Semi-supervised Deraining Method and Its Application to Object Detection

阮正鑫 120032910012 邱天宇 120032910162
谢宇辉 120032910142 祁殷俏 120032910130
王俊达 020032910016

Abstract

Single image rain removal is a typical inverse problem in computer vision, which is very important for object detection to resist the influence of noise provide by the rain. The deep learning technique has been verified to be effective for this task. However, these methods are limited in the sense that they can be trained only on fully labeled data. Due to various challenges in obtaining real world fully-labeled image deraining datasets, existing methods are trained only on synthetically generated data and hence, generalize poorly to real world images. We propose a Gaussian Process-based semi-supervised learning framework which enables the network in learning to derain using synthetic dataset while generalizing better using unlabeled real-world images. Our experiment shows that the proposed method is able to effectively leverage unlabeled data and achieves high performance. Further we shows that the deranging indeed helps object detection achieving higher performance.

Index Terms—*Deraining, rainy image, rain residue, semi-supervision learning, gaussian process*

1. Introduction

Developing autonomous driving systems that can assist drivers in making decisions is one of the most active and challenging research areas [1]. The goal is to improve safety, reduce traffic accidents, and move closer towards fully autonomous cars and intelligent transportation systems. Among the solutions that have been developed over the past few years, the computer vision-based approach offers the most cost effective solution as it uses cameras rather than other types of more costly sensors.

One way to exploit the information captured from vision information is object detection. The goal of object detection is to predict a set of bounding boxes and category labels for each object of interest. Modern detectors address this set prediction task in an indirect way, by defining surrogate regression and classification problems on a large set of proposals [2, 3], anchors [4], or window centers [5, 6]. However, their performances are significantly influenced by environment, for instance rainy day. Rain streaks and rain drops often occlude or blur the key information of the images captured outdoors. Thus the rain removal task for an image or a video is useful and necessary, which can be served as an important pre-processing step for outdoor visual system. An effective rain removal technique can often help an image/video better deliver more accurate detection or recognition results [7].

Current rain removal tasks can be mainly divided into two categories: video rain removal (VRR) and single image rain removal (SIRR). Compared to VRR, which utilize the temporal correlation among consecutive frames, SIRR is generally much more difficult and challenging without the aid of much prior knowledge capable of being extracted from a single image. Since being firstly proposed by Kang et al. [7], the SIRR problem has been attracting much attention. Recently, deep learning methods[8, 9, 10, 11] have been empirically substantiated to achieve state-of-the-art performance for SIRR by training an appropriate, carefully designed network to detect and remove the rain streaks simultaneously.

The task of rain removal is plagued with several issues such as (i) large variations in scale, density and orientation of the rain streaks, and (ii) lack of real-world labeled training data. Most of the existing work in image deraining have largely focused towards addressing the first issue. The use of synthetic datasets results in sub-optimal performance on the real-world images, typically because of the distributional-shift between synthetic and rainy images [12]. Despite this gap in performance, this issue remains relatively unexplored in the literature.

We address the issue of incorporating unlabeled real-world images into the training process for better generalization. In this paper, we propose a method to generate pseudo label using the Gaussian-process (GP). In the total process, we train the model with label data and unlabeled data iteratively.

To summarize, this paper makes the following contributions:

- We propose a non-parametric approach for performing semi-supervised learning to incorporate unlabeled real-world data into the training process.

- Through our experiment we show that our proposed is able to achieve great performance with limited training data.
- Through our experiment we show that our proposed successfully help object detection to reach higher accuracy in the dataset of rainy image.

The rest of this paper is organized as follows. In Section 2 we present our model as well as the Gaussian-process(GP) which is used to generate pseudo label for unlabeled real world image. In Section 3 we show the experimental results and we make conclusion in Section 4.

2. Method

In this section, we provide a formulation of the problem statement. Then, we will describe the proposed algorithm and elaborate on the details

2.1. Image de-raining

Existing image deraining methods assume the additive model where the rainy image (x) is considered to be the superposition of a clean image (y) and a rain component (r), i.e.,

$$x = y + r.$$

Single image deraining task is typically an inverse problem where the goal is to estimate the clean image y, given a rainy image x. This can be achieved by learning a function that extracts the rain component from the rainy image which can then be subtracted from the rainy image to obtain the clean image.

2.2. Semi-supervised learning

In semi-supervised learning, we are given a labeled dataset of input-target pairs $(\{x, y\} \in D_L)$ sampled from an unknown joint distribution $p(x, y)$ and unlabeled input data points $x \in D_U$ sampled from $p(x)$. The goal is to learn a function $f(x|\theta)$ parameterized by θ that accurately predicts the correct target y for unseen samples from $p(x)$. The parameters θ are learned by leveraging both labeled and unlabeled datasets. Since the labeled dataset consists of input-target pairs, supervised loss functions such as mean absolute error or cross entropy are typically used to train the networks. The unlabeled datapoints form D_U are used to augment $f(x|\theta)$ with information about the structure of $p(x)$.

We employ the semi-supervised learning framework to leverage unlabeled real-world data to obtain better generalization performance. Specifically, we consider the synthetically generated rain dataset consisting of input-target pairs as the labeled dataset D_L and real-world unlabeled images as the unlabeled dataset D_U.

2.3. Gaussian processes

A Gaussian process (GP) $f(v)$ is an infinite collection of random variables, of which any finite subset is jointly Gaussian distributed. A GP is completely specified by its mean function and covariance function which are defined as follows

$$m(v) = \mathbb{E}[f(v)],$$
$$K(v, v') = \mathbb{E}[(f(v) - m(v))(f(v') - m(v'))],$$

where $v, v' \in V$ denote the possible inputs that index the GP. The covariance matrix is constructed from a covariance function, or kernel, K which expresses some prior notion of smoothness of the underlying function. GP can then be denoted as follows

$$f(v) \sim GP(m(v), K(v, v') + \sigma_\epsilon^2 I).$$

where is I identity matrix and σ_ϵ^2 is the variance of the additive noise.

2.4. Method description

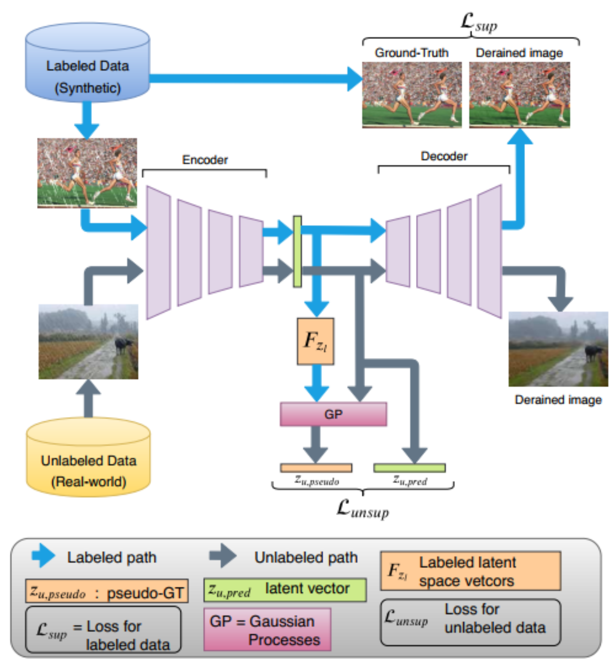

Fig.1 Overview of the proposed GP-based SSL framework

As shown in Fig. 1, the proposed method consists of a CNN based on the UNet structure, where each block is constructed using a Res2Block. In summary, the network is made up of an encoder $h(x, \theta_{enc})$ and a decoder $g(z, \theta_{dec})$. Here, the encoder and decoder are parameterized by θ_{enc} and θ_{dec}. Furthermore, x is the input to the network which is then mapped by the encoder to a latent vector z. In our case, x is the rainy image from which we want to remove the rain streaks. The latent vector is then fed to the decoder to produce the output r, which in our case is the rain streaks. The rain streak component is then subtracted form the rainy image (x) to produce the clean image (y), i.e.,

$$y = x - r,$$

Where

$$r = g(h(x, \theta_{enc}), \theta_{dec}).$$

In our problem formulation, the training dataset is $D = D_L \cup D_U$, where $D_L = \{x_l^i, y_l^i\}_{i=1}^{N_l}$ is a labeled training set consisting of N_l samples and $D_U = \{x_u^i\}_{i=1}^{N_u}$ is a set consisting of N_u unlabeled samples.

The goal of the proposed method is to learn the network parameters by leveraging both labeled (D_L) and unlabeled dataset (D_U). The training process iterates over labeled and unlabeled datasets. The network parameters are learned by minimizing the supervised loss function (L_{sup}) in the labeled training phase, and the unsupervised loss function (L_{unsup}) in the unlabeled training phase. For the unlabeled training phase, we generate pseudo GT using GP formulation, which is used in the unsupervised loss function. The two training phases are described in detail in the following sections.

2.5. Labeled training phase

In this phase, we use the labeled data D_L to learn the network parameters. Specifically, we minimize the following supervised loss function

$$L_{sup} = L_1 + \lambda_p L_p$$

where λ_p is a constant, and L_1 and L_p are $l_1 - loss$ and perceptual loss functions, respectively. They are defined as follows,

$$L_1 = \left\| y_l^{pred} - y_l \right\|_1$$

$$L_p = \left\| \Phi_{VGG}\left(y_l^{pred}\right) - \Phi_{VGG}(y_l) \right\|_2^2$$

where $y_l^{pred} = g(z, \theta_{dec})$ is the predicted output, y_l is the ground-truth, $z = h(x, \theta_{enc})$ is the intermediate latent space vector and $\Phi_{VGG}(.)$ represents the pre-trained VGG-16 network.

In addition to minimizing the loss function, we also store the intermediate feature vectors z_l^i for all the labeled training images x_l^i in a matrix F_{zl}. That is $F_{zl} = \{z_l^i\}_{i=1}^{N_l}$. It is used later in the unlabeled training phase to generate the pseudo-GT for the unlabeled data.

2.6. Unlabeled training phase

In this phase, we leverage the unlabeled data D_U to improve the generalization performance. Specifically, we provide supervision at the intermediate latent space by minimizing the error between the predicted latent vectors and the pseudo-GT obtained by modeling the latent space vectors of the labeled sample images F_z^l and z_u^{pred} jointly using GP

Pseudo-GT using GP: After the first iteration on D_L, we store the latent space vectors of the labeled data in a list F_z^l. These vectors lie on a low dimension manifold. During the unlabeled phase, we project the latent space vector z_u of the unlabeled input onto the space of labeled vectors $F_{zl} = \{z_l^i\}_{i=1}^{N_l}$. We express the unlabeled latent space vector z_u^k corresponding to the k^{th} training sample from D_U as

$$z_u^k = \sum_{i=1}^{N_l} \alpha_i z_l^i + \epsilon$$

where α_i are the coefficients, and ϵ is additive noise $\mathcal{N}(0, \sigma_\epsilon^2)$.

With this formulation, we can jointly model the distribution of the latent space vectors of the labeled and the unlabeled samples using GP. Conditioning the joint distribution will yield the following conditional multi-variate Gaussian distribution for the unlabeled sample

$$P(z_u^k | D_L, F_{zl})| = \mathcal{N}(\mu_u^k, \Sigma_u^k)$$

where

$$\mu_u^k = K(z_u^k, F_{zl})[K(z_u^k, F_{zl}) + \sigma_\epsilon^2 I]^{-1} F_{zl}$$

$$\Sigma_u^k = K(z_u^k, z_u^k) - K(z_u^k, F_{zl})[K(z_u^k, F_{zl}) + \sigma_\epsilon^2 I]^{-1} K(F_{zl}, z_u^k) + \sigma_\epsilon^2$$

Where K is defined by the kernel function as follows

$$K(Z, Z)_{k,i} = \kappa\left(z_u^k, z_l^i\right) = \frac{< z_u^k, z_l^i >}{|z_u^k|\left|z_l^i\right|}$$

We use the mean predicted as the pseudo-GT ($z_{u,pseudo}^k$) for supervision at the latent space level. By minimizing the error between $z_{u,pred}^k = h(x_u, \theta_{enc})$ and $z_{u,pseudo}^k$, we update the weights of the encoder $h(x_u, \theta_{enc})$, thereby adapting the network to unlabeled data which results in better generalization. Using GP we are approximating z_u^k, latent vector of an unlabeled image using the latent space vectors in F_{zl}, by doing this we may end up computing incorrect pseudo-GT predictions because of the dissimilarity between the latent vectors. This dissimilarity is due to different compositions in rain streaks like different densities, shapes, and directions of rain streaks. In order to address this issue we minimize the variance Σ_u^k computed between z_u^k and the N_n nearest neighbors in the latent space vectors using GP. Additionally, we maximize the variance $\Sigma_{u,f}^k$ computed between z_u^k and the N_f farthest vectors in the latent space using GP, in order to ensure that the latent vectors in F_{zl} are dissimilar to the unlabeled vector z_u^k and do not affect the GP prediction, as defined below

$$\Sigma_{u,f}^k = K(z_u^k, z_u^k) - K\left(z_u^k, F_{zl,f}\right)\left[K\left(F_{zl,f}, F_{zl,f}\right) + \sigma_\epsilon^2 I\right]^{-1} K\left(F_{zl,f}, z_u^k\right) + \sigma_\epsilon^2$$

Thus, the loss used during training using the unlabeled data is defined as follows

$$L_{unsup} = \left\| z_{u,pred}^k - z_{u,pseudo}^k \right\|_2 + log\Sigma_{u,n}^k + \log(1 - \Sigma_{u,f}^k)$$

Dataset	Input	Methods that use only synthetic dataset							Methods that use synthetic and real-world dataset					
		DSC [28]	LP [29]	JORDER [30]	DDN [31]	JBO [32]	DID-MDN [33]	UMRL [34]	SIRR [13] (CVPR '19)			Ours		
									DL	DL + DU	Gain	DL	DL + DU	Gain
Dense	17.95	19.00	19.27	18.75	19.90	18.87	18.60	20.11	20.01	21.60	1.59	20.24	22.36	2.12
Sparse	24.14	25.05	25.67	24.22	26.88	25.24	25.66	26.94	26.90	26.98	0.08	26.15	27.26	1.11

Table.1 PSNR results on the test set. D_L indicates using only labeled dataset during training and $D_L + D_U$ indicates using both labeled and unlabeled dataset during training.

(a) (b) (c) (d) (e) (f)

Figure.2 Results on the test set. (a) Input rainy image (b) DID-MDN [33] (c) DDN [31] (d) SIRR [13] (e) Ours (f) ground-truth image.

2.7. Total loss

The overall loss function used for training the network is defined as follows

$$L_{total} = L_{sup} + \lambda_{unsup} L_{unsup}$$

where λ_{unsup} is a pre-defined weight that controls the contribution from L_{sup} and L_{unsup}.

3. Experiments and results

In this section, we introduce the details of the datasets and various experiments conducted to demonstrate the effectiveness of the proposed framework and its application value in object detection. Specifically, we conducted two sets of experiments. In the first set, we compare the proposed method with other methods and verify the effectiveness of using the unlabeled real-world data to train the proposed framework. Particularly, we compare the performance of our method with a recent SSL framework for rain removal in images(SIRR) [13]. In the second set of experiments, we apply the proposed method to object detection.

3.1. Datasets

DDN-SIRR dataset: DDN-SIRR[13] consists of labeled synthetic training set and unlabeled real-world dataset. This dataset functions as the evaluation tool of semi-supervised learning frameworks. The labeled training set is derived from Fu et al. [9] and it comprises 9,100 image pairs obtained by adding raindrops of different intensities to the clean images from the UCID dataset [23]. The unlabeled real-world training set is derived from [24, 25, 26] and Google image search. Moreover, the test set can be divided into two categories: (i) Dense raindrops, and (ii) Sparse raindrops. Each category comprises 10 synthetic images.

MS COCO dataset: MS COCO[27] is one of the most commonly used dataset in the field of object detection. It contains 82,783 training, 40,504 validation, and 40,775 testing images. Each image is attached to an annotation file which contains categories of objects in the images and the four corners' coordinates of bounding boxes of the objects. We use the 40,504 validation images for our experiments. We first synthesize rain streaks on the cleaning images. Then we use the proposed method to remove rain.

3.2. Results on synthetic test set from DDN-SIRR

The experiment aims at evaluating the effect of using unlabeled real-world data along with labeled synthetic dataset during training and showing the superiority of the proposed method through comparing with other methods. Following the protocol set by [13], we label the labeled synthetic training set from the DDN-SIRR dataset as D_L and the unlabeled real-world training set from the DDN-SIRR dataset as D_U. All methods will be evaluated on the synthetic test set from DDN-SIRR.

The evaluation results on the test set are shown in Table. 1. We use PSNR as the evaluation metric. We compare the proposed method with approaches which only use synthetic dataset in the training framework, such as DSC[28], LP [29], JORDER [30], DDN [31], JBO [32] , DIDMDN [33] and UMRL[34]. Because our proposed method leverages unlabeled real-world data, it achieve significantly better results as compared to the above approaches.

Particularly, we also compare our method with a GMM-based semi-supervised method (SIRR) [13]. It can be seen from Table. 1 that our proposed method outperforms SIRR. In addition, we also illustrate the gains[1] of the two methods due to the use of unlabeled real-world data. The proposed method achieves larger gains comparing with SIRR, indicating that our method has better capacity to leverage unlabeled data.

Example results on one image of the test set are shown in Fig. 2. As can be seen from this figure, the proposed method achieves better reconstructions comparing to other methods.

3.3. Application in object detection

The goal of this experiment is to analyze the effect of using image deraining when detecting objects in rain. We use the original evaluation set from MS COCO as C1, the set with raindrops added to C1 as C2, and the set derived from removing rain in the C2 as C3. And we use YOLO v3 to perform object detection on the three set.

The detection results is showed in Table.2 It can be observed from Table.2 the existence of raindrops leads to a sharp decline in the accuracy of objection detection. And in this case, image deraining can considerably improve the accuracy of detection.

Table.2 the detection results of YOLO v3 on the three dataset. The detection metric we used is mAP at IOU= 0.5.

	C1	C2	C3
mAP	0.54135	0.41993	0.44982

4. Conclusion

We presented a GP-based SSL framework to leverage unlabeled data during training for the image deraining task. We use supervised loss functions such as l1 and the perceptual loss to train on the labeled data. For the unlabeled data, we estimate the pseudo-GT at the latent space by jointly modeling the labeled and unlabeled latent space vectors using the GP. The pseudo-GT is then used to supervise for the unlabeled samples. In our experiment, we prove that the proposed method shows good performance in the deraining job, and it successfully help the object detection model to have a better performance in the image with rain.

References

[1] A. Geiger, P. Lenz, R. Urtasun, Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite, in: Proceedings IEEE International Conference Comput. Vis. Pattern Recognit. (CVPR), 2012, pp. 3354–3361.

[2] Cai, Z., Vasconcelos, N.: Cascade R-CNN: high quality object detection and instance segmentation. PAMI (2019).

[3] Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. PAMI 39 (2015).

[4] Lin, T.Y., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV (2017).

[5] Tian, Z., Shen, C., Chen, H., He, T.: FCOS: fully convolutional one-stage object detection. In: ICCV (2019)

[6] Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv:1904.07850 (2019).

[7] Li-Wei Kang, Chia-Wen Lin, and Yu-Hsiang Fu. Automatic single-image-based rain streaks removal via image decomposition. IEEE Transactions on Image Processing, 21(4):1742–1755, 2012.

[8] Xueyang Fu, Jiabin Huang, Xinghao Ding, Yinghao Liao, and John Paisley. Clearing the skies: A deep network architecture for single-image rain removal. IEEE Transactions on Image Processing, 26(6):2944–2956, 2017.

[9] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In The IEEE Conference of Computer Vision and Pattern Recognition (CVPR), 2017.

[10] 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 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1357–1366, 2017.

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

[12] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. V. Gool, "Domain adaptive faster r-cnn for object detection in the wild," 2018 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3339–3348, 2018.

[13] W. Wei, D. Meng, Q. Zhao, Z. Xu, and Y. Wu, "Semi-supervised transfer learning for image rain removal," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3877–3886.

[14] W. Wei, L. Yi, Q. Xie, Q. Zhao, D. Meng, and Z. Xu, "Should we encode rain streaks in video as deterministic or stochastic?" in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2516–2525

[15] W. Wei, L. Yi, Q. Xie, Q. Zhao, D. Meng, and Z. Xu, "Should we encode rain streaks in video as deterministic or stochastic?" in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2516–2525

[16] H. Zhang and V. M. Patel, "Image de-raining using a conditional generative adversarial network," arXiv preprint arXiv:1701.05957, 2017

[17] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 3397–3405.

[18] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2736–2744.

[1] The gain is computed by subtracting the performance obtained using only D_L from the performance obtained using $D_L + D_U$.

[19] W. Wei, D. Meng, Q. Zhao, Z. Xu, and Y. Wu, "Semi-supervised transfer learning for image rain removal," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3877–3886.

[20] X. Fu, J. Huang, D. Zeng, X. Ding, Y. Liao, and J. Paisley, "Removing rain from single images via a deep detail network," In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1715–1723, 2017.

[21] L. Zhu, C. W. Fu, D. Lischinski, and P. A. Heng, "Joint bi-layer optimization for single-image rain streak removal," In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2536–2534, 2017

[22] H. Zhang and V. M. Patel, "Density-aware single image de-raining using a multi-stream dense network," In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. abs/1802.07412, 2018

[23] Gerald Schaefer and Michal Stich. Ucid: An uncompressed color image database. In Storage and Retrieval Methods and Applications for Multimedia 2004, volume 5307, pages 472–480. International Society for Optics and Photonics, 2003. 6

[24] Wei Wei, Lixuan Yi, Qi Xie, Qian Zhao, Deyu Meng, and Zongben Xu. Should we encode rain streaks in video as deterministic or stochastic? In Proceedings of the IEEE International Conference on Computer Vision, pages 2516–2525,2017. 6

[25] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zong ming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1357–1366, 2017. 1, 2, 3, 6, 8

[26] He Zhang and Vishal M Patel. Image de-raining using aconditional generative adversarial network. arXiv preprintarXiv:1701.05957, 2017. 2, 3, 6, 8

[27] Lin, T. Y. , et al. "Microsoft COCO: Common Objects in Context." European Conference on Computer Vision Springer International Publishing, 2014.

[28] Yu Luo, Yong Xu, and Hui Ji. Removing rain from a single image via discriminative sparse coding. In Proceedings of the IEEE International Conference on Computer Vision, pages 3397–3405, 2015. 6

[29] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using layer priors. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2736–2744, 2016. 6

[30] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun.Deep continuous fusion for multi-sensor 3d object detection.In Proceedings of the European Conference on Computer Vision (ECCV), pages 641–656, 2018. 1

[31] X Fu, J Huang, D Zeng, X Ding, Y Liao, and J Paisley. Removing rain from single images via a deep detail network.In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1715–1723, 2017. 1, 3, 6, 7

[32] L Zhu, C W Fu, D Lischinski, and P A Heng. Joint bilayer optimization for single-image rain streak removal. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2536–2534, 2017. 1, 3, 6

[33] H. Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), abs/1802.07412, 2018. 1, 3, 6, 7

[34] Rajeev Yasarla and Vishal M Patel. Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining. arXiv preprint arXiv:1906.11129, 2019. 3, 6

5. Group Contribution

Ruan Zhengxin: paper writing, network training
Qiu Tianyu: paper writing, presentation
Xie Yuhui: paper writing, network trainning
Qi Yinqiao: paper writing, evaluation
Wang Junda: paper writing, Make PPT