$ is output of our refine network (*i.e.* final output of our proposed method). We set the $\alpha = 0.3$.

Dilated Mask: In experiments, we find that our proposed raindrop detection network gets a relatively low recall at the edge of raindrops. Hence, the B will preserve some raindrop edge pixels if using the original binary mask $\hat{M}$. In order to reduce the number of raindrop pixels in B, we apply dilation, a basic mathematical morphology operation, to $\hat{M}$. It is implemented as a parameter-free layer after the output of raindrop detection network. Fig. 5 shows the effect of dilation operation on a binary mask and the blended input of refine network.

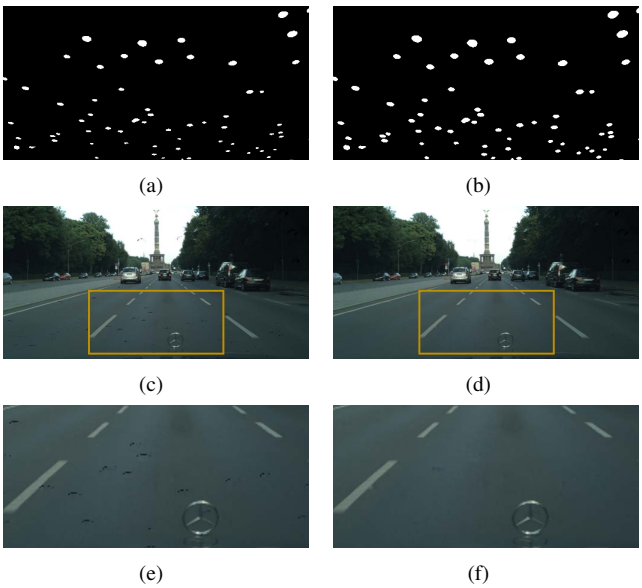


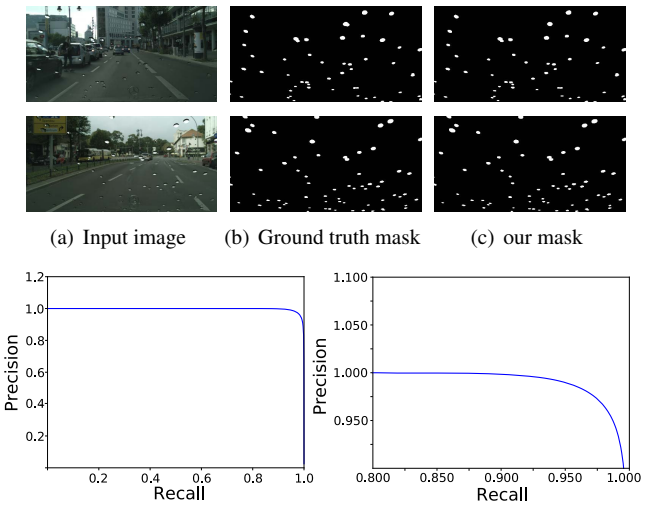
Figure 5. The effect of dilation operation. **(a):** the original binary mask produced by raindrop detection network. **(b):** the dilated binary mask. **(c):** blended image produced by (a). Note that there are many raindrop edge pixels in the road of the blended image. **(d):** blended image produced by (b). The raindrop edge pixels in road are removed. **(e), (f)**: local regions of blended images.

4.4. Implementation Details

Two-Stage Training: Our complete network consists of three different sub-networks, and we train them in a two-stage fashion. In the first stage, we train raindrop detection network and raindrop region reconstruction network respectively because their outputs are dependencies of the refine network. In the second stage, we combine the networks trained on the first stage with refine network to construct our whole network, and we only update the parameters in refine network when training. Limiting the number of trainable parameters in the second stage can be beneficial to prevent our network from overfitting.

Data Augmentation: We do online data augmentation for all training stages. All the images and masks in our training set have resolution of 256×512. Before feeding into the network, a training pair or its horizontal flip will be randomly cropped to 100×100.

Our implementation is built on TensorFlow [1] and trained on a single NVIDIA TITAN Xp GPU with 12GB memory. All trainable parameters in proposed networks are initialized by Xavier initializer [8]. The batch size is set to 32, and all three sub-networks are trained for 50 epochs which consists of 50K steps per epoch. We adopt Adam [18] optimizer with initial learning rate $= 0.001$ which is decayed linearly from 20 to 40 epoch until reaching the ending learning rate $= 0.0001$.



(a) Input image (b) Ground truth mask (c) our mask

Figure 6. Raindrop detection on our synthetic dataset. Above: Visual result. Below: Precision-Recall curve. We compute pixel-level accuracy. Curve figure in the right is magnification of figure in the left. Other methods do not output raindrop detection results and are therefore not compared.

5. Experiments

In this section, we compare our proposed method to Eigen [6], Qian [24], DID-MDN [42] and Pix2Pix [17] along with ablation experiments. Note that Eigen [6] and Qian [24] are the only methods in the literature dedicated to this problem to the best of our knowledge. For this reason, we add a general image-to-image translator Pix2Pix [17] and a rain streak removal method DID-MDN [42] in the comparison. We report the PSNR and SSIM metrics in our synthetic dataset and Qian's dataset.

5.1. Experiments on Synthetic Image

Quantitative Evaluation: Our synthetic test set contains 1525 rainy/clear images pairs and corresponding rain masks. The test set is synthesized based on the official test set of Cityscapes. We train all the existing methods on our synthetic training set to compare to our proposed method.

First, we evaluate the performance of our raindrop detection network on synthetic dataset which contains the ground truth of binary raindrop masks. Fig. 6 shows Precision-Recall curve of our raindrop detection network. The average precision (AP) of our network prediction is 0.9973. It indicates that our raindrop detection network works very well on synthetic images. Because other methods do not predict the raindrop mask, we cannot compare our raindrop detection network to them.

Table 1 shows quantitative raindrop removal results. It is clearly that our complete model outperforms other methods in terms of both PSNR and SSIM by a large margin.

	PSNR	SSIM
Eigen [6]	29.02	0.9560
Pix2Pix [17]	31.00	0.9471
DID-MDN [42]	31.32	0.9576
Qian [24]	37.79	0.9692
Ours (3RN only)	37.73	0.9852
Ours (3RN + RDN + RFN)	39.24	0.9848
Ours (3RN + Dilated RDN + RFN)	**41.29**	**0.9921**

Table 1. Quantitative evaluation results of raindrop removal on synthetic dataset. **3RN:** Our Raindrop Region Reconstruction Network. **RDN:** Our Raindrop Detection Network. **RFN:** Our Refine Network. The setting in the last row is our complete model.

	PSNR	SSIM
Eigen [6]	28.59	0.6726
Pix2Pix [17]	30.14	0.8299
DID-MDN [42]	27.06	0.8830
Qian [24]	**30.82**	0.9050
Ours (3RN only)	29.28	0.9016
Ours (rough mask)	30.17	**0.9128**

Table 2. Quantitative evaluation results of raindrop removal on dataset proposed in Qian [24]. **Ours (3RN only):** Our Raindrop Region Reconstruction Network only. **Ours (rough mask):** Our model trained with *low-quality* rough mask which is produced by subtracting raindrop image with the ground truth.

Qualitative Evaluation: Fig. 7 shows the qualitative results of different methods. Our proposed method, Qian [24] and DID-MDN [42] remove the most of raindrops successfully while the results produced by Pix2Pix [17] and Eigen [6] still contain many raindrops. In the first row of Fig. 7, there are artifacts in the center of Qian and DID-MDN results. In the second row, there are some raindrops in the bottom of results of Qian and DID-MDN. Our method achieves better performance in both removing raindrops and avoiding artifacts.

Ablation Study: We run a number of ablations in order to demonstrate the effectiveness of different modules in our proposed method. The quantitative results are shown in Table 1. We use the raindrop region reconstruction network as baseline. By adding the raindrop detection network and refine network to the baseline, both PSNR and SSIM are increasing. Then, by replacing the mask with dilated mask, we get our complete proposed architecture which archives the highest PSNR and SSIM. The results of ablation study indicate that all our proposed modules do contribute to the raindrop removal. Our complete architecture improve PSNR by 9.4% compare to the 3RN only. It is a substantially large improvement.

Efficiency: The running speed is critical in many outdoor computer vision systems. In order to show the efficiency of

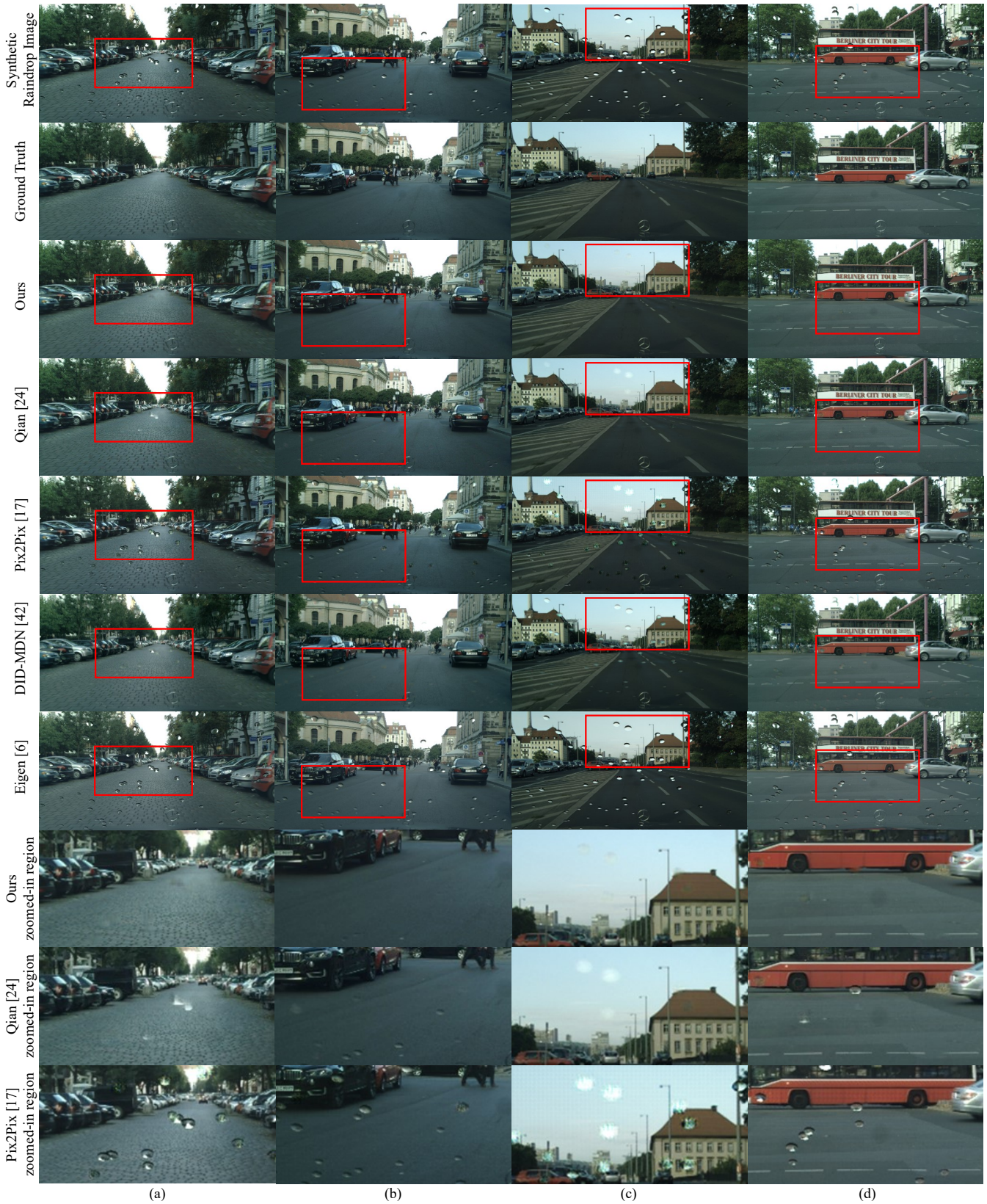


Figure 7. Comparison of raindrop removal on our synthetic dataset. It can be seen that the proposed method achieves better performance in both removing raindrops and avoiding artifacts. Best viewed with zoom.

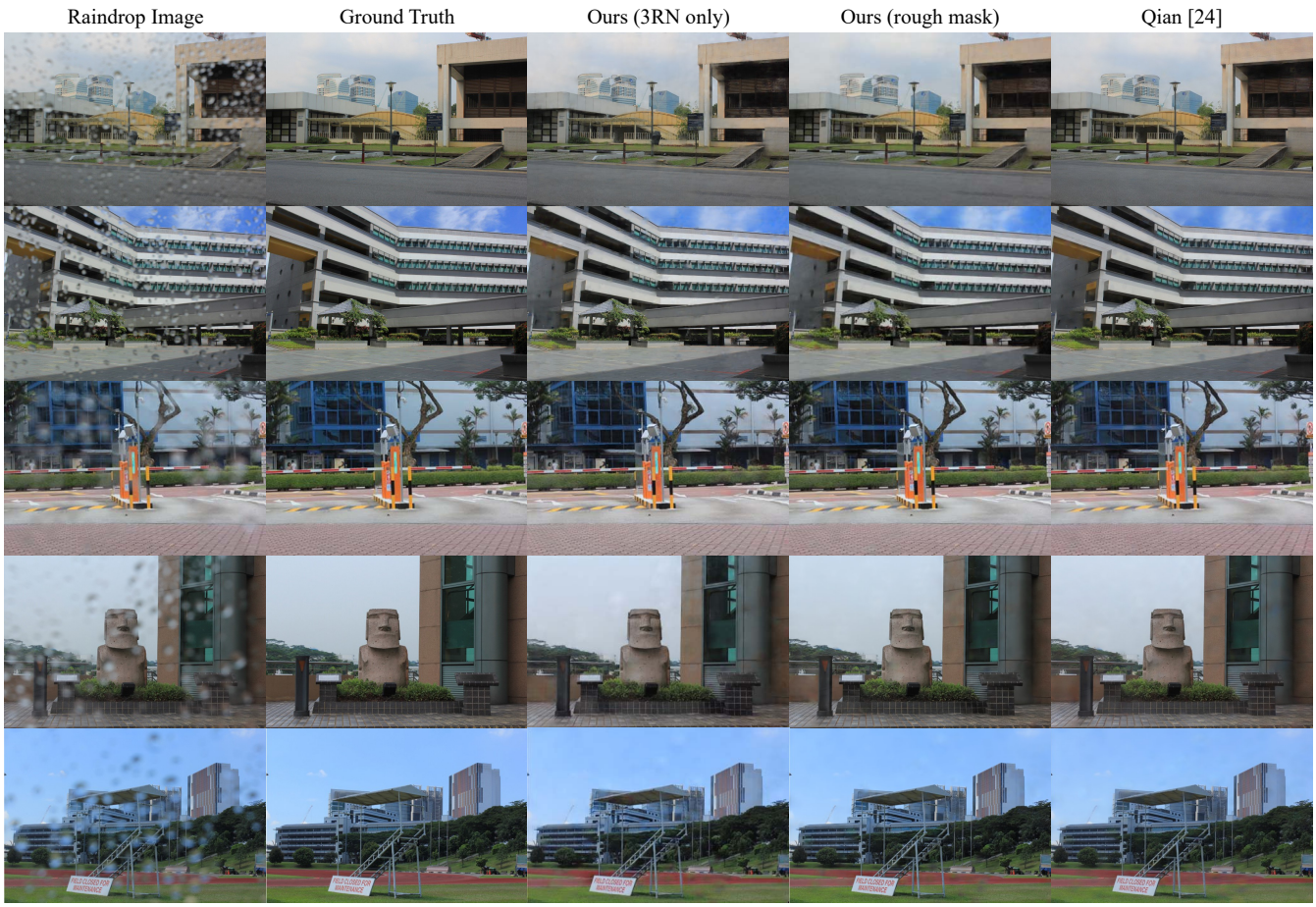


Figure 8. Comparison of raindrop removal results on Qian's dataset [24]. All three models remove raindrops successfully and produce clear images.

our method, we also evaluate the inference speed of prior methods and ours. For a single 256×512 image, the inference time of our method is 69.74ms, Pix2Pix [17] is 79.1ms, Qian [24] is 97.16ms, Eigen [6] is 107.712ms and DID-MDN [42] is 133.64ms. The results indicate that our method is not only effective but also efficient. All the methods are tested on a single NVIDIA TITAN Xp GPU, and the time reported is the average of 20 repeats.

5.2. Experiments on Real-world Image

We evaluate the proposed method on real-world dataset introduced in Qian [24]. Due to the lack of the ground-truth raindrop mask, we cannot train our complete architecture directly. Instead, we only use *low-accuracy* masks roughly estimated by subtracting raindrop images with ground truth. We also train our Raindrop Region Reconstruction network (3RN) which do not need raindrop masks. As shown in Table 2, our methods get competitive results quantitatively that is on par with Qian [24]. Note the dataset is not very challenging because the data in training set and test set is too similar. Thus all three models can produce compelling visual quality raindrop removal as shown in Fig. 8.

6. Conclusions

In this paper, we propose a novel end-to-end network to detect and remove adherent raindrop jointly. Since the supervised deep learning in raindrop removal suffers from the lacking of sufficient paired training data, we also develop a practical and realistic dataset for adherent raindrop which contains the pixel-level raindrop binary masks. There are two stages in our proposed multi-task network. In the first stage, two sub-networks detect raindrop locations and restore adherent raindrop regions respectively. In the second stage, a blended image is produced by using the location clues, then a refine network is employed to smooth the blended image. Our experiment results show that the proposed method outperforms the state-of-the-art and can handle both synthetic data and real-world data. In the future, we would like to enhance our raindrop imagery model and extend our experiments to further validate the benefit of using our method on computer vision systems working under the rainy scenes.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: a system for large-scale machine learning. In *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016. 5

[2] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao. Dehazenet: An end-to-end system for single image haze removal. *IEEE Transactions on Image Processing (TIP)*, 25(11):5187–5198, 2016. 2

[3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(4):834–848, 2018. 4

[4] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3

[5] M. Danelljan, G. Bhat, F. S. Khan, M. Felsberg, et al. Eco: Efficient convolution operators for tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[6] D. Eigen, D. Krishnan, and R. Fergus. Restoring an image taken through a window covered with dirt or rain. In *IEEE International Conference on Computer Vision (ICCV)*, 2013. 1, 2, 6, 8

[7] Q. Fan, J. Yang, G. Hua, B. Chen, and D. P. Wipf. A generic deep architecture for single image reflection removal and image smoothing. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 4

[8] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010. 5

[9] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems (NIPS)*, 2014. 2

[11] J. C. Halimeh and M. Roser. Raindrop detection on car windshields using geometric-photometric environment construction and intensity-based correlation. In *Intelligent Vehicles Symposium*, pages 610–615, 2009. 2, 3

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 4

[13] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(12):2341–2353, 2011. 2

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4

[15] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision (ECCV)*, 2016. 4

[16] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 2015. 4

[17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 6, 8

[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 5

[19] A. Levin and Y. Weiss. User assisted separation of reflections from a single image using a sparsity prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29(9), 2007. 5

[20] Y. Li and M. S. Brown. Exploiting reflection change for automatic reflection removal. In *IEEE International Conference on Computer Vision (ICCV)*, 2013. 5

[21] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown. Rain streak removal using layer priors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2

[22] Y. Luo, Y. Xu, and H. Ji. Removing rain from a single image via discriminative sparse coding. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[23] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning (ICML)*, 2010. 4

[24] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu. Attentive generative adversarial network for raindrop removal from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2, 3, 6, 8

[25] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[26] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (NIPS)*, 2015. 1

[27] M. Roser and A. Geiger. Video-based raindrop detection for improved image registration. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2009. 1, 2

[28] M. Roser, J. Kurz, and A. Geiger. Realistic modeling of water droplets for monocular adherent raindrop recognition using bezier curves. In *Asian Conference on Computer Vision (ACCV)*, 2010. 2, 3

[29] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)*, 47(1-3):7–42, 2002. 1

[30] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006. 1

[31] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr. End-to-end representation learning for correlation filter based tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[32] A. von Bernuth, G. Volk, and O. Bringmann. Rendering physically correct raindrops on windshields for robustness verification of camera-based object recognition. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 922–927. IEEE, 2018. 2

[33] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 13(4):600–612, 2004. 5

[34] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 1

[35] L. Xu, J. Ren, Q. Yan, R. Liao, and J. Jia. Deep edge-aware filters. In *International Conference on Machine Learning (ICML)*, 2015. 5

[36] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan. Deep joint rain detection and removal from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[37] S. You, R. T. Tan, R. Kawakami, and K. Ikeuchi. Adherent raindrop detection and removal in video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 1

[38] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa, and K. Ikeuchi. Raindrop detection and removal from long range trajectories. In *Asian Conference on Computer Vision (ACCV)*, 2014. 1

[39] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa, and K. Ikeuchi. Adherent raindrop modeling, detectionand removal in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(9):1721–1733, 2016. 1, 2

[40] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa, and K. Ikeuchi. Waterdrop stereo. *arXiv preprint arXiv:1604.00730*, 2016. 2, 3

[41] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2016. 4

[42] H. Zhang and V. M. Patel. Density-aware single image deraining using a multi-stream dense network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 6, 8

[43] H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2017. 5