Two-stage single image reflection removal with reflection-aware guidance

Yu Li · Ming Liu · Yaling Yi · Qince Li · Dongwei Ren · Wangmeng Zuo

Abstract
Removing undesired reflection from an image captured through a glass surface is a very challenging problem with many practical applications. For improving reflection removal, cascaded deep models have been usually adopted to estimate the transmission in a progressive manner. However, most existing methods are still limited in exploiting the result in prior stage for guiding transmission estimation. In this paper, we present a novel two-stage network with reflection-aware guidance (RAGNet) for single image reflection removal (SIRR). To be specific, the reflection layer is firstly estimated due to that it generally is much simpler and is relatively easier to estimate. Reflection-aware guidance (RAG) module is then elaborated for better exploiting the estimated reflection in predicting transmission layer. By incorporating feature maps from the estimated reflection and observation, RAG can be used (i) to mitigate the effect of reflection from the observation, and (ii) to generate mask in soft partial convolution for mitigating the effect of deviating from linear combination hypothesis. A dedicated mask loss is further presented for reconciling the contributions of encoder and decoder features. Experiments on five commonly used datasets demonstrate the quantitative and qualitative superiority of our RAGNet in comparison to the state-of-the-art SIRR methods. The source code and pre-trained model are available at https://github.com/liyucs/RAGNet.

Keywords Reflection removal · Soft partial convolution

1 Introduction

Reflections are commonly observed in images taken through glass, where the light reflected by the glass (i.e., reflection) and that from the objects behind the glass (i.e., transmission) are both captured by the camera. Undesired reflections inevitably degrade image visual quality and hinder many subsequent vision applications such as autonomous driving [1, 2], making reflection removal practically valuable but very challenging in low-level vision [3–6]. To mitigate the ill-posedness of reflection removal, early studies usually resort to multiple input images captured under different illumination conditions, focal lengths, polarizer angles, and camera viewpoints. On the contrary, single image reflection removal (SIRR) [7–10] is more common and practical in task setting, and has received considerable research interests in the recent few years.

Following the formulation in [5, 8], the input image, i.e., an observation $I$, can be viewed as a linear combination of a transmission layer $T$ and a reflection layer $R$, i.e., $I = T + R$. The goal of SIRR is to recover the transmission layer $T$ for the observation $I$. Some single-stage methods [5, 8, 11] have been proposed to learn a direct mapping from $I$ to $T$. However, the intrinsic ill-posedness of SIRR makes such process very challenging. To ease the difficulty of training SIRR, cascaded deep models are usually adopted.
Fig. 1 An example of real-world observation from Real 20 dataset and the reflection predicted by our model. It can be seen that the linear combination is violated in regions with strong reflection. The results of competing methods are given in the second row to mitigate the uncertainty of transmission estimation [4, 9, 12]. For example, [4] first estimates the edge map and then the transmission, while [9, 12] progressively refine the estimation of reflection and transmission layers. However, there remain several interesting issues to further investigate (i) what should be estimated in the prior stage and (ii) how to exploit the result in prior stage for guiding succeeding transmission estimation, thereby leaving some leeway in improving SIRR.

In this paper, we take a step forward to address the above issues, and present a novel two-stage network with reflection-aware guidance (RAGNet). To be specific, our RAGNet first estimates the reflection layer \( \hat{R} \) in the first stage. Then, it takes both the estimated reflection \( \hat{R} \) and observation \( I \) as the input to predict the transmission layer \( \hat{T} \). Figure 1 shows an example of an input image as well as the estimated reflection by our RAGNet. The reasons to estimate the reflection layer in the first stage can be explained from three aspects. First, the reflection layer generally is much simpler than the transmission layer, and thus is relatively easier to estimate. Second, the estimated reflection is beneficial to transmission estimation. For example, the difference between the observation and estimated reflection, i.e., \( I - \hat{R} \), can serve as a reasonable initialization of the transmission. Third, the linear combination \( I = T + R \) may not hold true for real-world observations, especially those images with heavy and bright reflection (see Fig. 1). Also, the illumination of estimated reflection is helpful in finding the regions violating the linear combination hypothesis.

Furthermore, we elaborate a reflection-aware guidance (RAG) module for better exploiting the estimated reflection in the second stage. As summarized in Table 1, the output of the prior stage is concatenated with the original observation as the input to the subsequent stage in [4, 9]. Li et al. [12] further applied a recurrent layer (i.e., LSTM [14]) to exploit the deep features across stages. Zhang et al. [8] and Peng et al. [15] propose to concurrently predict the transmission and reflection layers. Kim et al. [11] and Chang et al. [16] add an extra stage to reconstruct \( \hat{R} \) at the end of the pipeline. The predicted \( \hat{T} \) is then refined by minimizing the gap between \( \hat{R} \) and \( R \). In comparison, the second stage of

| Table 1 | Comparison of models of different types for SIRR |
|---------|-------------|
|         | Methods     | Scheme                        | Cross-stage fusion |
| Single-stage |             |                               |                   |
| Zhang et al. [8] | \( I \rightarrow [T, R] \) | – |
| ERRNet [5] | \( I \rightarrow T \) | – |
| Peng et al. [15] | \( I \rightarrow [T, R] \) | – |
| Multi-stage |             |                               |                   |
| CEILNet [4] | \( I \rightarrow E \rightarrow T \) | Concat |
| BDN [9] | \( I \rightarrow T_0 \rightarrow R \rightarrow T \) | Concat |
| IBCLN [12] | \((0, I) \rightarrow (R_1, T_1) \rightarrow \ldots \) | Concat+LSTM |
| Kim et al. [11] | \( I \rightarrow [T, R] \rightarrow R \) | BT-net |
| Chang et al. [16] | \( I \rightarrow E \rightarrow T_0 \rightarrow R_0 \rightarrow T \rightarrow R \) | Concat+Recurrent |
| Ours | \( I \rightarrow R \rightarrow T \) | RAG |

\( I, R, T \) and \( E \) denote observation, reflection, transmission and edge maps, respectively.
RAGNet includes an encoder for the estimated reflection and an encoder-decoder for the original observation. For each block of the decoder, an RAG module is elaborated for better leveraging the estimated reflection from two aspects. On the one hand, the encoder feature map of the reflection is utilized to suppress that of the observation, and their difference is then concatenated with the decoder feature map of the observation. On the other hand, the encoder feature maps of the reflection and observation are incorporated with the decoder feature map to generate a soft mask indicating the extent to which the transmission is corrupted by reflection. The mask is then collaborated with soft partial convolution to mitigate the effect of deviating from linear combination hypothesis. And a dedicated mask loss is further presented for reconciling the contributions of encoder and decoder features.

Extensive experiments are conducted on five commonly used real-world datasets to evaluate our RAGNet. With the RAG module, the estimated reflection is effective in guiding the transmission estimation and alleviating the effect of violating linear combination hypothesis. In comparison with the state-of-the-art models [4, 5, 8, 9, 12], our RAGNet performs favorably in terms of both quantitative metrics and visual quality.

The contributions of this work can be summarized from three aspects:

- An SIRR network RAGNet is presented, which involves two stages, i.e., first estimating reflection and then predicting transmission. Its rationality is also explicated to differentiate from existing cascaded SIRR models.
- For better leveraging the estimated reflection in the second stage, the reflection-aware guidance (RAG) module is further introduced to guide the transmission estimation and alleviate the effect of violating linear combination hypothesis.
- Experiments on five popular datasets show that our RAGNet performs favorably against the state-of-the-art methods quantitatively and qualitatively.

The remainder of this paper is organized as follows: In Section 2, we briefly review related works of reflection removal. In Section 3, the proposed RAGNet is elaborated in details along with its training strategies. Section 4 presents the experimental results and ablation studies. Finally, Section 5 ends this paper with concluding remarks.

2 Related work

In this section, we briefly review two categories of relevant methods on reflection removal tasks, i.e., multiple image reflection removal and single image reflection removal.

2.1 Multiple image reflection removal

Due to the ill-posed nature, multiple images have been exploited to solve the reflection removal problem. Some methods of this paradigm leverage the behavioral difference between the reflection and transmission layers under different settings [13, 17–21]. Agrawal et al. [17] used two separate images as input, which are captured with / without flash light, and Fu et al. [18] further took into account the high frequency illumination. Schechner et al. [19] treated the input image as a superimposition of two layers, and novel blur kernels were proposed to eliminate mutual penetration of the separated layers, using images taken with different focal lengths as a hint. Polarization camera is also used to take advantage of the polarization properties of light [13, 20, 21]. Based on the linear hypothesis \( I = T + R \) in raw RGB space, Lei et al. [13] adopted a simple two-stage model but took multiple polarized images as input.

Another line of works take images from different viewpoints to exploit rich information for reflection removal [20, 22–34]. In particular, Punnappurath et al. [3] leveraged dual pixel cameras, by which images captured from two sub-aperture views are readily accessible. Although methods in the multi-input paradigm can obtain impressive results, the necessity to prepare qualified input images with specially designed hardware or extra human efforts makes them inconvenient, and the prerequisites are unable to be satisfied in many practical scenarios, which boosts the recent prevalence of SIRR methods.

2.2 Single image reflection removal

SIRR methods merely require a single image as input. Traditional methods employ various priors observed in real-world images to reduce the ill-posedness, such as gradient sparsity prior [35–38], smoothness prior [39, 40] and ghosting cues [41]. However, the representation and generalization ability of such methods is limited, and they are prone to producing poor results without discriminative hints, which is often the case when using handcrafted priors.

As a remedy, learning based methods have been proposed to solve the SIRR problem by leveraging the capacity of deep convolutional neural networks (CNNs). Due to the large overhead to collect sufficient well-aligned real image pairs for model training, Fan et al. [4] generated reflection layers via smoothing and illumination intensity decay, which were then used to synthesize the observation by linear combination. Zhang et al. [8] further adapted the intensity decay to suit reflections with higher illuminance level, and applied random vignette to simulate different camera views. Besides, several loss functions have been particularly designed to leverage the characteristic of reflections. CEILNet [4] constrains the gradient maps of the predicted
transmission layer to avoid blurry output. Zhang et al. [8] proposed an exclusion loss by minimizing the correlation between the gradient maps of estimated transmission and reflection layers. Li et al. [12] took advantage of the synthesis model to reconstruct the observation, which is constrained to approximate the input. As to network structure, recent works follow two main schemes. On the one hand, [8] and [5] extracted multi-scale features via a pre-trained VGG-19 [42] model, hence being tolerant of mild misalignment between real-world training pairs. On the other hand, the encoder-decoder framework is exploited in [4, 5, 9, 12], where skip connections are usually integrated for better feature utilization.

Considering the difficulty of directly estimating transmission, cascaded models have been widely adopted. Fan et al. [4] first predicted potential edge maps, and then generated the transmission layer. Yang et al. [9] alternated between reflection and transmission estimation, while Li et al. [12] further incorporated an LSTM module [14] to generate the reflection and transmission layers in a progressive manner. Dong et al. [43] further regressed a probabilistic reflection confidence map to locate reflection and refined the reflection removal results in multiple iterations. Such multi-stage methods are closely related to our method, but they are still limited in exploiting the output of prior stage. For CEILNet, the smoothness prior for generating edge map E is often violated in real scenarios, such that the resultant inaccurate edge maps will lead to degraded performance. BDN [9] and IBCLN [12] adopt a simple concatenation between input and predicted reflection, which is insufficient in exploiting the inter-relationship between them. Consequently, BDN [9] often fails to tackle strong reflection regions (see Fig. 1). Despite of improved SIRR performance by the deployed LSTM module, IBCLN [12] is less efficient in terms of inference speed. Additionally, the accumulated error produced by IBCLN could lead to undesired artifacts in some scenarios (e.g., zoom in Fig. 1 for clear observation).

We note that the MIRR method [13] also uses a two-stage network but takes raw polarized images with different polarization angles as input. Moreover, it adopts the linear hypothesis \( I = T + R \), and its concatenation-based fusion is limited in exploiting the estimated reflection for the subsequent stage. Empirically, the retraining of adjusted MIRR by feeding a single unpolarized image is still limited in removing heavy and bright reflections (See Fig. 1).

### 3 The proposed method

In this section, we first present overall architecture design of our RAGNet, and then the dedicated mask loss and the learning objective are given.

#### 3.1 Overall architecture design

We present a two-stage model named RAGNet to address reflection removal problems. Given an input image \( I \), the first stage of RAGNet aims to estimate the reflection layer \( \hat{R} \), and then the estimation of transmission layer \( \hat{T} \) is predicted in the second stage on the basis of \( \hat{R} \). Unlike previous works such as BDN [9] and IBCLN [12], which directly concatenate \( \hat{R} \) with \( I \) and predict the transmission layer \( \hat{T} \) by using this concatenation, we argue that a more reasonable approach is making a subtraction between the features of \( I \) and \( \hat{R} \) in order to generate a strong visual clue for predicting the final transmission layer. However, as shown in Fig. 1, the linear hypothesis \( T = I - R \) could sometimes be violated especially in strong reflection regions, leading to potential problems such as over removal. In this regard, we propose reflection-aware guidance (RAG) modules in the second stage to conduct a soft inpainting process. We will further elaborate on this point in Section 3.2. Specifically, we use a plain U-Net [44] model as the first stage subnetwork, namely \( G_R \), which takes the observation \( I \) as input and predicts the reflection layer, i.e., \( \hat{R} = G_R(I) \). The structure of the second stage subnetwork \( G_T \) is shown in Fig. 2, where \( \hat{R} \) is fed into \( G_T \) together with the observation \( I \) to generate the transmission layer, i.e., \( \hat{T} = G_T(I, \hat{R}) \). In particular, \( G_T \) is composed of two encoders (\( E_I \) and \( E_R \)) and one decoder (\( D_T \)), and an RAG module is elaborated in each block of \( D_T \).

#### 3.2 Reflection-aware guidance

As shown in Fig. 1, reflections are spatially variant, whose location and intensity can be roughly predicted in the first stage. The estimated reflection \( \hat{R} \) can then be incorporated with the observation \( I \) for succeeding transmission estimation. However, the estimated reflection \( \hat{R} \) may be inaccurate. Furthermore, the linear combination hypothesis may not hold true in regions with high reflection intensities, where few informative features of the transmission are retained in \( I - \hat{R} \) (see Fig. 1). Therefore, estimating the transmission in such regions is more challenging, and is very similar to an image inpainting [45, 46] task. Taking these factors into account, we elaborate an RAG module in each decoder block for (i) generating feature complementary to the decoder and (ii) predicting a mask to indicate the strong reflection regions for guiding image inpainting.

Given the observation feature \( F_I \), the reflection feature \( F_R \) and the decoder feature \( F_{dec} \), the RAG module is designed as shown in Fig. 2. Using \( F_I \) and \( F_R \), the difference feature \( F_{diff} \) is generated via a subtraction operation to better suppress the reflections, i.e., \( F_{diff} = F_I - F_R \), which serves as a complement and enhancement.
to the decoder feature $F_{\text{dec}}$. Furthermore, taking $F_I$, $F_R$ and $F_{\text{dec}}$ as input, the RAG module generates a mask $M$ via two $1 \times 1$ convolution layers followed by a sigmoid operation. Note that $M = [M_{\text{diff}}, M_{\text{dec}}]$, where $[\cdot, \cdot]$ denotes the concatenation operation, and $M_{\text{diff}}$ and $M_{\text{dec}}$ are masks regarding to $F_{\text{diff}}$ and $F_{\text{dec}}$, respectively. With such mask, the model is aware of the regions deviating from the linear combination hypothesis, and recovers the transmission layer in such areas mainly relying on the surrounding decoder feature using image inpainting. The mask $M$ is visualized in Fig. 3.

As discussed in Section III-A, the linear hypothesis $T = I - R$ could be violated. The vanilla usage of $F_{\text{diff}} = F_I - F_R$ would result in over removal phenomena. We address this issue through a soft inpainting approach based on soft partial convolution. Here we emphasize the differences between our soft inpainting and the method used in classical image inpainting problems from two aspects. (i) The classical image inpainting deals with vacant regions which has no informative visual cues, while the inpainting in our setting deals with regions with different amounts of information. For strong reflection regions, the inpainting process relies more on $F_{\text{dec}}$, while for weak reflection regions, $F_{\text{diff}}$ is more exploited. That is why we call this process soft inpainting. (ii) Different from the hard coded 0/1 masks used in classical partial convolutions, we adopt soft masks ranging from 0 to 1, which are also predicted by RAG modules. These soft masks endow our network with the ability of being reflection-aware in the inpainting process. The soft partial convolution used in this paper can be formulated as,

$$F' = \begin{cases} (W \ast (F \circ M)) \circ \frac{1}{M_{3 \times 3}} + b, M_{3 \times 3} > 0, \\ 0, \text{ otherwise} \end{cases}$$

where $W$ and $b$ represent the weight and bias of the soft partial convolution, $F = [F_{\text{diff}}, F_{\text{dec}}]$, $\ast$ and $\circ$ are convolution and entry-wise product respectively. $M_{3 \times 3}$ contains the average values of $M$ in $3 \times 3$ neighborhood regions, which can be efficiently calculated by a $3 \times 3$ average pooling operation.

### 3.3 Mask loss

Furthermore, we design a novel mask loss to better exploit the mask $M$ during training RAGNet. As shown in Fig. 1, for regions with strong reflections, $F_{\text{diff}}$ can hardly provide
informative features. Therefore, we dispose the values of \( \mathbf{M}_{\text{diff}} \) to approximate 0 in such areas, i.e.,

\[
L_{\text{mask}}^{\text{diff}} = \sum_{i=1}^{4} \| \mathbf{M}_{\text{diff}}^{i}[R > \phi] \|_1,
\]

(2)

where \( i \) means the \( i \)-th layer, \( \| \cdot \|_1 \) denotes the \( \ell_1 \) norm, \( A[\text{condition}] \) represents the part of \( A \) that meets the condition in the square brackets, \( \phi \) is the threshold that delimit strong reflection regions.

However, with \( L_{\text{mask}}^{\text{diff}} \) only, \( \mathbf{M} \) may be optimized towards a trivial solution \( \mathbf{0} \), which diminishes the effect of \( F_{\text{diff}} \). Therefore, \( \mathbf{M} \) is supposed to be 1 for areas with few reflections, to avoid the trivial solution as a regularization term and force the soft partial convolution to exploit both \( F_{\text{diff}} \) and \( F_{\text{dec}} \), since both of them are reliable in such areas, i.e.,

\[
L_{\text{mask}}^{\text{reg}} = \sum_{i=1}^{4} \| \mathbf{M}^{i}[R < \xi] - 1 \|_1,
\]

(3)

where \( \xi \) means threshold for regions with few reflections.

Note that we do not constrain the mask values in remaining regions, which will be automatically optimized for better transmission estimation. Therefore, the mask loss is formulated as,

\[
L_{\text{mask}} = L_{\text{mask}}^{\text{diff}} + L_{\text{mask}}^{\text{reg}},
\]

(4)

and we empirically set \( \xi = 0.01 \) and \( \phi = 0.3 \). The effectiveness of mask loss is further proved in ablation studies.

### 3.4 Learning objective

To train RAGNet, several commonly used loss functions are collaborated, including reconstruction loss, perceptual loss [47], exclusion loss [8] and adversarial loss [48].

**Reconstruction loss** With synthetic image pairs, we are able to minimize the pixel-wise difference between network outputs \( \hat{T}, \hat{R} \) and the corresponding ground-truths \( T, R \),

\[
L_{\text{rec}} = \sum_{Y \in \{T,R\}} \| \hat{Y} - Y \|_1.
\]

(5)

**Perceptual loss** Given a pre-trained VGG-19 [42] model \( \phi \), we minimize the \( \ell_1 \) difference between \( \phi(\hat{T}), \phi(\hat{R}) \) and \( \phi(T), \phi(R) \) in the selected feature layers,

\[
L_{\text{perce}} = \sum_{Y \in \{T,R\}} \sum_{I} \kappa_l \| \phi_l(\hat{Y}) - \phi_l(Y) \|_1,
\]

(6)

where \( l \) indicates the index of \( \text{conv1.2}, \text{conv2.2}, \text{conv3.2}, \text{conv4.2} \) and \( \text{conv5.2} \) layers. The weights \( \{\kappa_l\} \) are used to balance different layers.

**Exclusion loss** Following [8], the exclusion loss is formulated as,

\[
L_{\text{excl}} = \frac{1}{N+1} \sum_{n=0}^{N} \sqrt{\| Y(T_{1n}, R_{1n}) \|}.\]

(7)

where \( Y(T, R) = \text{tanh}(\lambda_T |\nabla T|) \circ \text{tanh}(\lambda_R |\nabla R|) \), and \( \lambda_T \) and \( \lambda_R \) denote normalization factors. \( \nabla T \) and \( \nabla R \) are gradients of \( T \) and \( R \). \( \| \cdot \|_1 \) is the Frobenius norm. \( T_{1n} \) and \( R_{1n} \) represent the \( n \times \) down-sampling versions of \( T \) and \( R \), where \( T_{10} \) and \( R_{10} \) are the original inputs. In practice, we set \( N = 2, \lambda_T = \frac{1}{2} \), and \( \lambda_R = \frac{\| \nabla T \|}{\| \nabla R \|} \).

**Adversarial loss** Adversarial loss is adopted to further enhance the visual quality of the output images. We treat the whole RAGNet as the generator \( G \) and additionally build a 4-layer discriminator \( D \), whose parameters are updated via,

\[
L_D = \mathbb{E}_{I,T} \log D(I, T) + \mathbb{E}_{I} (1 - \log D(I, G(I))),(8)
\]

while the parameters of \( G \) are optimized by,

\[
L_{\text{adv}} = -\mathbb{E}_{I} \log D(I, G(I)).\]

(9)

Taking the above loss functions into account, the learning objective to train our RAGNet can be formulated as,

\[
L = \lambda_1 L_{\text{rec}} + \lambda_2 L_{\text{perce}} + \lambda_3 L_{\text{excl}} + \lambda_4 L_{\text{adv}} + \lambda_5 L_{\text{mask}}.\]

(10)

where \( \lambda_1 = \lambda_2 = \lambda_5 = 1, \lambda_3 = 0.2 \) and \( \lambda_4 = 0.01 \).

### 4 Experimental results

#### 4.1 Implementation details

**Training data** Following previous works, the proposed RAGNet is trained with both synthetic and real-world images, and we use the same data synthesis protocol as ERRNet [5]. Specifically, 7,643 image pairs are chosen from the PASCAL VOC dataset [49], and each pair is \( \{ \text{Real 45}, \text{Real 45} \} \) from [8], 45 images \( \{ \text{Real 45} \} \) from [4] and three subsets from SIR\(^2\) [50], i.e., i) SIR\(^2\) Wild with
Table 2 PSNR/SSIM/VSSIM results by different methods for reflection removal on four real-world datasets with ground truth

| Datasets       | Index | CEILNet [4] | Zhang et al. [8] | BDN [9] | ERRNet [5] | Lei et al. [13] | IBCLN [12] | Kim et al. [11] | Chang et al. [16] | Peng et al. [15] | Ours |
|----------------|-------|-------------|------------------|--------|------------|-----------------|------------|-----------------|-------------------|------------------|------|
| SIR² Solid    | PSNR  | 23.37       | 22.68            | 22.73  | 24.85      | 23.81           | 24.88      | 23.65           | 25.61             | 23.92            | 25.92 |
| 200           | SSIM  | 0.875       | 0.879            | 0.853  | 0.894      | 0.882           | 0.893      | 0.879           | 0.905             | 0.860            | 0.906 |
| SIR² Postcard | PSNR  | 20.09       | 16.81            | 20.71  | 21.99      | 21.48           | 23.39      | 23.02           | 22.73             | 23.29            | 23.80 |
| 199           | SSIM  | 0.786       | 0.797            | 0.857  | 0.874      | 0.873           | 0.875      | 0.857           | 0.860             | 0.890            | 0.876 |
| SIR² Wild     | PSNR  | 21.07       | 21.52            | 22.34  | 24.16      | 23.84           | 24.71      | 25.55           | 25.41             | 25.12            | 25.46 |
| 55            | SSIM  | 0.805       | 0.832            | 0.821  | 0.847      | 0.866           | 0.886      | 0.887           | 0.892             | 0.910            | 0.884 |
| Real 20       | PSNR  | 18.87       | 22.55            | 18.81  | 23.19      | 22.35           | 21.86      | 21.92           | 19.26             | 19.10            | 23.11 |
| 20            | SSIM  | 0.692       | 0.788            | 0.737  | 0.817      | 0.793           | 0.762      | 0.706           | 0.754             | 0.704            | 0.790 |
| Nature        | PSNR  | 19.23       | 19.54            | 18.91  | 20.57      | 20.42           | 23.57      | 20.24           | 20.35             | 20.49            | 23.62 |
| 20            | SSIM  | 0.699       | 0.732            | 0.735  | 0.754      | 0.747           | 0.783      | 0.755           | 0.746             | 0.760            | 0.800 |
| Average       | PSNR  | 21.44       | 20.05            | 21.56  | 23.38      | 22.68           | 24.09      | 23.29           | 24.03             | 23.50            | 24.81 |
| 494           | SSIM  | 0.817       | 0.831            | 0.842  | 0.872      | 0.868           | 0.875      | 0.859           | 0.873             | 0.867            | 0.882 |

The best and the second-best results in each dataset are highlighted with red and blue, respectively.

55 wild scene images, ii) SIR² Solid with 200 controlled scene images of a set of daily-life objects, and iii) SIR² Postcard with 199 controlled scene images on postcards, which are obtained by using one postcard as transmission and another as reflection. Note that we only present the quantitative results of Real 20 and three SIR² subsets, since the ground-truth transmission of Real 45 is unavailable.

**Training configuration** The model is optimized by the Adam [51] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and a fixed learning rate of $1 \times 10^{-4}$. For stable training, we first train $G_K$ for 50 epochs, and then the whole model is jointly trained for another 100 epochs. The values of $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$ in (10) are following the original setting in [8], which are widely adopted by the existing methods. All the experiments are conducted in the PyTorch [52] environment running on a PC with an Nvidia RTX 2080Ti GPU. The source code and pre-trained model are available at https://github.com/liyucs/RAGNet.

### 4.2 Comparison with state-of-the-arts

We compare the proposed RAGNet with nine state-of-the-art SIRR methods, i.e., CEIL.Net [4], Zhang et al. [8],...
BDN [9], ERRNet [5], IBCLN [12], Kim et al. [11], Chang et al. [16], Peng et al. [15], and the modified MIRR method [13] by only taking the observation \( I \) as input. We finetune these models on our training dataset and report the better result between the finetuned version and the released one. For Lei et al. [13], we adjust the network input and retrain the model with our training set.

The PSNR and SSIM metrics of all competing methods are reported in Table 2, from which one can see that the proposed RAGNet surpasses the other competing methods on average quantitative performance by a large margin (∼0.7dB in PSNR). The quantitative gain against other deep cascaded models [4, 9, 12, 13] also indicates the benefit of appropriate reflection-aware guidance. Figure 4 shows the qualitative results and the ground-truth on three SIR2 subsets and Real 20 dataset, and the results on Real 45 dataset are given in Fig. 5. CEILNet may fail when the reflections have clear edges, where the smoothness assumption is violated (e.g., the second row of Fig. 4). Zhang et al. [8] is prone to over-processing, which generates dim results with color aberration in most cases. BDN [9] and Peng et al. [15] tend to generate brighter output images than input ones. Moreover, some reflections are insufficiently removed in the results of ERRNet [5], Kim et al. [11], Chang et al. [16] and IBCLN [12]. Generally, RAGNet outperforms the competing methods by removing the reflection regions more accurately and thoroughly.

Additionally, we compare the visual results from Nature dataset with other compared methods in Fig. 6. When the image contains strong reflections or reflections of light sources, the compared methods always fail to remove the reflections. In contrast, RAGNet is able to remove the reflections in these scenarios.
4.3 Discussion on soft partial convolution and strong reflection

Here we elaborate on the difference between the soft partial convolution and the original one in classical image inpainting problem. As for image inpainting, partial convolution is required to fill missing regions without any information at all, and binary 0/1 mask is utilized. Therefore, massive training samples from the same categories or scenarios are required for image inpainting. In contrast, most reflection images contain visual information in the reflection regions. Therefore, soft partial convolution is adopted in RAG module, where the values in mask range from 0 to 1. In this paper, we define the information employed for soft partial convolution as the feature difference between input and reflection, i.e., $F_{\text{diff}} = F_I - F_R$. That is to say, our RAGNet can exploit useful information from $F_{\text{diff}}$ to benefit reflection removal. Thus, the soft partial convolution in RAGNet takes effect not only in strong reflection regions, but also in other regions with mild or weak reflection. As shown in the last row of Fig. 4, reflection intensities are spatially variant, and our method can well handle it. Especially, the texture of rock is much better recovered by our method in comparison to other SOTA methods.

As for very strong reflections, We give two real-world examples in Fig. 7. It can be seen that the reflections are successfully detected and removed by our method, whereas ERRNet fails in the first example and IBCLN fails in both examples. One may notice that for the regions with very strong reflections, the structures are not well recovered, e.g., Planks in first image and Black lines in second image. This is because that reflection intensities are so strong that nearly no visual hint can be exploited from $F_{\text{diff}} = F_I - F_R$, leaving it as an image inpainting problem. If we want to handle these cases well, massive training samples from the same categories or scenarios are needed during the training stage.

4.4 User study

Due to the lack of the ground-truth in Real45 dataset, a user study is additionally conducted to evaluate the visual quality of reflection removal. Our method is compared with five other methods, including ERRNet [5], IBCLN [12], Chang et al. and Peng et al. which perform better in terms of PSNR/SSIM metrics, and Zhang et al. [8] which performs well on Real20 dataset. We randomly select 20 image pairs from Real45 dataset. 30 participants are invited for the user study and each of them is given 20 questions. Each question contains the original input image and four choices, which are the results of the four candidate methods. The choices are arranged in a random order for a fair comparison. The users are instructed to choose the best result in terms of reflection removal ability and image quality. The results are reported in Table 3. It can be seen that our RAGNet has a much higher probability to be chosen as the best results among the four methods.

| Method   | Zhang et al. | ERRNet | IBCLN | Chang et al. | Peng et al. | Ours   |
|----------|--------------|--------|-------|--------------|-------------|--------|
| Real 45  | 18.30%       | 16.25% | 15.08%| 19.01%       | 4.08%       | 27.30% |

The best and the second-best results are highlighted with red and blue, respectively.
### Table 4  Comparison on running time between different methods

| Method        | CEILNet | Zhang | BDN   | ERRNet | IBCLN | Chang et al. | Peng et al. | Ours  |
|---------------|---------|-------|-------|--------|-------|--------------|-------------|-------|
| Time (s)      | 0.26    | 0.16  | 0.22  | 0.27   | 0.22  | 0.06         | 0.72        | 0.15  |

### Table 5  Comparison with state-of-the-art methods on regions with weak and strong reflection on the 4 test sets

| Reflection   | CEILNet | Zhang et al. | BDN   | ERRNet | IBCLN   | Ours     |
|--------------|---------|--------------|-------|--------|---------|----------|
| Intensity    | [4]     | [8]          | [9]   | [5]    | [12]    |          |
| Weak         | 21.51   | 19.38        | 21.86 | 23.79  | 24.28   | 25.18    |
| Strong       | 21.01   | 20.29        | 21.14 | 21.32  | 21.43   | 23.40    |

The PSNR for weak and strong reflection regions are respectively listed in the first and second row. The best and the second-best results are highlighted with red and blue, respectively.

### Table 6  Quantitative comparison on structure and loss functions

| Model                  | Real 20 | SIR$^2$ Wild |
|------------------------|---------|--------------|
|                        | PSNR / SSIM | PSNR / SSIM  |
| RAGNet$_{I \rightarrow T}$ | 20.73 / 0.753 | 24.37 / 0.871 |
| w/o $F_{diff}$         | 20.99 / 0.758 | 24.57 / 0.865 |
| w/o $L_{mask}$         | 21.32 / 0.764 | 25.09 / 0.878 |
| w/o $L_{reg}$          | 22.08 / 0.769 | 25.09 / 0.877 |
| w/o $L_{reg}^{mask}$   | 22.11 / 0.773 | 25.12 / 0.876 |
| RAGNet                 | 22.95 / 0.793 | 25.52 / 0.880 |

---

**Fig. 8**  Structure of the single-stage network RAGNet$_{I \rightarrow T}$, which directly estimates the transmission layer from the observation $I$. RAGNet$_{I \rightarrow T}$ follows the U-Net setting, whereas the concatenated feature through the skip-connection is $F_{diff} = F_I - F_{dec}$ rather than $F_I$. 

---

© Springer
4.5 Running time

The running time of RAGNet and the other competing methods are calculated and summarized in Table 4. It turns out that RAGNet consumes 0.15s per image (in 540×400 resolution) on an Nvidia RTX 2080Ti GPU, which ranks 2nd in speed among all the competing methods.

4.6 Ability of removing strong and weak reflection

While our RAGNet is particularly good at handling strong reflection regions, it also works well on the regions with weak reflection. To verify this point, we calculate the PSNR in terms of the weak and strong reflection regions respectively on the 4 test sets (SIR\textsuperscript{2} and Real 20 datasets). As shown in Fig. 3, the elements of \( M_{\text{diff}} \) tend to be 0 in strong reflection regions and 1 in weak reflection regions. Therefore, we can separately evaluate the network performance by calculating PSNR in regions with weak reflection based on \( M_w \). The PSNR for weak reflection regions is calculated as:

\[
\text{PSNR} = 10 \log_{10} \frac{I_{\text{max}}}{\text{MSE}_{\text{mask}}},
\]

where \( I_{\text{max}} \) is the maximum possible pixel value. Accordingly, the PSNR for strong reflection regions can be calculated by replacing \( M_w(i, j) \) with \( M_s(i, j) = 1 - M_w(i, j) \). The results are shown in Table 5. It can be seen that: 1) For weak / no reflection regions, our method can achieve 0.9 dB PSNR gain over the second best competitor. 2) For strong reflection regions, the PSNR by our method exceeds that of the second best method by 1.9 dB.

4.7 Ablation study

In this section, we perform ablation studies on two outdoor real-world datasets, i.e., Real 20 and SIR\textsuperscript{2} Wild.

4.7.1 Two-stage architecture

Our RAGNet is designed as a two-stage network, which takes reflection as an intermediate result to ease the reflection removal procedure. To show the necessity of the two-stage architecture, we additionally design a single-stage variant named RAGNet\(_{I \rightarrow T}\), which directly estimates the transmission layer from the input. The detailed structure of RAGNet\(_{I \rightarrow T}\) is shown in Fig. 8.

According to Fig. 2, the generation of \( T \) in RAGNet requires features from two sources, i.e., \( M_{\text{diff}} \circ F_{\text{diff}} \) and \( M_{\text{dec}} \circ F_{\text{dec}} \). We expect that \( M_{\text{diff}} \rightarrow 0 \) when reflection is strong as in (2), so that only \( F_{\text{dec}} \) provides information for recovering strong reflection regions. When designing RAGNet\(_{I \rightarrow T}\), we retain the functionality of the skip connection and \( F_{\text{dec}} \): one for providing auxiliary information via subtraction operations, the other focuses on

### Table 7: Quantitative comparison on mask utilization schemes

| Model       | Real 20 PSNR / SSIM | SIR\textsuperscript{2} Wild PSNR / SSIM |
|-------------|----------------------|----------------------------------------|
| RAGNet\(_F\) | 20.93 / 0.750        | 24.47 / 0.875                          |
| RAGNet\(_F \odot M\) | 21.60 / 0.772      | 24.86 / 0.875                          |
| RAGNet\(_F \odot M \circ\) | 21.69 / 0.774      | 24.69 / 0.871                          |
| RAGNet       | 22.95 / 0.793        | 25.52 / 0.880                          |
gradually recovering $\hat{T}$. The results in Table 6 and Fig. 9 show that the one-stage variant RAGNet$_{I\rightarrow T}$ suffers a huge performance degradation, indicating the essentiality of the two-stage architecture for our RAGNet.

4.7.2 RAG module

We discuss the effectiveness of key components in RAG module.

**Difference features** To evaluate the effectiveness of the difference features, we replace $F_{\text{diff}}$ with $F_I$ and keep the other parts of the model unchanged, i.e., the subtraction operation is discarded. As shown in Fig. 9, without $F_{\text{diff}}$, the model performs poorly in suppressing the reflection, leading to an obvious performance drop in Table 6.

**Masks** In order to verify the settings about masks, we conduct ablation studies on three variants, (i) RAGNet$_F$: the masks are totally discarded, and the soft partial convolution is accordingly replaced by vanilla convolution. (ii) RAGNet$_{F\circ M}$: the masks are multiplied with the feature $F$. In other words, the renormalization operation in soft partial convolutions is removed. (iii) RAGNet$_{F\circ M^c}$: a unified one-channel mask is predicted for $F_{\text{diff}}$ and $F_{\text{dec}}$ respectively, i.e., $M^c_{\text{diff}}$ and $M^c_{\text{dec}}$, where $M^c = [M^c_{\text{diff}}, M^c_{\text{dec}}]$.

As shown in Table 7, RAGNet outperforms RAGNet$_F$ and RAGNet$_{F\circ M}$ in both PSNR and SSIM, indicating the effectiveness of masks and soft partial convolution. Furthermore, the ablation study on RAGNet$_{F\circ M^c}$ shows that predicting a per-channel mask better suits the SIRR task in our framework.

Figure 10 shows the qualitative results of the ablation studies. It can be seen that all of RAGNet$_F, RAGNet_{F\circ M}$ and RAGNet$_{F\circ M^c}$ fail to remove the strong reflections, especially in complex environments (e.g., in the second row of Fig. 10, when the light on the ground is very similar to the reflections). RAGNet outperforms the others by recovering the strong reflection regions with contextual information, making the output image visually pleasant. Moreover, by integrating soft partial convolution, the reflection removal performance is also enhanced for images with mild reflections.
Two-stage single image reflection...  

4.7.3 Mask loss

To show the effectiveness of our mask loss, we conduct three experiments, i.e., discarding $L_{\text{diff}}$, $L_{\text{reg}}$, and both of them. Table 6 and Fig. 9 show that with the mask loss, the performance of our RAGNet is improved by a large margin. Furthermore, one can see that both items of the specially designed mask loss are essential for RAGNet by boosting the performance quantitatively and qualitatively. The thresholds $\xi$ and $\varphi$ in (2) and (3) are discussed in the ablation study.

4.7.4 Hyperparameters $\varphi$ and $\xi$

The parameters $\varphi$ and $\xi$ are the thresholds that define the intensity levels of reflection, i.e., regions with reflection intensity higher than $\varphi$ are defined as strong reflection regions, while regions with reflection intensity lower than $\xi$ are defined as weak reflection regions. We conduct an ablation study on $\varphi$ and $\xi$ on Real 20 and SIR2 test datasets with two groups of values, i.e., $\varphi = \{0.20, 0.25, 0.30, 0.35, 0.40\}$ and $\xi = \{0.005, 0.010, 0.015, 0.020\}$. Considering the PSNR and SSIM performance exhibited in Fig. 11, the final thresholds are set as $\varphi = 0.3$ and $\xi = 0.01$.

4.7.5 Robustness against estimated reflection

Empirically, the performance of our network is dependent on the quality of the estimated reflection $\hat{R}$. Thus, we make an investigation into the robustness of RAGNet against $\hat{R}$. In specific, we change the quality of $\hat{R}$ by training $G_R$ for different epochs and observe how the final performance is influenced. As shown in Fig. 12, only slight fluctuation of network performance is yielded given $\hat{R}$ generated in different epochs, indicating the decent robustness of RAGNet against $\hat{R}$.

5 Conclusion

Considering the difficulty in directly predicting the transmission layer from input, cascaded deep models have been developed to estimate the transmission in a progressive manner. However, the transmission estimation quality of most existing methods is limited because of insufficient exploitation of the prior stage’s output. In this paper, we proposed a two-stage framework for single image reflection removal task, and presented a RAGNet to exploit the reflection-aware guidance via the RAG module. The RAG module suppresses the reflection by using the difference between observation and reflection features. A mask is generated to indicate the extent to which the transmission is corrupted by strong reflection and to collaborate with soft partial convolution for mitigating the effect of deviating from the linear combination hypothesis. A dedicated mask loss is accordingly designed for reconciling the contributions of encoder and decoder features. Extensive experiments on five widely used real-world datasets indicate that the proposed RAGNet outperforms the state-of-the-art methods by a large margin. Nonetheless, RAGNet can still be promoted in terms of processing complicated images, for which the reflection regions can be sometimes inaccurately detected. In the future, we will delve into the physical mechanism of reflection formation and develop more effective methods for reflection removal.

Acknowledgements This work was partially supported by National Natural Science Foundation of China (Nos. 62172127 and U22B2035), Natural Science Foundation of Heilongjiang Province (No. YQ2022F004).

Data availability statement The datasets used during the training are available in http://host.robots.ox.ac.uk/pascal/VOC/ and https://github.com/ceciliavision/perceptual-reflection-removal. The datasets used during the testing are available in https://github.com/fqnchina/CEILNet and https://sir2data.github.io/.

References

1. Pal SK, Pramanik A, Maiti J, Mitra P (2021) Deep learning in multi-object detection and tracking: state of the art. Appl Intell 51(9):6400–6429
2. Yang J, Ge H, Yang J, Tong Y, Su S (2022) Online multi-object tracking using multi-function integration and tracking simulation training. Appl Intell 52(2):1268–1288
3. Punnappurath A, Brown MS (2019) Reflection removal using a dual-pixel sensor. In: IEEE conference on computer vision and pattern recognition, pp 1556–1565
4. Fan Q, Yang J, Hua G, Chen B, Wipf D (2017) A generic deep architecture for single image reflection removal and image smoothing. In: IEEE international conference on computer vision, pp 3238–3247
5. Wei K, Yang J, Fu Y, Wipf D, Huang H (2019) Single image reflection removal exploiting misaligned training data and
network enhancements. In: IEEE conference on computer vision and pattern recognition, pp 8178–8187
6. Asif M, Chen L, Song H, Yang J, Frangi AF (2021) An automatic framework for endoscopic image restoration and enhancement. Appl Intell 51(4):1959–1971
7. Chang Y, Jung C (2018) Single image reflection removal using convolutional neural networks. IEEE Trans Image Process 28(4):1954–1966
8. Zhang X, Ng R, Chen Q (2018) Single image reflection separation with perceptual losses. In: IEEE conference on computer vision and pattern recognition, pp 4786–4794
9. Yang J, Gong D, Liu L, Shi Q (2018) Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal. In: European conference on computer vision, pp 654–669
10. Heydecker D, Maierhofer G, Aviles-Rivero AI, Fan Q, Chen D, Schönhöfer C-B, Stürsumpf S (2019) Mirror, mirror, on the wall, who’s got the clearest image of them all?—a tailored approach to single image reflection removal. IEEE Trans Image Process 28(12):6185–6197
11. Kim S, Huo Y, Yoon S-E (2020) Single image reflection removal with physically-based training images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5164–5173
12. Li C, Yang Y, He K, Lin S, Hopcroft JE (2020) Single image reflection removal through cascaded refinement. In: IEEE conference on computer vision and pattern recognition, pp 3565–3574
13. Lei C, Huang X, Zhang M, Yan Q, Sun W, Chen Q (2020) Polarized reflection removal with perfect alignment in the wild. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1750–1758
14. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
15. Peng Y-T, Cheng K-H, Fang I-S, Peng W-Y, Wu J-S (2022) Single image reflection removal based on knowledge-distilling content disentanglement. IEEE Signal Process Lett 29:568–572
16. Chang Y-C, Lu C-N, Cheng C-C, Chiu W-C (2021) Single image reflection removal with edge guidance, reflection classifier, and recurrent decomposition. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 2033–2042
17. Agrawal A, Raskar R, Nayar S, K, Li Y (2005) Removing photograph artifacts using gradient projection and flash-exposure sampling. In: ACM SIGGRAPH, pp 825–835
18. Fu Y, Lam A, Sato I, Okabe T, Sato Y (2013) Separating reflective and fluorescent components using high frequency illumination in the spectral domain. In: IEEE international conference on computer vision, pp 457–464
19. Schechner Y, Y. Kiryati N, Basri R (2000) Separation of transparent layers using focus, pp 25–39
20. Sarel B, Irani M (2004) Separating transparent layers through layer information exchange. In: European conference on computer vision, pp 328–341
21. Wieschollek P, Gallo O, Gu J, Kautz J (2018) Separating reflection and transmission images in the wild. In: European conference on computer vision, pp 89–104
22. Xue T, Rubinstein M, Liu C, Freeman WT (2015) A computational approach for obstruction-free photography. ACM Trans Graph 34(4):1–11
23. Szeliski R, Avidan S, Anandan P (2000) Layer extraction from multiple images containing reflections and transparencies. In: IEEE conference on computer vision and pattern recognition, pp 246–253
24. Li Y, Brown MS (2013) Exploiting reflection change for automatic reflection removal. In: IEEE international conference on computer vision, pp 2432–2439
25. Guo X, Cao X, Ma Y (2014) Robust separation of reflection from multiple images. In: IEEE conference on computer vision and pattern recognition, pp 2187–2194
26. Gai K, Shi Z, Zhang C (2012) Blind separation of superimposed moving images using image statistics. IEEE Trans Pattern Anal Mach Intell 34(1):19–32
27. Sinha SN, Kopf J, Goesele M, Scharstein D, Szeliski R (2012) Image-based rendering for scenes with reflections. ACM Trans Graph 31(4):1–10
28. Yang J, Li H, Dai Y, Tan RT (2016) Robust optical flow estimation of double-layer images under transparency or reflection. In: IEEE conference on computer vision and pattern recognition, pp 1410–1419
29. Sun C, Liu S, Yang T, Zeng B, Wang Z, Liu G (2016) Automatic reflection removal using gradient intensity and motion cues. In: ACM international conference on multimedia, pp 466–470
30. Han B-J, Sim J-Y (2017) Reflection removal using low-rank matrix completion. In: IEEE conference on computer vision and pattern recognition, pp 5438–5446
31. Cheong JY, Simon C, Kim C-S, Park IK (2017) Reflection removal under fast forward camera motion. IEEE Trans Image Process 26(12):6061–6073
32. Simon C, Kyo Park I (2015) Reflection removal for in-vehicle black box videos. In: IEEE conference on computer vision and pattern recognition, pp 4231–4239
33. Li T, Chan Y-H, Lun DP (2020) Improved multiple-image-based reflection removal algorithm using deep neural networks. IEEE Trans Image Process 30:68–79
34. Han B-J, Sim J-Y (2018) Glass reflection removal using co-saliency-based image alignment and low-rank matrix completion in gradient domain. IEEE Trans Image Process 27(10):4873–4888
35. Levin A, Weiss Y (2007) User assisted separation of reflections from a single image using a sparsity prior. IEEE Trans Pattern Anal Mach Intell 29(9):1647–1654
36. Wan R, Shi B, Duan L-Y, Tan A-H, Gao W, Kot AC (2018) Region-aware reflection removal with unified content and gradient priors. IEEE Trans Image Process 27(6):2927–2941
37. Arvanitopoulos N, Achanta R, Susstrunk S (2017) Single image reflection suppression. In: IEEE conference on computer vision and pattern recognition, pp 4498–4506
38. Sandhan T, Young Choi J (2017) Anti-glare: tightly constrained optimization for eyeglass reflection removal. In: IEEE conference on computer vision and pattern recognition, pp 1241–1250
39. Li Y, Brown MS (2014) Single image layer separation using relative smoothness. In: IEEE conference on computer vision and pattern recognition, pp 2752–2759
40. Wang R, Shi B, Tan A-H, Kot AC (2016) Depth of field guided reflection removal. In: IEEE international conference on image processing, pp 21–25
41. Shih Y, Krishnan D, Durand F, Freeman W, T (2015) Reflection removal using ghosting cues. In: IEEE conference on computer vision and pattern recognition, pp 3193–3201
42. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556
43. Dong Z, Xu K, Yang Y, Bao H, Xu W, Lau RW (2021) Location-aware single image reflection removal. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 5017–5026
44. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention, pp 234–241
45. Chen Y, Zhang H, Liu L, Chen X, Zhang Q, Yang K, Xia R, Xie J (2021) Research on image inpainting algorithm of improved gan based on two-discriminations networks. Appl Intell 51(6):3460–3474
46. Zhang J, Liu Y, Guo C, Zhan J (2022) Optimized segmentation with image inpainting for semantic mapping in dynamic scenes. Appl Intell, pp 1–16
47. Johnson J, Alahi A, Fei-Fei L (2016) Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision, pp 694–711
48. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
49. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vis 88(2):303–338
50. Wan R, Shi B, Duan L-Y, Tan A-H, Kot AC (2017) Benchmarking single-image reflection removal algorithms. In: IEEE International Conference on Computer Vision, pp 3922–3930
51. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv:1412.6980
52. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L (2019) Pytorch: an imperative style, high-performance deep learning library. In: Advances in neural information processing systems, pp 8026–8037

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Yu Li received the M.E. degree from Harbin Institute of Technology, Harbin, China, in 2019. Currently, she is working towards the Ph.D. degree in computer science and technology at Harbin Institute of Technology. Her research interests are computer vision and deep learning, particularly focusing on reflection removal and image deblurring.

Ming Liu received the B.Sc. degree in computer science and technology from the Harbin Institute of Technology, Harbin, China, in 2017. He is currently working under the joint Ph.D. programmes in the School of Computer Science and Technology, Harbin Institute of Technology, and the Department of Computing, the Hong Kong Polytechnic University. His research interests include image restoration and deep learning.

Yaling Yi received the B.Eng degree in computer science and technology from Harbin Institute of Technology, in 2021. She is currently working toward the M.S. degree in the Faculty of Computing, HIT. She was an Intern with Megvii Tech in 2022. Her research interests include image deblurring and super-resolution.

Qince Li got his Ph.D. degree in Biophysics from the University of Manchester in 2013. He received the postdoctoral training from the University of Alabama at Birmingham from 2014 to 2015. He is currently an associate professor with the Faculty of Computing, Harbin Institute of Technology. His research interests include computational modeling of biological system, virtual physiological heart, medical image processing and ECG signal processing.
Dongwei Ren received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2017. From 2018 to 2021, he was an Assistant Professor with the College of Intelligence and Computing, Tianjin University. He is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology. His research interests include computer vision and deep learning.

Wangmeng Zuo (Senior Member, IEEE) received the Ph.D. degree in computer application technology from Harbin Institute of Technology, Harbin, China, in 2007. From 2004 to 2006, he was a Research Assistant with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. From 2009 to 2010, he was a Visiting Professor with Microsoft Research Asia. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology. He has published over 60 papers in top-tier academic journals and conferences. His current research interests include image enhancement and restoration, weakly supervised learning, visual tracking, and image classification. He has served as a Tutorial Organizer in ECCV 2016, an Associate Editor of the IET Biometrics, and the Guest Editor of Neurocomputing, Pattern Recognition, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.