6-2019

CoRRN: Cooperative Reflection Removal Network

Renjie WEN

Boxin SHI

Haoliang LI

Ling-Yu DUAN

Ah-hwee TAN
Singapore Management University, ahtan@smu.edu.sg

See next page for additional authors

Citation

WEN, Renjie; SHI, Boxin; LI, Haoliang; DUAN, Ling-Yu; TAN, Ah-hwee; and KOT, Alex C.. CoRRN: Cooperative Reflection Removal Network. (2019). *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1-14.
Available at: https://ink.library.smu.edu.sg/sis_research/5178

CoRRN: Cooperative Reflection Removal Network

Renjie Wan, *Student Member, IEEE,* Boxin Shi, *Member, IEEE,* Haoliang Li, *Student Member, IEEE,* Ling-Yu Duan, *Member, IEEE,* Ah-Hwee Tan, *Member, IEEE,* and Alex C. Kot, *Fellow, IEEE*

Abstract—Removing the undesired reflections from images taken through the glass is of broad application to various computer vision tasks. Non-learning based methods utilize different handcrafted priors such as the separable sparse gradients caused by different levels of blurs, which often fail due to their limited description capability to the properties of real-world reflections. In this paper, we propose a network with the feature-sharing strategy to tackle this problem in a cooperative and unified framework, by integrating image context information and the multi-scale gradient information. To remove the strong reflections existed in some local regions, we propose a statistic loss by considering the gradient level statistics between the background and reflections. Our network is trained on a new dataset with 3250 reflection images taken under diverse real-world scenes. Experiments on a public benchmark dataset show that the proposed method performs favorably against state-of-the-art methods.

Index Terms—Reflection removal, deep learning, statistic loss, cooperative framework.

1 Introduction

Reflections observed in front of the glass significantly degrade the visibility of the scene behind the glass. By obstructing, deforming or blurring the background scene, reflections cause many computer vision systems likely to fail. Reflection removal aims at enhancing the visibility of the background scene while removing the reflections.

Reflection removal is challenging due to its obviously ill-posed nature – the number of unknowns is twice the number of equations. Besides, the similarities between the properties of the background and reflections make it more difficult to simultaneously remove the reflections and restore the contents in the background. To solve this problem, many non-learning based reflection removal methods are proposed. They rely on the handcrafted image priors observed under special circumstances, *e.g.*, the gradient prior for different blur levels between background and reflection [1], [2], [3], the ghosting effects [4], the non-local content similarity [5], and so on.

While these methods can reasonably solve the problem when their assumptions are satisfied, the specific priors they rely on are often violated in real-world scenarios, since the low-level image priors only describe a limited range of the reflection properties and may project the partial observation as the whole truth. For example, when the structures and patterns of the background are similar to those of the reflections, the non-learning based methods have difficulty in simultaneously removing reflections and recovering the background [9]. On the other hand, many non-learning based methods [1], [10] adopt a two-stage pipeline, where they first locate the reflection regions (*e.g.*, by classifying the background and reflection edges [1], [10]) and then restore the background layers based on the edge information using the method proposed by Levin *et al.* [3], [11]. However, locating reflection regions itself is a very challenging task, so existing methods mainly rely on some heuristic observations [1], or have to involve user interaction [3], which are not applicable in many scenarios.

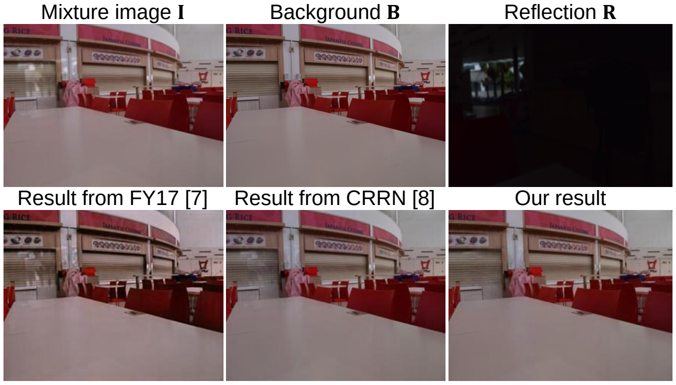

Fig. 1: An example of the image triplet with mixture image $\mathbf{I}$, background $\mathbf{B}$, and reflection $\mathbf{R}$ from 'SIR^2' dataset [6] and the reflection removal results obtained by using FY17 [7], CRRN [8], and our method.

To capture the reflection properties more comprehensively, recent methods start to explore deep learning techniques to solve this problem [7], [12]. The CEINet proposed by Fan *et al.* [7] is the seminal work that has introduced an end-to-end framework to solve the reflection removal problem for the first time.

- *This research was carried out at the Rapid-Rich Object Search (ROSE) Lab at the Nanyang Technological University, Singapore and is supported by the National Research Foundation, Prime Ministers Office, Singapore, under the NRF-NSFC grant NRF2016NRF-NSFC001-098. Boxin Shi is supported in part by National Natural Science Foundation of China under Grant No. 61872012. This work was supported in part by the National Natural Science Foundation of China under Grant 61661146005 and Grant U1611461.*

- *Renjie Wan, Haoliang Li, and Alex C. Kot is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore.*
- *Boxin Shi and Ling-Yu Duan are with the Institute of Digital Media, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China.*
- *Ah-Hwee Tan is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, Singapore*
- *Corresponding authors: Renjie Wan (e-mail: rjwan@ntu.edu.sg) and Ling-Yu Duan (email: lingyu@pku.edu.cn)*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2921574, IEEE Transactions on Pattern Analysis and Machine Intelligence

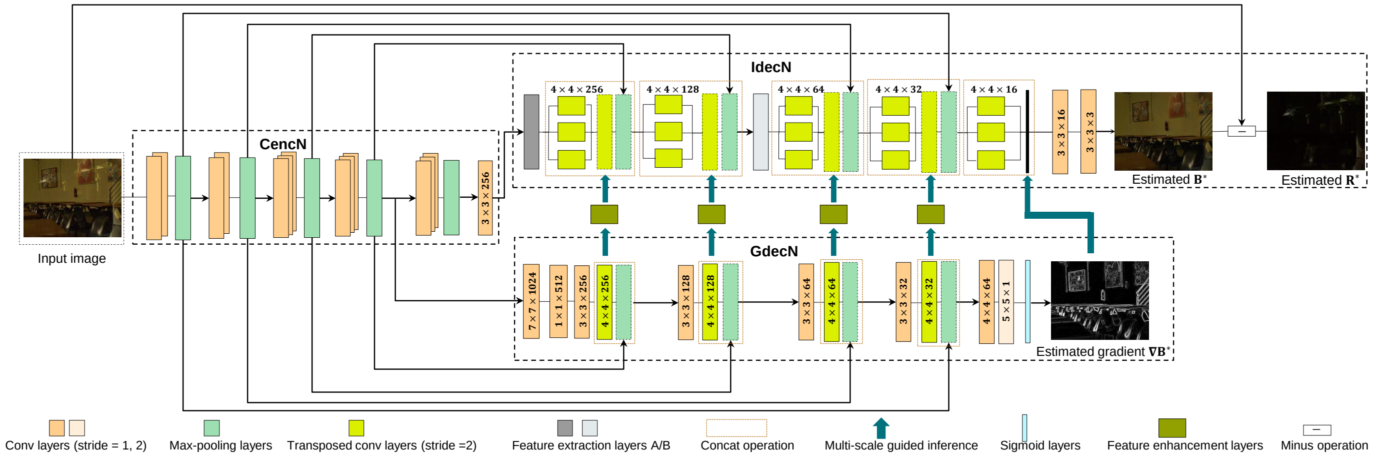

Fig. 2: The framework of CoRRN. It consists of three sub-networks to estimate the background and reflections in a cooperative manner. The IdecN is closely guided by the associated gradient features from GdecN with the same resolution at different upsampling stages.

Benefiting from the deep learning framework, they have shown improved modeling ability that captures a variety of reflection image characteristics [9], [13]. However, they still adopt the two-stage pipeline for gradient inference and image inference as many non-learning based methods [1], [3], which do not fully explore the multi-scale information for background recovery. Moreover, they mainly rely on the pixel-wise losses ($\mathcal{L}_2$ and $\mathcal{L}_1$ loss), that may generate blurry predictions [14], [15], [16]. Last but not least, the two methods [7], [12] are mainly trained with synthetic images which can never capture the comprehensive information in real-world image formation process completely.

To address the above issues, CRRN [8] proposes a concurrent framework to make use of the multi-scale information and introduces a perceptually motivated loss to suppress the blurring artifacts. However, in CRRN [8], the two concurrent network owns two independent but similar encoder networks, which cannot efficiently seek common ground and reserve differences across different tasks. On the other hand, though CRRN [8] introduces a framework guided by the multi-scale gradient features, the gradient features from the initial stage still contain artifacts caused by reflections, which result in the residual edges for the final estimated results. At last, as shown in Figure 1, though the perceptually motivated loss enhances the visual quality of the final estimated result, it is still difficult for CRRN [8] to remove the locally strong reflections due to the ignorance of the inherent relationship between the background and reflection.

In this work, in contrast to the conventional two-stage framework used in other methods [7], [12] and the concurrent model in CRRN [8], we propose the **Co**operative **R**eflection **R**emoval **N**etwork (CoRRN) by using one shared encoder network as illustrated in Figure 2. The proposed model consists of three cooperative sub-networks: the context encoder network (CencN) to extract contextual information, the gradient decoder network (GdecN) to estimate the gradient of the background, and the image decoder network (IdecN) to estimate the background and reflection. IdecN and GdecN share the feature information extracted by CencN. Such a feature sharing strategy is supposed to improve the learning efficiency and prediction accuracy [17]. On the other hand, we add the feature enhancement layers between IdecN and GdecN to partly suppress the reflection artifacts existing in the gradient features from GdecN. At last, besides the gradient constraints from GdecN, we introduce a new statistic loss for IdecN based on the gradient level statistics to better remove the locally strong reflections. Our major contributions are summarized as follows:

- We propose a unified reflection removal network with a multi-scale guided learning strategy. It is composed by three cooperative sub-networks with improved learning efficiency and prediction accuracy.
- We further introduce the gradient feature enhancement layers to suppress the reflection artifacts in the gradient features and design a statistic loss by considering the inherent relationship of the gradient level statistics to deal with the locally strong reflections.
- We capture a large-scale reflection image dataset to generate diverse and realistic training data, which improve the generality and practicability of our method.

As an extension of CRRN [8], CoRRN has better performances and generalization ability than CRRN [8] and the recent state-of-the-art methods [7], [18] based on the experimental results on a publicly available benchmark dataset with real-world images [6].

The remainder of this paper is organized as follows. Section 2 introduces relevant existing works. Section 3 and Section 4 introduce the preparation for the training dataset and our proposed method, respectively. Experimental results and discussions are prepared in Section 5. Finally, we conclude the paper in Section 6.

2 Related Work

2.1 Deep learning based image-to-image translation

In recent years, deep learning has achieved promising performances on image-to-image translation problems. Most current approaches design their method based on a paired model by using a dataset with input-output examples to learn a parametric translation function. For example, Cai *et al.* [19] proposed a dehazing method by using the maxout network. Yang *et al.* [20] proposed a multi-task learning framework to estimate the background and rain layer simultaneously. Recently, Qu *et al.* [21] also introduced an end-to-end and fully automatic framework to address the problems existed in the shadow removal problems. In addition, the paired model also shows its superiority in other different tasks, including image denoising [22] and super resolution [23].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2921574, IEEE Transactions on Pattern Analysis and Machine Intelligence

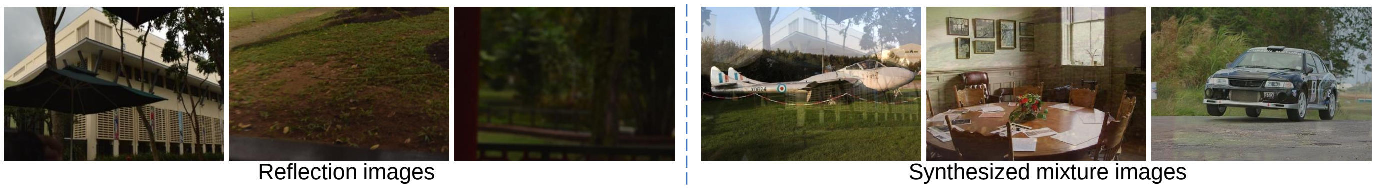

Fig. 3: Samples of captured reflection images in the 'RID' and the corresponding synthetic images using the 'RID'. From left to right column, we show the diversity of different illumination conditions, focal lengths, and scenes.

On the other hand, the unpaired settings are also introduced to solve these problems based on the generative adversarial network, when the training data with pixel-correspondence are difficult to obtain. For example, Qian *et al.* [24] designed a network to treat the rain streaks differently to model their distinctive characteristics. In this paper, we exploit a cooperative network to remove reflections by using the paired model.

2.2 Reflection removal

Reflection removal has been widely studied for several decades. Previous work can be roughly classified into two categories. The first category solves this problem using the non-learning based methods. Due to the ill-posed nature of this problem, different priors are employed to exploit the special properties of the background and reflection layers. For example, Levin *et al.* [3] adopted the sparsity priors to decompose the input image. However, their method relies on the users to label the background and reflection edges, which is quite labor-intensive and may fail in textured regions. Li *et al.* [2] made use of the different blur levels of the background and reflection layers. Nikolaos *et al.* [25] adopted the Laplacian data fidelity term to solve this problem. Shih *et al.* [4] used the GMM patch prior to remove reflections with the visible ghosting effects and they also show that the object recognition accuracy (*e.g.*, food, car, *etc.*) can be improved after the reflection removal. The handcrafted priors adopted by these methods are based on the observations of some special properties between the background and reflection (*e.g.*, different blur levels [1], [2]) which is often violated in the general scenes especially when these properties are weakly observed. Some other methods in this category remove reflections by using a set of images taken from different viewpoints [26], [27]. By exploiting the motion cues between the background and reflection from multiview captures and assuming the glass is closer to the camera, the projected motion of the two layers is different due to the visual parallax. The motion of each layer is represented by using parametric models, such as the translative motion [28], the affine transformation [27], and the homography [27]. Through the combination of the motion and traditional cues, the non-learning based methods using the multiple images as the input show more reliable results when the input data are appropriately prepared. However, the requirement for special facilities of capturing limits such methods for practical use, especially for mobile devices or images downloaded from the Internet.

Another category solves the problem by using the learning based methods. Since the deep learning has achieved promising results in both high-level and low-level computer vision problems, its comprehensive modeling ability also benefits reflection removal problems. For example, Paramanand *et al.* [12] proposed a two-stage deep learning approach to learn the edge features of the reflections by using the light field camera. The framework introduced by Fan *et al.* [7] exploited the edge information when training the whole network to better preserve the image details. Though the deep learning based methods better capture the image properties, the conventional two-stage framework they adopt as many non-learning based methods [1], [3], [11] ignores the inherent correlations, which also degrades their performances.

3 Dataset Preparation

3.1 Real-world reflection images for data-driven methods

Real-world image datasets play important roles in studying physics-based computer vision [29] and face anti-spoofing [30] problems. Although the reflection removal problem has been studied for several decades, publicly available datasets are rather limited. The data-driven methods need a large-scale dataset to learn the reflection image properties. As far as we know, 'SIR^2' [6] is the largest reflection removal image datasets, which provides approximately 500 image triplets composed of mixture, background, and reflection images, but its scale is still not sufficient for training a complicated neural network. Considering the difficulty in obtaining the image triplet like 'SIR^2', an alternative solution to the data size bottleneck is to use the synthetic image dataset. The recent deep learning based method [7] provides a reasonable way to generate the reflection images by taking the regional properties and blurring effects of the reflections into consideration to make their data similar to the images taken in the wild. However, the ignorance of other reflection image properties (*e.g.*, ghosting effects, various types of noise in the imaging pipeline) may degrade the training and thus limits its wide applicability to real-world scenes.

To facilitate the training of our proposed method for general compatibility on real data, we construct a large-scale **R**eflection **I**mage **D**ataset called '**RID**', which contains 3250 images in total. We then use the captured reflection images from the 'RID' to synthesize the input mixture images.

To collect reflection images, we use a Nikon D5300 camera configured with varying exposure parameters and aperture sizes under a fully manual mode to capture images in different scenes. The reflection images are taken by putting a black piece of paper behind the glass while moving the camera and the glass around, which is similar to what have been done in [6], [26].

The 'RID' has the following two major characteristics, with example scenes demonstrated in Figure 3:

- **Diversity.** We consider three aspects to enrich the diversity of the 'RID': 1) We take the reflection images at different illumination conditions to include both strong and weak reflections (the first row in Figure 3 left); 2) we adjust the focal lengths randomly to create different blur levels of reflection (the second row in Figure 3 left); 3) the reflection images are taken from a great diversity of both indoor and outdoor scenes, *e.g.*, streets, parks, inside of office buildings, and so on (the third row in Figure 3 left).

- **Scale.** The 'RID' has 3250 images in total with approximately 2000 reflection images from the bright scenes and other reflection images are from the relatively dark scenes to meet the request of data-driven methods.

3.2 Training data generation

The commonly used image formation model for reflection removal is expressed as:

$$\mathbf{I} = \mathbf{B} + \mathbf{R}, \tag{1}$$

where $\mathbf{I}$ is the mixture image, $\mathbf{B}$ is the background layer, and $\mathbf{R}$ is the reflection layer. The synthetic mixture image $\mathbf{I}$ used for training can be directly generated by adding the reflection image and the background image from the natural image dataset (*e.g.*, COCO dataset [31]) and the reflection image dataset. However, this setting makes the generated images too bright and unnatural, which may influence the generalization ability of the whole training dataset.

To address this issue, we employ a linearly additive weighting scheme by setting $\mathbf{B} = \alpha\mathbf{B}_o$ and $\mathbf{R} = \beta\mathbf{R}_o$, where $\mathbf{B}_o$ denotes the original background image from the natural image dataset, $\mathbf{R}_o$ denotes the reflection image from 'RID', and α and β are the mixing coefficients used to balance the combination of $\mathbf{B}_o$ and $\mathbf{R}_o$. During our training stage, we use the synthetic mixture image $\mathbf{I}$ as the input to the whole network, the weighted background image $\mathbf{B}$ and weighted reflection image $\mathbf{R}$ as the corresponding ground truth.

To ensure a sufficient amount of training data, α and β are randomly sampled in $[0.8, 1]$ and $[0.1, 0.5]$, respectively, and we further augment the generated image with two different operations: (1) *Rotation*: randomly rotate images by 90°, 180° or 270°; (2) *Flipping*: flip images horizontally with a probability of 0.5. In total, our training dataset includes 14754 images.

4 Proposed Method

In this section, we describe the design methodology of the proposed reflection removal network, the optimization using the proposed complementary loss functions, and the details for network training.

4.1 Network architecture

According to Equation (1), given the observed images with reflections $\mathbf{I}$, our task here is to estimate $\mathbf{B}$. Traditionally, reflection removal problems can be solved by a maximum-a-posteriori (MAP) estimation [5], [18] as follows:

$$\begin{aligned}\{\mathbf{B}^\star, \mathbf{R}^\star\} = \underset{\mathbf{B},\mathbf{R}}{\operatorname{argmin}}\, & L(\mathbf{I}|\mathbf{B}, \mathbf{R}, \sigma^2) + L_b(\mathbf{B}) \\ & + L_g(\nabla\mathbf{B}) + L_r(\mathbf{R}),\end{aligned} \tag{2}$$

where L_b, L_g, and L_r are the priors enforced on $\mathbf{B}$, $\nabla\mathbf{B}$, and $\mathbf{R}$, respectively. The multi-task learning framework in Equation (2) has been adopted by many previous methods [3], [18], [26] due to its ability in exploring the inherent correlations between the estimations of $\mathbf{B}$, $\mathbf{R}$, and $\nabla\mathbf{B}$. We embed such a multi-task learning framework into a cooperative model based on the U-Net structure [32]. Different from previous methods that mainly adopt the gradient priors [2] or content prior [18], our model learns the priors of $\mathbf{B}$, $\mathbf{R}$, and $\nabla\mathbf{B}$ from the large scale training dataset and then embed them into the estimation process implicitly.

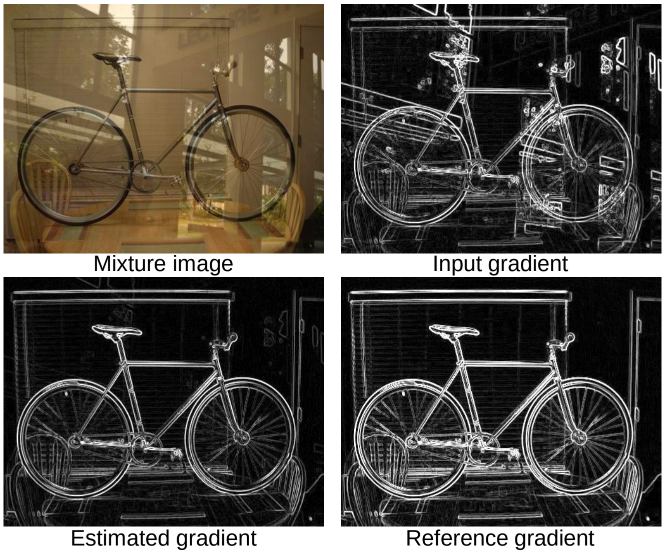
Mixture image
Input gradient
Estimated gradient
Reference gradient

Fig. 5: The estimated gradient generated by the gradient inference network, compared with the reference gradient obtained by using Sobel filter.

As shown in Figure 2, our model takes the mixture images as the input and contains three cooperative sub-networks: a context encoder network (CencN), an image decoder network (IdecN), and a gradient decoder network (GdecN). The three cooperative networks estimate $\mathbf{B}$ and $\mathbf{R}$ under the guidance of $\nabla\mathbf{B}$, which can be trained by using multiple loss functions based on their corresponding ground truths. Given the mixture image $\mathbf{I}$, the whole prediction process can be concluded as follows:

$$(\mathbf{B}^\star, \mathbf{R}^\star, \nabla\mathbf{B}^\star) = \mathcal{F}(\mathbf{I}, \theta), \tag{3}$$

where $\mathcal{F}$ is the network to be trained with θ consisting of all CNN parameters to be learned and $\mathbf{B}^\star$, $\mathbf{R}^\star$, and $\nabla\mathbf{B}^\star$ are the estimated results corresponding to their ground truth $\mathbf{B}$, $\mathbf{R}$, and $\nabla\mathbf{B}$.

To make each estimation tasks leverage information form other tasks, the three cooperative networks share the convolutional layers from each other. The whole network is constructed on the basis of CencN to extract the context information with the global structure and high-level semantic information of the scenes. For GdecN, it adopts the information from the shallower layers of CencN as the input and estimates $\nabla\mathbf{B}$. It can extract the image gradient information from the multiple scales and guide the whole image reconstruction process. IdecN takes the feature information from the deeper layer of CencN as the input and extracts background feature representations by using the multi-context information to estimate $\mathbf{B}$ and $\mathbf{R}$. The detailed architecture of CencN, GdecN, and IdecN will be introduced in the following sections.

4.1.1 Context encoder network (CencN)

CencN is on the basis of the VGG16 model [13] originally designed for the high-level computer vision tasks. VGG16 model [13] contains five convolutional blocks with thirteen 3×3 convolutional layers, five max-pooling layers, and three fully connected layers. As illustrated in Figure 2, we replace the fully connected layers at the last stage of VGG16 model [13] with a

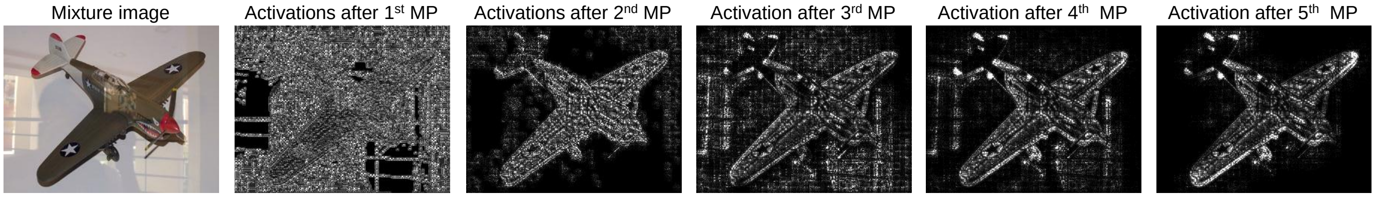

Fig. 4: Visualization of the gradient activations in our context encoder network after each max-pooling layer (MP). The activation maps from left to right correspond to the output maps from shallower to deeper layers in the context encoder network. Note that the activations from the reflections are suppressed through propagation while the activations closely related to the background are amplified. It shows that the learned filters in deeper convolutional layers tend to capture information related to the background while the details related to the reflection are kept in the shallower layers.

3×3 convolutional layer [21] in CencN to make it adapt to our image-to-image translation problems. From Figure 4, due to the substantially increasing receptive field size, CencN successfully suppresses the sparse reflection residues and extract the global information of the scenes from the shallower to deeper layers gradually. It can also facilitate the training of the successive networks.

4.1.2 Gradient decoder network (GdecN)

GdecN is designed to learn a mapping from $\mathbf{I}$ to $\nabla\mathbf{B}$. As illustrated in Figure 4, the shallower layer of CencN contains more local information related to small details of the whole images. To make a balance between the receptive field size and the available image information, we use the fourth layer of CencN as the input of GdecN to make full use of the image details to estimate the gradient. In GdecN, the features from CencN are upsampled and combined to reconstruct the output gradient without the reflection interference. In order to better preserve the sharp details and avoid the gradient vanishing problems, the features from CencN are linked to its corresponding layers in GdecN with the same spatial resolution. An example result shown in Figure 5 demonstrates that GdecN successfully removes the gradients from reflection and remain the gradient belonging to the background.

4.1.3 Image decoder Network (IdecN)

IdecN is a multi-task learning network to learn a mapping from $\mathbf{I}$ to $\mathbf{B}$ and $\mathbf{R}$. Since sparse reflection residues are gradually removed from shallower layers to deeper layers in CencN, we directly adopt the last layer of CencN as the input to IdecN to avoid the interference from the reflection. As shown in the part labeled as 'Feature extraction layers A/B' of Figure 2, IdecN consists of two feature extraction layers to extract multi-context and scale-invariant features and five transposed convolutional layers to upsample the feature maps gradually. We adopt the 'Reduction-A/B layers' from Inception-ResNet-v2 [33] as the 'Feature extraction layers A/B'[1] in IdecN. The two models are able to extract the scale-invariant and multi-context features due to its multi-size kernels [34]. However, they are seldom used in image-to-image problems because of its decimated features caused by pooling layers. To make it fit our problem, we make two modifications: First, the pooling layers in the original model are replaced by two convolutional layers with 1×1 and 7×7 filter sizes, respectively; second, the stride of all convolutions are decreased to 1. The transposed convolutional layers in this part have a parallel framework which is composed of three sub-layers, as shown in the part labeled as 'Transposed conv layers' in IdecN of Figure 2. Similar to GdecN, the feature maps from CencN are linked to its corresponding layers in IdecN with the same spatial resolution to avoid the gradient vanishing problem. In the last stage, due to the narrow intensity range of the residual $(\mathbf{I} - \mathbf{B})$ [9] and the close similarity between the input mixture images and output background images, we adopt the residual network to estimate the reflection images $\mathbf{R}$ by regarding it as the differences between $\mathbf{I}$ and $\mathbf{B}$, which increases the stability of the final estimation.

4.1.4 Multi-scale guidance

Gradient features are widely adopted in existing layer separation problems (*e.g.*, reflection removal [2] and intrinsic image decomposition [35], [36]) based on the observation that one layer is with larger gradient values and the other layer is with smaller gradient values. This observation suggests that the background pixels with larger gradient values and reflection pixels with smaller gradient values can be better differentiated in the gradient domain. Thus, we also embed such gradient priors into the image decoder network to increase the stability of the final solutions. Due to the multi-scale strategy used in our network, we extend the traditional single-scale gradient embedding scheme into a multi-scale scheme by concatenating the output of each transposed convolutional layers in GdecN with the output of transposed convolutional layers in IdecN at the same level, which is labeled as the 'Multi-scale guided inference' in Figure 2.

This multi-scale guidance strategy above has shown its superiority in CRRN [8]. However, we find that the feature maps from the GdecN still show obvious artifacts related to reflections if they were used directly, especially for those feature maps from the initial stage of GdecN. It may lead to the residual edges in the final estimated results. Inspired by the feature enhancement step proposed in [37], we add the feature enhancement layers with 7×7 kernel size between IdecN and GdecN to map the features with reflections to a feature space with relatively fewer reflections, which is labeled as the 'Feature enhancement layer' in Figure 2.

4.2 Loss function

Previous methods mainly adopt the pixel-wise loss function [7]. It is simple to calculate, but produces blurry predictions since it cannot capture the comprehensive property of the image distributions. To suppress the blurring artifacts and provide more visually pleasing results, we take the statistic similarity and human perception into considerations when designing our loss function.

[1] The details of the two layers can be found in the supplementary materials.

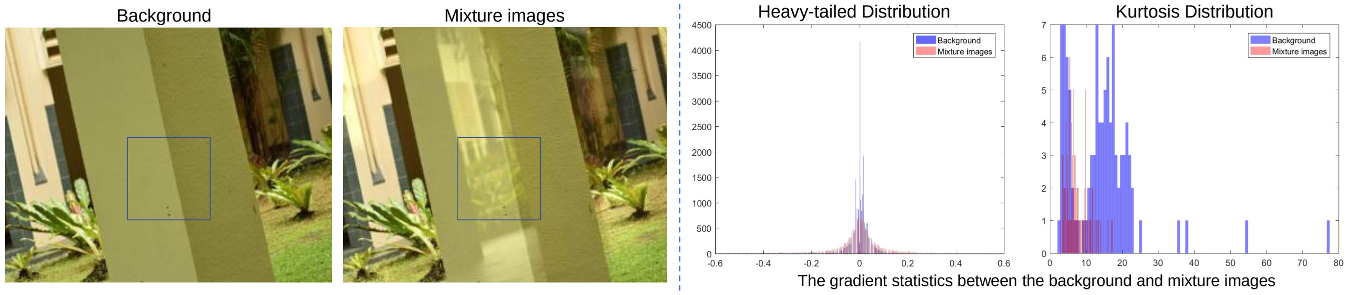

Fig. 6: An example of the background and its corresponding mixture images (left part) and the gradient level statistics comparison between the patches with and without reflection by using the heavy-tailed and kurtosis distribution (right part).

4.2.1 SSIM Loss

To generate the results consistent with the human perception. We adopt the perceptually motivated structural similarity index (SSIM) [38] to measure the similarity between the estimated $\mathbf{B}^\star$ and $\mathbf{R}^\star$ and their corresponding ground truth. SSIM is defined as

$$\mathbf{SSIM}(z, z^\star) = \frac{(2\mu_z\mu_{z^\star} + c_1)(2\sigma_{zz^\star} + c_2)}{(\mu_z^2 + \mu_{z^\star}^2 + c_1)(\sigma_z^2 + \sigma_{z^\star}^2 + c_2)}, \tag{4}$$

where c_1 and c_2 are the regularization constants, μ_z and $\mu_z^\star$ are the means of z and $z^\star$, σ_z and $\sigma_{z^\star}$ are the variances of z and $z^\star$, and $\sigma_{zz^\star}$ is their corresponding covariances. SSIM measures the similarity between two images from the luminance, the contrast, and the structure. To make the values compatible with the common settings of the loss function in deep learning, we define our loss function for IdecN as

$$\mathcal{L}^{\mathbf{SSIM}}(z, z^\star) = 1 - \mathbf{SSIM}(z, z^\star), \tag{5}$$

so that we can minimize it as that in the pixel-wise loss functions.

In GdecN, the luminance and contrast components in SSIM become undefined. We, therefore, omit the dependence of contrast and luminance in the original SSIM and define the loss function for GdecN as

$$\mathcal{L}^{\mathbf{SI}}(z, z^\star) = 1 - \mathbf{SI}(z, z^\star). \tag{6}$$

SI is used to measure the structural similarity between two images as demonstrated in [6], which is defined as

$$\mathbf{SI} = \frac{2\sigma_{zz^\star} + c_2}{\sigma_z^2 + \sigma_{z^\star}^2 + c_2}, \tag{7}$$

where all parameters share similar definitions as Equation (5).

4.2.2 Statistic Loss

Though the SSIM loss produces more visually pleasing results and suppresses the blurring artifacts, it is still difficult for them to deal with the locally strong reflections due to the ignorance of some essential differences between the images with and without reflections. As the example in Figure 1 shows, the results from CRRN [8] and FY17 [7] still display obvious residual edges, since they only consider pixel-wise loss or the SSIM loss.

One essential difference between the images with and without reflections can be found from their gradient level statistics. The general principle that gradient follows a heavy-tailed distributions has been known in this community for years [39]. By sampling image patches from the regions covered by reflections and their corresponding background, we plot their heavy-tailed distribution in Figure 6. Though their heavy-tailed distributions all have similar structures, the reflections widen the distributions and make the peak weaker since the strong reflections cover regions with rich textual information. Such phenomena have been utilized by previous methods [2], [3] by assuming the distributions with the same mean but different standard deviation values to discriminate the background and reflection. However, these lower-order statistics has limited ability to distinguish the reflection and background due to the large overlapping regions between the two probability distributions obtained by using this assumption. Thus, different from previous methods [1], [2] that mainly consider the lower-order statistics, we further take higher-order statistics into considerations. The higher-order statistics, such as the kurtosis, are mainly used to measure the deviation of a distribution. It is defined on the forth moments to describe the peakedness and shape of the heavy-tailed distribution, which can be described as follows:

$$K(\mathbf{a}) = \frac{\mathrm{E}[(\mathbf{a})^4]}{(\mathrm{E}[(\mathbf{a})^2])^2} - 3, \tag{8}$$

where $\mathrm{E}[\cdot]$ is the expectation operator for a data vector **a** and the -3 operator is to make normal-distribution kurtosis approach zero [39]. From Figure 6, since the reflections widen the heavy-tailed distributions, its corresponding kurtosis becomes smaller. The smaller overlapping regions in the kurtosis manifest the potential discriminative ability when applying it to our task.

We, therefore, propose a statistic loss to evaluate the similarity of the gradient level statistics between the estimated results and their corresponding ground truth by considering the lower- and higher-order moments on the basis of maximum mean discrepancy (MMD). As a kind of distribution divergence measurement derived from kernel embedding, MMD can measure the similarity of two distributions based on all-order moments as used in the two-sample testing problem [40], [41]. Given two images z and $z^\star$, MMD is defined as follows:

$$\begin{aligned} &\mathcal{L}^{\mathbf{MMD}}(z, z^\star) = \|\mathrm{E}[\phi(z)] - \mathrm{E}[\phi(z^\star)]\|^2 \\ &= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{i'=1}^{n} k(z_i, z_{i'}) + \frac{1}{m^2}\sum_{j=1}^{m}\sum_{j'=1}^{m} k(z_j^\star, z_{j'}^\star) \\ &- \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} k(z_i, z_j^\star), \end{aligned} \tag{9}$$

where i, i', j, and j' denote the pixel indices, m and n denote the number of elements in z and z^*, respectively, $k(\cdot,\cdot) = \phi(\cdot)\phi(\cdot)^\top$ is a Gaussian kernel, and $\phi(\cdot)$ is an implicit feature mapping [42], which can match different order moments of the statistics [43]. The example of ϕ can be found in the supplementary materials.

To build the statistic loss, we first translate the two input images z and $z^\star$ to their gradient domains as follows:

$$\begin{aligned} &p_{zx} = softmax([\nabla_x z]) \quad p_{zy} = softmax([\nabla_y z]), \\ &p_{z^\star x} = softmax([\nabla_x z^\star]) \quad p_{z^\star y} = softmax([\nabla_y z^\star]), \end{aligned} \tag{10}$$

where ∇_x and ∇_y denote the the derivative filters $[1, -1]$ and $[1, -1]^\top$, respectively, $[\cdot]$ denotes the vectorization operation, and the $softmax$ function normalizes the input into a tensor, where the summation of all elements equals to one. Then, our statistic loss is defined as follows:

$$\mathcal{L}^{\mathbf{SL}}(z, z^\star) = \mathcal{L}^{\mathbf{MMD}}(p_{zx}, p_{z^\star x}) + \mathcal{L}^{\mathbf{MMD}}(p_{zy}, p_{z^\star y}). \tag{11}$$

Finally, we observe that using SSIM loss or the statistic loss alone may cause changes of brightness and shifts of colors which makes the final results become dull [44], due to its insensitiveness to uniform bias, so we further introduce the $\mathcal{L}^{\mathbf{1}}$ loss for the background layer to better balance brightness and color.

Combining the above terms, our complete loss function becomes

$$\begin{aligned} \mathcal{L} =& \mathcal{L}^{\mathbf{SSIM}}(\mathbf{B}, \mathbf{B}^\star) + \gamma\mathcal{L}^{\mathbf{SL}}(\mathbf{B}, \mathbf{B}^\star) + \delta\mathcal{L}^{\mathbf{1}}(\mathbf{B}, \mathbf{B}^\star) \\ &+ \mathcal{L}^{\mathbf{SSIM}}(\mathbf{R}, \mathbf{R}^\star) + \epsilon\mathcal{L}^{\mathbf{SL}}(\mathbf{R}, \mathbf{R}^\star) + \mathcal{L}^{\mathbf{SI}}(\nabla\mathbf{B}, \nabla\mathbf{B}^\star), \end{aligned} \tag{12}$$

where γ, δ, and ϵ are the weighting coefficients and the coefficients for other three terms are all set to 1.

4.3 Implementation and training details

We have implemented our model using PyTorch[2]. In our training strategy, CencN is based on a pretrained VGG16 model [13], then it is connected with GdecN and IdecN, and the entire network is fine-tuned end-to-end, which grants the three sub-networks more opportunities to cooperate accordingly. The learning rate for the whole network training is initially set to 10^{-4} for the first 30 epochs and then decreases to 10^{-5} for the next 20 epochs.

Previous works that use deep learning to solve the inverse imaging problems [45], [46] or layer separation problems [20] mainly optimize the whole network on patches with resolution $n \times n$ cropped from the whole images. However, many real-world reflections only occupy some regions in an image like the regional 'noise' [6], we call it regional properties of reflections. Training with the patches without obvious reflections could potentially degrade the final performance. To avoid such negative effects, our model is trained using whole images with different sizes. We adopt a multi-size training strategy by feeding images of two sizes: coarse-scale 96×160 and fine-scale 224×288, to make the network scale-invariant. For the weighting coefficients in Equation (12), we empirically set γ, δ, and ϵ in Equation (12) to 0.6, 0.5, and 0.6, respectively.

5 Experiments

To evaluate the performance of our method, we conduct the comparison between our method with state-of-the-art reflection removal methods on the SIR^2 dataset [6]. The SIR^2 dataset [6] contains image triplets taken by using the postcards, solid objects, and objects from the wild scenes. We first use all image triplets to evaluate both quantitative benchmark scores and visual qualities. We then conduct experiments to compare the influence of the reflection blur levels to the final performances and the generalization ability with FY17 [7]. Finally, we compare our methods with CRRN [8] and also do a self-comparison experiment to justify the necessity of the new strategies proposed in our method. We resize images from SIR^2 dataset [6] to 224×288 for landscape images and 288×224 for portrait images. For the images used in the generalization comparison with FY17 [7], we set the image size to 224×320 for landscape images and 320×224 for portrait images.

TABLE 1: Quantitative evaluation results using five different error metrics, and compared with Baseline, FY17 [7], NR17 [25], WS18 [18], and LB14 [2].

	SSIM	SI	SSIM$_r$	SI$_r$	PSNR
Baseline	0.887	0.919	0.828	0.871	22.761
Ours	**0.903**	**0.923**	**0.880**	**0.905**	**24.194**
FY17 [7]	0.864	0.882	0.822	0.850	22.650
NR17 [25]	0.858	0.894	0.844	0.877	21.987
WS18 [18]	0.856	0.893	0.841	0.869	21.576
LB14 [2]	0.827	0.896	0.804	0.867	18.585

We adopt SSIM [38] and SI [6] as error metrics for our quantitative evaluation, which are widely used by previous reflection removal methods [2], [6], [26]. Due to the regional properties of reflections, we experimentally observe that many existing reflection removal methods [1], [2], [25] may downgrade the quality of whole images, although they can remove the local reflections cleanly. The original definitions of SSIM and SI, which evaluate the similarity between $\mathbf{B}$ and $\mathbf{B}^\star$ in the whole image plane, may not reflect the performance of reflection removal unbiasedly. We, therefore, define the regional SSIM and SI, denoted as SSIM$_r$ and SI$_r$, to complement the limitations of global error metrics. We manually label the reflection dominant regions and evaluate the SSIM and SI values at these regions similar to the evaluation method proposed in [18], [21].

5.1 Comparison with the state-of-the-arts

We compare our method with state-of-the-art single-image reflection removal methods, including FY17 [7], NR17 [25], WS18 [18], and LB14 [2]. We also adopt the comparisons between the mixture images and the ground truth as the baseline. For a fair comparison, we use the codes provided by their authors and set the parameters as suggested in their original papers. For FY17 [7], we follow the same training protocol introduced in their paper to train their network using our training dataset.

5.1.1 Quantitative comparison.

The quantitative evaluation results using four different error metrics and compared with four state-of-the-art methods are summarized in Table 1, where the errors between the input images and the corresponding ground truth are used as the baseline. The numbers displayed are the mean values over all 500 image triplets in the SIR^2 dataset [6]. As shown in Table 1, our method consistently outperforms other methods and the baseline for all four error metrics. The higher SSIM values indicate that our method recovers

[2] Please refer to http://pytorch.org/

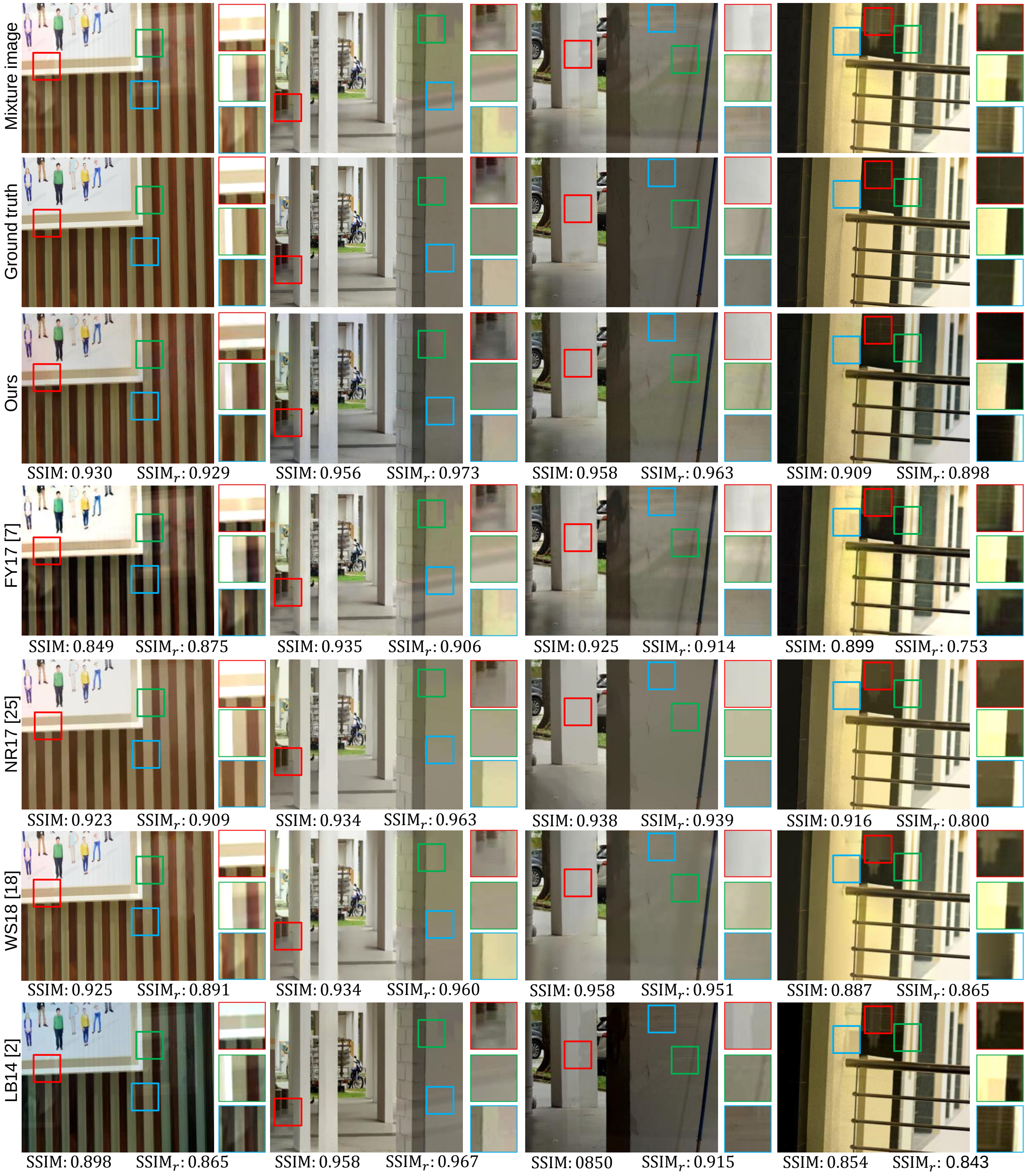
Mixture image
Ground truth
Ours
SSIM: 0.930 SSIM$_r$: 0.929
SSIM: 0.956 SSIM$_r$: 0.973
SSIM: 0.958 SSIM$_r$: 0.963
SSIM: 0.909 SSIM$_r$: 0.898
FY17 [7]
SSIM: 0.849 SSIM$_r$: 0.875
SSIM: 0.935 SSIM$_r$: 0.906
SSIM: 0.925 SSIM$_r$: 0.914
SSIM: 0.899 SSIM$_r$: 0.753
NR17 [25]
SSIM: 0.923 SSIM$_r$: 0.909
SSIM: 0.934 SSIM$_r$: 0.963
SSIM: 0.938 SSIM$_r$: 0.939
SSIM: 0.916 SSIM$_r$: 0.800
WS18 [18]
SSIM: 0.925 SSIM$_r$: 0.891
SSIM: 0.934 SSIM$_r$: 0.960
SSIM: 0.958 SSIM$_r$: 0.951
SSIM: 0.887 SSIM$_r$: 0.865
LB14 [2]
SSIM: 0.898 SSIM$_r$: 0.865
SSIM: 0.958 SSIM$_r$: 0.967
SSIM: 0850 SSIM$_r$: 0.915
SSIM: 0.854 SSIM$_r$: 0.843

Fig. 7: Examples of reflection removal results on four wild scenes from 'SIR^2' dataset [6], compared with FY17 [7], NR17 [25], WS18 [18], and LB14 [2]. Corresponding close-up views are shown next to the images (with patch brightness ×2 for better visualization), and SSIM and SSIM$_r$ values are displayed below the images. The complete results can be found in the supplementary materials.

the whole background image with better quality, whose global appearance is closer to the ground truth. The higher SI values indicate that our method preserves the structural information more accurately. The SSIM and SI values of other methods are all lower than the baseline, which indicate that they may more seriously distort the global structures when they remove reflections. The

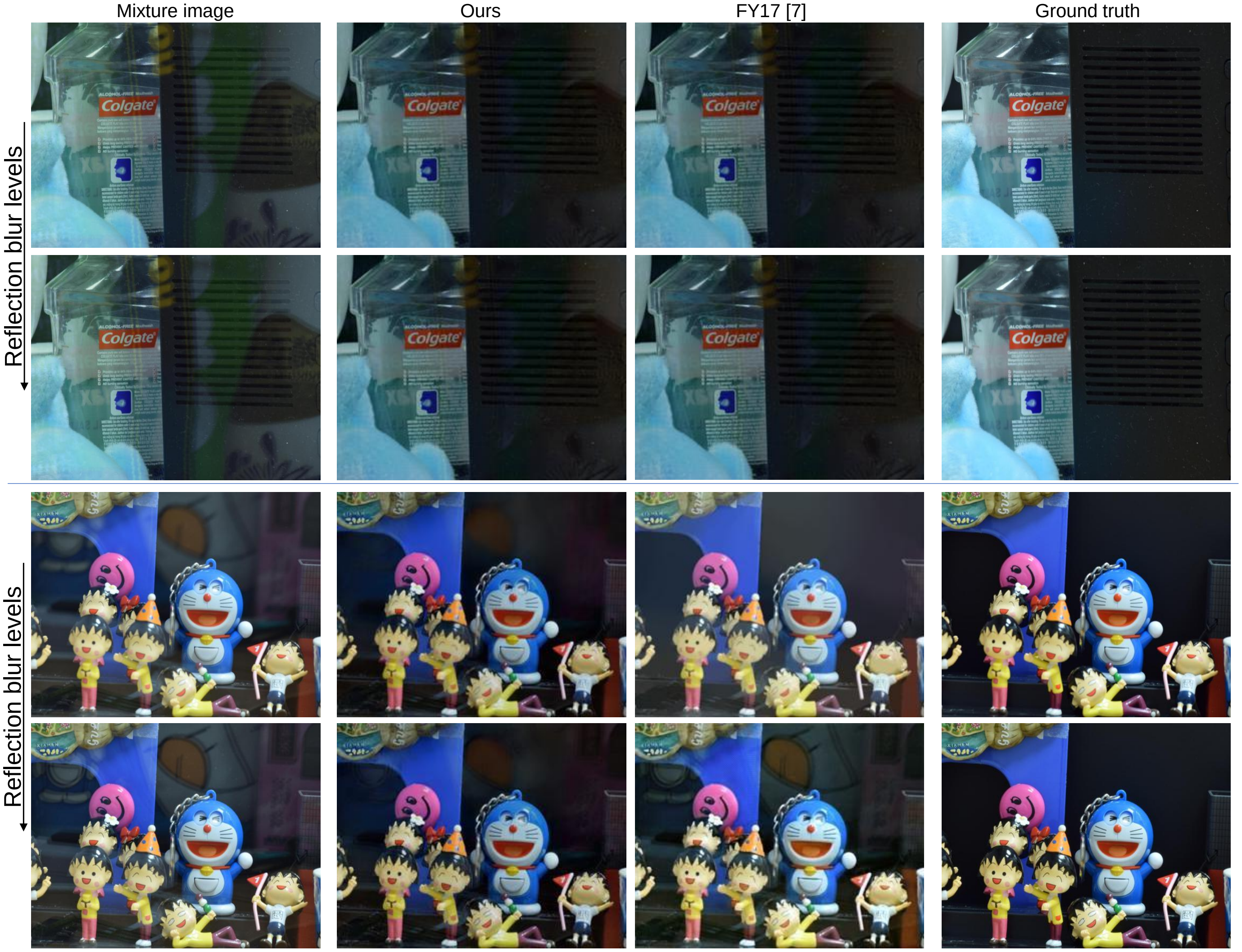

Fig. 8: Examples from the solid object dataset of 'SIR^2' [6] with different reflection blur levels. The down arrow means that the reflection blur levels increase. The complete results can be found in the supplementary materials.

higher SSIM$_r$ and SI$_r$ values mean that our method can remove strong reflections more efficiently in the regions overlaid with reflections than other methods. FY17 [7] shows the second best average performance on SSIM. WS18 [18] and NR17 [25] have similar performances.

5.1.2 Visual quality comparison.

We then show examples of estimated background images by our method and four state-of-the-art methods in Figure 7 to check their visual quality. In these examples, our method removes reflections more effectively and recovers the details of the background images more clearly, though some regions with small details are lost (*e.g.*, the grass regions in the second column) as the gradient guided framework may partly ignore the regions with small artifacts and the convolution is applied to the whole image and its feature maps, which may inevitably contaminate some regions with small artifacts. All the non-learning based methods (NR17 [25], WS16 [1], and LB14 [2]) remove the reflections to some extent, but residual edges remain visible for the reflections that are not out of focus. LB14 [2] causes some color change in the estimated results. It is mainly because of the insensitivity of the Laplacian data fidelity term to the spatial shift of the pixel values [25]. Though WS18 [18] and NR17 [25] can keep the color consistency and achieve similarly good quantitative values in SSIM (*e.g.*, the first example), they all show some over-smooth phenomena when they are not able to differentiate the background and reflection clearly (*e.g.*, the third examples in Figure 7 generated by NR17 [25] and WS18 [18]) or in some highly textured regions.

The deep learning based method FY17 [7] is good at preserving the image details and it does not cause the over-smooth artifacts as the non-learning based methods. However, the network in FY17 [7] is less effective in cleaning the residual edges comparing to our method. The SSIM and SSIM$_r$ values below each image also prove the advantage of our method.

5.1.3 The influence of the reflection blur levels

As a very important prior for the reflection removal problems, the different blur levels between the background and reflections play a very important role in the non-learning based methods, including LB14 [2] and NR17 [25]. Even in recent deep learning based method FY17 [7], they also follow such a blurring level assumption to create their training data. In general, when the reflection blur level increases, this problem becomes easier to be solved. To evaluate the influence of the reflection blur levels

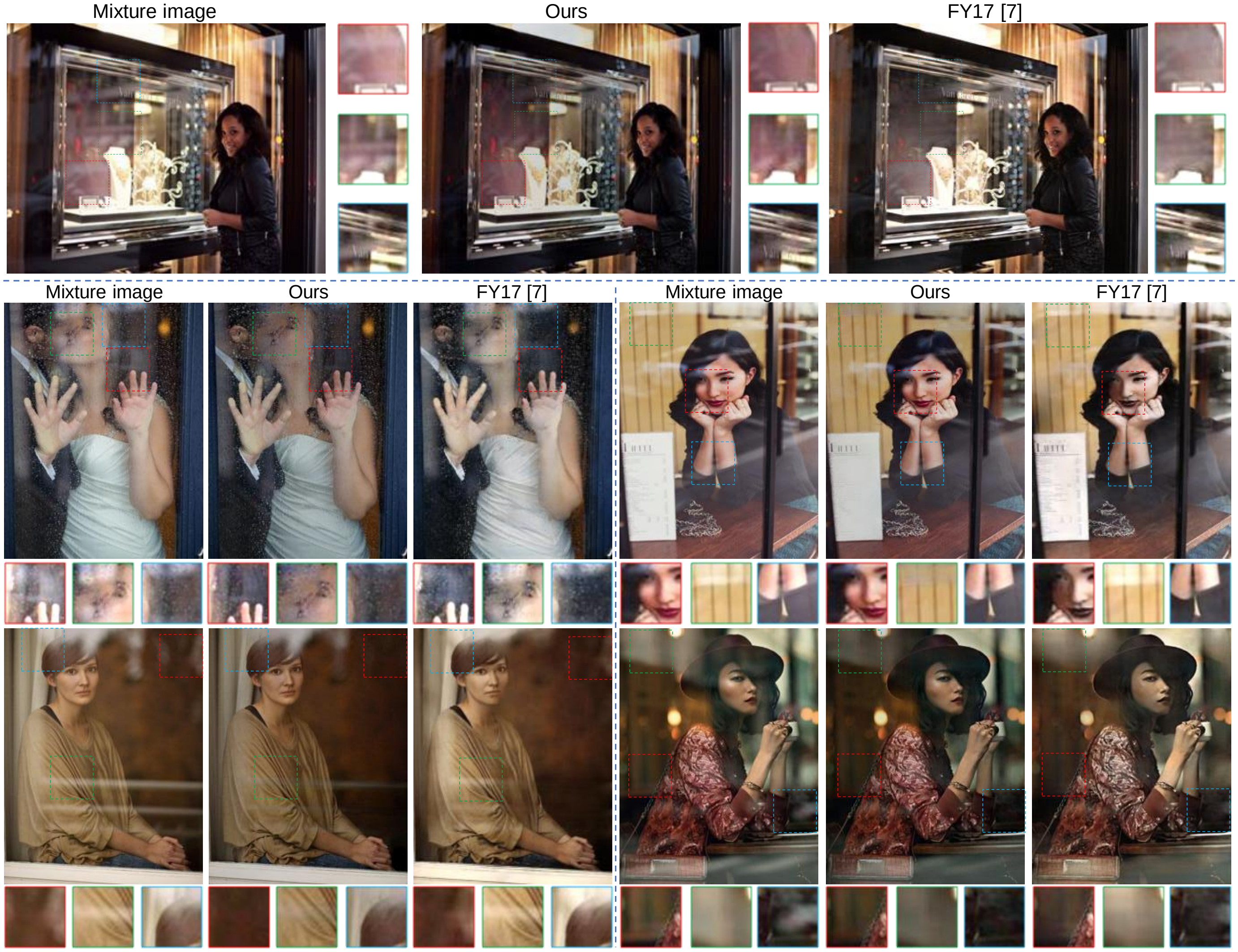

Fig. 9: The generalization ability comparison with FY17 [7] on their released validation dataset. Corresponding close-up views are shown below the images (the patch brightness ×2 for better visualization). The complete results can be found in the supplementary materials.

TABLE 2: Results by using images with different reflection blur levels. The reflection blur levles are inversely proportional to the values of 'F-variance'.

	F11	F16	F22	F32
Baseline	0.908	0.904	0.902	0.899
Ours	**0.920**	**0.916**	**0.909**	**0.903**
FY17 [7]	0.899	0.893	0.888	0.886
NR17 [25]	0.859	0.858	0.857	0.855
WS18 [18]	0.863	0.863	0.866	0.864
LB14 [2]	0.868	0.861	0.856	0.855

to the performance of our method, we conduct a more thorough experiment based on the images taken by using seven aperture sizes in the postcard and solid object dataset of SIR^2 [8]. From the results shown in Figure 8, though our method still remains some not very obvious residue edges when the reflections is sharp (the third and fifth column), it still performs much better than other methods. From Table 2, though the performances of all methods become worse when the reflections become sharp, our method achieves much better results than other methods and the baseline.

5.1.4 *Comparing generality with FY17 [7]*

The applicability to general complicated data of deep learning based methods is important yet challenging. Though the SIR^2 dataset [6] has covered many indoor and wild scenes, it is a dataset taken by using the purely flat surface and professional devices (*e.g.*, the DSLR camera) to take images. Such a strategy still cannot cover some daily scenarios (*e.g.*, the images taken through the window glass with curved degrees or taken by using mobile phones).

To better analyze the generalization ability of our method and FY17 [7], we evaluate the performances of the two methods by using the released validation dataset from the project website of FY17 [7][3]. Most images in their validation dataset are mainly downloaded from the Internet, which covers more challenging scenes. In this experiment, our network is still trained with our dataset described in Section 3.2 and strategy in Section 4.3, but

[3] https://github.com/fqnchina/CEILNet

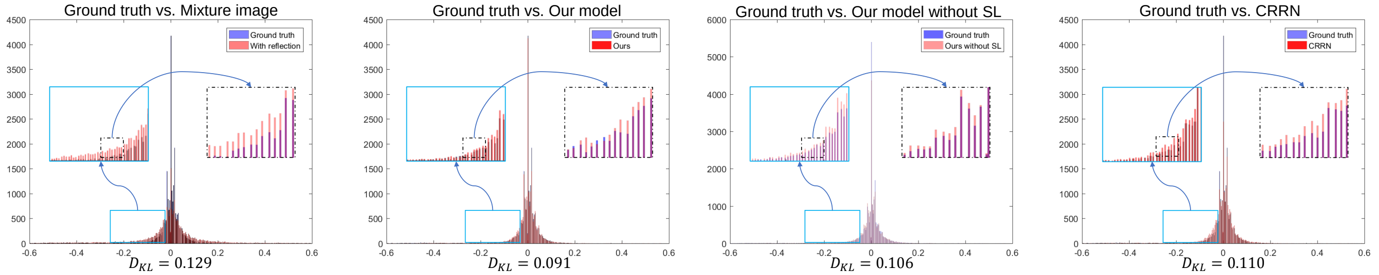

Fig. 10: The distribution comparisons between CRRN [8], our method, and our method without the statistic loss (SL). D_{KL} is the KL divergence between two distributions. The similarity of two distributions is inversely proportional to the D_{KL} values.

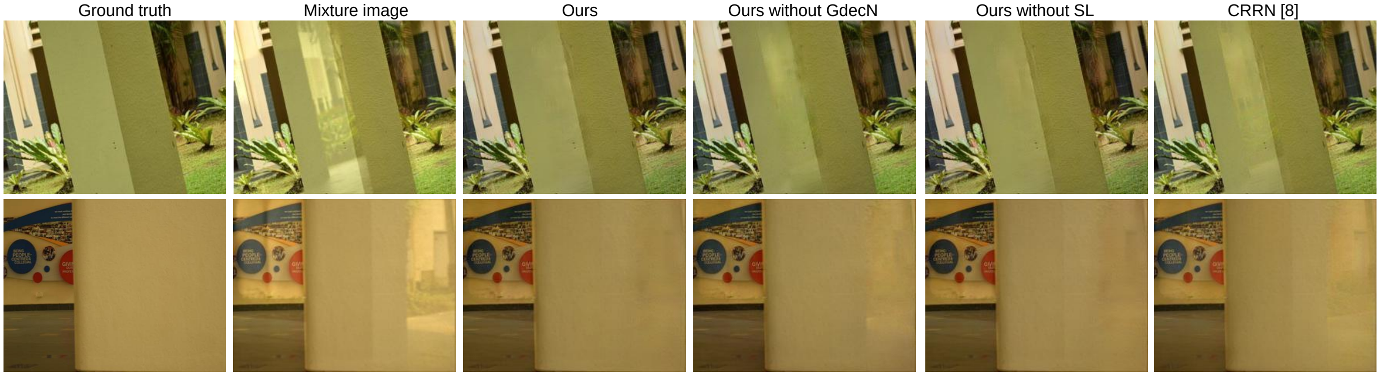

Fig. 11: Examples from the wild scene dataset with locally strong reflections compared with our model without statistic loss (SL) and CRRN [8].

TABLE 3: Result comparisons of our model against CRRN [8], our model without statistic loss (SL), with only $\mathcal{L}_1$ loss, and without GdecN, respectively.

	SSIM	SI	SSIM$_r$	SI$_r$
Ours	**0.903**	**0.923**	**0.880**	**0.905**
CRRN [8]	0.884	0.916	0.858	0.890
Ours without SL	0.896	0.913	0.877	0.891
Ours (only $\mathcal{L}_1$)	0.879	0.908	0.853	0.886
Ours (without GdecN)	0.869	0.889	0.849	0.879

for FY17 [7] we use the model released in their website (trained with their own data).

Due to the lack of ground truth, only the visual quality is compared for this part of images. From the results shown in Figure 9, it is not surprised that FY17 [7] performs well using their trained model on their validation dataset, but our method also achieves reasonably good results and performs even better in some images (*e.g.*, the examples in the first row and second row of Figure 9). Recall that when FY17 [7] is trained with our data and tested on the SIR^2 dataset, its quantitative and qualitative performances are below our method as shown in previous experiments.

5.2 Network analysis

In this section, we first compare our network structure with CRRN [8], the preliminary version of CoRRN with the similar structure which estimates $\mathbf{B}$, $\mathbf{R}$, and $\nabla\mathbf{B}$ concurrently. At the same time, we also analyze the influence of the proposed statistic loss to the final performances. At last, we conduct several experiments to evaluate the effectiveness of our cooperative framework.

5.2.1 Comparison with CRRN [8]

We first conduct several experiments to compare our network with CRRN [8]. The major differences between CRRN [8] and our method mainly exist in the encoder part and the statistic loss. We propose a feature sharing strategy where IdecN and GdecN share one same encoder network to make full use of the context information. On the other hand, instead of only using the perceptual and $\mathcal{L}_1$ loss like CRRN [8], we introduce a statistic loss by considering the inherent properties of reflections to better remove the locally strong reflections.

From the gradient level statistics shown in Figure 10, the distributions of the final estimated results obtained by using our model are the most similar to the ground truth both from the shape of the distributions and the KL divergence values shown below each figure.

At last, we evaluate the performances of our method and CRRN by using four error metrics as shown in Table 3. The quantitative scores illustrated in Table 3 have shown that our method performs better than CRRN [8] from an overall perspective. From another two examples shown in the last column of Figure 11, our method can better remove locally strong reflections when compared with CRRN [8].

5.2.2 Ablation study for loss functions and network designs

In this section, we conduct several ablation studies to further investigate the influence of different loss functions and network designs to the final performances and the contributions of GdecN to IdecN. From the results shown in Table 3 and the gradient level statistics in Figure 10, our complete model can achieve better performances than other models. The examples shown in Figure 11 and Table 3 also prove the ability of the statistics

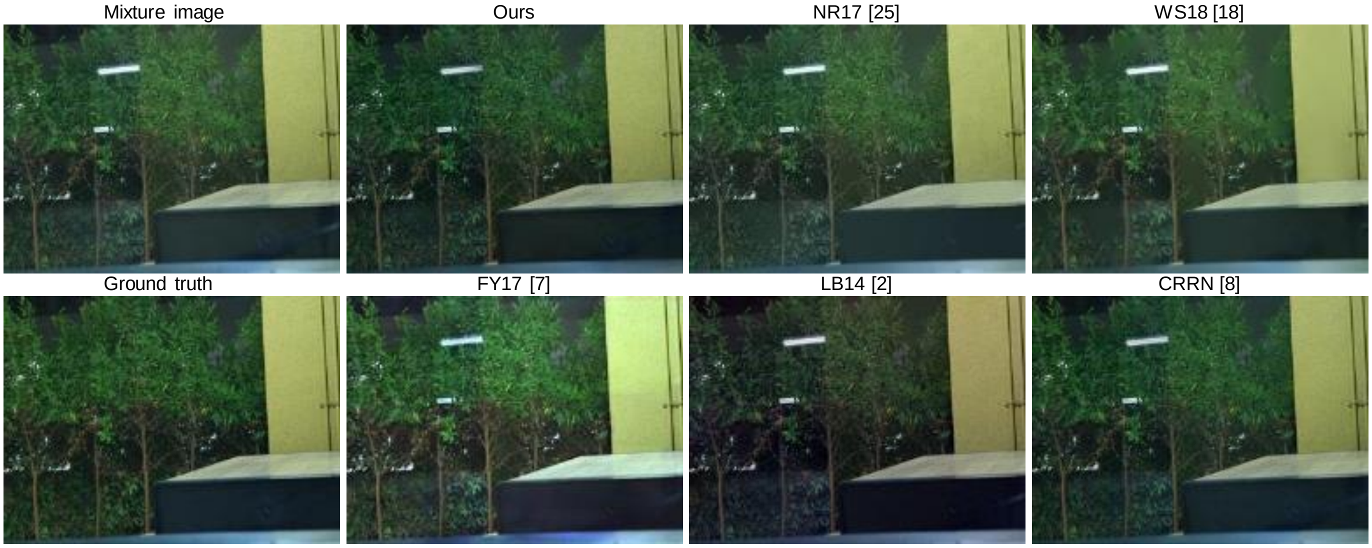

Fig. 12: Extreme examples with saturated reflections, compared with FY17 [7], NR17 [25], WS18 [18], LB14 [2], and CRRN [8].

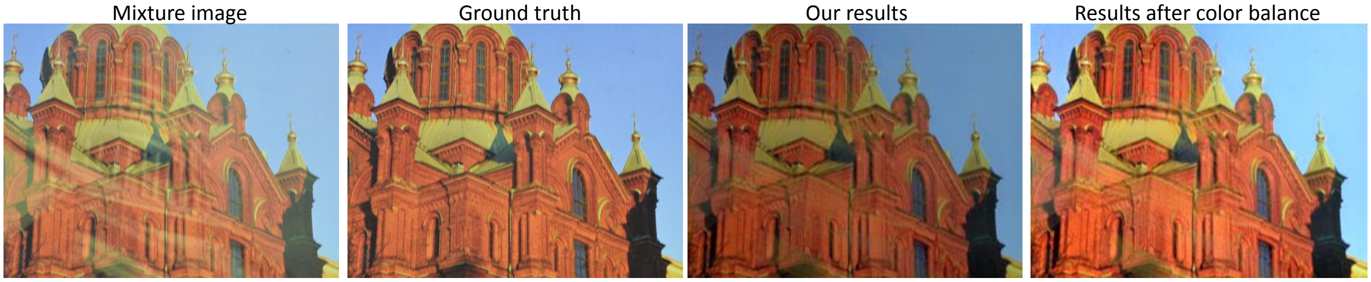

Fig. 13: One result from the postcard dataset obtained by our method and the corresponding results after color balance.

loss to remove locally strong reflections. When the statistics loss is removed, the overall performances become similar to CRRN.

Then, to analyze the contributions of GdecN to IdecN, we train another model by removing GdecN from the whole network. From the results shown in Table 3, though the SSIM$_r$ and SI$_r$ values are similar, the lower SSIM and SI values indicate that GdecN can contribute to the image reconstruction process in IdecN. The examples shown in Figure 11 also prove that the cooperative model with GdecN and IdecN can better handle the locally strong reflections.

6 Conclusion

We present a cooperative network to effectively remove reflection from a single image. Unlike the conventional pipeline that regards the gradient inference and image inference as two separate processes [7], [12] or a two-stream concurrent model [8], our network unifies them as a cooperative framework, which integrates high-level image context information and multi-scale low-level features. We further introduce a statistic loss based on the inherent relationship of the gradient level statistics to suppress the local strong reflections. Thanks to the collected real-world reflection image dataset and the corresponding training strategy, our method shows better performance than state-of-the-art methods for both the quantitative values and visual qualities and it is verified to be effectively generalized to other complicated data.

Limitations. The limitations of our method are as follows:

- **Saturated reflections.** Due to the loss of the background information in the regions with saturated reflections, the reflection removal problem has been degraded into an image inpainting problem, which has been regarded as a very challenging case for all reflection removal methods. For example, as shown in Figure 12, when the background information is completely lost in the white bulb area, almost all methods cannot remove the reflections efficiently. However, even in this challenging examples, our method still performs better than other methods.
- **Color shift.** With many convolutional layers in our network, our method suffers from the color shift problem in some specified situations, especially for the postcard dataset. From Figure 13, though the reflections have been efficiently removed, the estimated results become globally darker when compared with the ground truth. However, such kinds of problems can be easily alleviated by many existing color balance methods. For example, we simply try the method proposed in [47] to rescale the color information in the estimated images to its ground truth. The results after the rescaling become much better.
- **Data generation.** The diversity of capturing settings and scenarios for the reflection images needs to be further improved. These issues may limit the generalization ability of our training dataset, which will be specifically considered in our future work by including more diversified scenarios and setups.

References

[1] R. Wan, B. Shi, A. H. Tan, and A. C. Kot, “Depth of field guided reflection removal,” in *Proc. ICIP*, 2016.

[2] Y. Li and M. S. Brown, “Single image layer separation using relative smoothness,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2014.

[3] A. Levin and Y. Weiss, “User assisted separation of reflections from a single image using a sparsity prior,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, 2007.

[4] Y. Shih, D. Krishnan, F. Durand, and W. T. Freeman, “Reflection removal using ghosting cues,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2015.

[5] R. Wan, B. Shi, A. Tan, and A. C. Kot, “Sparsity based reflection removal using external patch search,” in *Proc. International Conference on Multimedia and expo (ICME)*, 2017.

[6] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, “Benchmarking single-image reflection removal algorithms,” in *Proc. International Conference on Computer Vision (ICCV)*, 2017.

[7] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, “A generic deep architecture for single image reflection removal and image smoothing,” *arXiv preprint arXiv:1708.03474*, 2017.

[8] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, “CRRN: Concurrent multi-scale guided reflection removal network.” *Proc. Computer Vision and Patter Recognition (CVPR)*, 2018.

[9] X. Fu, J. Huang, D. Z. Y. Huang, X. Ding, and J. Paisley, “Removing rain from single images via a deep detail network,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2017.

[10] Y. Li and M. S. Brown, “Exploiting reflection change for automatic reflection removal,” in *Proc. International Conference on Computer Vision (ICCV)*, 2013.

[11] A. Levin and Y. Weiss, “User assisted separation of reflections from a single image using a sparsity prior,” in *Proc. European Conference on Computer Vision (ECCV)*, 2004.

[12] P. Chandramouli, M. Noroozi, and P. Favaro, “Convnet-based depth estimation, reflection separation and deblurring of plenoptic images,” in *Proc. Asian Conference on Computer Vision (ACCV)*, 2016.

[13] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[14] J. Snell, K. Ridgeway, R. Liao, B. D. Roads, M. C. Mozer, and R. S. Zemel, “Learning to generate images with perceptual similarity metrics,” *arXiv preprint arXiv:1511.06409*, 2015.

[15] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, “Deep laplacian pyramid networks for fast and accurate super-resolution,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2017.

[16] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, “Pose guided person image generation,” in *Proc. Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[17] J. Dai, K. He, and J. Sun, “Instance-aware semantic segmentation via multi-task network cascades,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2016.

[18] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, W. Gao, and A. C. Kot, “Region-aware reflection removal with unified content and gradient priors,” *IEEE Transactions on Image Processing*, 2018.

[19] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, “Dehazenet: An end-to-end system for single image haze removal,” *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.

[20] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, “Joint rain detection and removal from a single image,” *arXiv preprint arXiv:1609.07769*, 2016.

[21] L. Qu, J. Tian, S. He, Y. Tang, and R. Lau, “Deshadownet: A multi-context embedding deep network for shadow removal,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2017.

[22] F. Agostinelli, M. R. Anderson, and H. Lee, “Adaptive multi-column deep neural networks with application to robust image denoising,” in *Proc. Conference on Neural Information Processing Systems (NeurIPS)*, 2013.

[23] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[24] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, “Attentive generative adversarial network for raindrop removal from a single image,” *arXiv preprint arXiv:1711.10098*, 2017.

[25] N. Arvanitopoulos, R. Achanta, and S. Susstrunk, “Single image reflection suppression,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2017.

[26] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman, “A computational approach for obstruction-free photography,” *ACM Transactions on Graphics*, vol. 34, no. 4, p. 79, 2015.

[27] K. Gai, Z. Shi, and C. Zhang, “Blind separation of superimposed moving images using image statistics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 19–32, 2012.

[28] E. Be’Ery and A. Yeredor, “Blind separation of superimposed shifted images using parameterized joint diagonalization,” *IEEE Transactions on Image Processing*, vol. 17, no. 3, pp. 340–353, 2008.

[29] B. Shi, Z. Mo, Z. Wu, D. Duan, S.-K. Yeung, and P. Tan, “A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[30] H. Li, W. Li, H. Cao, S. Wang, F. Huang, and A. C. Kot, “Unsupervised domain adaptation for face anti-spoofing,” *IEEE Transactions on Information Forensics and Security*, 2018.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *Proc. European Conference on Computer Vision (ECCV)*, 2014.

[32] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Proc. International Conference on Medical image computing and computer-assisted intervention (MICCAI)*. Springer, 2015.

[33] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning.” in *AAAI*, 2017.

[34] Y. Kim, I. Hwang, and N. I. Cho, “A new convolutional network-in-network structure and its applications in skin detection, semantic segmentation, and artifact reduction,” *arXiv preprint arXiv:1701.06190*, 2017.

[35] W.-C. Ma, H. Chu, B. Zhou, R. Urtasun, and A. Torralba, “Single image intrinsic decomposition without a single intrinsic image,” in *Proc. European Conference on Computer Vision (ECCV)*, 2018.

[36] S. Kim, K. Park, K. Sohn, and S. Lin, “Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields,” in *Proc. European Conference on Computer Vision (ECCV)*, 2016.

[37] K. Yu, C. Dong, C. C. Loy, and X. Tang, “Deep convolution networks for compression artifacts reduction,” *arXiv preprint arXiv:1608.02778*, 2016.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[39] J. Shi, L. Xu, and J. Jia, “Discriminative blur detection features,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2014.

[40] H. Li, S. J. Pan, S. Wang, and A. C. Kot, “Domain generalization with adversarial feature learning,” in *Proc. Computer Vision and Patter Recognition (CVPR)*, 2018.

[41] H. Li, S. J. Pan, R. Wan, and A. C. Kot, “Heterogeneous transfer learning via deep matrix completion with adversarial kernel embedding,” in *Proc. 33rd AAAI Conference on Artificial Intelligence (AAAI-19)*, 2019.

[42] A. Smola, A. Gretton, L. Song, and B. Schölkopf, “A hilbert space embedding for distributions,” in *International Conference on Algorithmic Learning Theory (ALT)*, 2007.

[43] Y. Li, K. Swersky, and R. Zemel, “Generative moment matching networks,” in *Proc. International Conference on Machine Learning*, 2015.

[44] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, “Loss functions for image restoration with neural networks,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 47–57, 2017.

[45] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep CNN denoiser prior for image restoration,” *arXiv preprint arXiv:1704.03264*, 2017.

[46] C. Dong, C. C. Loy, and X. Tang, “Accelerating the super-resolution convolutional neural network,” in *Proc. European Conference on Computer Vision (ECCV)*, 2016.

[47] L. A. Gatys, M. Bethge, A. Hertzmann, and E. Shechtman, “Preserving color in neural artistic style transfer,” *arXiv preprint arXiv:1606.05897*, 2016.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2921574, IEEE Transactions on Pattern Analysis and Machine Intelligence

Renjie Wan (S'14) received his B.S. degree from University of Electronic Science and Technology of China in 2012 and the Ph.D. degree from Nanayang Technological University, Singapore, in 2019. He is currently a research fellow in Nanyang Technological University, Singapore. He feels interested in computation photography and the history of ancient China.

Boxin Shi (M'14) received the BE degree from the Beijing University of Posts and Telecommunications, the ME degree from Peking University, and the PhD degree from the University of Tokyo, in 2007, 2010, and 2013. He is currently an assistant professor (1000 Youth Talents Professorship) with Peking University, where he leads the Camera Intelligence Group. Before joining PKU, he did postdoctoral research with MIT Media Lab, Singapore University of Technology and Design, Nanyang Technological University from 2013 to 2016, and worked as a researcher in the National Institute of Advanced Industrial Science and Technology from 2016 to 2017. He won the Best Paper Runner-up award at International Conference on Computational Photography 2015. He served/is serving as an Area Chair for ACCV 2018, MVA 2017. He is a member of the IEEE.

Haoliang Li received his B.S. degree from University of Electronic Science and Technology of China in 2013, the Ph.D degree from Nanyang Technological University, Singapore, in 2018. He is currently a research fellow in Nanyang Technological University, Singapore. His research interest is multimedia forensics and transfer learning.

Ling-Yu Duan (M'06) is a Full Professor with the National Engineering Laboratory of Video Technology (NELVT), School of Electronics Engineering and Computer Science, Peking University (PKU), China, and has served as the Associate Director of the Rapid-Rich Object Search Laboratory (ROSE), a joint lab between Nanyang Technological University (NTU), Singapore, and Peking University (PKU), China since 2012. He received the M.Sc. degree in automation from the University of Science and Technology of China, Hefei, China, in 1999, the M.Sc. degree in computer science from the National University of Singapore (NUS), Singapore, in 2002, and the Ph.D. degree in information technology from The University of Newcastle, Callaghan, Australia, in 2008. His research interests include multimedia indexing, search, and retrieval, mobile visual search, visual feature coding, and video analytics, etc. He received the EURASIP Journal on Image and Video Processing Best Paper Award in 2015, the Ministry of Education Technology Invention Award (First Prize) in 2016, the National Technology Invention Award (Second Prize) in 2017, China Patent Award for Excellence (2017), the National Information Technology Standardization Technical Committee "Standardization Work Outstanding Person" Award in 2015. He was a Co-Editor of MPEG Compact Descriptor for Visual Search (CDVS) Standard (ISO/IEC 15938-13), and is serving as a Co-Chair of MPEG Compact Descriptor for Video Analytics (CDVA). Currently he is an Associate Editor of ACM Transactions on Intelligent Systems and Technology (ACM TIST) and ACM Transactions on Multimedia Computing, Communications, and Applications (ACM TOMM).

Ah-Hwee Tan (SM'04) received the B.Sc. (Hons.) and M.Sc. degrees in computer and information science from the National University of Singapore, Singapore, in 1989 and 1991, respectively, and the Ph.D. degree in cognitive and neural systems from Boston University, Boston, MA, USA, in 1994. He is currently a Professor of Computer Science and the Associate Chair (Research) of the School of Computer Science and Engineering at Nanyang Technological University (NTU). Prior to joining NTU, he was a Research Manager with the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore, spearheading the Text Mining and Intelligent Agents research programs. His current research interests include cognitive and neural systems, brain inspired intelligent agents, machine learning, knowledge discovery, and text mining. Dr. Tan is an Editorial Board Member of IEEE ACCESS and Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS.

Alex C. Kot (S'85-M'89-SM'98-F'06) has been with Nanyang Technological University, Singapore, since 1991. He headed the Division of Information Engineering with the School of Electrical and Electronic Engineering for eight years and served as an Associate Chair/Research. He was the Vice Dean Research with the School of Electrical and Electronic Engineering and the Associate Dean for the College of Engineering for 8 years. He is currently a Professor at the School of Electrical and Electronic Engineering and a Director of the Rapid- Rich Object Search (ROSE) Lab. He has published extensively in the areas of signal processing for communication, biometrics, image forensics, information security and computer vision. He is a Fellow of IEEE, a Fellow IES and a Fellow of the Academy of Engineering, Singapore. He was awarded as the IEEE Distinguished Lecturer of the Signal Processing Society. He served as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE SIGNAL PROCESSING LETTERS, the IEEE SIGNAL PROCESSING MAGAZINE, the IEEE JOURNAL OF SPECIAL TOPICS IN SIGNAL PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I: FUNDAMENTAL THEORY AND APPLICATIONS, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: ANALOG AND DIGITAL SIGNAL PROCESSING. He has served the IEEE Signal Processing Society in various capacities, such as the General Co-Chair at the 2004 IEEE International Conference on Image Processing and the Vice President of the IEEE Signal Processing Society. He was a recipient of the Best Teacher of the Year Award and co-authored several best paper awards, including for ICPR, IEEE WIFS and IWDW.