Data-Driven Single Image Deraining: A Comprehensive Review and New Perspectives

Zhao Zhang $^{1,1,1}$, Yanyan Wei $^2$, Haijun Zhang $^2$, Yi Yang $^2$, Shuicheng Yan $^2$, and Meng Wang $^2$

$^1$Hefei University of Technology
$^2$Affiliation not available

November 8, 2023

Abstract

Abstract—Due to the powerful fitting ability of neural networks and massive training data, data-driven Single Image Deraining (SID) methods have obtained significant performance, and most of the existing studies focus on improving the deraining performance by proposing different kinds of deraining networks. However, the generalization ability of current SID methods may still be limited in the real scenario, and the deraining results also cannot effectively improve the performance of subsequent high-level tasks (such as object detection). The main reason may be because the current works mainly focus on designing new deraining neural networks, while neglecting the interpretation of the solving process. To investigate these issues, we in this paper re-examine the three important factors (i.e., data, rain model and network architecture) for the SID task, and specifically analyze them by proposing new and more reasonable criteria (i.e., general vs. specific, synthetical vs. mathematical, black-box vs. white-box). We will also study the relationship of the three factors from new perspectives of data, and reveal two different solving paradigms (explicit vs. implicit) for the SID task. Besides, we also profoundly studied one of the three factors, i.e., data, and measured the performance of current methods on different datasets via extensive experiments to reveal the effectiveness of SID data. Finally, with the above comprehensive review and in-depth analysis, we also draw some valuable conclusions and suggestions for future research.
Data-Driven Single Image Deraining: A Comprehensive Review and New Perspectives

Zhao Zhang\textsuperscript{a,b,*}, Yanyan Wei\textsuperscript{a,b}, Haijun Zhang\textsuperscript{c}, Yi Yang\textsuperscript{d}, Shuicheng Yan\textsuperscript{e} and Meng Wang\textsuperscript{a,b}

\textsuperscript{a}School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China.  
\textsuperscript{b}Key Laboratory of Knowledge Engineering with Big Data (Ministry of Education), Hefei University of Technology, Hefei 230009, China.  
\textsuperscript{c}Department of Computer Science, Harbin Institute of Technology (Shenzhen), Shenzhen, China.  
\textsuperscript{d}Centre for Artificial Intelligence, University of Technology Sydney, Sydney, NSW, Australia.  
\textsuperscript{e}Department of Electrical and Computer Engineering, National University of Singapore, Singapore.

ARTICLE INFO

Keywords:  
Data-driven single image deraining  
comprehensive review  
novel perspective of data  
model  
solving paradigms  
in-depth analysis  
effectiveness of data.

ABSTRACT

Single Image Deraining task aims at recovering the rain-free background from an image degraded by rain streaks and rain accumulation. For the powerful fitting ability of deep neural networks and massive training data, data-driven deep SID methods obtained significant improvement over traditional ones. Current SID methods usually focus on improving the deraining performance by proposing different kinds of deraining networks, while neglecting the interpretation of the solving process. As a result, the generalization ability may still be limited in real-world scenarios, and the deraining results also cannot effectively improve the performance of subsequent high-level tasks (e.g., object detection). To explore these issues, we in this paper re-examine the three important factors (i.e., data, rain model and network architecture) for the SID problem, and specifically analyze them by proposing new and more reasonable criteria (i.e., general vs. specific, synthetic vs. mathematical, black-box vs. white-box). We also study the relationship of the three factors from a new perspective of data, and reveal two different solving paradigms (explicit vs. implicit) for the SID task. We further discuss the current mainstream data-driven SID methods from five aspects, i.e., training strategy, network pipeline, domain knowledge, data preprocessing, and objective function, and some useful conclusions are summarized by statistics. Besides, we profoundly studied one of the three factors, i.e., data, and measured the performance of current methods on different datasets through extensive experiments to reveal the effectiveness of SID data. Finally, with the comprehensive review and in-depth analysis, we draw some valuable conclusions and suggestions for future research.

1. Introduction

Single image deraining (SID) or single image rain removal has been emerging as an important task in different areas of image processing and computer vision, which has a wide range of applications relating to single image processing and restoration. However, the quality of the images captured on rainy days by the outdoor vision system, such as autonomous driving, person/vehicle tracking and surveillance, are usually degraded by the rain streak, raindrop or rain accumulation (see Fig.1). This will directly degrade the subsequent high-level vision tasks, e.g., object detection [68, 89], image recognition [97, 163] and saliency detection [32, 132]. In addition, due to the irregular and complex rain information in practice and the ill-posed properties, the task of SID that aims to estimate the rain-free background from degraded image is still a challenging and unmanageable issue to date. Unlike the video deraining that can leverage temporal redundancy and dynamics of rain, SID mainly exploits the spatial information of neighboring pixels and the visual properties of rain and background scenes. As such, SID tasks usually confront with more difficulties than the video deraining task [4, 50, 69, 70].

To study the SID task, existing methods can be generally divided into two basic categories, i.e., traditional methods (such as filter-based, prior-based and model-based ones) [9, 12, 16, 66, 115, 130, 131] and deep learning-based data-driven methods [23, 42, 60, 64, 84, 88, 113, 121, 136, 153, 154]. The filter-based methods aim to filter the rain image to obtain the rain-removed image [130, 131]. Classical methods for prior- and model-based types include sparse representation [16, 115], low-rank representation [9, 12] and
gaussian mixture model [66], etc. Sparse coding finds the sparsest representation of the input in the form of a linear combination of basic elements (or dictionary atoms) as well as the basic elements themselves. Low-rank representation is a rank minimization problem, in which the cost function measures the fit between the given data matrix and approximating matrix (i.e., optimization variable), subject to a constraint that the approximating matrix has a reduced rank.

With the rapid development of deep learning [39, 44, 97], data-driven deraining methods have obtained impressive performance improvement over those traditional deraining methods. In this field, Rain1400 [23] and Rain100H/Rain100L [136] datasets were first proposed in 2017 for the task of data-driven SID. After that, more and more synthetic SID datasets have been constructed, and meanwhile, the study on data-driven SID enters into a new period. In addition to synthesizing rain image data, different kinds of side information have been developed based on the original data, such as rain streak, rain mask, rain density, and image depth. By designing specific SID methods [42, 60, 84, 136, 153] that can take advantage of these side information, state-of-the-art deraining performance has been obtained. It is noteworthy that, there are three important influencing factors that have to be discussed for the problem of SID, i.e., Data, Rain model and Network architecture:

- **Data.** Data are the core factor of data-driven SID methods. Massive training data can usually enable the deraining networks to obtain better performance. However, due to the lack of image pairs in the real world, most current SID methods use synthetic data for both training and testing.

- **Rain model.** The rain model mainly explores how to describe and model the rain mathematically, and how to synthesize the rain image. Rain is a complex optical phenomenon and the rain removal problem is usually ill-posed, so the rain model indicates how we view the phenomenon and how we wish to solve the problem.

- **Network architecture.** The network structure is the most changeful and charming factor in the SID problem. Current studies have designed lots of new modules and network architectures of different properties to improve the deraining performance.

In recent studies, the performance of SID is getting better, however, this is usually in the cases of processing synthetic image data rather than real-world data. Specifically, the generalization ability of current SID methods is still limited in real-world scenarios. Furthermore, the rain removal task still cannot effectively improve the subsequent high-level vision tasks (such as object recognition and detection [5, 89]). To this end, several researchers have conducted experiments to enable the SID methods to have better application potential. For example, a large-scale benchmark dataset MPID [61], which contains both synthetic and real rain images with various rain types, was proposed to evaluate the performance of existing SID methods. The experimental evaluation and result analysis reveal the performance gap between synthetic and real-world data to some extent.

It is noteworthy that current studies and analyses on SID mainly focus on measuring the **effectiveness of SID methods** [62, 107, 109, 137], and discussing the three influencing factors independently without clearly discussing their relationships and the mutual influence on the deraining task. In this paper, we therefore re-define the classification of the three factors, figure out their intrinsic relations, and moreover reveal different solving paradigms to address the SID task. Furthermore, we mainly examine the **effectiveness of SID data** based on new evaluation criteria. Overall, the main contributions of this paper are summarized as follows:

1. **New divisions on the three factors:** We provide new, more reasonable and easily understandable divisions on data (general vs. specific), rain model (synthesis vs. mathematical) and network architecture (black-box vs. white-box) through re-examining and rethinking the three influencing factors of the SID task.

2. **In-depth analysis:** We figure out the relations among the three factors and reveal two different solving paradigms (explicit vs. implicit). Besides, we also analyze and categorize the current 97 mainstream SID methods according to different properties and perspectives (i.e., training strategy, network pipeline, domain knowledge, data preprocessing and objective function).

3. **Novel experimental design:** We design and prepare new forms of experimental configurations to evaluate the effectiveness of SID data by providing novel quantitative evaluation criteria. To the best of our knowledge, this is the first work to evaluate and quantify the existing mainstream SID datasets.

4. **Instructive conclusion:** Based on the in-depth analysis and evaluations, we derive some practical and instructive conclusions on how to determine and choose data appropriately, which can help the researchers who are confused about the choice of SID data.

The rest of this paper is organized as follows. In Section 2, we introduce the background of single image deraining and related survey papers on rain removal task and describe the difference to our work. Section 3 illustrates the three factors, i.e., data, rain model and network architecture from new divisions. Section 4 explores the relationship between the three factors and discusses the solving paradigms of SID methods. In Section 5, we summarize the recent related data-driven SID methods from different aspects. In Section 6, we design novel forms of experiments to rank the public paired datasets and provide a detailed analysis. Finally, Section 7 provides some valuable conclusions and discusses some future directions.

2. Related Work

2.1. Single Image Deraining

Single image deraining technology has been developed rapidly in recent years, and existing SID methods can be roughly divided into two categories: traditional model optimization-based methods (i.e., non-deep learning methods) and deep learning-based data-driven methods. Model optimization-based SID methods include sparse coding [26, 75],
A brief introduction to the four closely-related survey papers and our survey paper on SID.

| Paper      | Description                                                                 | Key Words                                    | Year |
|------------|-----------------------------------------------------------------------------|----------------------------------------------|------|
| Wang et al. [107] | Present a comprehensive review for current rain removal methods for video and a single image. | Single image/Video deraining; lots of experiments; public github repository | 2022 |
| Li et al. [62] | Present a comprehensive study and evaluation of existing single image deraining algorithms, using a new largescale benchmark consisting of both synthetic and real-world rain images of various rain types. | A new benchmark dataset; lots of experiments; public Github repository | 2021 |
| Wang et al. [109] | Shortly discuss three categories of deraining approaches: filter-based, conventional prior-based, and deep learning-based approaches, and then proposing a new SID method. | Single image deraining; a new SID method | 2020 |
| Yang et al. [137] | Discuss two categories of deraining approaches: model-based and data-driven approaches from many perspectives. | Single image deraining; detailed analysis | 2020 |
| Ours       | (1) New divisions on the data, rain model and network architecture; (2) Reveal two different solving paradigms of SID; (3) Categorize current SID methods according to different properties and perspectives; (4) Evaluate the effectiveness of SID data through novel evaluation criteria. | Single image deraining; detailed analysis; lots of experiments; effectiveness of SID data | -   |

2.2. Related Survey Papers

So far, there are four most relevant survey papers to our work in recent years, i.e., [62, 107, 109, 137]. These works discuss and analyze the SID methods from different perspectives, and we summarize the four closely-related survey papers and our survey paper in Table 1. From the comparison, we can conclude that:

1) Wang et al. [107] (SCIS’22) simultaneously reviewed the video deraining and SID. Specifically, for the parts on SID, they mainly divide the existing methods into three categories according to the ways of problem-solving, i.e., filter-based, prior-based, and deep learning-based data-driven methods. However, data-driven SID methods only occupy a relatively small proportion. Specifically, the analysis part does not leave much space to introduce the data-driven methods, and the experimental part only evaluates seven data-driven methods, which are only trained on three datasets and tested on five datasets. Based on the evaluation results, they summarized the deficiencies of current deraining methods, and also presented some remarks to illumine some meaningful future research directions along this line.

2) Li et al. [62] (ICCV’21) presented a comprehensive study and evaluation of 6 existing SID algorithms and introduced a new large-scale benchmark dataset (MPID) that contained both synthetic and real-world rain images of various rain types. To be specific, the MPID dataset contains three types of synthetic rain models (rain streak, raindrop, and rain mist), as well as a rich variety of evaluation criteria (two full- and three no-reference objective evaluations, subjective evaluation, and task-specific evaluation). The evaluation and analysis can indicate the performance gap between the synthetic and real-world rain images to some extent.

3) Wang et al. [109] (IJMLC’20) provided an inextensive review of the current SID techniques and mainly categorized them into three classes, that is, early filter-based, conventional prior-based, and recent deep learning-based data-driven approaches. Furthermore, inspired by the rationality of the current deep learning-based method PReNet [88] and insightful characteristics underlying rain shapes, they also build a specific coarse-to-fine deraining network architecture, which can finely deliver rain structures and progress-
sively remove the rain streaks from the input images.

4) Yang et al. [137] (TPAMI’20) mainly focus on discussing the SID problem. Specifically, they divided the existing SID methods into model-based and data-driven methods, and further divided the deep learning-based data-driven methods into four sub-categories, i.e., deep convolutional neural networks (CNN-based), generative adversarial networks (GAN-based), semi-/unsupervised learning-based and benchmarks. Subsequently, they described the related methods in the four sub-categories in detail, including the network architectures, basic blocks, loss functions, and datasets. To evaluate the performance of each method, they selected eight deep learning-based data-driven methods and evaluated them using three synthetic datasets. However, this work did not provide detailed evaluations but only a performance graph. In addition, they also proposed some future directions, including the integration of physics models and real images, rain modeling, evaluation methodology, and more related tasks and real-world applications.

Remarks. After a brief introduction to the four related survey papers, we will present the difference between current works and ours. Wang et al. [109] propose a new SID method and leaves less space for reviewing current methods, so we will not present the details with this work here. We will mainly compare our work to the other three survey papers [62, 107, 137]. Firstly, the experiments in [62] focused on fairly evaluating the performance of SID methods using a new benchmark dataset. It reveals the shortcomings of current rain removal issue, including the high complexity of the rain removal task, lack of appropriate evaluation metrics, poor generalization on real images, and little help for upstream tasks. In a traditional sense, [109] is not a real survey paper, since it is ground-breaking work, but it only reveals some new features of the SID task. Inspired by [62], we have designed more experiments to reveal the two key issues mentioned above, to which less attention has been paid. From the contents of [107, 137], it is clear that they both lack investigation and analysis of the relationship between data and network structures. Particularly, their conducted experiments cannot deduct the following several issues: (1) superiority of different data, since appropriate data during the training phase can enhance the representation and generalization abilities of the networks, which can serve better for the real scenario and practical applications; (2) pros and cons of different solving paradigms, while we investigate them carefully via experiments on side information, different operations, and explicit or implicit paradigms. Specifically, different from [107, 137] that also focus on the video deraining and traditional SID methods, we mainly focus on the data-driven SID task, where the related methods are categorized from the perspective of data into general and specific ones based on the relationship between data and networks. Note that the conclusions of the two categories are derived from a large number of experimental verifications, and some discussions and suggestions are also provided on this basis. We believe this survey paper will be able to provide a more insightful view for understanding the
data-driven SID methods from a new perspective of data.

One of the most important purposes of exploring the SID task is that it can be potentially used in various applications of realistic scenarios, so the deraining results on real rain images (without ground truth) will determine the actual rain removal ability of each method. However, most existing data-driven SID methods are supervised, which needs all paired data for training, so they cannot be trained directly based on real images. Specifically, supervised methods are usually trained on synthetic datasets and tested on real images, i.e., the test result on real images represents the generalization ability of each method. Note that this result will be directly determined by the properties of synthetic datasets, including the rain direction, density, shapes, etc. In other words, to solve the real rain removal task well, the characteristics of synthetic data should be closer to those of the real rain as much as possible. However, most existing data-driven SID methods usually focus on the performance evaluation of synthetic datasets to reflect the deraining ability of the proposed rain models and are only tested on some real rain images to evaluate the so-called generalization ability. Clearly, this evaluation process is understandable and can be performed easily, however, the key and the most difficult issue is how to define a so-called “best” synthetic dataset. That is, how to select data and rain models appropriately for the task, and then design the corresponding network structures and optimization strategies will be worthy of discussion. But to the best of our knowledge, there is no prior study or discussion on these topics yet.

3. The Three Factors

In this section, we discuss the divisions of the three factors based on data (general vs. specific), rain model (synthetic vs. mathematical) and network architecture (black-box vs. white-box) by new and more reasonable criteria.

3.1. Data

As far as we can know from the literature, more than 30 paired datasets were proposed in current studies from 2017 to 2022. These datasets contain image pairs (i.e., rain image (O) and a clean ground-truth image (B)), as well as some side information, such as rain streak (S), rain mask (M), image depth (F), rain density (D), transmission map (T), and atmospheric light (A). According to whether the data contain extra side information, we divide the SID data into general data and specific data. Fig.2 and Fig.3 illustrate some representative general and specific datasets. Fig.4 shows an example of paired images with side information for SID. Table 2 summarizes these mainstream paired datasets.

3.1.1. General data

The datasets of this kind only contain the image pairs (i.e., rain image and the corresponding ground-truth image), without any side information. Some representative general datasets are described below:

- Rain800 [154]: 800 clean images are randomly chosen from the UCID [93] and BSD [1] datasets. The authors
add the rain streaks into these clear images using Photoshop [81]. After that, 700 pairs are chosen for training, while 100 pairs are chosen for testing.

- **Rain1400** [23]: 1,000 clean images are randomly chosen from UCID [93], BSD [1] and Google. Based on the Photoshop software [81], each clean image was used to generate 14 rain images with different streak orientations and magnitudes. They randomly select 12,600 image pairs for training and 1,400 for testing.
- **Rain12** [66]: It is only a test dataset that contains 12 rain images with one type of rain streak.
- **SPA-Data** [113]: 170 real rain videos are captured by a mobile phone or collected from the Internet. By using the video deraining method [113], 29,500 image pairs are generated, split into 28,500 for training and 1,000 for testing.
- **MPID** [61]: It contains three rain types, namely, rain streak, raindrop and rain&mist. The training set has 2,400, 861, and 700 image pairs, while the test set has 200, 149, and 70 image pairs.
- **Auto100L/Auto800** [123]: These two datasets are generated by a GAN-based model based on Rain100L [136] and Rain800 [154]. By automatically adding rain streaks in unsupervised mode, the rain streaks have more shapes and directions that will be adapted to natural rain streaks than the original datasets that are manually created by Photoshop.
- **RainDrop** [84]: Based on using a camera with two pieces of glass, the authors consist of 1,119 pairs of raindrops images with various background scenes, where 861 images are employed for training.
- **RainDS** [85]: The first real-world deraining dataset, including different types of rain (rain streak and raindrop) captured in various lighting conditions and scenes. It contains 250 real-world and 1,200 synthetic rain image pairs.
- **QSMD-Data** [116]: This dataset follows [63] to prepare the training set that contains 20,800 pairs. The synthetic rain images are synthesized with rendered rain layers and the ground truth by using the screen blend mode.
- **Rain-II** [117]: 400 images are synthesized resembling [81] and the synthetic rain images possess apparent vapor, called Rain-II. This dataset is only used for testing.
- **Rain20** [121]: Resembling Wei et al. [122], 20 rain images with complexity and multiformity of the rain streaks are synthesized. These testing images contain two different scenarios, i.e., sparse rain streaks and dense rain streaks, and each scenario contains 10 test images.
- **GT-RAIN** [2]: This dataset includes 26,124 training frames, 3,300 validation frames and 2,100 testing frames. These frames cover a large variety of background scenes from the urban locations to natural scenery, spanning a wide range of geographic locations, including varying degrees of

---

**Figure 2:** Illustration of representative general datasets for SID, where we also show some samples of image pairs.
Figure 3: Illustration of representative specific datasets for SID, where we also show some samples of image pairs.

Figure 4: Examples of paired data include the rain image and the corresponding clear image (the first and second in top row), and some types of side information (see others).

illuminations from different times of day and rain of varying densities, streak lengths, shapes and sizes.

3.1.2. Specific data

Besides the corresponding image pairs, specific datasets also contain side information, e.g., rain streak, rain mask, image depth, rain density, transmission map, and atmospheric light. Some representative specific datasets include:

- **Rain100H** [136]: 1,900 clear images are selected from BSD [1]. The rain streaks are synthesized by photorealistic rendering techniques [28] or by adding simulated sharp line streaks along a certain direction. Rain100H has 1,800 image pairs for training with three to five layers of rain streaks and 100 image pairs for testing.
- **Rain100L** [136]: 300 clear images are selected from BSD [1]. Rain100L has 200 image pairs with only one type of rain streak for training and 100 image pairs for testing.
- **Rain200H/Rain200L** [136]: Rain200H/Rain200L are two new versions based on the original Rain100H/Rain100L datasets. The clean backgrounds for these two datasets are
Table 2
Mainstream paired datasets for the SID task were proposed from 2017 to 2022. * denotes the name of the dataset defined by us. ∙ denotes that the dataset contains real-world paired images.

| Dataset               | Train/Test | Type         | Side Information          | Public | Venue   | Year   |
|-----------------------|------------|--------------|---------------------------|--------|---------|--------|
| ‘GT-RAIN’ [2]         | 26,124/2,100 | General      | -                         | Yes    | ECCV    | 2022   |
| UBCN-Data* [65]       | 2,975/1,525  | Specific     | semantic label            | No     | AAAI    | 2022   |
| Cityscapes_syn [127]  | 2,975/1,525  | Specific     | semantic label            | Yes    | ACM MM  | 2022   |
| Cityscapes_real [127] | 2,975/500    | Specific     | semantic label            | Yes    | ACM MM  | 2022   |
| QSMD-Data* [116]      | 20,800/100   | General      | -                         | No     | CVPR    | 2021   |
| RainLevel5 [78]       | 26,870/2,880 | Specific     | rain speed/fog             | No     | CVPR    | 2021   |
| ‘RainDS’ [85]         | 1,015/435    | General      | -                         | Yes    | CVPR    | 2021   |
| RainDirection [73]    | 2,920/430    | Specific     | rain direction            | Yes    | ICCV    | 2021   |
| Auto100L [123]        | 200/100      | General      | -                         | Yes    | TIP     | 2021   |
| Auto800 [123]         | 700/100      | General      | -                         | Yes    | TIP     | 2021   |
| Rain-II [117]         | -/400        | General      | -                         | No     | ECCV    | 2020   |
| RainKITTI2012 [155]   | 4,062/4,085  | Specific     | semantic label/stereo view | Yes    | ECCV    | 2020   |
| RainKITTI2015 [155]   | 4,200/4,189  | Specific     | semantic label/stereo view | Yes    | ECCV    | 2020   |
| RH [35]               | 1,619        | General      | -                         | No     | TIP     | 2020   |
| MPID [61]             | 3,961/419    | General      | -                         | Yes    | CVPR    | 2019   |
| RaindropCityscapes*  [37] | 30,000/1,525 | Specific | rain mask                 | No     | ICCVW   | 2019   |
| NYU-Rain [60]         | 13,500/2,700 | Specific     | rain streak/ts. map/as. light | No | CVPR    | 2019   |
| Outdoor-Rain [60]     | 9,000/1,500  | Specific     | rain streak/ts. map/as. light | Yes | CVPR    | 2019   |
| Rain20* [121]         | -/20         | General      | -                         | Yes    | CVPR    | 2019   |
| RainCityscapes [42]   | 9,432/1,188  | Specific     | depth map                 | Yes    | CVPR    | 2019   |
| ‘SPA-Data’ [113]      | 28,500/1,000 | General      | -                         | Yes    | CVPR    | 2019   |
| OxfordRaindrop* [83]  | 4,318/500    | Specific     | road marking segmentation | Yes    | ICRA    | 2019   |
| DDC-Data* [63]        | 10,400       | Specific     | rain streak                | Yes    | CVIU    | 2019   |
| Rain1200* [153]       | 12,000/4,000 | Specific     | rain density               | Yes    | CVPR    | 2018   |
| RainDrop* [84]        | 861/249      | General      | -                         | Yes    | CVPR    | 2018   |
| Rain1400* [23]        | 9,100/4,900  | General      | -                         | Yes    | CVPR    | 2018   |
| Rain800* [154]        | 700/100      | General      | -                         | Yes    | TCSVT   | 2017   |
| Rain100H [136]        | 1,800/100    | Specific     | rain mask/rain streak      | Yes    | CVPR    | 2017   |
| Rain100L [136]        | 200/100      | Specific     | rain mask/rain streak      | Yes    | CVPR    | 2017   |
| Rain200H [136]        | 1,800/200    | Specific     | rain mask/rain streak      | Yes    | CVPR    | 2017   |
| Rain200L [136]        | 1,800/200    | Specific     | rain mask/rain streak      | Yes    | CVPR    | 2017   |
| Rain12* [66]          | -/12         | General      | -                         | Yes    | CVPR    | 2016   |

also selected from the BSD [1], with the training set and test set containing 1,800 and 200 images, respectively.

- **Rain1200** [153]: By using the Photoshop software [81], each image from 4,400 clear images is selected to synthesize three rain density levels (i.e., light, medium, and heavy) rain images. After that, 12,000 image pairs are used to train, while 1,200 image pairs are used for testing.

- **RainCityscapes** [42]: 295 clear images are selected from Cityscapes [13] to synthesize rain images by using the camera parameters and scene depth information. The training set has 262 images, while the test set has 33 images.

- **NYU-Rain** [60]: For this dataset, a total of 16,200 clear images are chosen from the NYU-Depthv2 [96] to render the synthetic rain streaks and rain accumulation effects based on the provided depth information. 13,500 image pairs are then used for training, with 2,700 for testing.

- **Outdoor-Rain** [60]: Clear images are chosen from Qian et al. [84] to synthesize the rain images by the same method as NYU-Rain. The outdoor-Rain dataset contains 9,000 training samples and 1,500 testing samples.

- **DDC-Data** [63]: The clean images are chosen from the BSD [1], and the rain streak layers are generated following Photoshop software [81] with varying intensities, orientations, and overlaps. Finally, a dataset containing 10,400 image pairs is fused in the way of screen blending.

- **RainDirection** [73]: The rain images in RainDirection are obtained by adding clean images from Flick2K and DIV2K datasets [99] with synthetic labeled rain maps according to the rain model Eqn.(1). Each rain image is assigned a direction label. The training and test set of RainDirection contains 2,920 and 430 images, respectively.

- **RainLevel5** [78]: The clean images are selected from the Cityscapes dataset [13] and 5 levels (25, 50, 75, 100, and 200 mm/hr) of rain images with corresponding fog rendering is synthesized. The dataset includes 26,870 pairs for training and 2,880 pairs for testing.

- **Cityscapes_syn** [127]: Cityscapes [13] are used as clean images and a physics-based rendering way [36] is used...
to define this dataset with two different rain speeds, i.e., 100 mm (light rain) and 200 mm (heavy rain). Each sub-dataset has the same image numbers as Cityscapes (i.e., 2,975, 500, and 1,525 for training, validation, and testing, respectively).

- **Cityscapes_real** [127]: Cityscapes [13] are used as clean images and real rain streaks from SPA-Data [113] are used to synthesize real rain streak dataset. The dataset includes 2,975 pairs for training and 500 pairs for testing.

- **UBCN-Data** [65]: Cityscapes [13] are used as clean images and four levels of rain (approximate 0, 50, 150, 250 mm/hr) are synthesized on Cityscapes dataset. The dataset includes 2,975 pairs for training, 1,525 pairs for testing, and 500 pairs for validation, respectively.

- **RainKITTI2012** [155]: Photoshop is used to create a synthetic RainKITTI2012 dataset based on the public KITTI stereo 2012 dataset [30]. The training set contains 4,062 image pairs from various scenarios, and the testing set contains 4,085 image pairs.

- **RainKITTI2015** [155]: Similar to the RainKITTI2012, KITTI2015 dataset is another set from KITTI stereo 2015 dataset [30]. The authors also synthesize a RainKITTI2015 dataset, whose training and testing set contain 4,200 and 4,189 pairs of images, respectively.

- **OxfordRaindrop** [83]: The authors present the hardware used to record their narrow-baseline stereo dataset that allows one lens to be affected by real water droplets while keeping the other lens clear. They have collected approximately 50,000 pairs of images by driving in and around the city of Oxford. They have selected 4,818 image pairs to form a training, validation, and testing dataset. From the testing partition, they have created ground truth road marking segmentations for 500 images.

- **RH** [35]: A dataset called RH of 1,619 images is composed of two parts, including the dataset captured by Qian et al. [84] and 500 clean/corrupted pairs of images captured by the authors. They use Nikon D5300 to capture various background scenes which include the raindrops and the haze. The thickness of the glass slabs is 3 mm. To minimize the reflective effect of the glass, the distance between the glass slabs and the camera lens has been set between 2 to 8 cm to generate diverse raindrop images.

- **RaindropCityscapes** [37]: The first photo-realistic adherent raindrop dataset with pixel-level mask in autonomous driving settings based on Cityscapes dataset [13]. For each background image, the authors generate 50 to 70 raindrops. Finally, they make a dataset containing about 30,000 images based on the training set of Cityscapes for training and 1,525 images based on the test set of Cityscapes for testing.

### 3.2. Rain Model

A rain model mainly indicates two issues: 1) how to model the rain mathematically, and 2) how to synthesize the rain image. However, only some rain models can be used to synthesize rain images, while others are only mathematical models of rain. According to whether it can generate data, we divide current rain models into **synthetical rain model** and **mathematical rain model**. Fig.5 shows an example of generating a rain image by a synthetic rain model.

#### 3.2.1. Synthetical rain model

Both the synthetical and mathematical rain models can model the rain, while only the synthetical rain model can be used to synthesize rain images. In what follows, we introduce some representative synthetical rain models in detail:

- **Rain Streak Model** (RSM). This is the most fundamental rain model, and the formulation is described as

  \[ O = B + S, \]  

  where \( O \) denotes a rain image that is decomposed into a rain streak part \( S \) and a clean background \( B \). RSM is widely used in data-driven SID methods [49, 103, 113, 149, 153, 154]. Datasets like Rain800, Rain1400, and Rain1200 are synthesized based on this model. Note that, from Eqn.(1) to Eqn.(16), we use uniform mathematical characters.

- **Screen Blend Model** (SBM). Different from the RSM, SBM is a non-linear composite model formulated as

  \[ O = B \odot S, \]  

  where \( \odot \) is element-wise multiplication. Unlike the additive composite character in Eqn.(1), the background and rain streak influence the appearance of each other. Li et al. [63] uses SBM as the rain model and synthesizes the DDC-Data.

- **Image Depth Model** (IDM). According to [29], the visual intensity of a rain streak depends on the scene depth \( d \) from the camera to the underlying scene objects behind the rain. The mathematical model is formulated as follows:

  \[ O = \sum_{i}^{n} \tilde{S}_{i} + (1 - T) \odot A, \]

  where \( \tilde{S} \) is the atmospheric layer, which is assumed to be a constant [92], \( F \) denotes the fog layer with a range of [0,1], \( I \) is a matrix of ones. Hu et al. [42] use the IDM as the rain model and synthesize the RainCityscapes dataset.

- **Heavy Rain Model** (HRM). Heavy rain often causes rain accumulation and visual effects with haze. Rain accumulation or veiling affects the result of water particles in the atmosphere and distant rain streaks that cannot be seen individually. The mathematical model can be expressed as

  \[ O = T \odot \left( B + \sum_{i}^{n} \tilde{S}_{i} \right) + (1 - T) \odot A. \]
where each $S_i$ is a layer of rain streaks with the same direction, $i$ is the index of the rain-streak layers, and $n$ is the maximum number of rain streak layers. $T$ denotes the transmission map introduced by the scattering process of the tiny water particles and $A$ is the global atmospheric light of the scene. Li et al. [60] uses HRM as the rain model and synthesizes the NYU-Rain and Outdoor-Rain datasets.

- **Rain Accumulation Model** (RAM). RAM considers raining the accumulation effect like Heavy Rain Model except with adding rain mask information. The mathematical model can be expressed as follows:

$$
O = T \odot \left( B + \sum_{i}^{n} S_i \odot M \right) + (1 - T) \odot A,
$$

(5)

where $M$ is a rain mask image which is a binary image, i.e., 1 for RAM, and 0 denotes rain-free in pixel. Yang et al. [136] uses RAM as a rain model and synthesizes the Rain100H and Rain100L datasets.

### 3.2.2. Mathematical rain model

Unlike the synthetical rain models, mathematical rain models are designed to model and solve the physical phenomenon of rain from a mathematical point of view. However, the authors do not (or could not) accurately synthesize the phenomenon of rain from a mathematical point of view. Howev-er, the authors do not (or could not) accurately synthesize models are designed to model and solve the physical phenomenon of rain from a mathematical point of view. However, the authors do not (or could not) accurately synthesize models are designed to model and solve the physical phenomenon of rain from a mathematical point of view. However, the authors do not (or could not) accurately synthesize models are designed to model and solve the physical phenomenon of rain from a mathematical point of view. However, the authors do not (or could not) accurately synthesize.

- **Rain Residual Model** (RRM). This model is similar to RSM. The advantage of using the rain residuals is to obtain a cleaner background than using the rain streaks for training, but this formula cannot be used to synthesize rain images:

$$
O = B - R,
$$

(6)

where $R$ indicates the residual of rain steam, which is a negative matrix. RRM is wildly used in [23, 25, 88, 152].

- **Base Detail Model** (BDM). Instead of decomposing the rain image as a rain streak layer and a background layer, BDM decomposes the rain image into the sum of a base layer and a detail layer by using a low-pass filter:

$$
O = O_{base} + O_{detail},
$$

(7)

where the detail layer contains the structure information, and the base layer is similar to a clear image. Note that BDM is used in the SID method [22].

- **Raindrop Mask Model** (RMM). This model formulates the raindrop-degraded image as a combination of a background image and the effect of raindrops:

$$
O = I + (1 - M) \odot B,
$$

(8)

where $M$ is the binary mask, and $M(x) = 1$ means the pixel $x$ is part of a rain region, and otherwise means it is part of background regions. $I$ is the effect brought by the raindrops, representing the complex mixture of the background information and the light reflected by the environment and passing through the raindrops adhered to a lens or windscreen. SID method [84] uses RMM as the rain model.

- **Raindrop Transparency Model** (RTM). RTM considers a simple linear model with transparency:

$$
O = A \odot R + (1 - A) \odot B,
$$

(9)

where $A \in [0, 1]^{C \times M \times N}$ denotes the transparency matrix. Each entry of $A$ represents the percentage of the light path covered by raindrops for the corresponding pixel. The study in [86] has adopted RTM as the rain model.

- **Rain Convolutional Dictionary** (RCD). RCD considers the problem under the conventional prior-based methodology by exploiting the prior knowledge for representing the rain streaks, which is formulated as follows:

$$
O = B + \sum_{n=1}^{N} C_n \odot P_n,
$$

(10)

where $\sum_{n=1}^{N} C_n \odot P_n = S^c (c = 1, 2, 3)$. $S^c$ denotes the color channel of rain streaks, and $\{C_n\}_{n \in \mathbb{N}} \subseteq \mathbb{R}^{k \times k}$ is a set of rain kernels which describes the repetitive local patterns of rain streaks, and $\{P_n\}_{n \in \mathbb{N}} \subseteq \mathbb{R}^{h \times w}$ represents the corresponding rain maps representing the locations where local patterns repeatedly appear. $N$ is the number of kernels and $\odot$ is the 2-dimensional (2D) convolutional operation. RCD-Net [110] uses the RCD as the rain model.

- **Two Transmissions Model** (TTM). Because the rain streaks and vapors are entangled with each other, this mathematical model entangles the rain streaks and vapors properly from the transmission medium perspective:

$$
O = (T_s + T_v) \odot B + \left[1 - (T_s + T_v)\right] \odot A,
$$

(11)

where $T_s$ and $T_v$ are the transmission maps of rain streaks and vapors, respectively. TTM is adopted in the SID method [117] as the rain model.

- **Streak Drop Model** (SDM). In real-world rain weather scenarios, rain streaks and raindrops may co-occur in outdoor image capture. As such, this rain model considers both the rain streak and raindrop in one formula:

$$
O = aI + (1 - M) \odot (B + S),
$$

(12)

where $a$ is the global atmospheric lighting coefficient. SDM is adopted in the SID method [85] as the rain model.

- **Haze-Like Model** (HLM). The authors handle the hazy-like rain effect by modeling its influence directly using a new variable $T$ in their general model for rain images:

$$
O = T \odot B + S,
$$

(13)

where $T$ denotes the degradation caused by haze-like effect and $\odot$ denotes the pixel-wise multiplication, respectively. HLM is adopted in the SID method [114] as the rain model.

- **Rain Disentangled Model** (RDM). This model solves the deraining problem by distinguishing the special characteristics of $B$ and $S$ to alleviate the over-smoothness and artifacts in the final derained results:

$$
O = \alpha \odot B + \beta \odot S,
$$

(14)
where $\alpha$ and $\beta$ denote two sets of learnable parameters for $B$ and $S$ respectively. $\alpha$ represents the weighting factors of atmospheric reflected light and transparency in different luma and chroma channels under rain conditions, and $\beta$ represents a kind of mixed characteristic of the rain streaks (i.e., the region and amplitude of rain streaks). RDM is adopted in the SID method [51] as the rain model.

- **Uncertainty Raindrop Model (URM).** This model is based on the Raindrop Mask Model (Eqn.(8)) and considers the blur level of raindrops:

$$O = U \odot I + (1 - |U|) \odot B,$$

where $U \in [-1, 1]$ is a blurry confidence map with a value between $-1$ and $1$, which the authors call the uncertainty map. The larger the value of $U(x)$, the more blurred it is at the pixel $x$, which means that $x$ should be regarded as a raindrop and the raindrop effect $U \odot I$ will increase accordingly. The plus or minus sign of $U$ determines whether a raindrop reduces or increases image brightness. URM is adopted in the SID method [95] as the rain model.

- **Motion Blur Model (MBM).** Because of the high velocity of the raindrops, Wang et al. model the generation of rain streaks as the following motion blur process:

$$O = B + K(\theta, l) \odot M,$$

where $\theta$ and $l$ are the angle and length of the motion blur kernel $K(\theta, l)$, respectively. $M \in \mathbb{R}^{m \times n \times 3}$ is the raindrops mask, and $\odot$ denotes the spatial convolution operator. The two important factors, i.e., the length $l$ and the angle $\theta$, essentially encode the motion blur kernel and can be easily inferred from the rainy images by exploiting the repeatability of the rain streaks. MBM is adopted in the SID method [118] as the rain model.

- **Rain&Haze Model (RHM).** This model is similar to the Heavy Rain Model (Eqn.(4)), and considers joint raindrop and haze removal:

$$O = T \odot (B + (1 - M) \odot S) + (1 - T) \odot A,$$

where $A$ indicates the global atmospheric light, and $T$ denotes the transmission map. $I$ is an unit matrix (all-ones matrix), $(1 - M)$ indicates the locations of individually visible raindrops. Here, elements in $M$ are binary values, where $0$ indicates raindrop regions and $1$ indicates non-raindrop regions. RHM is used in the method [35] as rain model.

### 3.3. Network Architecture

The network architecture is always the most critical and concerning factor for SID. By designing new modules, researchers can construct a pipeline to solve the proposed rain model. According to whether the network derives all the rain model information, we divide the current SID network architectures into **white-box network architecture** and **black-box network architecture**. Fig.6 shows examples of two network architectures to learn information in different ways.
where $\vartheta$ represents the function of neural networks, $E$ denotes the rain feature maps extracted by previous CNN, $\hat{M}$ denotes the rain mask predicted from the convolutional process of $E$, and $\hat{S}$ denotes the rain streak predicted from the convolutional process based on the concatenation $[E, \hat{M}]$. In Eqn.(19), only $\hat{M}, \hat{S}$, and $\hat{B}$ are derived in the supervised training process, while the transmission map $T$ and atmospheric light $A$ in Eqn.(5) were not derived. Note that the existing SID methods [42, 64, 84, 88, 136] have adopted the black-box network architecture.

4. In-depth Analysis of Data-driven SID

We analyze the intrinsic relationships of the three influencing factors and then discuss the solving paradigms of existing SID methods based on the preceding studies.

![Diagram](image)

**Training goal:** $f(g(B, I); \vartheta) = \hat{B}, \hat{I}, \quad \text{s.t. } \operatorname{arg min} \mathcal{L}(B, \hat{B})$

**Figure 7:** The relationship among the three factors (i.e., data, rain model and network architecture) and optimizer (i.e., loss function) in the SID task.

4.1. Relationship of The Three Factors

To facilitate the analysis, we denote the set of data, rain model, and network architecture as $\mathcal{H}, \mathcal{G}$, and $\mathcal{F}$, respectively. The image pairs and side information are respectively represented as $[O, B, I_i] \in \mathcal{H}$, where $O$ and $B$ represent the rain image and clear ground-truth image, $I_i (i = 1, ..., n)$ denotes side information. Then, we can mathematically formulate the process of rainmaking and deraining as follows:

$$\begin{align*}
O &= g(B, I), \\
\hat{B}, \hat{I} &= f(O; \vartheta),
\end{align*}$$

(20)

where $g(\cdot) \in \mathcal{G}$ is a rain model for rainmaking or rain generation, by which the rain image $O$ is generated, and $f(\cdot) \in \mathcal{F}$ is a network architecture for deraining, by which the derived image $\hat{B}$ and side information $\hat{I}$ are predicted. $\vartheta$ are the learnable parameters of network. From Eqn.(20), we can see that the rain model $g(\cdot)$ is to make or generate rain, while the network architecture $f(\cdot)$ is an inverse function to remove rain. The three factors and loss function can be taken into account at the same time and the deep learning-based rain removal mechanism can be described as

$$f(g(B, I); \vartheta) = \hat{B}, \hat{I}, \quad \text{s.t. } \operatorname{arg min} \mathcal{L}(B, \hat{B}),$$

(21)

where $s.t.$ denotes the condition that the previous equation needs to satisfy. $\mathcal{L}(\cdot) \in \mathcal{L}$ denotes loss function. $\operatorname{arg min} \mathcal{L}(B, \hat{B})$ denotes the value of $\vartheta$ to make $\mathcal{L}(B, \hat{B})$ get the minimum value. Note that the relationship of the three factors is illustrated in Fig.7, from which we see that data is the core factor, while the rain model and network architecture are closely associated with data. Specifically, the rain model defined by some prior assumptions can be used to generate data. Then, the network architecture uses the data to solve the rain model by maximum posterior probability reversely. That is, rainmaking and deraining are inverse processes in the SID task. Since data play a core role in the three factors, we mainly study the effectiveness of SID data in Section 6.

4.2. Solving Paradigms of SID

Due to the complexity of the optical characteristics of rain [29], the uncertainty of prior knowledge and the flexibility of deep learning solution, the relationship of the three factors does not always follow Eqn.(21). The relationships between the three factors are often asymmetrical or inconsistent. The complexity of the relationships leads to different solving paradigms of SID methods. According to whether the solving paradigm follows Eqn.(21), we divide

| Paradigm | Method | Data | Model | Network |
|----------|--------|------|-------|---------|
| Li et al. [60] | Spe. | Syn. | W |
| DID-MDN [153] | Spe. | Syn. | W |
| RESCAN [64] | Gen. | Syn. | W |
| Wang et al. [117] | Gen. | Math. | W |
| MSPFN [49] | Gen. | Syn. | W |
| JDNNet [103] | Gen. | Syn. | W |
| GraNet [149] | Gen. | Syn. | W |
| Fu et al. [23] | Gen. | Math. | W |
| MPRNet [152] | Gen. | Math. | W |
| RCDNet [110] | Gen. | Math. | W |
| ReMAEN [140] | Gen. | Syn. | W |
| MOSS [46] | Gen. | Syn. | W |
| DCSFN [104] | Gen. | Syn. | W |
| DRD-Net [17] | Gen. | Syn. | W |
| QSM [116] | Gen. | Syn. | W |
| ReHEN [141] | Gen. | Syn. | W |
| VIRGNet [111] | Gen. | Syn. | W |
| RLNet [10] | Gen. | Syn. | W |
| UMR-L [143] | Gen. | Syn. | W |
| SSDRNNet [67] | Gen. | Syn. | W |
| DTDN [120] | Gen. | Math. | W |
| UMAN [95] | Spe. | Math. | W |
| KGCN [118] | Spe. | Math. | W |
| AI-GAN [51] | Spe. | Math. | W |
| AMPE-Net [114] | Spe. | Math. | W |
| Liu et al. [73] | Spe. | Syn. | W |
| Hao et al. [37] | Spe. | Math. | W |
the current SID methods into explicit and implicit solving paradigms. Table 3 and 4 summarizes the three factors and the corresponding solving paradigms of partial data-driven SID methods based on our proposed division criterion.

4.2.2. Implicit solving paradigm

We classify the solving paradigm as "explicit" only based on one judgment, i.e., the relationship of the three factors in the method is consistent with Eqn.(21). As described in Table 3, we see that existing SID methods with white-box network architecture are ascribed to the explicit solving paradigm. For example, DID-MDN [153] uses the RSM to synthesize the Rain1200 dataset, then uses a network architecture to solve it in white-box mode. Similarly, the method [60] builds the NYU-Rain and Outdoor-Rain data, while using Eqn.(4) to solve RAM.

4.2.2. Implicit solving paradigm

We classify the solving paradigm of a method as "implicit" based on two judgments: 1) the network architecture is not consistent with the rain model. For example, the methods [136] use RAM (Eqn.(5)) and neglect to predict the transmission maps and atmospheric light in their network architecture, while [42, 84, 88, 113, 154] cannot predict rain streak information; 2) the training data are not consistent with the rain models. For example, the method [25] adopt the RRM (Eqn.(6)) as its rain model, which only contains the clean image and rain residual layer, but also uses Rain100H and Rain100L as the training data. As shown in Table 4, existing SID methods with black-box network architecture are ascribed to the implicit solving paradigm.

4.3. Division of Recent SID Methods

Based on the above analysis, we can categorize some current data-driven SID methods from the six aspects of the three factors (i.e., general vs. specific, synthetical vs. mathematical, black-box vs. white-box). Fig.8 summarizes the division of partial data-driven methods based on the three factors. From the classification, we can conclude that:

1) Current methods usually use general data (i.e., Rain800, Rain1400, SPA-Data) and synthetic rain model (i.e., RSM), which is the most basic and simplest way to solve SID task. The main difference between those methods is the design of domain knowledge, which will be discussed in Section 5;

2) Among the reviewed 65 methods in Fig.8, there are 37 black-box and 28 white-box SID methods. More researchers tend to solve the rain model inaccurately, thanks to the amazing fitting ability of neural networks;

3) More and more methods use specific data (i.e., RainCitiscapes, NYU-Rain, DDC-Data) and mathematical rain model (i.e., RCD, TTM, SDM) to solve SID problem [42, 60, 85, 110, 117, 135], which results in better performance;

5. Further Analysis on SID Methods

In addition to the analysis of the three factors, SID methods still have some other aspects that can be further ex-
Table 5
The representative single image deraining methods proposed from 2020 to 2022.

| Method      | Training Strategy | Pipeline | Domain Knowledge | Data Processing | Objective Function | Code | Venue   | Year |
|-------------|-------------------|----------|------------------|-----------------|--------------------|------|---------|------|
| Ba et al. [2] | S              | Sequential | DK               | Random           | MAE/SSIM/Contra.   | ✓    | ECCV    | 2022 |
| DANet [48]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | IJCAI   | 2022 |
| SSID-KD [14] | S-S             | Hybrid    | PK/DK             | Random           | SSIM/KL/DC/TV      | ✓    | TCSVT   | 2022 |
| TransWeather [100] | S         | Hybrid    | PK/DK             | -                | MAE/MAE            | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Parallel  | DK               | Random           | MAE/Contrastive     | ✓    | CVPR    | 2022 |
| AirNet [57]  | S              | Hybrid    | DK               | -                | Adv                | ✓    | CVPR    | 2022 |
| Yu et al. [150] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
| AirNet [57]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
| AirNet [57]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
| AirNet [57]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
| AirNet [57]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
| AirNet [57]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
| AirNet [57]  | S              | Hybrid    | PK/DK             | Random           | Charbonnier/SSIM   | ✓    | CVPR    | 2022 |
| Zou et al. [164] | S            | Sequential | PK/DK           | -                | Adv                | ✓    | CVPR    | 2022 |
| FPNet [34]   | S              | Sequential | PK/DK           | Fixed            | MAE                | ✓    | IJCAI   | 2022 |
| Wang et al. [102] | S          | Hybrid    | DK/OK             | Random           | MAE/SSIM/KL        | ×    | AAAI    | 2022 |
| NCANet [98]  | S              | Hybrid    | DK               | -                | MSE                 | ×    | TPAMI   | 2022 |
Table 6
The representative single image deraining methods proposed from 2017 to 2019.

| Method            | Training Strategy | Pipeline | Domain Knowledge | Data Processing | Objective Function | Code | Venue | Year |
|-------------------|-------------------|----------|------------------|-----------------|--------------------|------|-------|------|
| DTDN [120]        | S                 | Parallel | PK/DK/OK         | Fixed           | MSE/Adv            | ✓    | ACM MM| 2019 |
| Liu *et al.* [72] | S                 | Sequential | DK              |                 | MAE/SSIM           | ✓    | CVPR  | 2019 |
| ERL-Net [105]     | S                 | Hybrid   | DK/OK            | Random          | MAE                | ×    | ICCV  | 2019 |
| GraNet [149]      | S                 | Hybrid   | DK/OK            | Fixed           | MAE                | ×    | ACM MM| 2019 |
| GCANet [11]       | S                 | Sequential | DK              |                 | MSE                | ✓    | WACV  | 2019 |
| LPNet [24]        | S                 | Hybrid   | PK/DK            | Random          | MAE/SSIM           | ✓    | TNNS  | 2019 |
| PreNet [88]       | S                 | Sequential | DK              |                 | SSIM               | ✓    | CVPR  | 2019 |
| DAF-Net [42]      | S                 | Hybrid   | DK/OK            | Fixed           | MSE                | ✓    | CVPR  | 2019 |
| ReHEN [141]       | S                 | Sequential | DK/OK           | Random          | MSE/SSIM           | ✓    | ACM MM| 2019 |
| DDC-Net [63]      | S                 | Hybrid   | DK/OK            | Fixed           | MSE/Adv            | ✓    | CVIU  | 2019 |
| JORDER-E [135]    | S                 | Hybrid   | PK/DK/OK         | Random          | MAE/SSIM/CE        | ×    | ICCW  | 2019 |
| Hao *et al.* [37] | S                 | Sequential | DK              |                 | MAE/Adv            | ✓    | ICRA  | 2019 |
| RR-GAN [158]      | U-S               | Parallel | DK/OK            | Random          | MSE/Adv            | ✓    | AAAI  | 2019 |
| MH-DerainNet [126]| S                 | Sequential | DK/OK          | Random          | MSE/SSIM           | ×    | ICDM  | 2019 |
| SIRR [121]        | S-S               | Parallel | PK/DK/OK         | Random          | MSE/KL/GMM/TV      | ✓    | CVPR  | 2019 |
| SPANet [113]      | S                 | Sequential | DK              | Random          | MSE/MAE/SSIM       | ✓    | CVPR  | 2019 |
| ReMAEN [140]      | S                 | Sequential | DK              | Random          | MSE                | ✓    | ICME  | 2019 |
| UD-GAN [53]       | U-S               | Parallel | PK/DK/OK         | Fixed           | MAE/Adv            | ×    | ICIW  | 2019 |
| UMRIL [143]       | S                 | Hybrid   | DK/OK            | Fixed           | MAE                | ✓    | CVPR  | 2019 |
| ID-CGAN [154]     | S                 | Parallel | DK              | Fixed           | MSE/Adv            | ×    | TCSVT | 2019 |
| Li *et al.* [60]  | S                 | Hybrid   | PK/DK/OK         | Fixed           | MSE/MAE/Adv        | ✓    | CVPR  | 2019 |
| Quan *et al.* [86]| S                 | Hybrid   | DK/OK            | Random          | MAE                | ✓    | ICCV  | 2019 |
| RWL [134]         | S                 | Hybrid   | PK/DK            | -               | MSE                | ×    | TIP   | 2019 |
| DualCNN [80]      | S                 | Parallel | DK              | Random          | MSE                | ✓    | CVPR  | 2018 |
| NLEDN [59]        | S                 | Sequential | DK              | Fixed           | MAE                | ✓    | ACM MM| 2018 |
| RESCAN [64]       | S                 | Sequential | DK              | Random          | MSE                | ✓    | ECCV  | 2018 |
| ResGuideNet [21]  | S                 | Sequential | DK              | Random          | MSE/SSIM           | ×    | ACM MM| 2018 |
| DDC-GAN [52]      | S                 | Parallel | DK              | Random          | MSE/Adv            | ×    | BMVC  | 2018 |
| Qian *et al.* [84] | S                 | Parallel | DK              | Fixed           | MSE/Adv            | ✓    | CVPR  | 2018 |
| DID-MDN [153]     | S                 | Hybrid   | DK/OK            | Fixed           | MSE/CE             | ✓    | CVPR  | 2018 |
| DerainNet [22]    | S                 | Hybrid   | PK/DK            | Random          | MSE/CE             | ✓    | TIP   | 2017 |
| JORDER [136]      | S                 | Hybrid   | DK/OK            | Random          | MSE/CE             | ×    | CVPR  | 2017 |
| Fu *et al.* [23]  | S                 | Sequential | PK              | Random          | MSE                | ×    | CVPR  | 2017 |

Table 7
Statistics of the training strategies in current SID methods.

| Training Strategy | Supervised | Semi-sup. | Un-super. |
|-------------------|------------|-----------|-----------|
| Count             | 86         | 8         | 3         |

5.1. Training Strategy
To improve the generalization ability of the SID methods to real-world scenarios, some methods also try to use real rain images as training data. According to whether unlabeled or unpaired data are involved during training, the training strategies can be divided into three learning modes, i.e., supervised, unsupervised and semi-supervised.

- **Supervised.** In supervised mode, the network is optimized by calculating the loss between the predicted image and the labeled ground truth image. For example, JORDER [136] can predict the rain mask, rain streak, and background information based on a fully-supervised training strategy.

- **Unsupervised.** There are no ground truth images in unsupervised mode, and the deraining network is usually optimized by using prior knowledge or self-supervision. For example, RR-GAN [158] uses consistency regularization to learn the rain distribution without using label information.

- **Semi-supervised.** By using both paired and unpaired images, the semi-supervised model can be optimized in the case of transfer learning, domain adaption, and sharing of network parameters. For example, SIRR [121] shares a CNN to learn both the synthetic and real-world rain distribution by regularizing the KL distance.

**Remarks.** We count the training strategies of the methods in Table 5 and 6, and listed the statistical results in Table 7, from which we can conclude that:

1) There are only 8 semi-supervised and 3 unsupervised...
SID methods, and 86 supervised SID methods. The main reason is that the SID problem is ill-posed and is difficult to solve via weak constraints in semi/un-supervised mode;

2) Most current methods use the supervised mode. A supervised method can significantly perform on test data with the same distribution by involving paired data. However, these pre-trained methods often failed in the case of real-world rain images due to the poor generalization ability;

3) Among those 11 semi-/un-supervised SID methods, 4 semi-supervised and 3 unsupervised methods are based on GAN [33, 160], since GAN can provide better image generation and restoration in weakly supervised mode. Besides, 8 semi-/un-supervised SID methods use the parallel pipeline to construct steady network architecture. Note that the parallel pipeline will be introduced below.

5.2. Network Pipeline
   Due to different training strategies and domain knowledge, data-driven SID methods have different network pipelines. According to the flow of data in the network, we can divide the network pipeline into three structures, i.e., sequential, parallel and hybrid, as illustrated in Fig. 9.

   - **Sequential.** This is the most basic and common network pipeline, where data flows linearly. For example, Fu et al. [23] adopts a linear way to build an end-to-end network. An architecture with multi-streams handling the same task can also be considered a sequential mode. For example, MH-DerainNet [126] uses different kernel sizes to get features with different receptive fields.

   - **Parallel.** Instead of simply using a single neural network, the parallel pipeline can handle equally important tasks. For example, DerainCycleGAN [123] uses parallel networks to process both rain and clean ground-truth images. An architecture with multi streams handling the different tasks can also be considered as parallel mode. For example, DerainNet [22] uses two streams to dispose of the detail and base layers, respectively.

   - **Hybrid.** A hybrid pipeline would have both sequential and parallel architectures inside. Since the modules can be assembled in a more complex way, in case that more side information needs to be processed simultaneously. For example, DAF-Net [42] uses a hybrid network to handle the tasks of learning depth map and image features at the same time; JORDER [136] firstly employs the sequential multi-streams to extract feature maps with different receptive fields, and then uses three parallel convolutions to learn the rain mask, rain streak, and background, respectively.

   **Remarks.** We count the network pipelines of the methods in Table 5 and 6, and listed the statistical results in Table 8, from which we can conclude that:

1) There are 32 sequential, 21 parallel and 44 hybrid pipelines used in current methods. The hybrid structure is more popular than the sequential and parallel structures;

2) More SID methods choose the sequential pipeline from 2017 to 2018 while choosing the parallel and hybrid pipelines to consider more side information from 2019 and later. Note that SID methods needing side information (i.e., rain density, image depth) prefer parallel and hybrid pipelines. That is because additional networks need to be designed for learning the side information;

3) From 2020 to 2022, 30 of 63 existing methods used the hybrid pipelines, compared to 14 with the hybrid pipeline in 34 SID methods proposed during 2017 and 2019. More and more hybrid pipelines are used, which means researchers begin to design more complicated architectures;

4) The pipeline of GAN-based deraining methods, such as RR-GAN [158] and Qian et al. [84], can be regarded as parallel, since the generator and discriminator are generally two independent networks.

5.3. Domain Knowledge
   Due to the complexity of the SID problem, people explored different domain knowledge to extract feature information and obtained better deraining performance. Generally, there are two main types of domain knowledge. One is called prior knowledge based on the digital signal/image processing and probability statistics, e.g., guided filter [38], mutual information [56] and Kullback-Leibler divergence [40]. The other one is deep-learning knowledge based on deep feature extraction, e.g., GAN [33], LSTM [18], high-level perceptual feature [54] and U-net [90]. Besides, some
other domain knowledge, e.g., parameter initialization and learning rate strategy, are used as knowledge.

- **Prior Knowledge (PK).** This knowledge is based on digital signal/image processing and probability statistics. For the SID task, researchers used a guild filter to decompose the original rain image into two parts: a high-frequency component (or detail layer) and a low-frequency component (or base layer). The former contains more rain information, while the latter contains more background information. For example, Fu et al. [23] decomposes the rain image into a base layer and a detail layer, while Yang et al. [134] decompose the rain image into high-frequency and low-frequency components. These decomposition operations are beneficial to the process of deraining, while preserving accurate background. Another example is the log histograms of filtered natural images below the straight line connecting the minimal and maximal values and naming this property as sparsity prior. Therefore, sparsity has great potential to separate rain and non-rain textures during deraining. For example, Wang et al. [116] develop a quasi-sparse distribution to approximate the sparsity to obtain a feasible loss function.

- **Deep-learning Knowledge (DK).** It uses deep neural networks to extract feature information. For example, many methods use perceptual features to preserve the image contents. VGG-16/VGG-19 [97] pre-trained on ImageNet [15] can obtain higher-level features. The perceptual loss [54] measuring the difference of high-level features has delivered better visual performance than the per-pixel loss used in traditional methods [23, 136]. In addition, by utilizing unique modules, SID methods can obtain more representative features. Some of these modules are universal modules, such as dilated convolution [148] used in methods [25, 136], LSTM [18] used in methods [84, 88], U-net [90] used in methods [63, 84], and ShuffleNet [156] used in QSMID [116]. Others are based on attention mechanisms and specifically designed, such as channel attention used in methods [17, 64], spatial attention used in SPANet [113], and confidence map network used in UMRL [143]. Recently, transformer [101] designs a self-attention mechanism to capture global interactions between contexts and has shown promising performance in SID methods [119, 129, 151].

- **Other Knowledge (OK).** In addition to the above two kinds of domain knowledge, there are still many operations based on the understanding of SID. For example, parameter initialization makes the initial network converge quickly and can avoid falling into the local saddle point or gradient cliff initially. A common approach is to use the Gaussian distributions to initialize the network parameters, which are superior to the random initialization. Another approach is to use the Xavier [31] method to initialize the network parameter, such as ResGuideNet [21]. Some networks, such as the VGG network pre-trained on ImageNet, can initialize the parameters of networks with the same structures, such as DAF-Net [42]. In addition, some other SID methods with complicated network architectures often train in different phases. For example, Li et al. [60] trains the network in the physics-based stage and model-free stage, respectively; Wang et al. [117] firstly pre-trains the ANet and SNet, and jointly trains the whole network at last. Learning rate is also considered to obtain better deraining performance. For example, Quan et al. [85] set the learning rate to 0.001 and adopt a cosine scheduler for training. CNN-based restoration models are usually trained on fixed-size image patches. However, training a Transformer model on small cropped patches may not encode the global image statistics, thereby providing suboptimal performance on full-resolution images at test time. For example, Zamir et al. [151] perform progressive learning where the network is trained on smaller image patches in the early epochs and on gradually larger patches in the later training epochs.

**Remarks.** We count the domain knowledge of the methods in Table 5 and 6. Besides, we also summarize the used representative deep-learning knowledge in Table 10:

1) Deep-learning knowledge is widely used in most current data-driven SID methods due to the strong representation ability. In addition, prior knowledge is only used in 39 methods, which means we can still explore more effective prior knowledge to help data-driven SID task;

2) Other kinds of knowledge are also important for the data-driven SID task since almost every method uses these tricks to obtain more stable and better performance;

3) Among the deep-learning knowledge, GAN [33], attention [101], encoder-decoder [3], Perceptual [54], Dilated Convolution [148] and multi-stage are used more frequently;

4) LSTM [18], Multi-scale, Two-branch, Multi-stream, Multi-task, Squeeze-and-excitation Network [41] and ShuffleNet [156] are often adopted in data-driven SID methods;

5) Neural Architecture Search [162] and Graph Convolutional Network [55] are used in CCN [85] and Fu et al. [25] in 2021, respectively. We suggest the researchers pay more attention to these new technologies, which have not
5.4. Data Preprocessing

Data preprocessing is also worthy of investigation for SID. Most methods read image data in PNG or JPG format, and then convert it to a tensor form. We divide the data preprocessing methods into fixed and random sampling according to the pattern of image clipping and augmentation.

- **Fixed Sampling (large resolution).** This sampling method aims at cropping a fixed-size image from the middle or random position of the input. For example, Zhang et al. [153] randomly crop a 512 × 512 image from the input (or its horizontal flip) of size 586 × 586. Due to the large proportion of cropping, data augmentation is rarely used. Fixed sampling can obtain an image with a larger receptive field and is suitable for the method with downsampling.

- **Random Sampling (small resolution).** Different from the fixed sampling, images of small size (namely, patches) are randomly or regularly cropped from the original image. For example, Fu et al. [23] randomly selected 9,100 images, generating three million 64 × 64 rain/clean patch pairs. Random sampling has two advantages: 1) it can augment data and therefore reduce the underfitting phenomenon caused by insufficient data; 2) it can greatly reduce GPU consumption and speed up the training process. A larger batch size can be set in training to reduce the over-fitting during the optimization process.

**Remarks.** We count the data preprocessing of the methods in Table 5 and 6, and listed the statistical results in Table 11, from which we can conclude that:

1) Most SID methods (a total of 56) chose random sampling, while only 27 methods used fixed sampling, from which we can infer that both data augmentation and large batch size are effective for the SID task;

2) Among those 11 SID methods [42, 60, 63, 78, 80, 84, 134, 135, 136, 153, 159] using side information, 5 have used the fixed sampling, and 5 have used the random sampling. We speculate that the correlation between data preprocessing and side information is not strong.

5.5. Objective Function

The objective function is also an important part of data-driven SID methods. The parameters of neural networks can be learned via the objective function in the training process. According to whether the constraint contains ground truth, current objective functions can be generally divided into Fidelity-Driven Metrics (with ground truth) and Probability&Statistics Models (without ground truth).

- **Fidelity-Driven Metrics.** Signal fidelity-driven metrics include Mean Absolute Error (MAE, L1), Mean Squared Error (MSE, L2), Charbonnier loss, and Structural Similarity (SSIM), which are technically defined as Eqn.(22) (23) (24) and (25), respectively.

\[
\mathcal{L}_{\text{MAE}}(x, \hat{x}) = \frac{1}{N} \sum_{i=1}^{N} |x_i - \hat{x}_i|, \quad (22)
\]

\[
\mathcal{L}_{\text{MSE}}(x, \hat{x}) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x}_i)^2, \quad (23)
\]

\[
\mathcal{L}_{\text{Charbonnier}}(x, \hat{x}) = \sqrt{\mathcal{L}_{\text{MSE}}(x, \hat{x}) + \epsilon^2}, \quad (24)
\]

\[
\mathcal{L}_{\text{SSIM}}(x, \hat{x}) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (25)
\]

where \(x\) and \(\hat{x}\) are ground-truth and derained images respectively in Eqn.(22) (23) (24) and (25). In Eqn.(24), \(\epsilon\) is a tiny constant, and this charbonnier loss is used in SID methods [48, 119]. In Eqn.(25), \(\mu_x\) and \(\mu_y\) denote the averages of \(x\) and \(\hat{x}\), respectively. \(\sigma(x)\) and \(\sigma(\hat{x})\) denote the variance of \(x\) and \(\hat{x}\), respectively. \(c_1\) and \(c_2\) are two numbers used to stabilize the division with a weak denominator.

- **Probability&Statistics Model.** Many objective functions are based on informatics, probability theory, and mathematical statistics, e.g., Adversarial loss (Adv), Cross Entropy loss (CE), Kullback-Leibler divergence (K-L), Gaussian Mixture Model (GMM), Gaussian Process (GP), Quasi-
Sparsity loss (QS), Autocorrelation loss (AC), Total Variation (TV), Triplet loss, Margin loss, Contrastive loss, Knowledge Distillation loss, and Dark Channel loss (DC). These objective functions are defined from Eqn.(26) to Eqn.(38):

\[ \min_{G} \max_{D} V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_{z}(z)} [\log (1 - D(G(z)))], \]  

(26)

\[ \mathcal{L}_{CE} = \frac{1}{N} \sum_{i} \left[ y_{i} \cdot \log (p_{i}) + (1 - y_{i}) \cdot \log (1 - p_{i}) \right], \]

\[ \mathcal{L}_{HOG} = \frac{1}{K} \sum_{k} \text{KL} \left( \bar{O}_{k}^{g}, \bar{O}_{k}^{d} \right), \]  

(27)

\[ D_{KL}(p \parallel q) = \sum_{i} p(x) \cdot \log \frac{p(x)}{q(x)}, \]  

(28)

\[ \mathcal{R} = \prod_{k} \pi_{k} N \left( \mathcal{R} \mid \mu_{k}, \Sigma_{k} \right), \]  

(29)

\[ m(v) = \mathbb{E} [f(v)], \]

(30)

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

\[ P_{q}(\bar{I}) = \prod_{i,k} P_{q} \left( \omega_{i,k} \ast \bar{I} \right), \]  

(31)

\[ \mathcal{L}_{ac} = \sum_{x,y} \sum_{t=1}^{P} \frac{\text{ac}_{xy} \left( \text{max}_{t} \right) - \text{mean}_{ac_{xy}}}{P}, \]  

(32)

\[ \mathcal{R}_{V, \beta}(x) = \sum_{i,j} \left( (x_{i,j+1} - x_{i,j})^{2} + (x_{i+1,j} - x_{i,j})^{2} \right)^{\frac{r}{2}}, \]  

(33)

\[ \mathcal{L}_{triplet} = \max \left( d(a, p) - d(a, n) + \text{margin}, 0 \right), \]  

(34)

\[ \mathcal{L}_{\text{margin}} = T_{k} \max \left( 0, m^{+} - \|v_{k}\|^{2} \right)^{2} + (1 - T_{k}) \max \left( 0, \|v_{k}\|^{2} - m^{-} \right)^{2}, \]  

(35)

\[ \mathcal{L}_{\text{contrastive}} = -\log \frac{\exp \left( q \cdot k_{i}^{+} / \tau \right)}{\sum_{i} \exp \left( q \cdot k_{i}^{-} / \tau \right)}, \]  

(36)

\[ \mathcal{L}_{\text{distillation}} = \mathbb{E}_{(x, y) \in P_{xy}} \mathcal{L}(\mathcal{T}(x), S(x)), \]  

(37)

\[ \mathcal{L}_{DC} = \| 1 - \mathcal{N}^{r}_{\text{max}} - 0 \|_{1} + \| 1 - \mathcal{N}^{r}_{\text{max}} - 0 \|_{1}. \]  

(38)

Table 12

| Objective Function | MSE | MAE | SSIM | Adv | KL | CE |
|--------------------|-----|-----|------|-----|----|----|
| Count              | 58  | 49  | 30   | 23  | 7  | 7  |
|                     | GMM | GP  | QS   | AC  | TV | Tri|
|                    | 1   | 1   | 1    | 1   | 4  | 1  |
|                    | Con | Cha | DC   | KD  | Mar|    |
|                    | 2   | 2   | 1    | 1   | 1  |    |

Eqn.(26) is the adversarial loss used in GAN-based SID methods, such as [84, 123, 158], where z is noise, D and G are discriminator and generator, respectively. In Eqn.(27), the first and second expressions are the dichotomous and multiple classification expressions, respectively. In Eqn.(28), p(x) and q(x) are two probability distributions on random variables, and Ni et al. [78] use K-L divergence to calculate the distance between the ground-truth and predicted image. In Eqn.(29), R is rain streak, K is the number of mixture components, \( \pi_{k}, \mu_{k} \), and \( \Sigma_{k} \) are mixture coefficients, Gaussian distribution means, and variance, respectively. Wei et al. [121] use the negative log-likelihood function to constrain the unsupervised samples, where Pi = \( \pi_{1},...\pi_{K}, \Sigma = \Sigma_{1},...\Sigma_{K} \), K is the number of mixture components, and N is the number of samples. In Eqn.(30), v, v’ \( \in \mathcal{V} \) denote the possible inputs that index the GP. In Eqn.(31), where I is rain image, \( \omega_{i,k} \) is the kth filter centered at the ith pixel. * is the convolution operation. Wang et al. [116] use it to construct a QS loss, where S(*) denotes the network inference. In Eqn.(32), max_i represents the index of the top i-th coefficient, and mean_{ac_{xy}} denote the mean values of ac_x and ac_y. In Eqn.(33), the total variation function \( R_{V, \beta}(x) \) encourages the images to include piece-wise constant patches, images are discrete (\( X \in \mathbb{R}^{H \times W} \)) and \( \beta = 1 \). Huang et al. [46] adopt a Total Variation regularizer term to smooth the recovered background image. In Eqn.(34), the input is a triplet, including the anchoring example (a), positive example (p), and negative example (n). Deng et al. [27] use a
triplet loss by optimizing the distance between the anchoring example and the positive example to be smaller than the distance between the anchoring example and the negative example, the similarity calculation between the samples is realized. In Eqn. (35), where $T_k$ is the target label (0, 1), $v_k$ is the vector of each capsule in the last layer; $m^+$ and $m^-$ represent the margins for the positive and negative predictions. Yang et al. [133] use this margin loss in their work. In Eqn. (36), $q$ is a degradation representation, $k^+$ and $k^-$ are the corresponding positive and negative counterparts, respectively. $r$ is a temperature hyper-parameter and $K$ denotes the number of negative samples. Li et al. [57] use this contrastive loss in their work. In Eqn. (37), $T(x)$ and $S(x)$ denote the output of the teacher model $T$ and student model $S$, respectively. $C(\cdot)$ is a similarity criterion loss (e.g., L1 loss). Zou et al. [164] use this knowledge distillation loss to force $S$ to approximate the original statistical modeling of $T$. In Eqn. (38), $N^s_{\max}$ and $N^r_{\max}$ is max value of pixel in synthetic streak feature $R^s$ and real-world streak feature $R^r$, respectively. Cui et al. [14] use this dark channel loss on the deraining images of the student model.

Remarks. We count the objective functions of the methods in Table 5 and 6, and listed the statistical results in Table 12, from which we can conclude that:

1) Fidelity-driven metrics are wildly used in data-driven SID methods. Specifically, MSE is the most popular loss function, as 58 of 97 methods chose it, while MAE is the second-popular since 49 SID methods selected it;

2) A single loss is used in 36 SID methods as the objective function. Specifically, MSE, MAE, and SSIM are used as a single loss function in 14, 13, and 6 current SID methods, respectively. This phenomenon implies that a single loss function can also make network convergence well;

3) The cross-entropy has been chosen by 7 SID methods. For example, it is used as a loss function to constrain the rain mask prediction ([136]) or rain density prediction ([153]);

4) Adversarial loss is widely selected by 23 GAN-based SID networks, such as ([53, 84, 120, 123, 124, 158]);

5) SSIM is used in 27 SID methods, such as [21, 24, 72, 88, 113, 141], and it was frequently used 9 times in 2022;

6) The probability&statistics models are used in 42 methods. Due to the complexity of rain distribution, probability&statistics models are usually hard to be guaranteed, which is usually used as an auxiliary constraint. This form of unconditional constraint may be a future research direction.

6. Data Ranking Experiment

In previous related works [61, 107, 109, 137], it is rather common to perform experiments on synthetic datasets and compare the results of different methods. Generally speaking, the researchers tend to perform a horizontal comparison, i.e., evaluating different SID methods on the same dataset, but rarely make a vertical comparison and analysis of the performance of each method on different synthetic datasets, mainly because the horizontal comparison can highlight the superiority and fairness of each method. Note that the authors of related papers on SID usually select three or so synthetic datasets for experimental evaluation. The main reasons are twofold. First, due to many synthetic datasets, it is time-consuming and laborious to evaluate the methods on all of them. Second, researchers may prefer to choose those favorable datasets for the experiments, so that the proposed method can perform better than other competitors. In this paper, we mainly vertically evaluate the contribution of the data generated by rain models on the task. Although the vertical study cannot highlight the characteristics of the method itself, it can reflect the difficulty of the dataset and the matching degree with the rain model. Since data play a core role among the three factors, we mainly focus on analyzing data and designing a group of experiments to study the impact of data and evaluate the effectiveness of SID data.

6.1. Experiment Design

The data ranking experiments are mainly prepared to figure out which paired dataset is most effective for the SID task, denoted by EOD (Effectiveness of Data). A higher EOD score can at least reflect two things: 1) this dataset is more challenging and can potentially stimulate the network to fit continuously, without causing rapid convergence; 2) a pre-trained method on this dataset will potentially have better generalization ability in the real scenario. To obtain more fair and reliable evaluation results, we will conduct four tests from different dimensions:

- **Test-1**: Objective evaluation on paired datasets by full-reference Image Quality Assessment, i.e., PSNR and SSIM;
- **Test-2**: Objective evaluation on real image datasets by using no-reference Image Quality Assessment metrics, i.e., SSEQ, NIQE, BRISQUE, and BLIINDS-II;
- **Test-3**: Subjective evaluation on real image dataset by the human subjective assessment, i.e., Bradley-Terry model;
- **Test-4**: High-level task evaluation on real image dataset by using object detection evaluation metrics, i.e., mean Average Precision (mAP).
Table 14
The numerical evaluation results of the data ranking experiment (from Test-1 to Test-4).

|               | Original | Rain100H [136] | Rain100L [136] | Rain800 [154] | Rain1400 [23] | Rain1200 [153] | SPA-Data [113] | RainCityscapes [42] | Outdoor-Rain [60] | MPID [61] |
|---------------|----------|----------------|----------------|---------------|---------------|---------------|----------------|---------------------|------------------|------------|
| **DetailNet**  |          |                |                |               |               |               |                |                     |                  |            |
| Test-1 PSNR † | -        | 26.78          | 34.54          | 25.23         | 30.05         | 29.20         | 36.93          | 27.52               | 20.19            | 34.65      |
| Test-1 SSIM † | -        | 0.810          | 0.956          | 0.841         | 0.904         | 0.903         | 0.960          | 0.904               | 0.833            | 0.953      |
| Test-2 SSEQ † | 30.61    | 34.10          | 31.25          | 34.15         | 33.98         | 34.39         | 31.76          | 37.68               | 29.14            | 31.28      |
| Test-2 NIQE † | 3.959    | 4.267          | 3.723          | 4.048         | 3.925         | 4.166         | 3.754          | 4.063               | 4.170            | 3.908      |
| Test-2 BRIS † | 25.69    | 31.36          | 24.88          | 29.17         | 29.08         | 30.97         | 24.06          | 33.61               | 36.41            | 26.20      |
| Test-2 BLIIN †| 80.46    | 86.09          | 81.67          | 81.39         | 82.12         | 78.38         | 82.30          | 69.29               | 77.54            | 82.71      |
| Test-3 Clear ↑| 0.27     | 0.07           | 0.15           | 0.12          | 0.14          | 0.07          | 0.04           | 0.05                | 0.08             |           |
| Test-3 Authe ↑| 0.12     | 0.09           | 0.15           | 0.18          | 0.15          | 0.15          | 0.02           | 0.03                | 0.11             |           |
| Test-4 FCNN † | 18.84    | 16.20          | 18.72          | 16.41         | 18.03         | 18.36         | 17.71          | 15.59               | 17.14            | 18.27      |
| Test-4 YOLO † | 19.61    | 18.47          | 19.57          | 17.86         | 18.94         | 19.90         | 19.27          | 12.51               | 14.17            | 19.13      |
| **RESCAN** [64] |         |                |                |               |               |               |                |                     |                  |            |
| Test-1 PSNR † | -        | 28.95          | 38.41          | 28.36         | 32.69         | 33.86         | 38.50          | 18.03               | 21.10            | 40.52      |
| Test-1 SSIM † | -        | 0.867          | 0.980          | 0.885         | 0.941         | 0.948         | 0.972          | 0.773               | 0.834            | 0.984      |
| Test-2 SSEQ † | 30.61    | 30.26          | 30.36          | 33.46         | 32.18         | 32.15         | 31.62          | 31.25               | 27.13            | 30.07      |
| Test-2 NIQE † | 3.959    | 4.007          | 3.698          | 4.087         | 4.233         | 4.108         | 3.924          | 3.871               | 4.080            | 3.821      |
| Test-2 BRIS † | 25.69    | 25.81          | 24.05          | 28.72         | 29.30         | 25.44         | 28.09          | 27.93               | 24.93            |           |
| Test-2 BLIIN †| 80.46    | 87.91          | 82.14          | 82.50         | 79.45         | 79.66         | 81.46          | 77.87               | 81.39            | 82.47      |
| Test-3 Clear ↑| 0.32     | 0.08           | 0.23           | 0.09          | 0.06          | 0.11          | 0.04           | 0.05                | 0.04             |           |
| Test-3 Authe ↑| 0.14     | 0.13           | 0.11           | 0.16          | 0.11          | 0.11          | 0.01           | 0.04                | 0.17             |           |
| Test-4 FCNN † | 18.84    | 18.02          | 18.63          | 18.71         | 18.55         | 18.92         | 18.47          | 17.36               | 18.01            | 18.96      |
| Test-4 YOLO † | 19.61    | 19.38          | 19.68          | 19.39         | 18.91         | 19.16         | 20.49          | 17.01               | 17.37            | 19.65      |
| **LPNet** [24] |         |                |                |               |               |               |                |                     |                  |            |
| Test-1 PSNR † | -        | 19.64          | 33.63          | 24.62         | 26.40         | 27.39         | 37.08          | 21.55               | 16.78            | 32.74      |
| Test-1 SSIM † | -        | 0.805          | 0.959          | 0.859         | 0.905         | 0.901         | 0.968          | 0.900               | 0.781            | 0.960      |
| Test-2 SSEQ † | 30.61    | 37.54          | 31.10          | 37.36         | 35.54         | 35.59         | 32.76          | 30.28               | 28.05            | 31.01      |
| Test-2 NIQE † | 3.959    | 4.733          | 3.690          | 4.886         | 4.313         | 4.696         | 3.850          | 4.070               | 4.731            | 3.820      |
| Test-2 BRIS † | 25.69    | 37.42          | 24.61          | 34.23         | 29.90         | 32.76         | 24.58          | 28.36               | 39.72            | 23.98      |
| Test-2 BLIIN †| 80.46    | 78.15          | 81.45          | 72.94         | 78.47         | 75.31         | 78.48          | 77.86               | 84.48            | 83.68      |
| Test-3 Clear ↑| 0.09     | 0.12           | 0.17           | 0.17          | 0.17          | 0.13          | 0.02           | 0.02                | 0.12             |           |
| Test-3 Authe ↑| 0.05     | 0.13           | 0.21           | 0.15          | 0.17          | 0.15          | 0.00           | 0.02                | 0.12             |           |
| Test-4 FCNN † | 18.84    | 14.15          | 17.89          | 16.32         | 17.19         | 17.75         | 17.90          | 16.63               | 16.16            | 17.99      |
| Test-4 YOLO † | 19.61    | 15.48          | 19.40          | 17.67         | 18.76         | 17.88         | 19.51          | 14.47               | 14.20            | 19.35      |

Figure 11: Illustration of some derained results of DetailNet [23] under our tests. From the top row to the bottom row denote the deraining examples of Test-1, Test-2, and Test-4, respectively. From (a) to (i) denote the results of methods pre-trained on the 9 public paired datasets, i.e., Rain100H [136], Rain100L [136], Rain800 [154], Rain1400 [23], Rain1200 [153], SPA-Data [113], RainCityscapes [42], Outdoor-Rain [60], and MPID [61], respectively.
For better understanding, we list the implementation details and emphasis on the four types of tests in Table 13, from which we see that different tests have different emphases, and a fair final score can be obtained by combining different subjective and objective evaluations and low-level and high-level tasks. Specifically, Test-1 is done for basic simulation experiments, i.e., training and testing on synthetic datasets, and is mainly used to verify the difficulty of the data itself. Test-2 and Test-3 evaluate the generalization abilities of data by using both objective and subjective evaluation metrics, which can indicate how large the difference between synthetic and natural scenarios. Finally, Test-4 is conducted on a high-level task, i.e., object detection, which can reflect the impact of data on practical vision tasks.

6.2. Dataset and Experiment Setting

Datasets. In this study, 9 public paired datasets are used in Test-1, which include Rain100H [136], Rain100L [136], Rain800 [154], Rain1400 [23], Rain1200 [153], SPA-Data [113], RainCityscapes [42], Outdoor-Rain [60] and MPID [61]. For both Test-2 and Test-3, we construct a new dataset called Real-Internet with 385 real-world images without ground-truth, which are collected from the Internet and previous studies [121, 136, 153]. For Test-4, an object detection dataset RID [61] is used for the task-driven evaluation.

Evaluation Metrics. To evaluate the deraining result for the images with ground truth in Test-1, we use PSNR [47] and SSIM [7] as evaluation metrics. To evaluate the authentic images without ground truth in Test-2, we use the no-reference IQA metrics, including SSEQ [71], NIQE [77], BRISQUE [76] and BLIINDS-II [91]. To evaluate the authentic images by subjective assessment in Test-3, we follow a standard-setting of the Bradley-Terry model [6] to estimate the subjective score of each method so that they can be ranked, with the same routine as [58]. For the high-level task evaluation in Test-4, we use the mean Average Precision (mAP) score to evaluate the detection performance.

Implementation Details. In this study, three representative methods are chosen for testing as examples, i.e., DetailNet [23], RESCAN [64], and LPNet [24]. All methods are trained on an NVIDIA GeForce GTX 1080Ti GPU with 12GB memory. We use the original implementation codes and follow the same experimental setup for these methods. All of these three methods are trained on the above 9 public paired datasets (i.e., Rain100H [136], Rain100L [136], Rain800 [154], Rain1400 [23], Rain1200 [153], SPA-Data [113], RainCityscapes [42], Outdoor-Rain [60] and MPID [61]), and are tested on the corresponding test sets.

6.3. Experiment Results

We will analyze the deraining results of each test and finally calculate the EOD score in this subsection.

6.3.1. Test-1: Objective evaluation on paired datasets

We evaluate the deraining results on 9 paired datasets, and then calculate the improvement compared with the original performance. For PSNR, the more significant the improvement, the less challenging the dataset will be. For SSIM, the closer the value is to 1, the less challenging the dataset will be. Taking DetailNet [23] as an example, the original PSNR/SSIM of Rain100H [136] and Rain100L [136] testing sets are 13.56/0.379 and 26.90/0.838; the derained PSNR/SSIM are 26.78/0.810 and 34.54/0.956. If we consider PSNR, the improvement ratios of Rain100H/Rain100L are 97%/28%. We can infer Rain100L is more challenging than Rain100H. If we consider SSIM, the SSIM value after deraining are 0.810 and 0.956, i.e., Rain100H still has some space to improve since the maximum value of a clean image is 1. In such a case, we can infer that Rain100H is more challenging than Rain100L. As such, we consider both PSNR and SSIM with the same weight. We tested three methods to reduce the error caused by one method, and the results are shown in Table 14. By calculating and normalizing the results, we can obtain the image for each dataset in Test-1, as shown in Fig.12. We see that RainCityscapes [42] obtains the highest score, while Rain100L obtains the lowest score. This result is because RainCityscapes is synthesized by Eqn.(3), which has a more complex rain streak and atmospheric light circumstance. We also illustrate some deraining results of DetailNet in the first row of Fig. 11, from which we see that DetailNet [23] can remove most rain streak information in Rain100L and MPID, but hardly remove the rain streak in the datasets like Rain800, Rain1400 and RainCityscapes. Note that these visual results are consistent with the test scores.

6.3.2. Test-2: Objective evaluation on real data

In this experiment, we evaluate the pre-trained model on a real scenario dataset (i.e., Real-Internet), and then evaluate the improvement compared with the original result. Since these authentic rain images have no corresponding clean ground-truth images, we choose four no-reference IQAs as the evaluation criteria. The results are shown in Table 14, from which we see that most of the numerical indices decline after rain removal (the lower the value, the better the effect), which is contrary to what we expected. We also illustrate some deraining results of DetailNet [23] in the sec-

---

**Figure 12:** The numerical evaluation results of the four tests (i.e., Test-1 to Test-4) in the data ranking experiment.

---

Zhao Zhang et al.: Preprint submitted to Elsevier
and the YOLO-V4 [5] pre-trained on MS COCO [8] are used for evaluating the performance. The mAP score is used as the quantitative evaluation metric. The object detection results are shown in Table 14, from which we see that only a few detection results are higher than the original detection results. That is because SID methods are not optimized for the task of object detection, and the deraining process might lose some discriminative and meaningful semantic information [82]. We also calculate the scores of the datasets according to the improvement proportion of the detection results after rain removal, as illustrated in Fig. 12. We can see that the scores of the dataset are fairly average, except for RainCity and Outdoor due to the remnants of rain streaks. We also show some visualization results of the object detection task in the third row of Fig.11, from which we see that in most cases there is a decrease in the detection performance after the rain removal in terms of detection category and detection accuracy.

6.3.5. Calculation of EOD

To ensure the diversity of data rank experiments and the reliability of EOD, we decided to retain the result of Test-2. After obtaining the results of the four tests, we can calculate the EOD score as follows:

\[
EOD = \lambda_1 S_1 + \lambda_2 S_2 + \lambda_3 S_3 + \lambda_4 S_4,
\]

where \(\lambda_1, \lambda_2, \lambda_3\) and \(\lambda_4\) are the weights of each test. \(S_i\) is the deraining performance of Test-\(i\) (\(i = 1, 2, 3, 4\)). Note that we conduct extensive experiments to determine the appropriate weight and finally choose each test’s weight as 0.3, 0.1, 0.3, and 0.3, respectively. We set the weight \(\lambda_2\) of Test-2 to 0.1 to reduce the unreliability of no-reference IQA. The calculated EOD scores on each dataset are shown in Fig.13. We see that Rain800 obtains the highest score while Outdoor-Rain obtains the lowest score. The EOD values shown in Fig.13 are comprehensively evaluated scores based on both synthetic and real data, objective and subjective metrics, low-level and high-level tasks, so it could help the researchers of interest to choose appropriate data for their tasks. The results of each test in Fig.12 can also be used as reference. From the results, we can draw some conclusions: 1) to obtain state-of-the-art deraining performance, datasets like RainCityscapes, Rain800 and Outdoor-Rain will be more challenging; 2) to obtain better deraining performance in real scenarios, datasets like Rain100H, Rain800 and Rain1400 will be a better choices; 3) to obtain better performance on the object detection task, datasets like Rain100L, SPA-DATA and MPID can be more effective.

**Remarks:** The values of the weights \(\lambda_i\) in Eqn.(39) are critical and influence the rating of each dataset. In our paper, the ratio is determined experimentally and empirically. We re-display the information of the four tests in Table 15 and discuss the reasons for the selections from three aspects:

1. **Matrix:** we mainly focus on the object matrices, since they are useful and cheap for evaluation (such as PSNR/SSIM/mAP), and the ratios of objective and subjective are respectively set to 0.7 and 0.3. Test-2 and Test-3
evaluated the SID performance in real-world scenarios via objective and subjective ways, respectively. However, for Test-2, the no-reference IQAs (SSEQ/NIQE/BRISQUE/BLIIDS-II) may be not reliable [61], so we reduce the ratio of Test-2 to 0.1 and set the ratio of Test-3 to 0.3;

(2) As for Data, we mainly consider the real-world scenario, since image restoration methods should be able to serve practical applications well. Hence, the ratio of real-world v.s. synthetic is set to 0.7 v.s. 0.3 in our method;

(3) As for Task, we focus more on the low-level vision task than the high-level vision task, since SID mainly solves the image restoration problem. As such, the ratios are set to 0.7 v.s. 0.3 for low-level v.s. high-level tasks.

In this paper, we consider these different aspects, and finally determine and set such settings of $\lambda_i$ for the four tests to obtain the ranking of every dataset.

7. Conclusions and Future Work

We have rethought the data-driven SID problem by providing comprehensive analysis and new division criteria from new perspectives. Specifically, we have re-examined the three factors (i.e., data, rain model and network architecture) and divided them under new and more reasonable criteria (i.e., general vs. specific, synthetic vs. mathematical, black-box vs. white-box). We also analyzed the relationships of the three factors from the effectiveness perspective of data and revealed two different solving paradigms (explicit vs. implicit) for the SID task. We further discussed the current mainstream data-driven SID methods from five aspects: training strategy, network pipeline, domain knowledge, data preprocessing and objective function. In addition, we conducted extensive data ranking experiments to evaluate and calculate the effectiveness of current SID datasets. According to in-depth analysis and test, we can draw the following remarks and hypotheses:

(1) Based on the relations of the three factors and the proposed new criteria (see Sections 3 and 4), we can infer:

- Most existing data-driven SID methods did not strictly follow Eqn.(21), which makes the solving paradigm of current SID methods not solid. From this viewpoint, Li et al. [60] is a good example, in which the three factors are combined to make the solution process more explainable. More analyses are concluded in Subsection 4.3.

(2) Based on the analysis of the five aspects of current data-driven SID methods (see Section 5), we can infer that:

- By analyzing from different aspects, we find current data-driven SID methods have some common preferences, such as the fully-supervised mode in training strategy, hybrid structure in network pipeline, random sampling in data preprocessing, and MSE/MAE in the objective function.

- Basic network structures can well handle the SID task, e.g., GAN, U-Net, and Encoder-decoder. Besides, SID methods are very fond of using specific attention modules to remove rain and restore the background effectively. As such, we suggest fresh researchers follow the basic network structure with specific attention modules. For more specific conclusions, please refer to the remarks in each subsection.

(3) Based on the vertical data ranking experiments over nine paired datasets (see Section 6), we can infer that:

- The current mainstream SID datasets cannot enhance the high-level task for the large gap between synthetic and real rain. As such, it is recommended to collect or build datasets with similar distribution to the real rain. Note that SPA-Data [113] is a good example with a satisfactory score from our data ranking experiment.

- The existing no-reference IQA metrics are not suitable for evaluating the SID tasks. However, subjective evaluation consumes a lot of time and labor. Therefore, it is very urgent to develop a specific and effective no-reference IQA metric for evaluating the SID tasks in real scenarios.

- Data play an important role in all data-driven deraining tasks. As such, how to produce/select/generate more effective data is also an important topic in the future. It is recommended to design more effective rain models to generate rain images that can be close to real ones, and we also suggest exploring new and effective automatic rain generation modes, such as DerainCycleGAN [123] or VRGNet [111].

Despite the presented results and analysis of this investigation, there is still much space to be explored for the SID task in the future. We believe that improving the interpretability of the solving process will be beneficial to improve the deraining performance and applications of SID methods, which will be important directions for future research. We have verified the effectiveness of the data through extensive experiments, and we will design more experiments to study the effectiveness of the rain model and network architecture to explore more interpretability in near future.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (62072151, 61732007, 61932009 and 62020106007), and Anhui Provincial Natural Science Fund for Distinguished Young Scholars (2008085J30). Zhao Zhang is the corresponding author of this paper.

References

[1] Arbelaez, P., Maire, M., Fowlkes, C., Malik, J., 2010. Contour detection and hierarchical image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 898–916.

[2] Ba, Y., Zhang, H., Yang, E., Suzuki, A., Pfahl, A., Chandrappa, C.C., de Melo, C.M., You, S., Soatto, S., Wong, A., et al., 2022. Not just streaks: Towards ground truth for single image deraining, in: Proceedings of the European Conference on Computer Vision, Springer. pp. 723–740. https://visual.ee.ucla.edu/gt_rain.htm.
A Survey on Data-Driven Single Image Deraining

[3] Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 2481–2495.

[4] Barnum, P.C., Narasimhan, S., Kanade, T., 2010. Analysis of rain and snow in frequency space. International Journal of Computer Vision 86, 256.

[5] Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

[6] Bradley, R.A., Terry, M.E., 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika 39, 324–345.

[7] Brooks, A.C., Zhao, X., Pappas, T.N., 2008. Structural similarity quality metrics in a coding context: Exploring the space of realistic distortions. IEEE Transactions on Image Processing 17, 1261–1273.

[8] Caesar, H., Uijlings, J., Ferrari, V., 2018. Coco-stuff: Thing and stuff classes in context, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1209–1218.

[9] Chang, Y., Yan, L., Zhong, S., 2017. Transformed low-rank model for line pattern noise removal, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1726–1734.

[10] Chen, F., Li, H., 2021. Robust representation learning with feedback for single image deraining, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7742–7751.

[11] Chen, D., He, M., Fan, Q., Liao, J., Zhang, L., Hou, D., Yuan, L., Hua, G., 2019. Gated context aggregation network for image dehazing and deraining, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, IEEE, pp. 1375–1383. https://github.com/cdyf/GCANet.

[12] Chen, Y.L., Hsu, C.T., 2013. A generalized low-rank appearance model for spatio-temporally correlated rain streaks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1968–1975.

[13] Cords, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B., 2016. The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3213–3223.

[14] Cui, X., Wang, C., Ren, D., Chen, Y., Zhu, P., 2022. Semi-supervised image deraining using knowledge distillation. IEEE Transactions on Circuits and Systems for Video Technology https://ieeexplore.ieee.org/document/9828828.

[15] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Ieee. pp. 248–255.

[16] Deng, L.J., Huang, T.Z., Zhao, X.L., Jiang, T.X., 2018. A directional global sparse model for single image rain removal. Applied Mathematical Modelling 59, 662–679.

[17] Deng, S., Wei, M., Wang, J., Feng, Y., Liang, L., Xie, H., Wang, F.L., Wang, M., 2020. Detail-recovvy image deraining via context aggregation networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14560–14569. https://github.com/Denggihub/DRD-Net.

[18] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T., 2015. Long-term recurrent convolutional networks for visual recognition and description, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2625–2634.

[19] Du, Y., Xu, J., Zhen, X., Cheng, M.M., Shao, L., 2020. Conditional variational image deraining. IEEE Transactions on Image Processing 29, 6288–6301. https://ieeexplore.ieee.org/document/8905969.

[20] Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A., 2010. The pascal visual object classes (voc) challenge. International Journal of Computer Vision 88, 303–338.

[21] Fan, Z., Wu, H., Fu, X., Huang, Y., Ding, X., 2018. Residual-guide network for single image deraining, in: Proceedings of the 26th ACM international conference on Multimedia, pp. 1751–1759.

[22] Fu, X., Huang, J., Ding, X., Liao, Y., Paisley, J., 2017a. Clearing the skies: A deep network architecture for single-image rain removal. IEEE Transactions on Image Processing 26, 2944–2956. https://github.com/jinnovation/rainy-image-dataset.

[23] Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J., 2017b. Removing rain from single images via a deep detail network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3855–3863. https://xueyangfu.github.io/projects/cvpr2017.

[24] Fu, X., Liang, B., Huang, Y., Ding, X., Paisley, J., 2019. Lightweight pyramid networks for image deraining. IEEE Transactions on Neural Networks and Learning Systems 31, 1794–1807. https://xueyangfu.github.io/projects/LPNet.

[25] Fu, X., Qi, Q., Zha, Z.J., Zhu, Y., Ding, X., 2021. Rain streak removal via dual graph convolutional network, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 1352–1360.

[26] Fu, Y.H., Kang, I.W., Lin, C.W., Hsu, C.T., 2011. Single-frame-based rain removal via image decomposition, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE. pp. 1453–1456.

[27] Gao, X., Wang, Y., Cheng, J., Xu, M., Wang, M., 2021. Meta-learning based relation and representation learning networks for single-image deraining. Pattern Recognition 120, 108124.

[28] Garg, K., Nayar, S.K., 2006. Photorealistic rendering of rain streaks. ACM Transactions on Graphics 25, 996–1002.

[29] Garg, K., Nayar, S.K., 2007. Vision and rain. International Journal of Computer Vision 75, 3–27.

[30] Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The kitti dataset. The International Journal of Robotics Research 32, 1231–1237.

[31] Glorot, X., Bengio, Y., 2010. Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, pp. 249–256.

[32] Goferman, S., Zelnik-Manor, L., Tal, A., 2011. Context-aware saliency detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 1915–1926.

[33] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2020. Generative adversarial networks. Communications of the ACM 63, 139–144.

[34] Guo, X., Fu, X., Zhou, M., Huang, Z., Peng, J., Zha, Z.J., 2022. Exploring Fourier prior for single image rain removal, in: Proceedings of the 30th International Joint Conferences on Artificial Intelligence, pp. 935–941. https://github.com/willinglucky/Exploring-Fourier-Prior-for-Single-Image-Rain-Removal.

[35] Guo, Y., Chen, J., Ren, X., Wang, A., Wang, W., 2020. Joint rain-haze removal and haze removal from a single image. IEEE Transactions on Image Processing 29, 9508–9519.

[36] Halder, S.S., Lalonde, J.F., Charette, R.d., 2019. Physics-based rain removal via image decomposition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1209–1218.

[37] Han, S., Fang, Z., Zou, Y., Wang, Y., Jiang, T., 2021. Exploring Fourier prior for single image rain removal, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pp. 1–10.

[38] Hershey, J.R., Olsen, P.A., 2007. Approximating the kullback leibler divergence between gaussian mixture models, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE. pp. IV–317.

[39] He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. IEEE Transactions on Computer Vision 86, 256.

[40] He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. IEEE Transactions on Computer Vision 86, 256.
[41] Hu, J., Shen, L., Sun, G. 2018. Squeeze-and-excitation networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141.

[42] Hu, X., Fu, C.W., Zhu, L., Heng, P.A., 2019. Depth-attentional features for single-image rain removal, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8022–8031. https://github.com/xu-hu/DAF-Net.

[43] Hu, X., Zhu, L., Wang, T., Fu, C.W., Heng, P.A., 2021. Single-image real-time rain removal based on depth-guided non-local features. IEEE Transactions on Image Processing 30, 1759–1770. https://github.com/xu-hu/DGNL-Net.

[44] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4700–4708.

[45] Huang, H., Yu, A., Chai, Z., He, R., Tan, T., 2021a. Selective wavelet attention learning for single image deraining. International Journal of Computer Vision 129, 1282–1300.

[46] Huang, H., Yu, A., He, R., 2021b. Memory oriented transfer learning for semi-supervised image deraining, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7732–7741.

[47] Huynh-Thu, Q., Ghanbari, M., 2008. Scope of validity of psnr in image/video quality assessment. Electronics Letters 44, 800–801.

[48] Jiang, K., Wang, Z., Wang, Y., Li, P., Jiang, J., Xiao, J., Lin, C.W., 2022. Danet: Image deraining via dynamic association learning, in: Proceedings of the 31th International Joint Conferences on Artificial Intelligence. https://github.com/kjjian94/DANet.

[49] Jiang, K., Wang, Z., Li, P., Chen, C., Huang, B., Lao, Y., Ma, J., Jiang, J., 2020. Multi-scale progressive fusion network for single image deraining, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8346–8355. https://github.com/kjjian94/MSPFN.

[50] Jiang, T.X., Huang, T.Z., Zhao, X.L., Deng, L.J., Wang, Y. 2017. A novel tensor-based video rain streaks removal approach via utilizing discriminatively intrinsic priors, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4057–4066.

[51] Jin, X., Chen, Z., Li, W., 2020. Ai-gan: Asynchronous interactive generative adversarial network for single image rain removal. Pattern Recognition 100, 107143.

[52] Jin, X., Chen, Z., Lin, J., Chen, J., Zhou, W., Shan, C., 2018. A decomposed dual-cross generative adversarial network for image rain removal, in: Proceedings of the British Machine Vision Conference, p. 119.

[53] Jin, X., Chen, Z., Lin, J., Chen, Z., Zhou, W., 2019. Unsupervised single image deraining with self-supervised constraints, in: Proceedings of the IEEE International Conference on Image Processing, IEEE, pp. 2761–2765.

[54] Johnson, J., Alahi, A., Fei-Fei, L., 2016. Perceptual losses for real-time style transfer and super-resolution, in: Proceedings of the European Conference on Computer Vision, Springer, pp. 694–711.

[55] Kipf, T.N., Welling, M., 2016. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

[56] Kraskov, A., Stögbauer, H., Grassberger, P., 2004. Estimating mutual information. Physical Review E 69, 066138.

[57] Li, B., Liu, X., Hu, P., Wu, Z., Lv, J., Peng, X., 2022a. All-in-one image restoration for unknown corruption, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17452–17462. https://github.com/XLearning-SCU/2023-CVP-R-AirNet.

[58] Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., Wang, Z., 2018a. Benchmarking single-image dehazing and beyond. IEEE Transactions on Image Processing 27, 492–505.

[59] Li, G., He, X., Zhang, W., Chang, H., Dong, L., Lin, L., 2018b. Non-locally enhanced encoder-decoder network for single image de-raining, in: Proceedings of the 26th ACM international conference on Multimedia, pp. 1056–1064. https://github.com/AlexHeX7/NLEDN.

[60] Li, R., Cheong, L.F., Tan, R.T., 2019a. Heavy rain image restoration: Integrating physics model and conditional adversarial learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1633–1642. https://github.com/Iruoteng/HeavyRainRemoval.

[61] Li, S., Araujo, I.B., Ren, W., Wang, Z., Tokuda, E.K., Junior, R.H., Cesar-Junior, R., Zhang, J., Guo, X., Cao, X., 2019b. Single image deraining: A comprehensive benchmark analysis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3838–3847. https://github.com/panda-lab/Single-Image-Deraining.

[62] Li, S., Ren, W., Wang, F., Araujo, I.B., Tokuda, E.K., Junior, R.H., Wang, Z., Cao, X., et al., 2021. A comprehensive benchmark analysis of single image deraining: Current challenges and future perspectives. International Journal of Computer Vision 129, 1301–1322.

[63] Li, S., Ren, W., Zhang, J., Yu, J., Guo, X., 2019c. Single image rain removal via a deep decomposition-composition network. Computer Vision and Image Understanding 186, 48–57. https://sites.google.com/view/xjguo/rain.

[64] Li, X., Wu, J., Lin, Z., Liu, H., Zha, H., 2018c. Recurrent squeeze-and-excitation context aggregation net for single image deraining, in: Proceedings of the European Conference on Computer Vision, pp. 254–269. https://github.com/XiaLiPKU/RESCAN.

[65] Li, Y., Chang, Y., Yu, C., Yan, L., 2022b. Close the loop: A unified bottom-up and top-down paradigm for joint image deraining and segmentation , 1438–1446.

[66] Lin, C.Y., Tao, Z., Xu, A.S., Kang, I.W., Akhyar, F., 2020. Sequential dual attention network for rain streak removal in a single image. IEEE Transactions on Image Processing 29, 9250–9265. https://github.com/fityanul/SDAN-for-Rain-Removal.

[67] Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2117–2125.

[68] Liu, J., Yang, W., Yang, S., Guo, Z., 2018a. D3r-net: Dynamic routing residue recurrent network for video rain removal. IEEE Transactions on Image Processing 27, 699–712.

[69] Liu, J., Yang, W., Yang, S., Guo, Z., 2018b. Erase or fill? deep joint recurrent rain removal and reconstruction in videos, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3233–3242.

[70] Liu, L., Liu, B., Huang, H., Bovik, A.C., 2014. No-reference image quality assessment based on spatial and spectral entropies. Signal Processing: Image Communication 29, 856–863.

[71] Liu, X., Suganuma, M., Sun, Z., Okatani, T., 2019. Dual residual networks leveraging the potential of paired operations for image restoration, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7007–7016. https://github.com/liu-vis/DualResidualNetworks.

[72] Liu, Y., Yue, Z., Pan, J., Su, Z., 2021. Unpaired learning for deep image deraining with rain direction regularizer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4753–4761. https://lewisyangliu.github.io/projects/UDRDR.

[73] Luo, W., Lai, J., Xie, X., 2020. Weakly supervised learning for raindrop removal on a single image. IEEE Transactions on Circuits and Systems for Video Technology 31, 1673–1683.

[74] Luo, W., Wang, K., Wang, W., 2018. Removing rain from a single image via discriminative sparse coding, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3397–3405.

[75] Lu, S., Araujo, I.B., Ren, W., Wang, Z., Tokuda, E.K., Junior, R.H., Cesar-Junior, R., Zhang, J., Guo, X., Cao, X., 2019b. Single image deraining: A comprehensive benchmark analysis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3838–3847. https://github.com/panda-lab/Single-Image-Deraining.

[76] Mittal, A., Moothoo, A.K., Bovik, A.C., 2011. Blind/referenceless image spatial quality evaluator, in: Proceedings of the Conference Record of the 35th Asilomar Conference on Signals, Systems and Computers, IEEE, pp. 723–727.
Zhao Zhang et al.: Preprint submitted to Elsevier
[112] Wang, Q., Sun, G., Dong, J., Zhang, Y., 2022c. Pfdn: Pyramid feature decoupling network for single image deraining. IEEE Transactions on Image Processing 31, 7091–7101.

[113] Wang, T., Yang, X., Xu, K., Chen, S., Zhang, Q., Lau, R.W., 2019b. Spatial attentive single-image deraining with a high quality real rain dataset, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12270–12279. https://github.com/stewevong/SPANet.

[114] Wang, Y., Gong, D., Yang, J., Shi, Q., Xie, D., Zeng, B., et al., 2020e. Deep single image deraining via modeling haze-like effect. IEEE Transactions on Multimedia 23, 2481–2492.

[115] Wang, Y., Liu, S., Chen, C., Zeng, B., 2017. A hierarchical approach for rain or snow removing in a single color image. IEEE Transactions on Image Processing 26, 3936–3950.

[116] Wang, Y., Ma, C., Zeng, B., 2021d. Multi-decoding deraining network and quasi-sparsity based training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13375–13384.

[117] Wang, Y., Song, Y., Ma, C., Zeng, B., 2020f. Rethinking image deraining via rain streaks and vapors, in: Proceedings of the European Conference on Computer Vision, Springer, pp. 367–382. https://github.com/yluestc/derain.

[118] Wang, Y.T., Zhao, X.L., Jiang, T.X., Deng, L.J., Chang, Y., Huang, T.Z., 2020g. Rain streaks removal for single image via kernel-guided convolutional neural network. IEEE Transactions on Neural Networks and Learning Systems 32, 3664–3676. https://github.com/munin718/KCNN.

[119] Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H., 2022d. Uformer: A general u-shaped transformer for image restoration, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17683–17693. https://github.com/ZhendongWang6/Uformer.

[120] Wang, Z., Li, J., Song, G., 2019c. Dtdn: Dual-task de-raining network, in: Proceedings of the 27th ACM International Conference on Multimedia, pp. 1833–1841. https://github.com/long-username/DTDN-DTDN-Dual-task-De-raining-Network.

[121] Wei, W., Meng, D., Zhao, Q., Xu, Z., Wu, Y., 2019a. Semi-supervised transfer learning for image rain removal, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3877–3886. https://github.com/wwzjer/Semi-supervised-IRR-CNN.

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

[123] Wei, Y., Zhang, Z., Wang, Y., Xu, M., Yang, Y., Yan, S., Wang, M., 2021a. Deraincycler: Rain attentive cyclegan for single image deraining and rainmaking. IEEE Transactions on Image Processing 30, 4788–4801. https://github.com/OaDisis/DerainCycleGAN.

[124] Wei, Y., Zhang, Z., Wang, Y., Zhang, H., Zhao, M., Xu, M., Wang, M., 2021b. Semi-deraining: A new semi-supervised single image deraining network, in: Proceedings of the IEEE International Conference on Multimedia and Expo, pp. 1–6.

[125] Wei, Y., Zhang, Z., Xu, M., Hong, R., Fan, J., Yan, S., 2022a. Robust attention deraining network for synchronous rain streaks and raindrops removal, in: Proceedings of the 30th ACM International Conference on Multimedia, pp. 6464–6472.

[126] Wei, Y., Zhang, Z., Zhang, H., Hong, R., Wang, M., 2019b. A coarse-to-fine multi-stream hybrid deraining network for single image deraining, in: Proceedings of the IEEE International Conference on Data Mining, IEEE, pp. 628–637.

[127] Wei, Y., Zhang, Z., Zheng, H., Hong, R., Yang, Y., Wang, M., 2022b. Sginet: Towards sufficient interaction between single image deraining and semantic segmentation, in: Proceedings of the 30th ACM International Conference on Multimedia, pp. 6202–6210. https://github.com/OaDisis/SGINet.

[128] Wu, X., Huang, T.Z., Deng, L.J., Zhang, T.J., 2022. A decoder-free transformer-like architecture for high-efficiency single image deraining, in: Proceedings of the 31th International Joint Conferences on Artificial Intelligence.

[129] Xiao, J., Fu, X., Liu, A., Wu, F., Zha, Z.J., 2022. Image de-raining transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[130] Xu, J., Zhao, W., Liu, P., Tang, X., 2012a. An improved guidance image based method to remove rain and snow in a single image. Computer and Information Science 5, 49.

[131] Xu, J., Zhao, W., Liu, P., Tang, X., 2012b. Removing rain and snow in a single image using guided filter, in: Proceedings of the IEEE International Conference on Computer Science and Automation Engineering, IEEE, pp. 304–307.

[132] Yan, Q., Xu, L., Shi, J., Jia, J., 2013. Hierarchical saliency detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1155–1162.

[133] Yang, F., Ren, J., Lu, Z., Zhang, J., Zhang, Q., 2022. Rain-component-aware capsule-gan for single image de-raining. Pattern Recognition 123, 108377.

[134] Yang, W., Liu, J., Yang, S., Guo, Z., 2019a. Scale-free single image deraining via visibility-enhanced recurrent wavelet learning. IEEE Transactions on Image Processing 28, 2948–2961.

[135] Yang, W., Tan, R.T., Feng, J., Guo, Z., Yan, S., Liu, J., 2019b. Joint rain detection and removal from a single image with contextualized deep networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 42, 1377–1393. https://github.com/frjyh/ JORDER-E-Deep-Image-Deraining-TPAMI-2019-Journal.

[136] Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S., 2017. Deep joint rain detection and removal from a single image, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1357–1366.

[137] Yang, W., Tan, R.T., Wang, S., Fang, Y., Liu, J., 2020a. Single image deraining: From model-based to data-driven and beyond. IEEE Transactions on Pattern Analysis and Machine Intelligence 43, 4059–4077.

[138] Yang, W., Wang, S., Xu, D., Wang, X., Liu, J., 2020b. Towards scale-free rain streak removal via self-supervised fractal band learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 12629–12636.

[139] Yang, Y., Guan, J., Huang, S., Wan, W., Xu, Y., Liu, J., 2021. End-to-end rain removal network based on progressive residual detail supplement. IEEE Transactions on Multimedia 24, 1622–1636.

[140] Yang, Y., Lu, H., 2019a. Single image scene understanding using a recurrent multi-scale aggregation and enhancement network, in: Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE, pp. 1378–1383. https://github.com/nnUyi/ReMAEN.

[141] Yang, Y., Lu, H., 2019b. Single image deraining via recurrent hierarchy enhancement network, in: Proceedings of the 27th ACM International Conference on Multimedia, pp. 1814–1822. https://github.com/nntui/ReHEN.

[142] Yang, Y., Ran, W., Lu, H., 2020e. Rddan: A residual dense dilated aggregated network for single image deraining, in: Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE, pp. 1–6.

[143] Yasarla, R., Patel, V.M., 2019. Uncertainty guided multi-scale residual learning using cycle spinning can for single image de-raining, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8405–8414. https://github.com/ rajeeyyasla/UMRL–using-Cycle-Spinning.

[144] Yasarla, R., Patel, V.M., 2020. Confidence measure guided single image de-raining. IEEE Transactions on Image Processing 29, 4544–4555.

[145] Yasarla, R., Sindagi, V.A., Patel, V.M., 2020. Syn2real transformer learning for image deraining using gaussian processes, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2726–2736. https://github.com/ rajeeyyasla/Syn2Real.

[146] Ye, Y., Chang, Y., Zhou, H., Yan, L., 2021. Closing the loop: Joint rain generation and removal via disentangled image translation, in:
A Survey on Data-Driven Single Image Deraining

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2053–2062.

[147] Yi, Q., Li, J., Dai, Q., Fang, F., Zhang, G., Zeng, T., 2021. Structure-preserving deraining with residue channel prior guidance, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4238–4247. https://github.com/Joyies/SPDNet.

[148] Yu, F., Koltun, V., 2015. Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122.

[149] Yu, W., Huang, Z., Zhang, W., Feng, L., Xiao, N., 2019. Gradual network for single image de-raining, in: Proceedings of the 27th ACM International Conference on Multimedia, pp. 1795–1804.

[150] Yu, Y., Yang, W., Tan, Y.P., Kor, A.C., 2022. Towards robust rain removal against adversarial attacks: A comprehensive benchmark analysis and beyond, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6013–6022. https://github.com/yuyi-sd/Robust_Rain_Removal.

[151] Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., 2022. Restormer: Efficient transformer for high-resolution image restoration, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5728–5739. https://github.com/swz30/Restormer.

[152] Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L., 2021. Multi-stage progressive image restoration, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14821–14831. https://github.com/swz30/MPRNet.

[153] Zhang, H., Patel, V.M., 2018. Density-aware single image deraining using a multi-stream dense network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 695–704. https://github.com/hezhangsprinter/DID-MDN.

[154] Zhang, H., Sindagi, V., Patel, V.M., 2019. Image de-raining using a conditional generative adversarial network. IEEE Transactions on Circuits and Systems for Video Technology 30, 3943–3956. https://github.com/hezhangsprinter/ID-CGAN.

[155] Zhang, K., Luo, W., Ren, W., Wang, J., Zhao, F., Ma, L., Li, H., 2020. Beyond monocular deraining: Stereo image deraining via semantic understanding, in: Proceedings of the European Conference on Computer Vision, Springer, pp. 71–89. https://github.com/HDCVLab/Stereo-Image-Deraining.

[156] Zhang, X., Zhou, X., Lin, M., Sun, J., 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6848–6856.

[157] Zheng, Y., Yu, X., Liu, M., Zhang, S., 2020. Single-image deraining via recurrent residual multiscale networks. IEEE Transactions on Neural Networks and Learning Systems.

[158] Zhu, H., Peng, X., Zhou, J.T., Yang, S., Chandrasekhar, V., Li, L., Lim, J.H., 2019. Single image rain removal with unpaired information: A differentiable programming perspective, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 9332–9339.

[159] Zhu, H., Wang, C., Zhang, Y., Su, Z., Zhao, G., 2020a. Physical model guided deep image deraining, in: Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE, pp. 1–6. https://github.com/Ohraincub/PHYSICAL-MODEL-GUIDED-DEEP-IMAGE-DERAINING.

[160] Zhu, J.Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2223–2232.

[161] Zhu, L., Deng, Z., Hu, X., Xie, H., Xu, X., Qin, J., Heng, P.A., 2020b. Learning gated non-local residual for single-image rain streak removal. IEEE Transactions on Circuits and Systems for Video Technology 31, 2147–2159.

[162] Zoph, B., Le, Q.V., 2016. Neural architecture search with reinforcement learning. arXiv preprint arXiv:1611.01578.

[163] Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V., 2018. Learning transferable architectures for scalable image recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8697–8710.

[164] Zou, W., Wang, Y., Fu, X., Cao, Y., 2022. Dreaming to prune image deraining networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6023–6032.