An investigation of rain streak removal models based on expert experience and deep learning

Wanlin Du

College of Science, China Agriculture University, Beijing, China.

1225974412@qq.com

Abstract. Computer vision technology has a wide range of applications in today's society, and image rain removal is of great importance in outdoor vision capture. Today's image de-rain techniques are divided into video de-rain, and image de-rain, with the image de-rain task being more difficult than the video de-rain task due to the lack of a time factor. Current image rain removal methods are divided into three main types: filter-based methods, a priori knowledge-based methods and deep learning methods. Although these methods can achieve the image rain removal requirements to a certain extent, there is still no highly generalized method that can better solve the image rain removal problem in all cases. This paper first considers a filter-based approach, which takes less time to run but is difficult to remove cleanly for complex rain streaks. Secondly, this paper examines an a priori knowledge-based approach, which requires the study of how rain images are constructed as a priori knowledge and then uses the existing prior knowledge to remove rain from the images. This approach has a high reliance on prior knowledge and poor generalization. Finally, this paper investigates a deep learning-based method that requires a large number of supervised samples for training and has a better rain removal effect, but ignores the prior knowledge of rain streaks and is prone to overfitting. Based on these three methods we collected some data and experimental results in this field and summarized and analyzed them, giving reasons for the strengths and weaknesses of these three models and presenting new perspectives for future improvements in image de-rain methods.

Keywords: Computer vision technology, rain streak removal models, deep learning.

1. Introduction

In the current context of rapid technological development, the application of computer vision technology in object tracking [1], scene analysis [2], person re-identification [3] ,and event detection [4] is very promising, and images as the processing object of computer vision technology, its clarity directly affects the effect of subsequent processing analysis. Images captured from outdoor vision systems often need to be pre-processed to achieve the required clarity standards, where rain, as one of the most common weather conditions, is one of the most common causes of image clarity, so image de-rain is particularly important.

The task of image rain removal is to convert an image with rain into an image without rain, which is usually divided into two cases: video rain removal and single-image rain removal. In contrast to video rain removal, single-image rain removal is more difficult due to the lack of a time factor. In general, existing single-image rain removal methods can be divided into three categories: filter-based methods, a priori-based methods ,and deep learning-based methods. Early filter-based methods used bootstrap filters to decompose the image into low and high-frequency parts, using different bootstrap images to obtain the final rain-free result. Prior knowledge-based methods focus on introducing the study of the physical properties of rain streaks - such as sparsity and self-similarity - into the rain streak processing process and combine it with approaches such as dictionary learning and sparse coding for rain-containing images. More recent deep learning approaches have instead built deep neural networks, designed supervised learning algorithms, adversarial neural network algorithms ,and some semi-supervised/unsupervised learning methods for rain removal processing of images.

One of the first early approaches to image de-rain was the idea of using filters by Fu et al [5]. They used a bilateral filter to decompose the image into low and high-frequency components and then used dictionary learning and sparse coding to decompose the high-frequency components into "rain

components" and "non-rain components", successfully achieving rain removal from a single black and white image. After that, Xu et al [6] extracted a reference image from the original image based on an imaging model that was not affected by rain and snow and used the reference image as a guide image to de-rain the original image using a guide filter to achieve rain removal of the image with color retention. Zheng et al [7] proposed a de-rain method based on the low-frequency part of the image based on the previous method. Using the low-frequency part of the image without rain and snow streaks as the guide image, and the high-frequency part as the input image, the image was processed using a guide filter to obtain a de-rain image with high definition. Ding et al [8] proposed an improved rain and snow removal method, in which the principle of L0 gradient minimization was applied. In the method, the observed image and the low-frequency part are first used as input to a bootstrap filter to obtain coarse results and then refined to distinguish the rain and snow parts of the image by classifying the high-frequency edges of the image. The L0 gradient minimization bootstrap filter is then further used to achieve clarity of the background edges and smoothing of the rain and snow streaks to achieve a clearer de-rain image. Huang et al [9] proposed a single-image rain removal method combining adaptive morphological filtering and multiscale convolutional sparse coding to address the problem of incomplete rain streak information removal and texture detail loss. The method first combines the structural characteristics of rain patterns and proposes an adaptive multi-scale morphological filtering method to filter out the rain pattern information and obtain the low-frequency component of the image while preserving the image texture information as much as possible. An MS-CSC method is proposed to accurately detect the rain pattern information and obtain the rain layer by considering the directional characteristics of the rain pattern. Finally, the reconstructed rain layer is subtracted from the input rain image to obtain a rain-free image.

Although rain removal using bootstrap filters has the advantage of being faster and does not require additional data to learn, there is still room for improvement in rain removal. So instead of relying solely on different bootstrap filters to decompose rain images, some researchers have focused on introducing more physical properties of rain streaks, i.e. a priori information, to process the rain streak layers in an image. Sun et al [10] proposed a single-image rain removal method that exploits the structural similarity of images. An incremental dictionary learning strategy is first used to process high-frequency images to better separate their rain-containing components from non-rain components. The structural similarity of the image library is then used to formulate the high-frequency part of the image as an optimization problem so that the dictionary atoms representing the rain characteristics are automatically extracted. The remaining dictionary atoms, as well as the original low-frequency image are eventually synthesized and output as a rain-removal image, and the resulting image has good retention of background details. Luo et al [11] developed a single-image rain removal method based on discriminative sparse coding. Combining a non-linear screen mixture model used to model rain images, and then learning a mutually exclusive dictionary, which in turn accurately separates the rain and non-rain layers, the de-rain image is finally obtained using a greedy tracking algorithm. The experimental results are more complete for image background details retention. Li et al [12] proposed an effective method that uses a Gaussian mixture model trained on blocks of natural images, on top of which an additional gradient sparsity constraint is imposed thereby further regularising the image and using this to process the background and rain layers. This processing method can adapt to multiple directions and scales of rain patterns, and can qualitatively and quantitatively remove rain patterns better than existing methods. Zhu et al [13] proposed a new method for removing rain marks from the input image by decomposing the input image into a rain-free background layer and a rain mark layer. Using a joint optimization process, non-stripe details are removed from the rain-free background layer and the rain-scarred layer alternatively. Three images a priori methods are successively utilized in the process. Firstly, local gradients in the rain image are analyzed to separate the rainstain layer, and in this process, Zhu et al. introduce a centralized sparse representation to improve the performance of rainstain removal while retaining background detail. Next, the rain direction is constructed a priori by considering the angular deviation of the pixel gradient from the rain direction, and a set of rain-dominated patches are extracted. Finally based on the similarity

between the rain streak features themselves and the extracted rain streak features, Zhu et al. introduced a rain layer prior specifically for the rain streak layer, aiming to smooth the non-rain streak background details in the background layer by using the rain patches automatically extracted from the rain layer image. The method is effective in removing rain streaks and better-preserving background detail in both quantitative comparisons and visual results. Mu et al [14] introduced a flexible energy function model to characterize the feasibility constraint. Unlike existing optimization models, they added new domain knowledge related to the rain streak removal region as a feasibility constraint to work with. A learning-based two-layer a priori model is then formulated for the rain stripe removal problem, which provides a collaborative mechanism to jointly study feasibility and optimality, thus providing a learnable framework for the research problem. The method retains more background detail while working well for all types of image datasets. Sun et al [15] proposed an a priori guided rainwater perception network for single-image rain removal, which uses the Prewitt operator to guide the convolution and embed a priori knowledge related to rainwater direction into the model. It further constructs a multi-layer rainwater sensing system by connecting multiple detail units to learn rainwater details of different densities, and introduces a sparse prior of rain, peak signal-to-noise ratio ,and a structure similarity-based regulariser in the loss function to improve the rain removal results. The method better solves the problem of lack of rain distribution perception in the a priori method.

However, there are unavoidable problems with using a priori knowledge to de-rain a single image. For example, when processing rain images from real-life scenes, rain images collected from different sources often contain complex and variable structures, and using prior knowledge to process rain images from all scenes is usually subjective and incomplete, so this direct modeling approach to rain removal is often only applicable to certain pre-assumed scenarios. And due to certain shortcomings of the above two methods, data-based deep learning methods are gradually developed. Fu et al [16] introduced a deep learning method called DerainNet to remove rain marks from images. The method uses a convolutional neural network to learn the mapping relationship between the detail layers of images with and without rain by training synthetic rain images. The difference between this method and other common strategies for increasing the depth or breadth of the network is that Fu et al. choose to train DerainNet on the detail layer rather than the image domain. In addition, Fu et al. optimize the convolutional neural network framework by image enhancement to improve the visual effect. This method is faster than previous machine learning-related methods to process images. Zhang et al [17] proposed a new single-image de-rain method, called Image De-rain Conditional Generation Adversarial Network (ID-CGAN), which takes quantitative, visual ,and discriminative performance into account in the objective function and generates the corresponding adversarial network (CGAN) by enforcing additional constraints to minimize the difference between the de-rain image and its corresponding non-rain image. Where the adversarial loss of the GAN provides additional regularisation and helps to achieve superior results. In addition to proposing a new method for rain removal images, Zhang et al. also improved the "generator-discriminator" to reduce the artifacts introduced by the GAN and to ensure better visual results. Li et al [18] proposed a new deep-learning network architecture based on deep convolutional and recurrent neural networks. They treated the rain-stripe image as a superposition of multiple rain-stripe layers, applied Squeeze-and-Excitation (SE) blocks to assign different alpha values to each rain-stripe layer depending on the intensity and transparency, and stripped each rain-stripe layer in stages; At the same time, a recursive neural network is introduced to effectively guide the rain removal in the next stage while preserving the information of the previous stage. This method performs well in recovering rainfall images with complex rain patterns, especially heavy rainfall images. However, as the performance of deep networks for rain removal improves, their structure and learning become increasingly complex and diverse, making it difficult to analyze the usefulness of various network modules when developing new rain removal networks. Ren et al [19] addressed this problem by reconsidering the network architecture, inputs and outputs, and loss functions to form a better and simpler rain removal network. The method introduces a progressive residual network (PRN) by recursively expanding a shallow

residual network (ResNet) while introducing a recurrent layer to form another progressive recurrent network (PReNet). The application of these two networks enables the gradual removal of rain traces from the images with good rain removal results. In addition, Ren et al. used the intra-stage recursive calculation of ResNet in PRN and PReNet to achieve a significant reduction in the network parameters without a significant reduction in the rain removal effect. Wei et al [20] proposed a semi-supervised learning paradigm. Unlike traditional deep learning methods that use only rain-free images and images to which rain patterns are artificially added for supervised learning, Wei et al. use real rain images for training the deep learning network. The image de-rain process is achieved by processing the residuals between the input rain image and the expected output de-rain image into a specific parameterized rain streak distribution. The network is therefore able to significantly alleviate the problem of insufficient training samples as well as supervised sample bias. Yu et al [21] focused on a new issue emerging from the single-image rain removal approach using deep learning, namely its robustness against adversarial attacks in the image recognition process. Their study showed that when images are severely degraded due to heavy rainfall, for example, small perturbations in the images become less visible or difficult to detect, which in turn leads to rain removal methods being more vulnerable to adversarial attacks. After a systematic evaluation of the key modules in existing methods, Yu et al. integrated the modules with better rain removal performance in the evaluation to construct a more robust deep learning-based rain removal method. Liang et al [22] proposed an end-to-end rain removal network. Their proposed method extracts high-level semantic information and low-level detail information through the feature aggregation module and the residual channel attention mechanism, respectively, while introducing discriminators and adversarial loss to generate more realistic rain removal images, and fuses the above feature modules to propose a single-image rain removal network based on the attention mechanism and residual structure.

Based on the above summary of previous work, the work in this paper will focus on the following three areas:

(1) To summarise and analyze the advantages and disadvantages of the three image rain removal methods described above.

(2) To investigate the effectiveness of rain removal in the published literature in recent years.

(3) To give future directions for the improvement of image rain removal methods and the development of hybrid models.

A summary of the methods related to image rain removal is shown in Table 1.

Table 1. Summary of rain removal methods

Species	Method	Advantage	Disadvantage
Filters	Fu et al. [5] Xu et al. [6] Zheng et al. [7] Ding et al. [8]	Faster, no additional data required for learning	More visible rain residue
A priori knowledge	Sun et al. [10] Luo et al. [11] Li et al. [12] Zhu et al. [13] Mu et al. [14]	A priori knowledge of rain is used to provide a more targeted approach to rain mark removal	Difficult and time-consuming iterations to handle different directions and patterns of rain marks
Deep learning	Fu et al. [16] Zhang et al. [17] Li et al. [18] Ren et al. [19]	Better perception of rain image distribution, more targeted and more effective than other methods	Need to find a large number of supervised samples, ignore intrinsic prior knowledge of rain streaks, prone to overfitting

2. Model framework

2.1 Filtering framework

Xu et al [6] used a guided filter approach to de-rain a single image. They used an imaging model of a raindrop to obtain a guided image and then used a guided filter based on the guided image to achieve a single-image de-rain.

(1) Raindrop imaging model:

Due to the fast falling speed of raindrops, no spherical raindrops are observed in the image at normal exposure speeds, but rather rain patterns are formed by the rapid movement of raindrops [23]. The pixel brightness of the rain pattern in an image is a linear combination of the background irradiance and the irradiance of the raindrops, and the model equation for rain pattern imaging is formula (1)

$$\mathbf{I}_{rs} = \int_0^{\tau} \overline{\mathbf{E}}_{rs} dt + \int_{\tau}^{T} \overline{\mathbf{E}}_{b} dt \tag{1}$$

$\mathbf{I}_{rs}$ indicates the pixel brightness of the image at the point where the rain pattern is covered, $\overline{\mathbf{E}}_{rs}$ indicates the irradiance of the raindrop as it passes through the pixel location ,and $\overline{\mathbf{E}}_{b}$ indicates the average irradiance of the background pixels. T is the camera exposure time, τ is the time it takes for a raindrop to fall through a pixel location. Further, we let $\mathbf{I}_{b}$ be the background intensity, Define $\alpha = \tau / T$,Then (1) can be reduced to formula(2)

$$\mathbf{I}_{rs} = \alpha \cdot T \cdot \overline{\mathbf{E}}_{rs} + (1-\alpha)\mathbf{I}_{b} \tag{2}$$

We further define $\mathbf{I}_{E}$ as the equivalent ideal luminance value formed by assuming that the raindrops are always stationary over s during the exposure time T, then you can get $\mathbf{I}_{E} = T \cdot \overline{\mathbf{E}}_{rs}$, Combining (2) enables the final reduced imaging formula to be obtained formula(3):

$$\mathbf{I}_{rs} = \alpha \mathbf{I}_{E} + (1-\alpha)\mathbf{I}_{b} \tag{3}$$

(2) Guided images:

The lead image is a greyscale image depicting the general outline of $\mathbf{I}_{rs}$. It needs to retain the background outline similar to $\mathbf{I}_{rs}$ while not containing the rain outline in $\mathbf{I}_{rs}$. Since α in the raindrop imaging model given in A is difficult to find accurately, a bootstrap image model that does not require α is needed.

Since still, raindrops appear almost identical to white in an image, we can approximate the three RGB channels of $\mathbf{I}_{E}$ as equal, denoted as I_0. Let I_{b-max} be the maximum value in the RGB channel of $\mathbf{I}_{b}$ and I_{rs-max} be the maximum value in the RGB channel of $\mathbf{I}_{rs}$. And $(1-\alpha)$ is positive, so combining (3) gives formula(4):

$$I_{rs-\max} = \alpha I_0 + (1-\alpha) I_{b-\max} \tag{4}$$

Similarly, we define I_{b-min}, I_{rs-min} as the minimum values in the RGB channels of $\mathbf{I}_{b}$ and $\mathbf{I}_{rs}$ respectively, which gives us formula(5):

$$I_{rs-\min} = \alpha I_0 + (1-\alpha) I_{b-\min} \tag{5}$$

Further, subtracting (4) from (5) and eliminating the αI_0 in them gives formula(6):

$$I_f = I_{rs-\max} - I_{rs-\min} = (1-\alpha)\left(I_{b-\max} - I_{b-\min}\right) \tag{6}$$

I_f is the guide image pixel brightness. Due to the actual situation $\tau \ll T$, α converges to 0 and $(1-\alpha)$ converges to 1, so the I_f value is determined by $\mathbf{I}_{b}$ only and is independent of α.

(3) Guidance filter:

Next Xu et al. used the contour-preserving property of the bootstrap filter to find the image $\mathbf{I}'$, which needed to satisfy the requirement of having the same pixel contour as I_f and similar pixel

brightness to $\mathbf{I}_{rs}$, so that it had a similar background contour to $\mathbf{I}_{rs}$ but was unaffected by raindrops. Taking each RGB channel of $\mathbf{I}_{rs}$ as an input image p and combining it with the guide image I_f (denoted as I in the formula), the output image $\mathbf{I}'$ is obtained as following formula(7), formula(8), and formula(9):

$$q_i = \overline{a}_k I_i + \overline{b}_k \tag{7}$$

$$a_k = \frac{\frac{1}{|\omega|}\sum_{i\in\omega_k} I_i p_i - \mu_k \overline{p}_k}{\sigma_k^2 + \varepsilon} \tag{8}$$

$$b_k = \overline{p}_k - a_k \mu_k \tag{9}$$

where q_i is the RGB channel value for $\mathbf{I}'$ and a_k, b_k and p_k are the coefficients used. $|\omega|$, μ_k and σ_k^2 are the number of pixels in the ω_k, the mean of I_f and the variance of I_f, respectively. Further applying the bootstrap filter again, with $\mathbf{I}_{rs}$ as the input image and $\mathbf{I}'$ as the bootstrap image, then the de-rain image is obtained.

2.2 A priori framework

There are various algorithms for rain removal of images based on a priori knowledge, of which the single image rain removal method based on discriminative sparse coding used by Luo et al [11] is one of the more classical algorithms. The method first chooses to decompose the rain-containing image using a layer mixture model, and further uses sparsity-based prior knowledge as the core of the rain removal method.

(1) Layer blending model:

Rain images can be constructed in a variety of ways, one of the most common being the linear summation model (10):

$$J = I + R \tag{10}$$

Where J is the rain image, I is the de-rain image layer (background layer) and R is the rain image layer. However, the rain image synthesized by this model differs significantly from the rain image in real-life situations, so a modelling method that can more accurately reproduce the rain image is needed as the basis for the rain removal method. The layer blending model can be used to create a more realistic image of rain by inverting, multiplying, and inverting again the I and R layers as model(11):

$$J = 1 - (1 - I) * (1 - R) = I + R - I * R \tag{11}$$

"$*$" denotes the point-by-point multiplication operator. The model has been experimentally shown to be more realistic than the linear summation model for the construction of rain images.

(2) Sparse coding of separation layers:

Methods to achieve image recovery using sparse prior knowledge of layers are widely used and work well. In the method proposed by Luo et al. they also consider the sparsity prior of image layers and rain layers under the learning dictionary. Firstly, Y_1 is defined as the set of background layer image patches and Y_2 is the set of rain layer image patches, both of which are vectors. Defining $\mathcal{P}$ as the operator of the linear mapping between layers and the array of image patches, we have patch(12) and patch(13)

$$Y_1 = \mathcal{P}I \tag{12}$$

$$Y_2 = \mathcal{P}R \tag{13}$$

Then assuming that the dictionary D is learned in an unsupervised setting, there is a sparse approximation of the patches:

$$Y_1 \approx DC_1 \tag{14}$$

$$Y_2 \approx DC_2 \tag{15}$$

where C_1 and C_2 are sparse codes.

In the next step Luo et al. need to separate the background layer I and the rain layer R from the original image, and currently only have the composite data of Y_1, Y_2, so they need to learn a dictionary with mutually exclusive properties to achieve the separation of I and R. The principle of which is as following formula(16):

$$\mathcal{B}(C_k) = \sum_j C_{kj}{}^2 \tag{16}$$

where $\mathcal{B}(C_k)$ denotes the weight vector that calculates the l_2 parametric number of each row of the array $C_k(k=1,2)$. When the correlation between $\mathcal{B}(C_1)$, $\mathcal{B}(C_2)$ defined by $|\mathcal{B}(C_1)^\top \mathcal{B}(C_2)|$ is small enough, the two layers are better separated.

Following the above conclusions, a variational model(17) for image rain removal was proposed by Luo et al:

$$\min_{I,R,D,C_1,C_2} \|\mathcal{P}I - DC_1\|_F^2 + \|\mathcal{P}R - DC_2\|_F^2 \tag{17}$$

Satisfy the formula(18):

$$\begin{cases} J = I + R - I * R; \\ 0 \le I \le 1; \quad 0 \le R \le 1 \\ \|C_1[:,j]\|_0 \le T_1, \|C_2[:,j]\|_0 \le T_2, \quad \text{for all } j \\ |\mathcal{B}(C_1)^\top \mathcal{B}(C_2)| \le \epsilon_0. \end{cases} \tag{18}$$

where T_1 and T_2 are sparse constraints for each column of sparse codes C_1 and C_2 respectively.

2.3 Deep learning framework

The characteristics of deep learning lead to its wide application in data processing, and there are various methods for image rain removal using deep learning, among which the more classical ones include the method of applying convolutional neural networks by Fu et al [16], the method of generating adversarial networks by Zhang et al [17], and so on. Among them, Ren et al [19] argued that a simple deep network is usually not successful in removing rain marks from multiple rain images, and then considered solving the image rain removal problem in multiple stages, effectively reducing the complexity of the network.

(1) Progressive Residual Network (PRN):

Ren et al. first proposed a Progressive Residual Network (PRN) in which the basic shallow residual network contains three components:

(i) Convolutional layer (f_{in}): accepts the input image;

(ii) Combination of several residual blocks (ResBlocks): extraction of depth feature information f_{res};

(iii) Convolution layer (f_{out}): output rain removal result image.

These three components are stage invariant, i.e. they can be reused at different stages of the network. The processing of the PRN in the flowchart at the stage can be expressed as following formula(19):

$$\begin{aligned} \mathbf{x}_{t-0.5} &= f_{\text{in}}(\mathbf{x}_{t-1}, \mathbf{y}) \\ \mathbf{x}_t &= f_{\text{out}}(f_{\text{res}}(\mathbf{x}_{t-0.5})) \end{aligned} \tag{19}$$

The input to this process is the result of concatenating the $t-1$ stage image $\mathbf{x}_{t-1}$ with the rain image $\mathbf{y}$, which improves the rain removal effect compared to the previous rain removal method. The cycle section is shown in Figure 1.

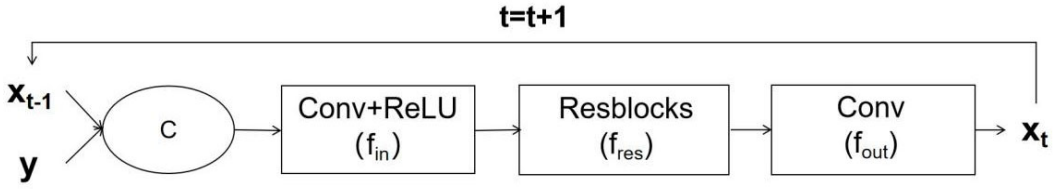


Figure 1. PRN cycle section

(2) Progressive Recurrent Network (PReNet):

Ren et al. also introduced a recurrent layer (20) into the PRN, the progressive recurrent network PReNet, which, in contrast to the PRN, contains recurrent states $\mathbf{s}_t$:

$$\begin{aligned}\mathbf{x}_{t-0.5} &= f_{\text{in}}\left(\mathbf{x}_{t-1}, \mathbf{y}\right) \\ \mathbf{s}_t &= f_{\text{recurrent}}\left(\mathbf{s}_{t-1}, \mathbf{x}_{t-0.5}\right) \\ \mathbf{x}_t &= f_{\text{out}}\left(f_{\text{res}}\left(\mathbf{s}_t\right)\right)\end{aligned} \tag{20}$$

where $f_{\text{recurrent}}$ is the recurrent layer, where a convolutional long short-term memory (LSTM) is used as the recurrent unit. The loop section of PReNet is shown in Figure 2, and this loop structure can effectively propagate the depth feature information of rain removal.

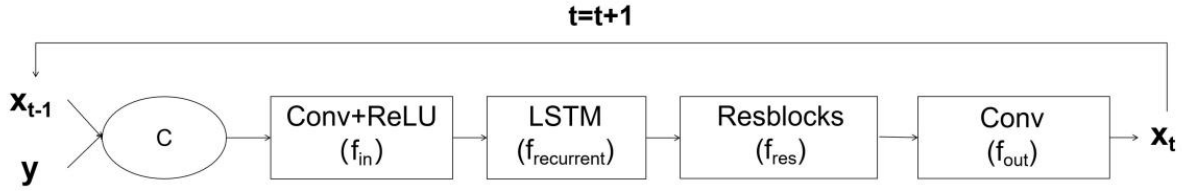


Figure 2. PReNet cycle section

(3) Network architecture:

PRN and PReNet have similar network architectures, with all convolutional kernels of size 3×3 and a full size of 1×1. f_{in} consists of ReLU [24] and a convolutional layer; f_{res} consists of five ResBlocks, each of which consists of two convolutional layers containing ReLU; and f_{out} consists of one convolutional layer. The convolution in f_{in} has 6 input channels and 32 output channels respectively, as all three RGB channels satisfy the series relationship between y and x_{t-1}. f_{out} takes the output of f_{res} with 32 channels as input and outputs a 3-channel RGB image for the PRN (or PReNet). There are two implementations of f_{res}: traditional ResBlocks and recursive ResBlocks, with PRN and PReNet using traditional ResBlocks. the difference between the two is that traditional ResBlocks are 5 consecutive ResBlocks, while recursive ResBlocks loop a ResBlock 5 times, reducing the network volume while ensuring de-rain effect. The network architecture of the two methods is shown in Figure 3.

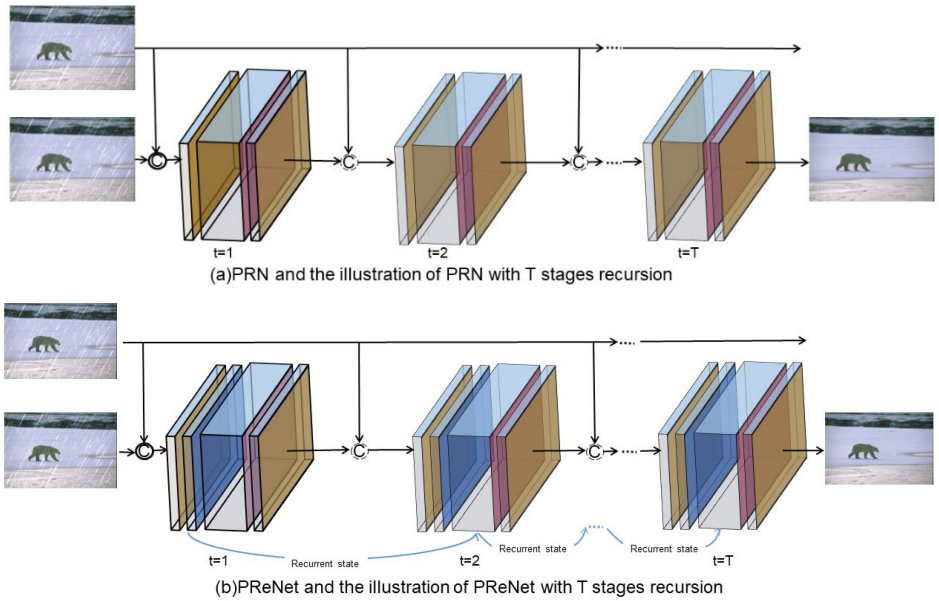


Figure 3. Network architecture of PRN, PReNet

(4) Target loss function:

Compared to the hybrid loss function, the single loss function requires fewer tuned parameters and is sufficient for training PRNs and PReNet, where the MSE loss or negative SSIM loss is better for training PRNs and PReNet.

For a model with T stages, there will be T outputs, i.e. $\mathbf{x}_1$, $\mathbf{x}_2$, … , $\mathbf{x}_T$. By applying supervision to the final output x_T only, the MSE loss(21) is:

$$\mathcal{L} = \|\mathbf{x}_T - \mathbf{x}_{gt}\|^2 \tag{21}$$

Negative SSIM losses(22) are:

$$\mathcal{L} = -SSIM(\mathbf{x}_T, \mathbf{x}_{gt}) \tag{22}$$

where $\mathbf{x}_{gt}$ is the corresponding rain-free background image.

In addition, recursive supervision can be imposed on each intermediate result(23):

$$\mathcal{L} = -\sum_{t=1}^{T} \lambda_t SSIM(\mathbf{x}_t, \mathbf{x}_{gt}) \tag{23}$$

where λ_t is the trade-off parameter for stage t. Recursive supervision can produce better visual results in the pre-processing of images.

3. Experimental results and discussions

3.1 Evaluation criteria

(1) SSIM:

SSIM (Structural similarity), a measure of how similar two images are. Given two images and two images, the formula for solving for structural similarity is as following formula(24):

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{24}$$

where μ_x and μ_y are the means of x and y respectively, σ_x^2 and σ_y^2 are the variances of x and y respectively, and σ_{xy} is the covariance of x and y. $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$ ($k_1 = 0.01$, $k_2 = 0.03$) is the constant that maintains stability. L is the dynamic range of the pixel values.

The value of SSIM is $[0,1]$. When the two images are identical in structure, $\mathrm{SSIM}=1$. SSIM has a higher value indicating better image quality.

(2) PSNR:

PSNR (Peak signal-to-noise ratio), which measures the level of quality of an image after some processing, is based on the definition of MSE (mean squared error): for a given original image I of size $m\times n$ and a noisy image K to which noise has been added, the MSE can be defined as formula (25):

$$\mathrm{MSE}=\frac{1}{\mathrm{mn}}\sum_{\mathrm{i}=0}^{\mathrm{m}-1}\sum_{\mathrm{j}=0}^{\mathrm{n}-1}[\mathrm{I}(i,j)-\mathrm{K}(i,j)]^{2} \tag{25}$$

Further, PSNR is defined as formula (26):

$$\mathrm{PSNR}=10\cdot\log_{10}\left(\frac{\mathrm{I}_{\max}^{2}}{\mathrm{MSE}}\right)=20\cdot\log_{10}\left(\frac{\mathrm{I}_{\max}}{\sqrt{\mathrm{MSE}}}\right) \tag{26}$$

Larger PSNR values indicate better image quality. In general:

(1) Above 40dB: indicates excellent image quality (very close to the original image).
(2) 30-40dB: usually indicates good image quality (distortion is perceptible but acceptable).
(3) 20-30dB: indicates poor image quality.
(4) Below 20dB: unacceptable image quality.

3.2 Datasets

In recent years many people have proposed various image datasets for rain removal, mainly divided into synthetic and real datasets, as shown in Table 2. The main difference between these datasets is the scale, shape, direction, and intensity of the rain streaks, and the more commonly used ones are summarised below. Of these, Rain12, Rain100L, Rain100H, Rain14000, Rain12000, Rain800, NYU-Rain, Outdoor-Rain, and RainCityscapes are all synthetic rain datasets, SPA-data is a real rain dataset, and MPID is a mixed dataset of real and synthetic rain datasets. Rain100L is a typical small rain dataset, meaning that the rain streaks are sparse, while Rain100H is a typical large rain dataset with denser rain streaks and relatively complex types. The exact format is shown in Figure 4.

Table 2. Summary of commonly used rain removal datasets

Dataset	Number (#train/#test)	Real/Synthetic	feature	publication
Rain12	12	Synthetic	Only for testing	Li *et al*[12]
Rain100L	1800/200	Synthetic	Only one type of rain streaks (light rain case)	Yang *et al*[25]
Rain100H	1800/200	Synthetic	Five types of rain streaks (heavy rain case)	Yang *et al*[25]
Rain14000	9100/4900	Synthetic	Clean images are selected from BSD500 and UCID	Fu *et al*[26]
Rain12000	12000/1200	Synthetic	The data has three kinds of densities	Zhang *et al*[27]
Rain800	700/100	Synthetic	The ground truth data is synthesized	Zhang *et al*[17]
NYU-Rain	13500/2700	Synthetic	Background images and the depth information are selected from NYU-Depth	Li *et al*[28]
Outdoor-Rain	9000/1500	Synthetic	Images are consisted of fog and rain	Li *et al*[28]
RainCityscapes	9432/1188	Synthetic	The rain-free images are selected from the training and validation sets of Cityscape.	Hu *et al*[29]
SPA-data	28500/1000	Real	Images with light interactions	Wang *et al*[30]
MPID	2400/250	Synthetic & Real	The dataset covers a much larger diversity of rain models	Li *et al*[31]

No rain images Rain100L Rain100H

Figure 4. Examples of common rain removal images

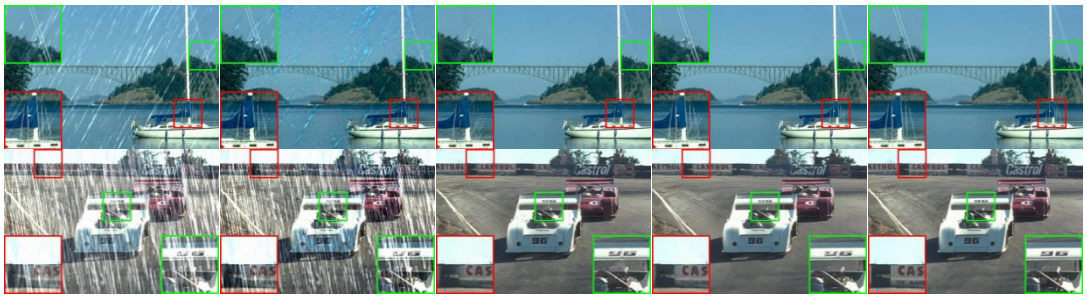
3.3 Performance assessment

(1) Synthetic rain datasets:

This section evaluates the rain removal effectiveness of some classical single-image rain removal methods on synthetic datasets [33][34], including methods using prior knowledge: Luo et al [11] (DSC), Li et al [12] (GMM), Gu et al [32] (JCAS), filter-based methods: Kang et al [35], deep learning-based methods: Fu et al [16] (Clear), Fu et al. [26] (DDN), Li et al. [18] (RESCAN) and Ren et al. [19] (PReNet), Wang et al. [30] (SPANet), Yang et al. [25] (JORDER-E), Wei et al. [20] (SIRR), Eigen D et al. [36].

Table 3. Comparison of PSNR and SSIM values for different rain removal methods under four synthetic datasets

Datasets	**Rain100L**		**Rain100H**		**Rain14000**		**Rain12**	
Metrics	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Input	26.90	0.838	13.56	0.371	25.24	0.810	30.14	0.856
ID [35]	23.13	0.670	13.78	0.397			27.21	0.753
DSC [11]	27.34	0.849	13.77	0.320	27.88	0.839	30.07	0.866
CNN [24]	23.70	0.814	13.21	0.371			26.65	0.783
ID-CGAN [12]	23.39	0.819	16.86	0.492			20.78	0.852
UGSM [25]	28.83	0.882	13.40	0.509			33.30	0.932
GMM [1]	29.05	0.872	15.23	0.450	27.78	0.859	32.14	0.916
JCAS [32]	28.54	0.852	14.62	0.451	26.20	0.847	33.10	0.931
Clear [16]	30.24	0.934	15.33	0.742	26.21	0.895	31.24	0.935
DDN [26]	32.38	0.926	22.85	0.725	28.45	0.889	34.04	0.933
SRCNN [37]	32.63	0.939	18.29	0.612			34.41	0.942
PReNet [19]	37.45	0.979	**30.11**	0.905	32.55	**0.946**	36.66	0.961
RESCAN [18]	38.52	0.981	29.62	0.872	32.03	0.931	36.43	0.952
SPANet [30]	34.46	0.962	25.11	0.833	29.76	0.908	34.63	0.943
JORDER-E [25]	**38.61**	**0.982**	30.04	**0.906**	**32.68**	0.943	**36.69**	0.962
SIRR [20]	32.37	0.926	22.47	0.716	28.44	0.889	34.02	0.935
DiG-CoM [38]	30.68	0.965					32.91	**0.964**
DSCAUD-RDN [39]	37.51	0.97	27.71	0.87			34.75	0.96



Input DSC PReNet JORDER-E Groundtruth

Figure 5. Several typical methods of rain removal in Rain100L (above) and Rain100H (below) are shown (The boxed section shows where the background of the image blends in with the rain streaks)

Overall, the PSNR and SSIM of the same method in the small rain dataset are significantly smaller than those in the large rain dataset, implying that the image quality and clean rain streak removal of the rain removal method are significantly better in the small rain dataset than in the large rain dataset.

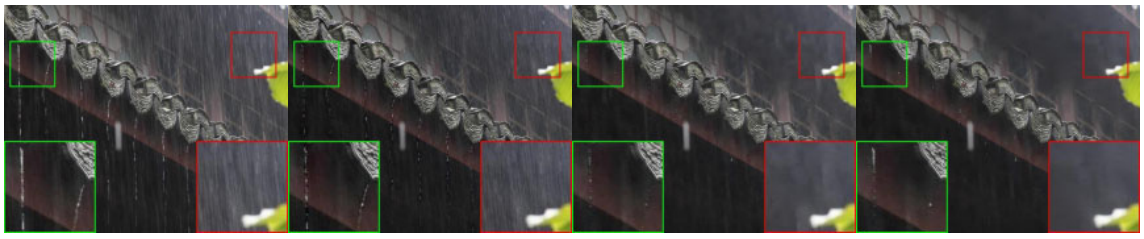
Taking the PReNet and JORDER-E with better rain removal effect and the DSC with poorer effect shown in the data in the table as examples, in the case of light rain (Rain100L), the model DSC obtained from prior knowledge still leaves obvious rain streaks remaining in the processed image, and its SSIM value in the table is 0.849, which indicates that the prior knowledge is not sufficient to completely identified and removed. Compared to this, the deep learning-based methods PReNet and JORDER-E show a more significant improvement in rain removal, with SSIM values reaching 0.979 and 0.982 respectively, and as can also be seen in Figure 2 there are almost no rain streaks present in the processed images. At the same time, however, as can be seen in the red enlargement, the PReNet-processed image also loses some of its detail, making the background more blurred, while JORDER-E retains more detail in the image, which is confirmed by the greater PSNR values of the JORDER-E method in the table.

In the case of heavy rain (Rain100H), the a priori knowledge model-driven method is even less effective in removing rain streaks due to the dense rain streaks and the variability of the rain streak shape and direction. Some of the deep learning-based methods still contain blurred rain streak outlines in the rain removal images. In comparison, the PReNet and JORDER-E methods are more effective in removing rain streaks and retaining background information, and both can still achieve SSIM values of 0.905 and 0.906 respectively at Rain100H, indicating that they can basically remove the rain streaks cleanly, while both methods can also retain good details for the images, with PSNR values of 30.11 and 30.04.

In general, the a priori knowledge-based model does not remove the rain streaks cleanly when removing them. In contrast, methods such as deep learning lose some of the background detail in the rain removal process, but in general, are significantly more effective than the a priori knowledge-based models.

(2) Real rain datasets:

As shown in Figure 6, the performance of each method decreases in the real rain image dataset compared to that in the synthetic rain dataset [33]. In terms of comparison between the different methods, the PReNet and JORDER-E methods can remove rain streaks better than the DSC method. The PReNet method outperforms the JORDER-E method in terms of rain image detail retention.



Input DSC PReNet JORDER-E

Figure 6. Demonstration of the rain removal effect of several typical methods on a real rain dataset (red boxes are where rain streaks are dense, green boxes are where they are sparse)

4. Summary and outlook

The task of image rain removal is to convert images with rain into images without rain. This paper summarises three main types of methods for rain removal of single images: filter-based methods, prior knowledge-based methods, and deep learning-based methods. The performance of the three methods on synthetic rain datasets and real rain datasets is also summarised, and their rain removal effects are evaluated through data and images.

(1) Filter-based methods: The advantages of this type of method are mainly that it is faster and does not require additional data for learning, but at the same time it is difficult to remove rain streaks cleanly.

(2) A priori knowledge-based methods: These methods (e.g. DSC) take the structure of the rain streaks into account, making the removal of rain streaks a clearer objective, but the iterations take longer. Their SSIM values are usually low, making it difficult to remove rain streaks cleanly.

(3) Deep learning-based methods: These methods tend to be more targeted in removing rain streaks from images, resulting in cleaner removal of rain streaks. For example, the PReNet and JORDER-E methods in 3.3 have higher SSIM values, the rain streaks in the output image are largely removed, and the retention of image details is better than the other two methods. However, the disadvantage is that a large amount of data is required for training, and there is still room for improvement in the performance in real rain images.

Based on the above generalizations, this paper suggests that subsequent rain removal methods could be improved in the following ways:

(1) An attempt was made to train with a real rain dataset so that rain streaks in real rain images could be better removed.

(2) Pre-classification of heavy and light rain images is combined with a priori knowledge, and the images are further fed into the model for iteration, thus enabling more accurate construction of rain removal models for various types of rain streaks.

(3) Optimisation of the iterative model, making the model simple and reducing the network parameters.

References

[1] Comaniciu D, Ramesh V, Meer P. Kernel-based object tracking[J]. Pattern Analysis & Machine Intelligence, 2003, 25(5):564-575.

[2] Itti L. A model of saliency-based visual attention for rapid scene analysis[J]. IEEE Trans, 1998, 20.

[3] Farenzena M, Bazzani L, Perina A, et al. Person re-identification by symmetry-driven accumulation of local features[C]// Proc IEEECONFERENCE on Computer Vision & Patternrecognition. IEEE, 2010:2360-2367.

[4] Shehata M S, Cai J, Badawy W M, et al. Video-Based Automatic Incident Detection for Smart Roads: The Outdoor Environmental Challenges Regarding False Alarms[J]. IEEE Transactions on Intelligent Transportation Systems, 2008, 9(2):349-360.

[5] Fu Y H, Kang L W, Lin C W, et al. Single-frame-based rain removal via image decomposition[C]// IEEE International Conference on Acoustics. IEEE, 2014.

[6] Jing X, Wei Z, Peng L, et al. Removing rain and snow in a single image using guided filter[C]// 2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE). 0.

[7] Zheng X, Liao Y, Guo W, et al. Single-image-based rain and snow removal using multi-guided filter[J]. Springer Berlin Heidelberg, 2013.

[8] Ding X, Chen L, Zheng X, et al. Single image rain and snow removal via guided L0 smoothing filter[J]. Multimedia Tools & Applications, 2016, 75(5):2697-2712.

[9] Huang S Y, Xu Y T, Yang Y, et al. Image deraining algorithm based on morphological filtering and convolution sparse coding[J]. Chinese Journal of Image Graphics, 2022, 27(5): 1522-1536

[10] Sun S H, Fan S P, Wang Y C F. Exploiting image structural similarity for single image rain removal[C]//2014 IEEE International Conference on Image Processing (ICIP). IEEE, 2014: 4482-4486.

[11] Luo Y, Xu Y, Ji H. Removing rain from a single image via discriminative sparse coding[C]//Proceedings of the IEEE international conference on computer vision. 2015: 3397-3405.

[12] Li Y, Tan R T, Guo X, et al. Rain streak removal using layer priors[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2736-2744.

[13] Zhu L, Fu C W, Lischinski D, et al. Joint bi-layer optimization for single-image rain streak removal[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2526-2534.

[14] Mu P, Chen J, Liu R, et al. Learning bilevel layer priors for single image rain streaks removal[J]. IEEE Signal Processing Letters, 2018, 26(2): 307-311.

[15] Sun G, Shao H, Cattani C. A priori-guided multi-layer rain-aware network for single image deraining[J]. Knowledge-Based Systems, 2022, 235: 107613.

[16] Fu X, Huang J, Ding X, et al. Clearing the skies: A deep network architecture for single-image rain removal[J]. IEEE Transactions on Image Processing, 2017, 26(6): 2944-2956.

[17] Zhang H, Sindagi V, Patel V M. Image de-raining using a conditional generative adversarial network[J]. IEEE transactions on circuits and systems for video technology, 2019, 30(11): 3943-3956.

[18] Li X, Wu J, Lin Z, et al. Recurrent squeeze-and-excitation context aggregation net for single image deraining[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 254-269.

[19] Ren D, Zuo W, Hu Q, et al. Progressive image deraining networks: A better and simpler baseline[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 3937-3946.

[20] Wei W, Meng D, Zhao Q, et al. Semi-supervised transfer learning for image rain removal[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 3877-3886.

[21] Yu Y, Yang W, Tan Y P, et al. Towards Robust Rain Removal Against Adversarial Attacks: A Comprehensive Benchmark Analysis and Beyond[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 6013-6022.

[22] Liang X, Zhao F. Single-Image Rain Removal Network Based on an Attention Mechanism and a Residual Structure[J]. IEEE Access, 2022.orks[J]. IEEE Transactions on Industrial Electronics, 2019, 67(8): 6473-6482.

[23] Liu P, Xu J, Liu J F, et al. An Algorithm for Real-time Analysis of Rain-affected Videos[J]. ACTA AUTOMATICA SINICA, 2010, 36(10):1371-1378.

[24] Nair V, Hinton G E. Rectified Linear Units Improve Restricted Boltzmann Machines Vinod Nair[C]// Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel. 2010.

[25] Yang W, Tan R T, Feng J, et al. Deep joint rain detection and removal from a single image[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1357-1366.

[26] Fu X, Huang J, Zeng D, et al. Removing rain from single images via a deep detail network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 3855-3863.

[27] Zhang H, Patel V M. Density-aware single image de-raining using a multi-stream dense network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 695-704.

[28] Li R, Cheong L F, Tan R T. Heavy rain image restoration: Integrating physics model and conditional adversarial learning[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 1633-1642.

[29] Hu X, Fu C W, Zhu L, et al. Depth-attentional features for single-image rain removal[C]//Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. 2019: 8022-8031.

[30] Wang T, Yang X, Xu K, et al. Spatial attentive single-image deraining with a high quality real rain dataset[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 12270-12279.

[31] Li S, Araujo I B, Ren W, et al. Single image deraining: A comprehensive benchmark analysis[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 3838-3847.

[32] Gu S, Meng D, Zuo W, et al. Joint convolutional analysis and synthesis sparse representation for single image layer separation[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1708-1716.

[33] Wang H, Wu Y, Li M, et al. A survey on rain removal from video and single image[J]. arXiv preprint arXiv:1909.08326, 2019.

[34] Yang W, Tan R T, Feng J, et al. Joint rain detection and removal from a single image with contextualized deep networks[J]. IEEE transactions on pattern analysis and machine intelligence, 2019, 42(6): 1377-1393.

[35] Kang L W, Lin C W, Fu Y H. Automatic Single-Image-Based Rain Streaks Removal via Image Decomposition[J]. IEEE Transactions on Image Processing, 2011, 21(4):1742-1755.

[36] Eigen D, Krishnan D, Fergus R. Restoring an Image Taken through a Window Covered with Dirt or Rain[C]// IEEE International Conference on Computer Vision. IEEE, 2014.

[37] Dong C, Chen C L, He K, et al. Image Super-Resolution Using Deep Convolutional Networks[J]. arXiv e-prints, 2014(2).

[38] Shang Z, Wu J. Research progress on single image rain removal technology[C]// 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC). IEEE, 2020.

[39] Li P, Tian J, Tang Y, et al. Model-Based Deep Network for Single Image Deraining[J]. IEEE Access, 2020, 8:14036-14047.