Rain-removing Algorithm for Image Based on Deep Learning

QingPing Zou*, Jing Wang, ShiTing Luo

HeChi College, Guangxi, China, 546300

*Corresponding author e-mail: qingping@163.com

Abstract. Aiming at the problem that the decrease of driver’s visual definition is affected by rainy weather, a new deep learning image rain removal algorithm is proposed. The network consists of a rainwater detection subnetwork and a rainwater removal subnetwork in series. The rainwater detection subnetwork adopts residual learning network, which can accurately learn the difference between rain images and non-rain images. The rain-removing subnetwork uses a U-shaped network with dense connections. On the one hand, it uses the U-shaped network to retain the details of the background, and on the other hand, the DenseNet is used to multiplex the lower layer features to higher layer features to improve the accuracy of rain removal, by combining them, it alleviates the contradiction between the loss of background details caused by excessive rain removal and the incomplete rain-removing. Experimental results show that this kind of deep rain-removing network can detect and remove the rainwater in the image well.

Key words: Rain-removing, Deep Learning, Image Processing

1. Preface

Visibility refers to the maximum horizontal distance that a person with normal vision can see the outline of the target clearly under the weather conditions at that time. The weather conditions that directly affect the visibility include fog, heavy rain, heavy snow, hail, etc., among them, fog and rain weather have the greatest impact on drivers with the longest circle. In rainy conditions, the driver’s visual vision is very difficult, resulting in vehicle collision accidents occur from time to time.

In order to solve this problem, early people put forward a priori-based method to deal with rainwater images by designing a statistically-based artificial priori. For example, Zhu et al. [1] proposed a method based on sparse coding, Chen et al. [2] proposed a method based on low-rank representation, Li et al. [3] proposed a method based on GMM (Gaussian mixture model) to separate the rain fringe layer. Kurihata et al. [4] used principal component analysis (PCA) to learn the shape of raindrops and tried to match an area in the test image to the shape of raindrops they learned. However, since raindrops are transparent and have different shapes, it is not clear how many raindrops need to be
learned, how to ensure that principal component analysis can simulate the various appearances of raindrops, and how to prevent other local areas which similar to raindrops from being detected as raindrops. Therefore, these prior-based methods limit the removal ability of rainwater and fail to recover high-frequency texture background and image details.

With the drive of big-data and vastly improved computing power, algorithms based on deep learning have been hugely successful in almost every computer vision task, including single-image rain-removing. For example, Eigen et al. [5] solved the problem of single-image raindrop removal. The basic idea of this method is to train a convolutional neural network containing two raindrop-degraded images and corresponding raindrop-free images. While this method is effective, especially for relatively sparse and smaller raindrops and dirt, it does not produce clean results for larger and dense raindrops. Chen et al. [6] recently proposed an end-to-end gated context aggregation network (called GCA-Net) for image defogging and rain removal. Unfortunately, we cannot apply this method to raindrop removal alone, because the image formation and the constraints of raindrops attached to glazing or lenses are different from fog.

Based on the rain model, this paper designs a new deep learning network for detecting and removing rain. When the rain image is known, there is a certain relationship between the rain fringe and the non-rain image. The detection of rain streaks is the same as the removal of the image area of interest, the detected rain streaks can guide the removal of rainwater. Therefore, this paper adopts a series method to detect and remove rainwater, so that the accurate estimation of rainwater can promote the removal of rain streaks, and the rain-removing subnetwork can also regulate the detection of rainwater through back propagation, so that the detection and removal of rainwater can promote each other.

2. The rain model
Considering factors such as rainfall density, wind speed and raindrop size, rainfall can cause several types of reduced visibility. The previous image rain-removing methods mainly focus on the task of rain mark removal, which obviously cannot cover all the problems of rain pattern. We divide rainfall models into two categories: rain tracks and rain drops.

The rain-mark image $I_s$ can be regarded as a combination of rainless background scene $B$ and linear rain-mark layer $S$.

$$I_s = B + S$$

(1)

Rain streaks can obscure and reduce the visibility of the background scene, clear image $B$ can be obtained by removing streaks layer $S$. In addition, we modeled the raindrop degradation image as a combination of background image and raindrop effect:

$$I_s = (1 - M) \odot B + R$$

(2)

Where $I_s$ is the input image and $M$ is the binary mask. $M(x) = 1$ means that pixel $x$ is part of the raindrop area, otherwise it is part of the background area. $B$ is the background image, and $R$ is the effect of raindrops attached to a lens or windshield, representing a complex mixture of background information and the light reflected by the environment. The symbol $\odot$ represents multiplication by element. In deep learning algorithms, these priors are fitted by training data and automatically
embedded into the network.

3. Train dataset
Similar to current deep learning methods, our approach requires a relatively large amount of paired tag data for training. To get a tagged dataset, we created our own composite dataset. For our example, we need a set of image pairs, each containing exactly the same background scene, but one blurred by rain and the other without raindrops.

We first downloaded and photographed a clean background image without rain from the Internet, and then used raindrops and rain streaks of different densities, shapes, sizes and transparency to synthesize rainy images. Our synthetic dataset contains 2000 sets of rain, background and rain streak images, including 1700 sets of training sets and 300 sets of test sets. Figure 1 shows a partial sample of the dataset. Each line from top to bottom represents light rain, moderate rain and heavy rain.

![Figure 1. synthetic dataset](image)

4 Depth rain-removal network
According to the rain model, rain and background are a linear combination to get the image of rain. Generally, after a rain image is given, the rain image and the background image are unknown, so the solution of the rain model is an ill-posed problem. Deep learning network needs to estimate these two unknown quantities at the same time, so this paper adopts two functional modules to detect and remove rainwater at the same time, and uses a series network structure so that the detection and removal of rainwater can promote each other. In order to learn the differences between the rainy images and the non-rainy images, the rain detection subnetwork adopts residual learning network, and the rain-removal subnetwork adopts the U-shaped network with dense connections. The U-shaped network can well retain the details of the image, and the multiplexing characteristics of DenseNet features with dense connection can improve the accuracy of the network. Our network structure is shown in figure 2.
4.1. **Subnetwork of rainwater detection**

The dynamic rain layer in the rain model contains not only the information of the location and shape of the rain, but also the information of the distribution, transparency and density of the rain. Therefore, the dynamic rainwater detection requires both low-level information and accurate estimation of high-level information. Generally, the bottom layer of the convolutional neural network extracts the low-level information and the high-level extracts the semantic information, and the more layers there are, the richer and abstract the extracted features are. However, simply increasing the number of network layers to extract high-level semantic information will lead to problems such as gradient disappearance and degradation. Therefore, this paper prevents the gradient from disappearing by introducing regularization and regularization intermediate layers.

The degradation problem is that the accuracy of the training set reaches saturation or decline with the increase of network layers. To solve this problem, the network uses the idea of residual learning\([7]\). If the layers behind the deep network are the identity mapping, then the model degenerates into a shallow network. It is difficult to fit some layers into an identity mapping function \(H(x)=x\), if the network is designed as \(H(x)=F(x)+x\), it can be converted to learning a residual function \(F(x)=H(x)-x\), as long as \(F(x)=0\), then it constitutes an identity map \(H(x)=x\) and it usually becomes easier to fit the residuals.

In order to accurately estimate the rainwater image, the detection sub-network uses residual network with 30 layers to detect the rainwater. The network consists of three 10-layer residual learning modules, the number of channels is adjusted through the convolution of \(1*1\) between the learning modules, the number of channels in each module is twice that of the previous module, each module contains five residual blocks, and each residual block consists of two layers of \(3*3\) convolutional layers, the residual connection method is used between the two layers.

4.2. **Subnetwork of rainwater removal**

The rainwater area needs to be restored after rainwater is detected, in order to retain the background details while restoring, this article uses U-net composed of Dense block to remove rainwater.

U-net\([8]\) is a deep learning network with a structure similar to the letter U, its schematic diagram is shown in figure 3. The U-net consists of two parts: contraction path and expansion path. The contraction path is mainly used to capture semantic information in the image, while the expansion path is mainly used to precisely locate the parts concerned in the image. By fusing the high pixel feature
extracted from the contraction path with the new feature map on the expansion path, the accurate location is realized and some important feature information in the previous sampling process is retained to the maximum extent.

Figure 3. U-net diagram

Dense block[9] is a network that maximizes the information flow between all layers of the network, and its structure diagram is shown in figure 4. It connects all layers in the network in pairs, so that each layer in the network accepts the features of all the layers in front of it as the input. Dense block adopts a dense connection mode so that each layer will receive the gradient signal of all subsequent layers during the back propagation of the network, so the gradient of the convolutional layer close to the input will not become smaller and smaller with the increase of the network depth, reducing the problem of gradient dissipation in the training process. At the same time, since a large number of features are multiplexed, a large number of features can be generated by using a small number of convolution kernels, ultimately making the size of the module smaller.

Figure 4. schematic diagram of a Dense block

In this paper, the U-net and Dense block are combined to design a U-Dense-Net network as the
rain-removal subnetwork. The network uses U-net to better retain the image details, while DenseNet multiplexes the features of the lower layer to the higher layer to improve the accuracy of rain-removal, so the combination of them alleviates the contradiction between the loss of background details due to excessive rain removing and the incomplete rain removing. The U-DenseNet contraction path is composed of three down-sampling modules, which include the Denseblock, bath normalization and pooling layer, among which the bath normalization contains the standardized operation and convolution operation with Relu activation, and the pooling layer is the down-sampling operation with a step size of 2 for the original image. The U-DenseNet extension path consists of three corresponding up-sampling modules. Each module consists of a deconvolution layer, a Denseblock and a bath normalization. First, the deconvolution layer is fused with the feature map on the corresponding contraction path, and then is sent to the Denseblock for convolution. Finally, the U-DenseNet network converts any number of feature vectors into 3d output images through a layer of 1*1 convolution.

4.3. Training details
The network uses one by one pixel-weighed MSE loss as the loss function, and the whole training process is to constantly update the network through error back propagation to minimize the loss function. In order to prevent problems such as overfitting and gradient disappearance during training, data were enhanced by rotation, clipping and transformation, and leakage, parameter regularization and normalization were used in most convolutional layers. The whole network is implemented by Tensorflow framework, and the objective function is optimized by inertial optimizer and stochastic gradient descent method. During the training process, all network parameters are initialized using a normal distribution with mean value of 0 and variance of 0.01; the inertia coefficient of the inertial optimizer is 0.9; the attenuation coefficient of network parameters is $10^{-6}$; the initial learning rate is $10^{-2}$.

5 Experiments and results
In order to demonstrate the effectiveness of the rain-removal network in this paper, quantitative analysis tests are made on the synthetic dataset, and qualitative tests are made on the real image, and compared with the latest rain-removal algorithm.

5.1. Composite image comparison experiment
In order to qualitatively evaluate the effect of rain removal, we compared the methods in literature [5,10] and used peak signal to noise ratio (PSNR) and structural similarity (SSIM) as the evaluation criteria. PSNR 和 SSIM can better measure the difference between the estimated rain removal image and the real non-rain image. Table 1 shows the test results on three different sizes of rain images and the entire test dataset. It can be seen from the table that the algorithm in this paper has the best rain removal effect. From figure 5, it can be seen that the algorithm in this paper can remove different types of rainwater and keep the background details well. Although the method of Eigen et al. can remove the rainwater, it can’t be removed well in response to heavy rain scenarios.
input synthetic rain image results of method in literature
results in this paper original non-rain image

**Figure 5.** comparison of rain removal effect of composite rain image

**Table 1.** comparison of PSNR and SSIM of different algorithms on datasets

| Light rain | Moderate rain | Heavy rain |
|------------|---------------|------------|
| PSNR       | SSIM          | PSNR       | SSIM          | PSNR       | SSIM          |
| Reference 5 methods | 26.48 | 0.86 | 25.22 | 0.85 | 24.81 | 0.86 |
| Reference 10 methods | 23.48 | 0.81 | 23.03 | 0.79 | 21.52 | 0.78 |
| Methods of this paper | 29.78 | 0.93 | 27.56 | 0.92 | 25.91 | 0.91 |

5.2. **Real image comparison experiment**

In order to further verify the rain removal effect of our network on real heavy rain images, we collected some real rain images and tested the algorithm in these images. Figure 6 shows the effect before and after removal. It can be seen from the figure that the algorithm in this paper can remove heavy rain well and the network has strong robustness.

**Figure 6.** comparison of real rain (heavy rain) image before and after rain removal
6. Conclusion
In order to improve the visual clarity of rainy day images, this paper proposes an algorithm based on deep learning to remove rainwater. The network consists of the rainwater detection subnetwork and the rainwater removal subnetwork. The rainwater detection subnetwork adopts the residual network to detect the location and transparency information of the rainwater, while the rainwater removal subnetwork adopts the U-DenseNet network combined with the U-net and DenseNet to retain the background details and remove the rainwater. Experiments show that the proposed algorithm achieves good rain-removal effect both in the synthetic images and the real images.

References
[1] Lei Zhu, Chi-Wing Fu, Dani Lischinski, and Pheng-Ann Heng. Joint bi-layer optimization for single-image rain streak removal. In Proceedings of the IEEE international conference on computer vision, pages 2526–2534, 2017. 3.
[2] Yi-Lei Chen and Chiou-Ting Hsu. A generalized low-rank appearance model for spatio-temporally correlated rain streaks. In Proceedings of the IEEE International Conference on Computer Vision, pages 1968–1975, 2013. 1, 3.
[3] Yu Li and Michael S Brown. Single image layer separation using relative smoothness. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages2752–2759, 2014. 3.
[4] H. Kurihata, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, and T. Miyahara. Rainy weather recognition from in-vehicle camera images for driver assistance. In Intelligent Vehicles Symposium, 2005. Proceedings. IEEE, pages 205–210. IEEE, 2005. 1, 2.
[5] D. Eigen, D. Krishnan, and R. Fergus. Restoring an image taken through a window covered with dirt or rain. In Proceedings of the IEEE International Conference on Computer Vision, pages 633–640, 2013. 2,3, 6, 7.
[6] Chen, D.; He, M.; Fan, Q.; Liao, J.; Zhang, L.; Hou, D.; Yuan, L.; and Hua, G. 2019. Gated context aggregation network for image dehazing and deraining. In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), 1375–1383. IEEE.
[7] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Deep residual learning for image recognition[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2016: 770-778.
[8] RONNEBERGER O, FISCHER P, BROX T. U-net: Convolutional networks for biomedical image segmentation[C]// International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, 2015: 234-241.
[9] HUANG Gao, LIU Zhuang, MAATEN L V D, et al. Densely connected convolutional networks[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2017: 2261-2269.
[10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial net-works. arXiv preprint arXiv:1611.07004, 2016. 3, 6,7.