# Benchmarking Single-Image Reflection Removal Algorithms

Renjie Wan, *Member, IEEE,* Boxin Shi, *Senior Member, IEEE,* Haoliang Li, Yuchen Hong, Ling-Yu Duan, *Member, IEEE* and Alex C. Kot, *Fellow, IEEE*

**Abstract**—Reflection removal has been discussed for more than decades. This paper aims to provide the analysis for different reflection properties and factors that influence image formation, an up-to-date taxonomy for existing methods, a benchmark dataset, and the unified benchmarking evaluations for state-of-the-art (especially learning-based) methods. Specifically, this paper presents a SIngle-image Reflection Removal Plus dataset "SIR$^{2+}$ " with the new consideration for in-the-wild scenarios and glass with diverse color and unplanar shapes. We further perform quantitative and visual quality comparisons for state-of-the-art single-image reflection removal algorithms. Open problems for improving reflection removal algorithms are discussed at the end. Our dataset and follow-up update can be found at https://sir2data.github.io/.

**Index Terms**—Reflection removal, benchmark dataset, deep learning

## 1 Introduction

How to obtain a reflection-free image from a mixture image taken through glass has attracted attention from computer vision researchers. Removing the undesired reflection enhances the visibility of target objects and benefits the performance of image classification [1], and face recognition [2]. Similar to other important image restoration problems [3], [4], reflection removal aims at obtaining an estimation of clear background $\mathbf{B}$ from its corrupted mixture image $\mathbf{I}$ by removing the interference caused by reflection $\mathbf{R}$ using one or more shots.

Reflection removal has been studied for more than two decades. Due to its ill-posed nature, earlier methods [5], [6] address the difficulty by rotating the polarizer. With the polarizing filter in different angles, the visibility of the background scene among the captured images can be changed to some degree. This polarizer-based setting provides independent observations of the mixtures and makes the problem less ill-posed. In more recent works, polarization cues are integrated with a deep learning pipeline to increase the robustness [7], [8].

The assumption about the layer independence between $\mathbf{B}$ and $\mathbf{R}$ proposed by pioneering polarizer-based works [5], [6] inspires solving this problem using ordinary RGB image. One of the most representative methods is proposed by Levin and Weiss [9]. They show that the property of $\mathbf{B}$ and $\mathbf{R}$ can be well fitted in the gradient domain by the Gaussian or Laplacian distribution. By modeling the independence assumption using the probability theory, Levin's method relaxes the requirement for extra devices, and a large number of follow-up methods inherits its core assumption. For example, by better exploring the relationship of $\mathbf{B}$ and $\mathbf{R}$ in the gradient domain, earlier methods [10], [11] differentiate $\mathbf{B}$ and $\mathbf{R}$ by using their different blur levels caused by the non-uniform depth-of-field. The recently proposed deep-learning-based methods [12], [13], [14], [15] also use gradient information as the auxiliary information for network inference. The gradient property also plays a crucial role in reflection removal methods using multiple images (*e.g.*, [16]).

The previous version [17] of this paper mainly focuses on the solutions of non-learning-based methods, while there are several new open problems and trends needed to be discussed nowadays. For example, since deep learning becomes the silver bullet for most reflection removal methods (*e.g.*, [2], [12], [18]), and different datasets (*e.g.*, [2], [15], [19]) are also proposed for various scenarios along with the data-driven solutions. The trend brought by deep learning raises the need for a more up-to-date taxonomy. Besides, since existing deep-learning-based methods show degraded performance for unseen examples due to their dependency on synthetic training data, it also becomes necessary to evaluate their performance under different settings and scenarios with a unified and systematic dataset.

To address above issues, this paper extends from its previous version [17] in the following aspects:

- A reflection image formation analysis for clearly categorizing existing methods.
- An up-to-date taxonomy by taking mainstream methods for discussions.
- An extended dataset SIR$^{2+}$ with the new consideration for in-the-wild scenarios and glass with diverse color and unplanar shapes.
- A more comprehensive evaluation including the cross-dataset investigation for existing methods and the corresponding open problems considering the latest progress.

- *This work was done at Rapid-Rich Object Search (ROSE) Lab, Nanyang Technological University. This project is supported in part by National Natural Science Foundation of China under Grant No. 62136001, 62088102, 61872012, and in part by the PKU-NTU Joint Research Institute (JRI) sponsored by a donation from the Ng Teng Fong Charitable Foundation. Haoliang Li is supported by CityU New Research Initiatives/Infrastructure Support from Central under the grant APRC 9610528.*
- *Renjie Wan is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China.*
- *Boxin Shi, Yuchen Hong, and Ling-Yu Duan are with the National Engineering Research Center of Visual Technology, School of Computer Science, Peking University, Beijing, China.*
- *Haoliang Li is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong SAR, China.*
- *Alex C. Kot is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, Singapore.*
- *Corresponding author: Boxin Shi (e-mail: shiboxin@pku.edu.cn)*

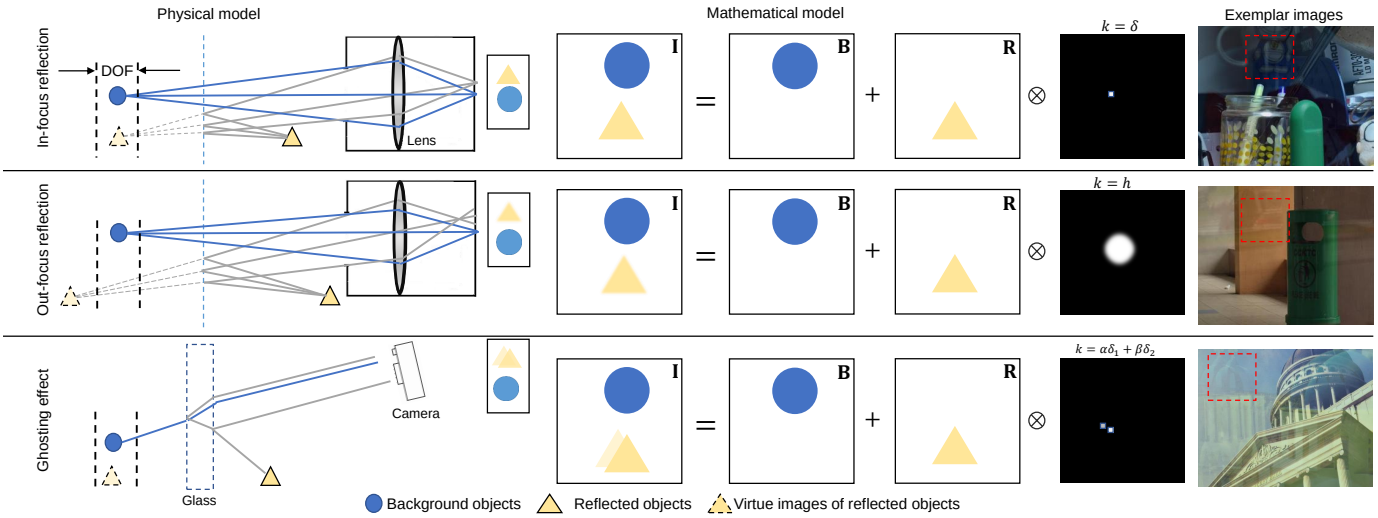



Fig. 1: The physical (left), mathematical (middle) image formation models, and exemplar images (right) for three kinds of reflections. The physical models for the in-focus reflection and out-of-focus reflection ignore the refractive effect of glass. For simplicity, we omit the lens and use only one ray for the physical model of the ghosting effect. $\mathbf{I}$, $\mathbf{B}$, $\mathbf{R}$, and kernel $k$ are defined in Section 2.2. The red boxes in exemplar images denote the regions with in-focus reflection, out-of-focus reflection, and ghosting effect, respectively.

## 2 Analysis on the Image Formation Process

Reflection removal aims at recovering $\mathbf{B}$ from the mixture image $\mathbf{I}$. The image formation process for the reflection removal problem can be concluded as follows:

$$\mathbf{I} = g(\mathbf{B}) + f(\mathbf{R}), \tag{1}$$

where $\mathbf{B}$ and $\mathbf{R}$ denote the light emitted by background scenes and reflection scenes, respectively; $g(\cdot)$ and $f(\cdot)$ denote the various degradation for $\mathbf{B}$ and $\mathbf{R}$ during light transmission, respectively; $g(\mathbf{B})$ and $f(\mathbf{R})$ denote background and reflection irradiance that finally reach the camera sensor, respectively; and a mixture of them forms $\mathbf{I}$.

There are different definitions for $\mathbf{I}$, $\mathbf{B}$, and $\mathbf{R}$ among existing methods. Most methods [12], [13], [14] call them as "image" or "layer" by directly inheriting the meaning from the work proposed by Levin *et al.* [9]. For them, they solve the reflection removal problem from the aspect of image separation. Another category of methods (*e.g.*, [7], [8], [72], [73]) inherits the meanings from the method proposed by Faried *et al.* [5], where the three parameters are defined as the irradiance reflected off the scenes and received by the sensors. In this section, we specifically denote them as irradiance to discuss the potential influence during light transmission. In other sections, we do not deliberately differentiate the two meanings.

### 2.1 Factors that influence $f(\cdot)$ and $g(\cdot)$

Different factors may influence $f(\cdot)$ and $g(\cdot)$, which can be summarized as follows:

**Refractive effect.** The refractive effect is related to the density of glass [20] and mainly affects $g(\cdot)$ by causing pixel shifts between $g(\mathbf{B})$ and $\mathbf{B}$. In general, such shifts cause the pixel misalignment between the images captured with and without glass. The pixel misalignment makes it challenging to directly utilize real image pairs as the training data since pixel-wise loss functions heavily rely on the well-aligned training pairs [14].

**Absorption and reflectivity effect.** When light travels through a piece of glass, the light's intensity is typically influenced by the absorption and reflectivity effect. For colorless glass, the low reflectivity of glass makes only a limited amount of $\mathbf{R}$ be reflected by glass and received by the camera, and it also leads to $f(\mathbf{R}) < \mathbf{R}$ [21]. On the contrary, the high transmittance rate and low absorption of glass [22] make $g(\mathbf{B})$ almost a copy of $\mathbf{B}$. For tinted glass, since it selectively absorbs different frequencies of visible light [23], $g(\mathbf{B})$ is with the obvious difference to $\mathbf{B}$. For example, a piece of gray glass in Figure 2 transmits mainly blue wavelengths of light while absorbing in other wavelengths, which makes the captured image appear globally blue color.

**Depth of field (DoF).** Depth of field is the zone of acceptable sharpness within a photo that appears in focus [24]. As an important property of the camera, DoF influences blur levels of $\mathbf{B}$ and $\mathbf{R}$. Reflection scenes show different blur levels when they are with different distances to the camera sensor. Since photographers mainly focus on background scenes [11], the influence from $g(\cdot)$ can be ignored in some scenarios with uniform distances from background scenes to the camera sensor. However, if background scenes are with non-uniform distances to the camera sensor, its blur levels also become non-uniform, which makes it difficult for existing methods (*e.g.*, [10], [11]) designed for uniform blur differentiate $\mathbf{B}$ and $\mathbf{R}$ properly.

Fig. 2: Three examples with low-transmitted reflections (labelled by red boxes) caused by the regional property. The third one is captured through tinted glass.

### 2.2 Reflection property

Most methods (*e.g.*, [2], [9], [11], [13]) also formulate Equation (1) using the following analytical form:

$$\mathbf{I} = \alpha\mathbf{B} + \beta(\mathbf{R} \otimes k), \tag{2}$$

where $\alpha$ and $\beta$ are weighting coefficients, and $k$ is a convolutional kernel representing the potential degradation. By studying the factors in Section 2.1, we find that most reflections can be roughly classified into four types.

**In-focus reflection.** As shown in Figure 1, the in-focus reflection is mainly caused by DoF discussed in Section 2.1. When the objects behind glass and the virtue images of reflected objects are approximately in the same focal plane, their corresponding background and reflection layers are more likely to have the same sharp levels [11], [25]. In this situation, $\mathbf{I}$ becomes a linear additive mixture of $\mathbf{B}$ and $\mathbf{R}$, and the kernel $k$ degenerates into a one-pulse kernel $\delta$.

**Out-of-focus reflection.** It is reasonable to assume that virtue images of reflected objects and background objects behind glass have different distances from the camera. Since taking the objects behind glass in focus is more likely to be a common behavior for most photographers [25], the observed image $\mathbf{I}$ is an additive mixture of the sharp background layer and the blurred reflection layer. The kernel $k$ depends on the camera's point spread function, which is parameterized by a 2D Gaussian function denoted as $h$ [26].

**Ghosting effect.** Both types above assume that glass's refractive effect is negligible, while a more realistic physical model should also consider the influence from glass thickness. The ghosting effect is linked to the refractive effect [27]. As illustrated in Figure 1 (third row), the light rays from background scenes are partially reflected on the outside facet of the glass. Then, the remaining rays penetrate the glass and are reflected again from the inside facet of the glass [28]. Such ghosting effects caused by thick glass make the observed image $\mathbf{I}$ be a mixture of $\mathbf{B}$ and the convolution of $\mathbf{R}$ with a two-pulse ghosting kernel $k = \alpha\delta_1 + \beta\delta_2$, where $\alpha$ and $\beta$ are combination coefficients and $\delta_2$ is a spatial shift of $\delta_1$.

**Regional property.** Caused by the absorption and reflectivity effect, the regional property makes the reflection only dominate limited regions. As shown in Figure 2, for the regions with $g(\mathbf{B}) > f(\mathbf{R})$, the background may dominate the final captured image. Inversely, for the regions with $g(\mathbf{B}) \approx f(\mathbf{R})$, the reflection may become obvious in the final captured image [21]. Furthermore, if $g(\mathbf{B}) < f(\mathbf{R})$, the reflection may even dominate the final captured image. Specifically, we call the reflections in the regions with $g(\mathbf{B}) > f(\mathbf{R})$ and $g(\mathbf{B}) \approx f(\mathbf{R})$ as *transmitted reflections*, since the light rays from background objects can transmit through these regions. Correspondingly, the reflections in the regions with $g(\mathbf{B}) < f(\mathbf{R})$ are denoted as *low-transmitted reflections*.

## 3 A Reflection Removal Taxonomy

This section first presents the mathematical form of the layer independence assumption. Then, we categorize existing methods hierarchically and intuitively. We first classify each method according to the number of input images and then by different constraints imposed for solving this problem.

### 3.1 Single-image methods

From Equation (1), the difficulties of this problem are mainly from two parts - the number of unknowns is twice the number of equations. Besides, the similarities between background and reflection's properties make it difficult to remove the reflection and restore the background simultaneously.

Single-image methods mainly address the two difficulties based on the independence assumption between the background layer and the reflection layer [9] as follows:

$$P(\mathbf{B}, \mathbf{R}) = P_1(\mathbf{B}) \cdot P_2(\mathbf{R}), \tag{3}$$

where $P$ is the joint probability distribution, and $P_1$ and $P_2$ are the distributions imposed on $\mathbf{B}$ and $\mathbf{R}$, respectively. Equation (3) assumes that the properties of background and reflection layers can be approximated by specific probability distributions and the two distributions are mutually independent. Then, by imposing different constraints and priors on $\mathbf{B}$ and $\mathbf{R}$, different kinds of methods are proposed.

#### 3.1.1 Non-learning-based methods

The non-learning-based methods aim at modeling $P_1$ and $P_2$ by using handcrafted priors and then solve Equation (3) based on various mathematical tools. We summarize non-learning-based methods in Table 1 and classify them based on the priors they use.

**Sharpness priors** are motivated by the fact that reflection and background layers may have different sharpness due to different DoF settings. The non-learning-based methods embed the sharpness prior into their optimization process by utilizing the natural image gradient with heavy-tailed distributions. The pioneer method [9] imposes two same Laplacian mixture distributions ($P_1 = P_2$) on $\mathbf{B}$ and $\mathbf{R}$ to separate the two parts by manually labeling their edges. Despite its effectiveness, its requirement for manual annotations of background and reflection edges limits its practicality. With more assumptions for background and reflection edges, the manually labeling process can be done more automatically by the DoF difference [11] (WS16), the gradient profile sharpness [52], [53], the total variation [29], and multi-view images [54], [55].

However, as shown in Figure 1 (out-of-focus reflection), since we are likely to focus on background objects when taking photos, it is reasonable to assume that the background and reflection layers in the final captured image have different sharpness levels. Since the sharp background $\mathbf{B}$ and the blurred reflection $\mathbf{R}$ usually have different shapes in the gradient domain [10], it becomes inappropriate to impose the same distributions on $\mathbf{B}$ and $\mathbf{R}$. Some methods [10], [30], [56] introduce a more robust statistical model for background and reflection gradients by assuming $P_1 \neq P_2$. $P_1$ is designed to model the gradient distribution of $\mathbf{B}$ with large gradient values, so it drops slower than $P_2$ designed to model the gradient distribution of $\mathbf{R}$ with small gradient values.

Gradient-based sharpness priors show promising results, especially in examples with inconsistent sharpness or blur levels. By combing it with other specific priors, it can also be used to remove reflection with special property (*e.g.*, the repetitive pattern [28] caused by the ghosting effect). Besides the non-learning-based methods discussed above, it is also adopted by the deep-learning-based methods [12], [13], [15], [35], [37], [39], [50], [57] to differentiate reflection and background layers better. However, since it cannot deal with more complex situations (*e.g.*, the in-focus reflection and the low-transmitted reflection in Figure 1), where the reflection and background are with similar sharpness, more specific priors are needed to remove the reflection with complex properties. However, even for the methods using more specific

TABLE 1: Summary of non-learning-based single-image reflection removal methods. The third column denotes the methods or priors that may inspire the method in the first column. "N.C." means that the method in the first column proposes new strategies or scenarios, which are Not Covered by previous methods.

| Estimate $\mathbf{B}$ from $\mathbf{I} = \alpha\mathbf{B} + \beta(\mathbf{R} \otimes k)$ by using different assumptions on $k$ and constraints on $\mathbf{B}$ and $\mathbf{R}$<br>Notations: $P(\mathbf{B}, \mathbf{R}) = P_1(\mathbf{B}) \cdot P_2(\mathbf{R})$, where $P_1$ and $P_2$ are the priors enforced on $\mathbf{B}$ and $\mathbf{R}$, respectively | | | |
|---|---|---|---|
| **Method** | **Gradient** | **Related to** | **Key Assumptions** |
| AY07 [9] | Not used | N.C. | $P_1 = P_2$, user assistance to label the edges of $\mathbf{B}$ and $\mathbf{R}$ |
| CC09 [29] | Used | AY07 [9] | $P_1 = P_2$, blur differences to label the edges of $\mathbf{B}$ and $\mathbf{R}$ automatically |
| LB14 [10] | Used | N.C. | $P_1 \neq P_2$, different gradient priors to describe to describe $P_1$ and $P_2$ |
| SK15 [1] | Not used | N.C. | $P_1 \neq P_2$, $P_2$ is described by the ghosting priors |
| WS16 [11] | Used | AY07 [9] | $P_1 = P_2$, blur differences to label the edges of $\mathbf{B}$ and $\mathbf{R}$ automatically |
| NR17 [30] | Used | LB14 [10] | $P_1 \neq P_2$, remove reflection in Laplacian domain |
| WS17 [31] | Used | N.C. | $P_1 = P_2$, estimate $\mathbf{B}$ by using the information from the reference image |
| WS18 [32] | Used | WS17 [31] | $P_1 \neq P_2$, estimate $\mathbf{B}$ by using the regional property and self-similarity inside the image |
| YW19 [33] | Used | LB14 [10], NR17 [30] | $P_1 \neq P_2$, a partial differential equation with gradient thresholding |
| HQ20 [28] | Used | SK15 [1] | $P_1 \neq P_2$, a wavelet-transform-based regularization to distinguish repetitive patterns |

TABLE 2: Summary of deep-learning-based reflection removal methods. "T-data" and "E-data" denote the training and evaluation datasets, respectively. For "T-data" column, "S", "Half-S", and "Mixed" denote "Synthetic data", "Half-synthetic data", and "Mixed data", respectively. The second column denotes the methods or priors that may inspire the method in the first column. "N.C." means that the method in the first column proposes new strategies or scenarios, which are Not Covered by previous methods.

| Estimate $\mathbf{B}$ from $\mathbf{I} = \alpha\mathbf{B} + \beta(\mathbf{R} \otimes k)$ by using the data-driven methods<br>Notations: $P(\mathbf{B}, \mathbf{R}) = P_1(\mathbf{B}) \cdot P_2(\mathbf{R})$, where $P_1$ and $P_2$ are all leaned by the convolutional neural network. | | | | |
|---|---|---|---|---|
| Single-image methods | | | | |
| **Method** | **Related to** | **T-data** | **E-data** | **Key Assumptions** |
| CN17 [34] | AY07 [9] | S | Net images | **Supervised**\| CNN to label edges and then a non-learning way to remove reflection |
| CEILNet [12] | AY07 [9] | S | Net images | **Supervised**\| A two-stage CNN framework under the supervision of edges |
| LY18 [35] | CycleGAN [36] | S | Net images | **Weakly-Supervised**\| The categories of $\mathbf{B}$ and $\mathbf{R}$ are known |
| CRRN [37] | AY07 [9] | Half-S | $SIR^2$ | **Supervised**\| A concurrent CNN framework under the guidance of gradients |
| ZN18 [38] | GP | Mixed | Real20 | **Supervised**\| Estimate $\mathbf{B}$ based on the adversarial framework |
| CoRRN [13] | CRRN [37] | Half-S | $SIR^2$ | **Supervised**\| Explore the higher order statistics to differentiate $\mathbf{B}$ and $\mathbf{R}$ |
| ERRNet [14] | N.C. | Mixed | $SIR^2$, Real20 | **Supervised**\| Use the high level features to handle the pixel misalignment |
| MW19 [39] | CycleGAN [36] | Half-S | $SIR^2$ | **Weakly-Supervised**\| A joint model to generate and separate reflection |
| SIRRBL [40] | N.C. | S | $SIR^2$ | **Supervised**\| A learning-based strategy to synthesize mixture image |
| CR19 [41] | FY20 [42] | Half-S | Real Text images | **Supervised**\| A specific model for text image by embedding the text priors |
| KH19 [43] | N.C. | S | $SIR^2$, Real100 | **Supervised**\| A physically based rendering to synthesize the training images |
| LL19 [44] | N.C. | S | $SIR^2$, Real20 | **Supervised**\| A high-level semantic guided framework |
| IBCLN [19] | RNN | S | $SIR^2$, Nature | **Supervised**\| LSTM network to *iteratively* refine the reflection removal process |
| CL21 [45] | GP | S | $SIR^2$, Real20 | **Supervised**\| Reflection classifier to distinguish $\mathbf{R}$ from $\mathbf{B}$ |
| RAGNet [46] | IBCLN [19] | S | $SIR^2$, Real20, Nature | **Supervised**\| $\mathbf{R}$-aware guidance to distinguish the $\mathbf{R}$-dominated regions |
| ZX20 [47] | IBCLN [19] | S | $SIR^2$, Real20, Net45 | **Supervised**\| $\mathbf{R}$-aware framework to distinguish the $\mathbf{R}$-dominated regions |
| ZS21 [48] | N.C. | S | $SIR^2$, Nature | **Supervised**\| A two-step solution that considers the absorption effect |
| HZ21 [49] | N.C. | S | Natural | **Supervised**\| Utilize panoramic images with auxiliary contextual cues of $\mathbf{R}$ |
| Multiple-image methods | | | | |
| FY20 [42] | N.C. | S | $SIR^2$ | **Unsupervised**\| Two images with same $\mathbf{B}$ and different $\mathbf{R}$ as the input |
| FIRR [2] | N.C. | Half-S | Real face images | **Supervised**\| A specific model for face images by embedding the facial priors |
| LL20 [50] | XR15 [16] | S | Real sequences [50] | **Supervised**\| Estimate the motion flow field based on deep neural network |
| OF21 [51] | N.C. | Mixed | Real157 | **Supervised**\| Guided by an image taken in a dark room with only a flash on |

priors, the gradient-based sharpness prior is still considered by many methods as a complementary constraint to improve the model robustness.

**Semantic priors** focus on the restoration of missing contents [31], instead of simply separating $\mathbf{B}$ and $\mathbf{R}$. Along this direction, by modeling $P_1$ and $P_2$ in Equation (3) using more discriminative priors, various frameworks are proposed, including the GMM patch prior [1] (SK15), sparsity-based framework [31] (WS17), and region-aware framework with non-local priors [32] (WS18). For example, WS18 [32] adopts the non-local image prior to better approximate the background statistics $P_1$ in Equation (3) and a complementary gradient prior to model the reflection statistics $P_2$, respectively. The non-local prior borrows information from regions surrounding the reflection to recover the missing background.

Compared with sharpness priors, semantic priors are more effective in removing the reflection with similar gradient properties to the background. For example, SK15 [1] is claimed to be effective for the reflection with ghosting effects. WS17 [31] and WS18 [32] can handle all situations shown in Figure 1 if the clean patches with similar contents can be found. However, the methods using semantic priors are mainly patch-based. Instead of regarding the input image as a whole, they divide the whole image into several patches and process them one by one. This strategy is computationally expensive and causes additional artifacts in the final result.

#### 3.1.2 Deep-learning-based methods

Deep learning techniques are employed by recent methods to model reflection properties more comprehensively. Despite the

TABLE 3: Summary of motion-based reflection removal methods.

| Estimate $\mathbf{B}$ from $\{\mathbf{I}_i = \alpha_i \mathbf{B}_i + \beta_i(\mathbf{R}_i \otimes k_i)\}_{i=1\cdots n}$, where $n$ denotes the number of input images | | |
|---|---|---|
| **Method** | **Motion estimation** | **Key Assumptions** |
| GS12 [58] | Affine | Estimate the motions and mixing coefficients of each layer |
| LB13 [55], SL16 [54], SC15 [59] | SIFT flow | Use multiple images to label the edges of $\mathbf{B}$ and $\mathbf{R}$ |
| GC14 [60] | Homography | Low rank property to remove $\mathbf{R}$ from aligned $\mathbf{B}$ |
| XR15 [16] | Pixel-wise flow | Same gradient priors to describe $P_1$ and $P_2$ |
| YL16 [61] | Optical flow | A double-layer brightness consistency assumption to constrain $\mathbf{B}$ and $\mathbf{R}$ |
| HS18 [62] | Co-saliency | Low-rank matrix completion to remove reflections from aligned images |
| AB19 [63] | Defocus-disparity | The defocus-disparity cues to align image pairs captured by a dual-pixel sensor |
| AC19 [64] | N.C. | 3D CNN to capture appearance and motion patterns |
| FY20 [42] | N.C. | Two images with same $\mathbf{B}$ and different $\mathbf{R}$ to learn the background property |
| FIRR [2] | Optical flow | A specific model for face images by embedding facial priors |
| LL20 [50] | Dense optical flow | Estimate the motion flow field based on deep neural network |

effectiveness of sharpness priors and semantic priors, they are often violated in real-world scenarios since they only describe a limited range of reflection and background properties and may project the partial observation as the whole truth. For example, the performance of SK15 [1] largely drops if the ghosting effect is not observed, and LB14 [10] has difficulties in removing the in-focus reflection. Actually, in real-world scenarios, the situations in Figure 1 and more complex situations (*e.g.*, $\mathbf{R}$ in focus and $\mathbf{B}$ not in focus) may all exist in one image. Thus, it is inappropriate to cover all possibilities by using one or two specific priors. As advanced learning techniques, deep learning techniques can learn $P_1$ and $P_2$ in a data-driven manner, which improves modeling ability.

**Low-level-based methods** embed low-level image features into deep learning networks by finding optimal solutions from the image formation process and its resulted physical properties. One category continues to leverage advantages from the gradient difference between $\mathbf{B}$ and $\mathbf{R}$ by using the gradient information as the auxiliary features during the network inference process (*e.g.*, CEILNet [12], CRRN [37], CoRRN [13], and WM19 [39]) or measuring the feature difference in the gradient domain [15], [81]. Some recent methods try to jump out from the constraints of the gradient prior. For example, by considering the absorption effect, ZS21 [48] proposes a two-step solution by first estimating the absorption effect from a mixture image and then taking the mixture image and its corresponding absorption effect as the input for the second stage.

**High-level-based methods** utilize high-level computer vision to refine reflection removal results. It is important for the methods with the requirement for the accuracy of recovered results. For example, the reflection removal model for face images [2] utilizes the features estimated by the face recognition model to keep face identity consistency between the estimated result and its corresponding reference. The text image reflection removal method [41] also makes use of the text recognition model to achieve a similar goal. Besides the models for specific purposes, some recently proposed methods (*e.g.*, [44]) also utilize the semantic information estimated by the segmentation network to ameliorate the performance. CL21 [45] further utilizes the image classifier to differentiate the background and reflection. Besides, some methods also estimate the background by relying on data-driven features. For example, ERRNet [14], ZN18 [15], and Chi *et al.* [82] all embed the features from the pre-trained VGG model [83] into the whole estimation process. Hong *et al.* [49] relieve the content ambiguity problem in reflection removal by utilizing panoramic images containing auxiliary contextual cues of reflection scenes.

**Region-aware methods** address the difficulties by considering the regional property. To avoid the potential degradation to non-reflection areas, region-aware methods restrict reflection removal to be effective in reflection-dominated areas. The earlier method [32] localizes reflection-dominated regions by using the gradient differences between $\mathbf{B}$ and $\mathbf{R}$. To complement the limitations of handcrafted features, the recent methods [46], [47] localize reflection-dominated regions using the recurrent neural network.

From Table 2, deep-learning-based methods can also be classified by their training strategy. The majority of deep-learning-based methods solve this problem in a supervised manner (*e.g.*, CEILNet [12], CRRN [37], CoRRN [13], ZN18 [15], YD18 [84], and CR19 [41]), which requires a massive number of paired samples as training data. Due to the difficulty in obtaining enough paired images from the real world, most methods use synthetic images for training, while it again causes the domain gap problem, which leads to performance degradation on unseen examples. To complement the limitation of synthetic data, ERRNet [14] train the supervised scheme by measuring the difference of unaligned real-world training pairs in the feature domain. Recent methods leverage advantages from image translation [36] to recover the background based on the weakly supervised framework [35], [39], and the unsupervised framework [42]. The weakly supervised frameworks in [35], [39] leverage advantages from CycleGAN [36] to estimate the background $\mathbf{B}$ without the requirement for paired training data. Due to the ill-posed nature of reflection removal, the unsupervised framework in [42] relies on the reflection motion difference between two images with the same $\mathbf{B}$ but different $\mathbf{R}$ for their separation.

#### *3.1.3 Strategy for training data generation*

For deep-learning-based methods, the training dataset plays a vital role in network training. We discuss each strategy for training data generation in this section and summarize them in Table 2.

**Synthetic strategy.** The synthetic strategy aims at synthesizing a mixture image by adding a background image and a reflection image. Based on Equation (2), earlier non-learning-based methods (*e.g.*, LB14 [10]) directly utilize images from public datasets (*e.g.*, COCO [85]) by adding some blurring effects to reflection images. This simple strategy has obvious limitations. For example, it lacks consideration for the regional property. To address this issue, instead of simply adding two images, the CLIP method proposed in CEILNet [12] subtracts a value from the mixture image to simulate the reflection layers' regional property. The

CLIP method [12] is used by CEILNet [12], ZN18 [15], ERRNet [14], FY20 [42], and IBCLN [19]. Besides, instead of only considering the regional property, the recently proposed rendering-based strategy [43] renders four different images by considering the glass-effect and lens-effect.

Though the synthetic strategy shows promising results, it is still difficult to cover all complicated real-world scenarios. Besides, its blurring effect for different reflection images mainly comes from fixed parameters, leading to over-fitting issues.

**Half-synthetic strategy.** Instead of solely relying on two images without relationship to the "capturing-though-glass" process, the half-synthetic strategy uses the real-world reflection images captured by putting a black sheet of paper behind glass. Since the reflection images are all from the real world, they cover almost all reflection properties. Such coverage helps overcome the domain gap caused by the synthetic strategy. However, the linear additive relationship used to add background and real reflection images still leads to the domain gap problem since it cannot model the non-linear relationship of light rays received by the camera sensor. The half-synthetic strategy is considered by CRRN [37], CoRRN [13], MW19 [39], CR19 [41], and FIRR [2].

**Learning-based strategy.** Instead of using fixed weighting coefficients, the learning-based strategy proposed in [43] synthesizes mixture images using deep networks. By considering the non-linear relationship between $\mathbf{B}$ and $\mathbf{R}$, the learning-based synthetic strategy can add two images with spatially varying weighting coefficients estimated by deep networks.

**Mixed strategy.** The mixed strategy utilizes real and synthetic images as training data. The real images from the real world help alleviate the domain gap issues. The challenge for its utilization mainly comes from the curse of the refractive effect discussed in Section 2.1. Since the pixel shifts caused by the refractive effect may lead to artifacts during the training process, real images cannot be directly used to train the network. Some methods alleviate this issue by using the perceptual loss [14] or training the network in a weakly supervised manner [39]. This strategy is considered by ERRNet [14], ZN18 [15], and MW19 [39].

### 3.2 Multiple-image methods

The multiple-image methods adopt more than one image as the input. The images are taken with different conditions (*e.g.*, illuminations, viewpoints, different focuses, or varied polarizer angles). Due to more available information, the limitations that exist in single-image methods are partially solved. We summarize existing motion-based multiple-image methods in Table 3.

The first category of multiple-image methods exploits the motion cues between the background and reflection using at least two images of the same scene from different viewpoints. Assuming glass is closer to the camera, the projected motion of the background and reflection is different due to the visual parallax. The motion of each layer can be represented using parametric models, such as the translative motion [86], the affine transformation [58] and the homography [60]. In contrast to the fixed parametric motion, the dense motion field provides more general modeling of layer motions represented by per-pixel motion vectors. Existing reflection removal methods estimate the dense motion field for each layer using optical flow [61], [87], SIFT flow [54], [55], [59], the pixel-wise flow field [16] and co-saliency [62]. Recently, there are also methods [2] using FlowNet [88] to estimate the motion relationship between different input images. For the image sequence from a video clip, AC19 [64] models the appearance and motion patterns among different frames by using 3D CNN. Besides, the recent method [50] also utilizes a deep network to estimate the dense optical flow.

The second category of multiple-image methods can be represented as a linear combination of the background and reflection: The $i$-th image is represented as $\mathbf{I}_i = \alpha_i \mathbf{B}_i + \beta_i(\mathbf{R}_i \otimes k_i)$, where weighting coefficients $\alpha_i$ and $\beta_i$ can be estimated by taking a sequence of images using special devices or in different environments, *e.g.*, by rotating the polarizer [6], [7], [8], [69], [70], [71], [72], [73], repetitive dynamic behaviors [74], and different illuminations [75].

The third category of multiple-image methods takes a set of images under special conditions and camera settings, such as using flash and non-flash images [76], [77], [78], different focuses [79], light field camera [65], [66], [67], [68], images captured by dual-pixel sensors [63], and two images taken by the front and back camera of a mobile phone [80]. The recently proposed method [51] solves this problem by using a pair of images with a normal image and a flash-only image taken in a dark environment with only a flash on.

Due to the additional information from multiple images, the problem becomes less ill-posed or even well-posed. However, special data capture requirements (*e.g.*, observing different layer motions or using polarizers) limit such methods for practical use, especially for mobile devices or images downloaded from the Internet.

### 3.3 Image vs. Irradiance

As discussed in Section 2, the meanings of $\mathbf{I}$, $\mathbf{B}$, and $\mathbf{R}$ differ depending on context. The relationship among the three parameters can either be defined by looking at from the images after camera ISP (nonlinear w.r.t. scene radiance) or raw images (linear w.r.t. scene radiance) that record irradiance received by the sensor (*e.g.*, works [71], [89] following [5]). Among the latter category, the two methods proposed by Lei *et al.* [51], [73] further explore the physical and linear relationship using the RAW data. They also introduce a new dataset [90] with the consideration for RAW images. The two categories are all reasonable under specific context. Obviously, if the three parameters are defined as the irradiance, the RAW data can avoid the influence from camera ISP. Based on our surveys, the polarizer-based methods [6], [7], [8], [69], [70], [71], [72], [73] denote $\mathbf{I}$, $\mathbf{B}$, and $\mathbf{R}$ as "irradiance" to better discuss the influences caused by the polarizer, while almost all other methods denote the three parameters as "image" or "layer".

## 4 Benchmark Dataset

To investigate the image formation process and the performance of existing methods, we extend our previous $\text{SIR}^2$ to $\text{SIR}^{2+}$ . We summarize the three parts of $\text{SIR}^{2+}$ in Table 4.

### 4.1 Indoor dataset

The indoor dataset is purposely designed to include common priors with organized parameters for the thorough evaluations of state-of-the-art methods (*e.g.* [1], [9], [10], [11]). The images inside are captured using a Nikon D5300 camera with a 300 mm lens. For the indoor dataset, we use three steps to capture a triplet of images: 1) The mixture image captured through glass; 2) the

TABLE 4: Details of the proposed dataset. "Original res." denotes the original image resolution. "Resized res." denotes the image resolution for experiments. TIN denotes the number of total images. SN denotes the number of scenes.

| | Original res. | Resized res. | TIN | SN |
|---|---|---|---|---|
| Outdoor | $540 \times 400$ | $224 \times 288$ | 300 | 100 |
| Indoor | $540 \times 400$ | $224 \times 288$ | 1200 | 40 |
| In-the-wild | $540 \times 400$ | $224 \times 288$ | 200 | 100 |

ground truth of $\mathbf{R}$ by putting a sheet of black cloth behind glass; and 3) the ground truth of $\mathbf{B}$ by removing glass. The indoor dataset contains 40 indoor scenes composed of solid objects (20 scenes) and postcards (20 scenes) with 1200 images in total. 20 scenes among the indoor dataset are composed of a set of solid objects. We use commonly available daily-life objects (*e.g.*, ceramic mugs, plush toys, fruits, *etc.*) for both background and reflection scenes. Another 20 scenes of the indoor dataset use different postcards as the background and reflection scenes. The indoor dataset mainly explores the influence of the in-focus reflection, the out-of-focus reflection, and the ghosting effect by considering the following aspects:

**Camera DoF.** We use seven different aperture sizes {F11, F13, F16, F19, F22, F27, F32} to create various DoFs for our data capture and choose seven different exposure time {1/3 s, 1/2 s, 1/1.5 s, 1 s, 1.5 s, 2 s, 3 s} corresponding to the seven aperture settings to make the brightness of each picture approximately constant. We denote such variation as "F-variance" for short and keep using the same glass with 5 mm thickness when varying DoF. The "F-Variance" mainly influences reflection blur levels. The reflection layers taken under F32 are the sharpest, and the reflection layers taken under F11 have the greatest blur.

**Glass thickness.** To explore how different glass thickness affects the effectiveness of existing methods, we place three different glass with the thickness of {3 mm, 5 mm, 10 mm} (denoted as {T3, T5, T10} and "T-variance" for short thereafter) one by one during the data capture under a fixed aperture size F32 and exposure time 3 s. The reflections taken with T10 and T3 show the largest and smallest spatial shift, respectively.

### 4.2 Outdoor dataset

For the outdoor dataset, we bring our setup out of the lab to capture images with real-world objects of complicated reflectance (car, tree leaves, glass windows, *etc.*), various distances and scales (residential halls, gardens, and lecture rooms, *etc.*), and different illuminations (direct sunlight, cloudy skylight, and twilight, *etc.*). The outdoor dataset contains 100 image triplets with 300 images. To better investigate the influence of the properties and effects discussed in Section 2.1, we categorize examples into four categories as: in-focus reflection (35), out-of-focus reflection (62), ghosting effect (16), and low-transmitted reflection (23)[1].

### 4.3 In-the-wild dataset

The in-the-wild dataset is proposed to cover more diverse properties resulted from real-world scenarios (*e.g.*, the glass color and curvatures). The glass in this dataset could be from various scenarios (*e.g.*, showcase, window, or windshield), which facilitate the evaluation for different algorithms under purely wild scenarios. Due to the special environment, the in-the-wild dataset contains more examples captured through tinted glass. From the examples shown in Figure 2, the blue and light gray glass changes the property of the light rays from background scenes. The in-the-wild dataset contains 100 image pairs with 200 images. We categorize examples into four categories: In-focus reflection (26), out-of-focus reflection (58), low-transmitted reflection (16), and tinted glass (55).

1. Due to each category's overlap, the total number of image triplets in all categories does not equal the number of image triplets in this dataset.

### 4.4 Differences between SIR$^{2+}$ and other datasets

One recent work [90] also proposes a new large-scale dataset, which specifically considers the RAW data. Besides, we both feel it is essential to address the influence of tinted and curved glass. However, the categorization for different glass thickness and reflection types in SIR$^{2+}$ can also help find the bottleneck of mainstream methods under specific settings. On the other hand, our dataset also contains the images captured through obscure glass.

## 5 Experiments

In this section, we use the SIR$^{2+}$ dataset to evaluate representative single-image reflection removal algorithms, LB14 [10], WS16 [11], NR17 [30], CEILNet [12], CoRRN [13], ZN18 [15], YD18 [84], YW19 [33], ERRNet [14], and IBCLN [19] for both quantitative accuracy (w.r.t. to the ground truth) and visual quality.

For each non-learning-based method, we use their original codes. Since almost all deep-learning-based methods have used SIR$^2$ dataset in their experiments, we first test their performance on the indoor and outdoor datasets inherited from SIR$^2$ [17]. Then, we evaluate their cross-dataset performance by testing their released models on the in-the-wild dataset. Finally, we also fine-tune their models using new images to evaluate different strategies for training data generation. To investigate the damage to non-reflection regions, we use the error metric values between the mixture image and the ground truth image as the baseline similar to previous settings [13], [32]. The evaluation images are all resized to $224 \times 288$. We show the execution time for this image size of each method in Table 6.

### 5.1 Error metrics

We consider the perceptual similarity as the complement to PSNR, which cannot faithfully measure the similarity of two images in a way that coincides with human judgment [18].

We first adopt the perceptually-motivated error metric SSIM [91], which evaluates the similarity of two images from the luminance, contrast, and structure components. By considering the structure distortion for the image assessment, SSIM models the human visual system with high sensitivity to structure distortion [92], [93]. On the other hand, we also use Structure Index (SI) [13], [32] to focus more on the structural similarity by removing the components for luminance and contrast in SSIM. Besides, we further adopt LPIPS [18], which better models the humane judgement by extracting the features from the pre-trained image classification network.

Moreover, due to the regional property of the reflection, SSIM designed for the whole image plane may not reflect the performance of reflection removal unbiasedly on local regions. We,

TABLE 5: Summary for existing reflection removal datasets. The following properties are considered: the image are captured manually or downloaded from the Internet (Source); whether the dataset has the background ground truth (Background GT) and Reflection GT or not; whether the dataset considers regular factors and tinted glass or not; the dataset is designed for single- or multiple-image methods; the number of image sets (Image set) and the total number of images (Image number). Regular factors in this place mean the gradually changing settings for the image capture (*e.g.*, the glass with different thickness).

| | Source | Backgound GT | Reflection GT | Regular factors | Tinted glass | Single/Multiple | Image set | Image number |
|---|---|---|---|---|---|---|---|---|
| Seq12 [10] | Captured | × | × | × | × | Multiple | 12 | 55 |
| Seq2 [16] | Captured | ✓ | ✓ | × | × | Multiple | 2 | 14 |
| Real20 [15] | Captured | ✓ | × | × | × | Single | 109 | 218 |
| Net45 [14] | Internet | × | × | × | × | Single | 45 | 45 |
| $SIR^2$ [17] | Captured | ✓ | ✓ | ✓ | × | Single | 500 | 1500 |
| Seq162 [63] | Captured | ✓ | × | × | × | Multiple | 162 | 636 |
| Face90 [2] | Captured | × | × | × | × | Multiple | 90 | 180 |
| Nature [19] | Captured | × | × | × | × | Multiple | 200 | 400 |
| Real100 [43] | Captured | ✓ | × | × | × | Single | 100 | 200 |
| CDR [90] | Captured | ✓ | ✓ | × | ✓ | Single | 1063 | 3189 |
| P&N [49] | Captured | ✓ | ✓ | × | × | Single | 40 | 80 |
| $SIR^{2+}$ [17] | Captured | ✓ | ✓ | ✓ | ✓ | Single | 600 | 1700 |

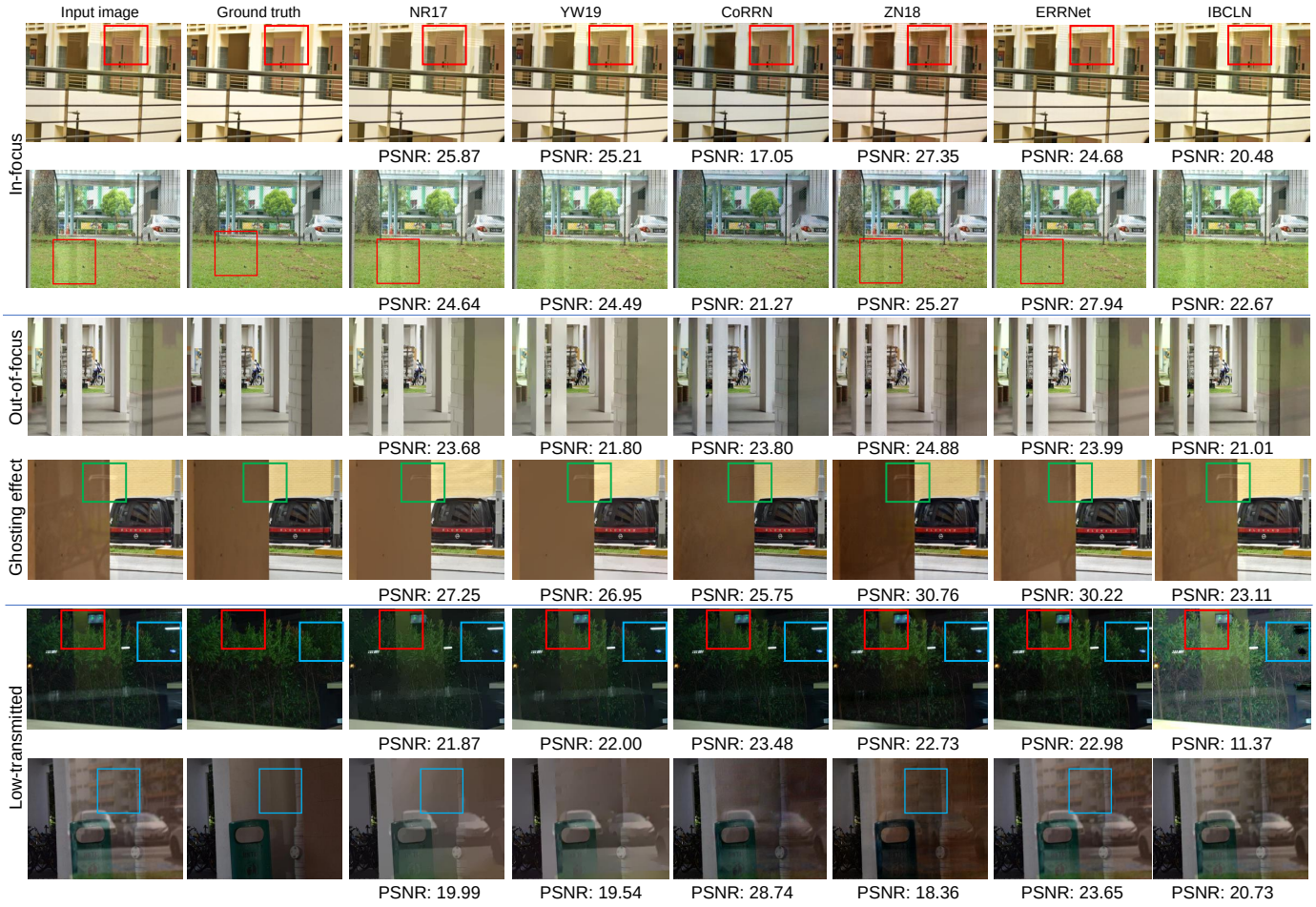



Fig. 3: Examples from the outdoor dataset with four types of reflections. We show the results obtained by NR17 [30], YW19 [33], CoRRN [13], ZN18 [15], ERRNet [14], and IBCLN [19]. We adjust the contrast of the result obtained by IBCLN [19] in low-transmitted reflection (fifth row) to show the dark-hole effect lablled by blue box in this case.

therefore, adopt the regional SSIM, denoted as $SSIM_r$, to complement the limitations of SSIM. We manually label reflection-dominated regions and evaluate SSIM values at these regions similar to previous methods [13], [32]. At last, we also adopt LMSE and NCC as error metrics for reference.

### 5.2 Evaluations on the outdoor dataset

We evaluate the overall performance on the outdoor dataset using the error metrics above and show their quantitative performance in Table 6.

From the overall performance in Table 6, deep-learning-based

TABLE 6: The average error metric values on the outdoor and in-the-wild dataset. The execution speed on an image with size $224 \times 288$ is shown next to each method in the first column. ↑ and ↓ denote the higher and lower values are better, respectively.

| | Outdoor dataset | | | | | | | In-the-wild dataset | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SSIM↑ | SI↑ | $SSIM_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SSIM↑ | SI↑ | $SSIM_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ |
| Baseline | 0.895 | 0.923 | 0.815 | 0.089 | 24.49 | 0.930 | 0.187 | 0.799 | 0.864 | 0.750 | 0.162 | 19.13 | 0.891 | 0.285 |
| LB14 [10] (1.112s) | 0.877 | 0.946 | 0.838 | 0.111 | 21.14 | 0.913 | 0.213 | 0.687 | 0.856 | 0.659 | 0.203 | 14.90 | 0.873 | 0.325 |
| WS16 [11] (7.012) | 0.901 | 0.938 | 0.859 | 0.095 | 23.49 | 0.906 | 0.222 | 0.775 | 0.851 | 0.745 | 0.192 | 18.12 | 0.847 | 0.351 |
| NR17 [30] (29.539s) | 0.858 | 0.894 | 0.844 | 0.093 | 23.80 | 0.934 | 0.160 | 0.785 | 0.852 | 0.749 | 0.178 | 18.99 | 0.886 | 0.294 |
| YW19 [33] (0.129s) | 0.877 | 0.922 | 0.846 | 0.096 | 22.34 | 0.899 | 0.221 | 0.780 | 0.859 | 0.750 | 0.177 | 17.83 | 0.843 | 0.365 |
| CoRRN [13] (0.024s) | 0.902 | 0.929 | 0.878 | 0.071 | 24.59 | 0.947 | 0.126 | 0.741 | 0.861 | 0.715 | 0.175 | 15.57 | 0.892 | 0.280 |
| ZN18 [15] (0.288s) | 0.883 | 0.926 | 0.834 | 0.093 | 23.45 | 0.935 | 0.149 | 0.758 | 0.860 | 0.722 | 0.178 | 17.09 | 0.876 | 0.313 |
| YD18 [84] (0.018s) | 0.871 | 0.924 | 0.849 | 0.094 | 21.01 | 0.941 | 0.243 | 0.757 | 0.863 | 0.721 | 0.185 | 17.82 | 0.872 | 0.443 |
| ERRNet [14] (3.179s) | 0.895 | 0.922 | 0.832 | 0.085 | 25.18 | 0.935 | 0.188 | 0.764 | 0.862 | 0.727 | 0.164 | 18.11 | 0.899 | 0.248 |
| SIRRBL [40] (0.013s) | 0.830 | 0.867 | 0.768 | 0.161 | 20.85 | 0.883 | 0.275 | 0.746 | 0.826 | 0.707 | 0.219 | 17.43 | 0.851 | 0.358 |
| IBCLN [19] (1.692s) | 0.896 | 0.918 | 0.823 | 0.086 | 24.33 | 0.940 | 0.940 | 0.784 | 0.858 | 0.739 | 0.168 | 18.846 | 0.87 | 0.348 |

TABLE 7: Benchmark results using indoor dataset with F-variance and T-variance. The light blue and gray rows indicate non-learning-based methods and deep-learning-based methods, respectively.

| F-var | SSIM ↑ | | SI↑ | | $SSIM_r$↑ | | LPIPS↓ | | PSNR↑ | | NCC↑ | | LMSE↓ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | F11 | F32 | F11 | F32 | F11 | F32 | F11 | F32 | F11 | F32 | F11 | F32 | F11 | F32 |
| Baseline | 0.884 | 0.867 | 0.917 | 0.898 | 0.837 | 0.811 | 0.111 | 0.136 | 21.966 | 22.155 | 0.959 | 0.958 | 0.101 | 0.101 |
| LB14 [10] | 0.832 | 0.812 | 0.907 | 0.883 | 0.815 | 0.783 | 0.135 | 0.161 | 17.604 | 18.029 | 0.953 | 0.951 | 0.116 | 0.115 |
| SK15 [1] | 0.820 | 0.811 | 0.867 | 0.857 | 0.803 | 0.808 | 0.190 | 0.199 | 19.177 | 19.458 | 0.867 | 0.874 | 0.231 | 0.218 |
| WS16 [11] | 0.869 | 0.840 | 0.916 | 0.891 | 0.837 | 0.819 | 0.121 | 0.146 | 20.988 | 21.106 | 0.952 | 0.949 | 0.110 | 0.115 |
| NR17 [30] | 0.863 | 0.844 | 0.902 | 0.880 | 0.843 | 0.810 | 0.112 | 0.142 | 21.426 | 21.578 | 0.960 | 0.957 | 0.107 | 0.109 |
| YW19 [33] | 0.873 | 0.859 | 0.910 | 0.894 | 0.858 | 0.832 | 0.111 | 0.137 | 21.173 | 21.330 | 0.953 | 0.948 | 0.113 | 0.118 |
| CoRRN [13] | 0.909 | 0.888 | 0.933 | 0.912 | 0.891 | 0.854 | 0.085 | 0.107 | 21.064 | 21.230 | 0.973 | 0.968 | 0.080 | 0.088 |
| ZN18 [15] | 0.882 | 0.853 | 0.922 | 0.897 | 0.866 | 0.819 | 0.124 | 0.162 | 19.937 | 19.518 | 0.965 | 0.963 | 0.083 | 0.085 |
| YD18 [84] | 0.893 | 0.876 | 0.917 | 0.902 | 0.865 | 0.840 | 0.105 | 0.122 | 21.872 | 21.900 | 0.958 | 0.958 | 0.117 | 0.123 |
| ERRNet [14] | 0.904 | 0.874 | 0.925 | 0.890 | 0.864 | 0.815 | 0.100 | 0.133 | 23.453 | 23.078 | 0.961 | 0.955 | 0.085 | 0.100 |
| SIRRBL [40] | 0.813 | 0.793 | 0.848 | 0.828 | 0.771 | 0.743 | 0.206 | 0.229 | 18.857 | 18.849 | 0.889 | 0.886 | 0.219 | 0.227 |
| IBCLN [19] | 0.902 | 0.879 | 0.922 | 0.908 | 0.898 | 0.865 | 0.115 | 0.127 | 23.162 | 22.962 | 0.970 | 0.964 | 0.067 | 0.079 |

| T-var | SSIM ↑ | | SI↑ | | $SSIM_r$↑ | | LPIPS↓ | | PSNR↑ | | NCC↑ | | LMSE↓ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | T3 | T10 | T3 | T10 | T3 | T10 | T3 | T10 | T3 | T10 | T3 | T10 | T3 | T10 |
| Baseline | 0.868 | 0.869 | 0.898 | 0.899 | 0.816 | 0.819 | 0.138 | 0.134 | 22.143 | 22.160 | 0.956 | 0.956 | 0.104 | 0.103 |
| LB14 [10] | 0.811 | 0.812 | 0.882 | 0.884 | 0.783 | 0.782 | 0.177 | 0.168 | 17.737 | 18.000 | 0.949 | 0.948 | 0.121 | 0.123 |
| SK15 [1] | 0.803 | 0.815 | 0.849 | 0.862 | 0.781 | 0.793 | 0.204 | 0.193 | 19.368 | 19.648 | 0.860 | 0.873 | 0.236 | 0.219 |
| WS16 [11] | 0.842 | 0.854 | 0.889 | 0.880 | 0.820 | 0.824 | 0.149 | 0.144 | 21.045 | 21.061 | 0.947 | 0.947 | 0.118 | 0.117 |
| NR17 [30] | 0.845 | 0.847 | 0.880 | 0.881 | 0.813 | 0.816 | 0.144 | 0.141 | 21.584 | 21.560 | 0.955 | 0.955 | 0.112 | 0.112 |
| YW19 [33] | 0.884 | 0.860 | 0.893 | 0.895 | 0.832 | 0.834 | 0.140 | 0.137 | 21.283 | 21.260 | 0.947 | 0.948 | 0.120 | 0.119 |
| CoRRN [13] | 0.890 | 0.892 | 0.915 | 0.903 | 0.864 | 0.864 | 0.108 | 0.105 | 21.273 | 21.318 | 0.966 | 0.965 | 0.089 | 0.091 |
| ZN18 [15] | 0.853 | 0.884 | 0.893 | 0.897 | 0.821 | 0.820 | 0.162 | 0.158 | 19.490 | 19.741 | 0.962 | 0.964 | 0.086 | 0.081 |
| YD18 [84] | 0.897 | 0.875 | 0.903 | 0.902 | 0.845 | 0.841 | 0.123 | 0.123 | 21.841 | 21.909 | 0.957 | 0.960 | 0.125 | 0.120 |
| ERRNet [14] | 0.874 | 0.873 | 0.881 | 0.889 | 0.821 | 0.822 | 0.134 | 0.134 | 23.140 | 23.132 | 0.956 | 0.955 | 0.098 | 0.102 |
| SIRRBL [40] | 0.796 | 0.793 | 0.829 | 0.827 | 0.752 | 0.751 | 0.138 | 0.134 | 18.848 | 18.932 | 0.884 | 0.889 | 0.234 | 0.224 |
| IBCLN [19] | 0.878 | 0.879 | 0.885 | 0.900 | 0.898 | 0.835 | 0.123 | 0.120 | 22.868 | 23.149 | 0.961 | 0.963 | 0.085 | 0.081 |

methods achieve more promising results than non-learning-based methods since the handcrafted prior cannot describe the complicated properties of outdoor scenes (*e.g.*, non-uniform reflection blur levels). Moreover, since almost all methods achieve lower values than the baseline performance, they may cause damage to non-reflection regions.

From the comparison in Table 8, the increasing reflection intensity or decreasing reflection blur level leads to worse performance. For example, almost all methods achieve better results on out-of-focus reflections than that on both in-focus and low-transmitted reflections. It can also be observed from Figure 3, where the out-of-focus reflection is more effectively removed. Besides, the low-transmitted reflection also poses challenges for each method, since $SSIM_r$ values for this type are obviously lower than the values for other types in Table 8 and it also cannot be effectively removed from the results in Figure 3. Though IBCLN [19] successfully removes a part of the low-transmitted reflection, it fails to recover the background and leads to the dark-hole effects in Figure 3. Finally, though each method achieves the best results on examples with ghosting effects in Table 8, it is partly because the examples with ghosting effects are mainly with small intensity. From the fourth row in Figure 3, the ghosting effects do not benefit the reflection removal process. Most methods effectively remove the out-of-focus reflection but fail to remove the reflection with the ghosting effect (labeled by the green box).

### 5.3 Evaluations for generalization ability

From discussions in Section 5.2, the performance of each method tends to decline as the reflection becomes sharp. To better investigate the generalization ability of each method to different

TABLE 8: The overall evaluations for the categorized outdoor dataset. The light blue and gray rows indicate non-learning-based methods and deep-learning-based methods, respectively. ↑ and ↓ denote the higher and lower values are better, respectively.

| | In-focus reflection | | | | | | | Out-of-focus reflection | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ |
| Baseline | 0.878 | 0.777 | 0.119 | 23.964 | 0.919 | 0.246 | 0.904 | 0.908 | 0.838 | 0.076 | 24.947 | 0.945 | 0.136 | 0.937 |
| LB14 [10] | 0.819 | 0.768 | 0.130 | 20.721 | 0.909 | 0.226 | 0.903 | 0.843 | 0.814 | 0.099 | 21.418 | 0.934 | 0.169 | 0.933 |
| WS16 [11] | 0.880 | 0.816 | 0.113 | 23.584 | 0.917 | 0.243 | 0.913 | 0.894 | 0.857 | 0.084 | 23.598 | 0.902 | 0.206 | 0.934 |
| NR17 [30] | 0.868 | 0.809 | 0.116 | 23.346 | 0.917 | 0.221 | 0.901 | 0.897 | 0.872 | 0.079 | 24.203 | 0.947 | 0.120 | 0.932 |
| YW19 [33] | 0.860 | 0.804 | 0.119 | 22.263 | 0.901 | 0.245 | 0.901 | 0.889 | 0.868 | 0.082 | 22.485 | 0.899 | 0.207 | 0.932 |
| CoRRN [13] | 0.884 | 0.836 | 0.093 | 23.218 | 0.934 | 0.168 | 0.911 | 0.913 | 0.900 | 0.059 | 25.415 | 0.956 | 0.102 | 0.941 |
| ZN18 [15] | 0.868 | 0.801 | 0.115 | 22.672 | 0.918 | 0.185 | 0.909 | 0.894 | 0.851 | 0.079 | 23.950 | 0.948 | 0.125 | 0.937 |
| YD18 [84] | 0.855 | 0.813 | 0.115 | 21.205 | 0.931 | 0.276 | 0.908 | 0.884 | 0.870 | 0.080 | 21.090 | 0.949 | 0.219 | 0.936 |
| ERRNet [14] | 0.871 | 0.783 | 0.114 | 24.268 | 0.909 | 0.296 | 0.901 | 0.913 | 0.861 | 0.068 | 25.817 | 0.953 | 0.124 | 0.938 |
| SIRRBL [40] | 0.810 | 0.714 | 0.182 | 20.351 | 0.870 | 0.297 | 0.847 | 0.847 | 0.797 | 0.147 | 21.290 | 0.893 | 0.255 | 0.883 |
| IBCLN [19] | 0.875 | 0.785 | 0.113 | 23.666 | 0.923 | 0.220 | 0.899 | 0.911 | 0.844 | 0.071 | 24.853 | 0.952 | 0.128 | 0.933 |

| | Ghosting effect | | | | | | | Low-transmitted reflection | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ |
| Baseline | 0.919 | 0.862 | 0.078 | 25.305 | 0.940 | 0.141 | 0.936 | 0.860 | 0.724 | 0.122 | 24.096 | 0.867 | 0.285 | 0.862 |
| LB14 [10] | 0.864 | 0.838 | 0.078 | 21.358 | 0.930 | 0.165 | 0.938 | 0.791 | 0.721 | 0.154 | 21.671 | 0.856 | 0.255 | 0.864 |
| WS16 [11] | 0.913 | 0.902 | 0.079 | 24.036 | 0.886 | 0.225 | 0.941 | 0.864 | 0.775 | 0.136 | 24.007 | 0.863 | 0.298 | 0.872 |
| NR17 [30] | 0.905 | 0.900 | 0.085 | 24.106 | 0.936 | 0.145 | 0.928 | 0.856 | 0.777 | 0.132 | 23.756 | 0.866 | 0.248 | 0.869 |
| YW19 [33] | 0.893 | 0.898 | 0.089 | 22.365 | 0.859 | 0.271 | 0.928 | 0.847 | 0.774 | 0.139 | 22.576 | 0.847 | 0.284 | 0.869 |
| CoRRN [13] | 0.900 | 0.901 | 0.074 | 25.000 | 0.931 | 0.162 | 0.932 | 0.868 | 0.815 | 0.110 | 23.957 | 0.883 | 0.190 | 0.873 |
| ZN18 [15] | 0.879 | 0.834 | 0.099 | 23.048 | 0.914 | 0.202 | 0.926 | 0.864 | 0.773 | 0.128 | 23.665 | 0.873 | 0.194 | 0.873 |
| YD18 [84] | 0.904 | 0.895 | 0.083 | 20.892 | 0.931 | 0.271 | 0.934 | 0.845 | 0.806 | 0.117 | 21.582 | 0.890 | 0.297 | 0.878 |
| ERRNet [14] | 0.920 | 0.873 | 0.068 | 25.197 | 0.937 | 0.165 | 0.938 | 0.855 | 0.732 | 0.139 | 24.288 | 0.850 | 0.368 | 0.856 |
| SIRRBL [40] | 0.857 | 0.810 | 0.157 | 21.757 | 0.886 | 0.261 | 0.880 | 0.792 | 0.693 | 0.208 | 20.814 | 0.808 | 0.340 | 0.809 |
| IBCLN [19] | 0.914 | 0.860 | 0.078 | 24.374 | 0.937 | 0.170 | 0.930 | 0.865 | 0.744 | 0.133 | 24.064 | 0.873 | 0.249 | 0.858 |

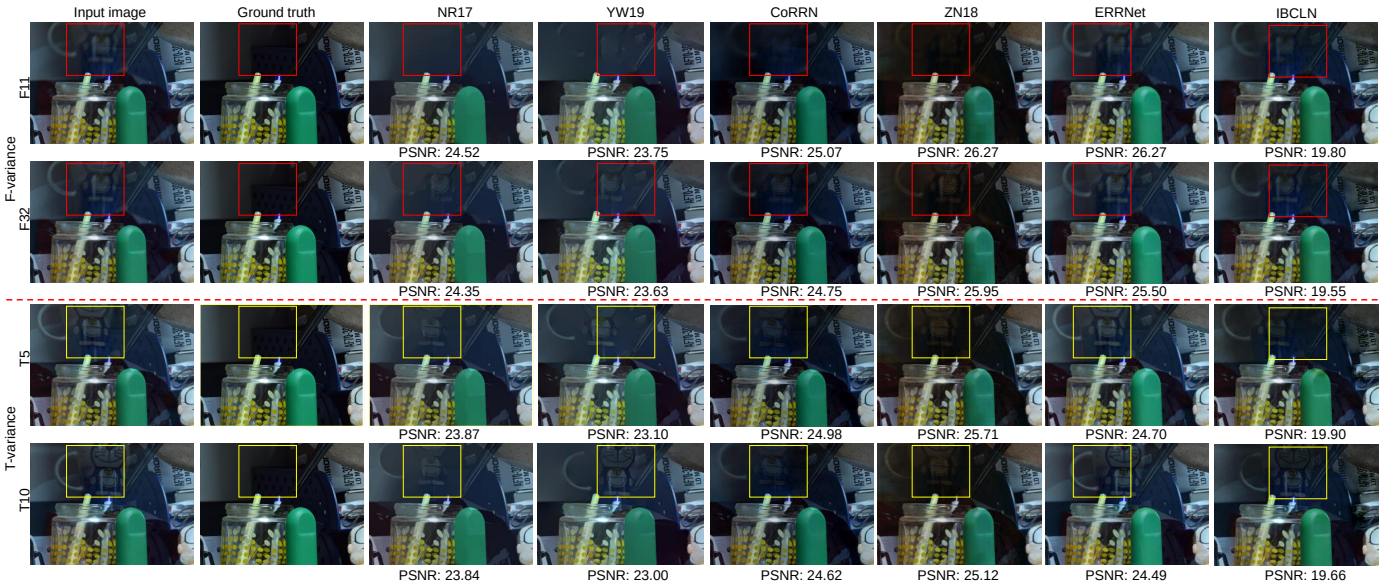



Fig. 4: Examples from the indoor dataset with regular settings. We show the results obtained by NR17 [30], YW19 [33], CoRRN [13], ZN18 [15], ERRNet [14], and IBCLN [19].

reflections, we first conduct more experiments by using the indoor dataset with regular settings. Then, we conduct several experiments to evaluate the generalization ability of each method using our in-the-wild dataset.

#### 5.3.1 *Generalization ability to reflection regular settings*

As an important prior for reflection removal, the reflection blur level is considered by both non-learning and deep-learning-based methods. Since our indoor dataset has considered both "T-variance" and "F-variance", we test the performance of each method on the categorized indoor dataset to evaluate their generalization ability to different blur levels.

From the results in Table 7, except CoRRN [13], YD18 [84], and ERRNet [14], almost all methods introduce new artifacts to final results due to the lower error metric values compared with the baseline. Besides, deep-learning-based methods achieve generally better performance than non-learning-based methods.

For results on F-variance, except SK15 [1], all methods show

TABLE 9: The overall evaluations for the categorized in-the-wild dataset. The light blue and gray rows indicate non-learning-based methods and deep-learning-based methods, respectively. ↑ and ↓ denote the higher and lower values are better, respectively.

| | In-focus reflection | | | | | | | Out-of-focus reflection | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ |
| Baseline | 0.767 | 0.692 | 0.216 | 16.988 | 0.835 | 0.401 | 0.830 | 0.840 | 0.814 | 0.111 | 19.652 | 0.945 | 0.135 | 0.899 |
| LB14 [10] | 0.669 | 0.616 | 0.247 | 14.242 | 0.819 | 0.426 | 0.820 | 0.734 | 0.725 | 0.154 | 15.699 | 0.931 | 0.177 | 0.893 |
| WS16 [11] | 0.749 | 0.693 | 0.230 | 16.568 | 0.812 | 0.428 | 0.815 | 0.809 | 0.803 | 0.154 | 18.539 | 0.889 | 0.228 | 0.884 |
| NR17 [30] | 0.752 | 0.695 | 0.224 | 16.802 | 0.831 | 0.404 | 0.817 | 0.825 | 0.813 | 0.133 | 19.634 | 0.941 | 0.151 | 0.887 |
| YW19 [33] | 0.754 | 0.701 | 0.221 | 16.396 | 0.811 | 0.422 | 0.825 | 0.817 | 0.811 | 0.133 | 18.324 | 0.889 | 0.245 | 0.893 |
| CoRRN [13] | 0.719 | 0.675 | 0.222 | 14.927 | 0.839 | 0.364 | 0.826 | 0.782 | 0.776 | 0.129 | 16.313 | 0.943 | 0.172 | 0.893 |
| ZN18 [15] | 0.731 | 0.672 | 0.225 | 16.010 | 0.821 | 0.413 | 0.828 | 0.806 | 0.790 | 0.128 | 17.788 | 0.929 | 0.159 | 0.893 |
| YD18 [84] | 0.743 | 0.685 | 0.220 | 16.133 | 0.815 | 0.526 | 0.833 | 0.789 | 0.780 | 0.147 | 7.232 | 0.916 | 0.222 | 0.895 |
| ERRNet [14] | 0.717 | 0.654 | 0.231 | 16.158 | 0.833 | 0.419 | 0.814 | 0.808 | 0.793 | 0.118 | 18.718 | 0.948 | 0.122 | 0.897 |
| SIRRBL [40] | 0.715 | 0.654 | 0.264 | 15.679 | 0.802 | 0.465 | 0.792 | 0.785 | 0.770 | 0.176 | 17.936 | 0.901 | 0.213 | 0.859 |
| IBCLN [19] | 0.752 | 0.679 | 0.223 | 16.653 | 2.090 | 0.310 | 0.823 | 0.828 | 0.808 | 0.116 | 19.327 | 0.937 | 0.139 | 0.895 |

| | Low-transmitted | | | | | | | Tinted glass | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | LMSE↓ | SI↑ |
| Baseline | 0.758 | 0.666 | 0.209 | 16.288 | 0.826 | 0.497 | 0.828 | 0.750 | 0.667 | 0.204 | 14.753 | 0.852 | 0.367 | 0.848 |
| LB14 [10] | 0.631 | 0.575 | 0.257 | 12.603 | 0.811 | 0.530 | 0.824 | 0.645 | 0.592 | 0.235 | 12.419 | 0.832 | 0.404 | 0.843 |
| WS16 [11] | 0.740 | 0.662 | 0.226 | 15.829 | 0.755 | 0.582 | 0.816 | 0.732 | 0.664 | 0.217 | 14.388 | 0.803 | 0.427 | 0.837 |
| NR17 [30] | 0.747 | 0.662 | 0.214 | 16.250 | 0.823 | 0.488 | 0.818 | 0.740 | 0.670 | 0.206 | 14.955 | 0.847 | 0.373 | 0.838 |
| YW19 [33] | 0.741 | 0.659 | 0.218 | 15.526 | 0.740 | 0.602 | 0.823 | 0.734 | 0.667 | 0.208 | 14.321 | 0.809 | 0.427 | 0.843 |
| CoRRN [13] | 0.685 | 0.617 | 0.219 | 13.626 | 0.822 | 0.458 | 0.822 | 0.690 | 0.636 | 0.206 | 12.966 | 0.862 | 0.331 | 0.850 |
| ZN18 [15] | 0.696 | 0.624 | 0.226 | 14.291 | 0.811 | 0.557 | 0.823 | 0.710 | 0.646 | 0.213 | 13.871 | 0.842 | 0.379 | 0.848 |
| YD18 [84] | 0.723 | 0.629 | 0.222 | 15.177 | 0.819 | 0.979 | 0.826 | 0.719 | 0.654 | 0.213 | 14.626 | 0.840 | 0.452 | 0.850 |
| ERRNet [14] | 0.749 | 0.684 | 0.175 | 15.574 | 0.872 | 0.286 | 0.843 | 0.719 | 0.651 | 0.200 | 14.210 | 0.864 | 0.345 | 0.847 |
| SIRRBL [40] | 0.701 | 0.618 | 0.262 | 15.342 | 0.807 | 0.565 | 0.786 | 0.698 | 0.631 | 0.254 | 13.975 | 0.812 | 0.419 | 0.810 |
| IBCLN [19] | 0.739 | 0.649 | 0.218 | 15.607 | 0.791 | 0.761 | 0.820 | 0.736 | 0.655 | 0.208 | 14.526 | 0.847 | 0.389 | 0.843 |

a clear decreasing tendency on it. It is mainly because these methods all assume different blur levels between the background and reflection when designing priors or generating training data. Moreover, with the decreasing blur levels, the reflection becomes more similar to the background, and such similarity also increases the difficulty of differentiating the two layers. For SK15 [1], it is difficult to tell how F-variance influences its results because it relies on the ghosting effect instead of reflection blur levels. However, it achieves better performance in F32 of SSIM$_r$ results due to more apparent spatial shifts of reflections.

For the performance on T-variance, from Table 7, since the performance on glass with different thicknesses does not show noticeable differences, the ghosting effect seems not to be a key influential factor for existing methods. As a method specifically designed for the ghosting effect, SK15 [1] shows generally better performance on T10, where the ghosting effect of reflection is the most obvious. Since SK15 [1] needs to estimate the spatial shift distances of reflections with ghosting effects, the larger distance may make this detection easier and lead to better performance.

The images in Figure 4 also prove the observations from Table 7. With decreasing reflection blur levels, the refections labeled by the red boxes in Figure 4 become more difficult to be removed. Meanwhile, from the regions labeled by the yellow boxes in Figure 4, it is difficult to find an obvious difference from the results obtained under different "F-variance".

#### 5.3.2 Generalization ability to in-the-wild scenes

Since most existing deep-learning-based methods are fine-tuned by considering the reflection properties of existing datasets, we further evaluate their cross-dataset generalization ability on the newly collected in-the-wild dataset. From the results in Table 6 and Figure 5, the performance on this unseen dataset becomes relatively worse. Besides, from error metric values in Table 6, the gap between non-learning-based methods and deep-learning-based methods becomes closer. The examples in Figure 5 also prove this observation. Both the learning and non-learning-based methods cannot remove low-transmitted reflections (the bottom part in Figure 5). The non-learning-based methods even achieve better results than the deep-learning-based methods (*e.g.*, the regions labeled by the red boxes in the first part of Figure 5). Then, since existing methods are mainly designed for colorless glass, they cannot handle tinted glass, where the light rays from background scenes are also seriously corrupted (*e.g.*, the last example in Figure 5).

From Figure 5 and Table 9 , the results on the categorized in-the-wild dataset also show that tinted glass introduces more difficulty to existing methods since they are mainly designed for colorless glass. Besides, since all scenes in this dataset are unseen to each method, they do not show apparent differences for each type.

#### 5.3.3 The influences of reflection types

In this section, we summarize the influences of reflection types on different methods. Almost all methods show superior performance on out-of-focus reflections due to their assumption for this type. For example, as the non-learning-based methods, LB14 [10], WS16 [11], NR17 [30], and YW19 [33] all utilize the gradient difference caused by different blur levels between $\mathbf{R}$ and $\mathbf{B}$. Other deep-learning-based methods also assume blurred reflection when generating their training data.

The ghosting effect does not obviously affect the results since all methods show similar performance on the glass with different

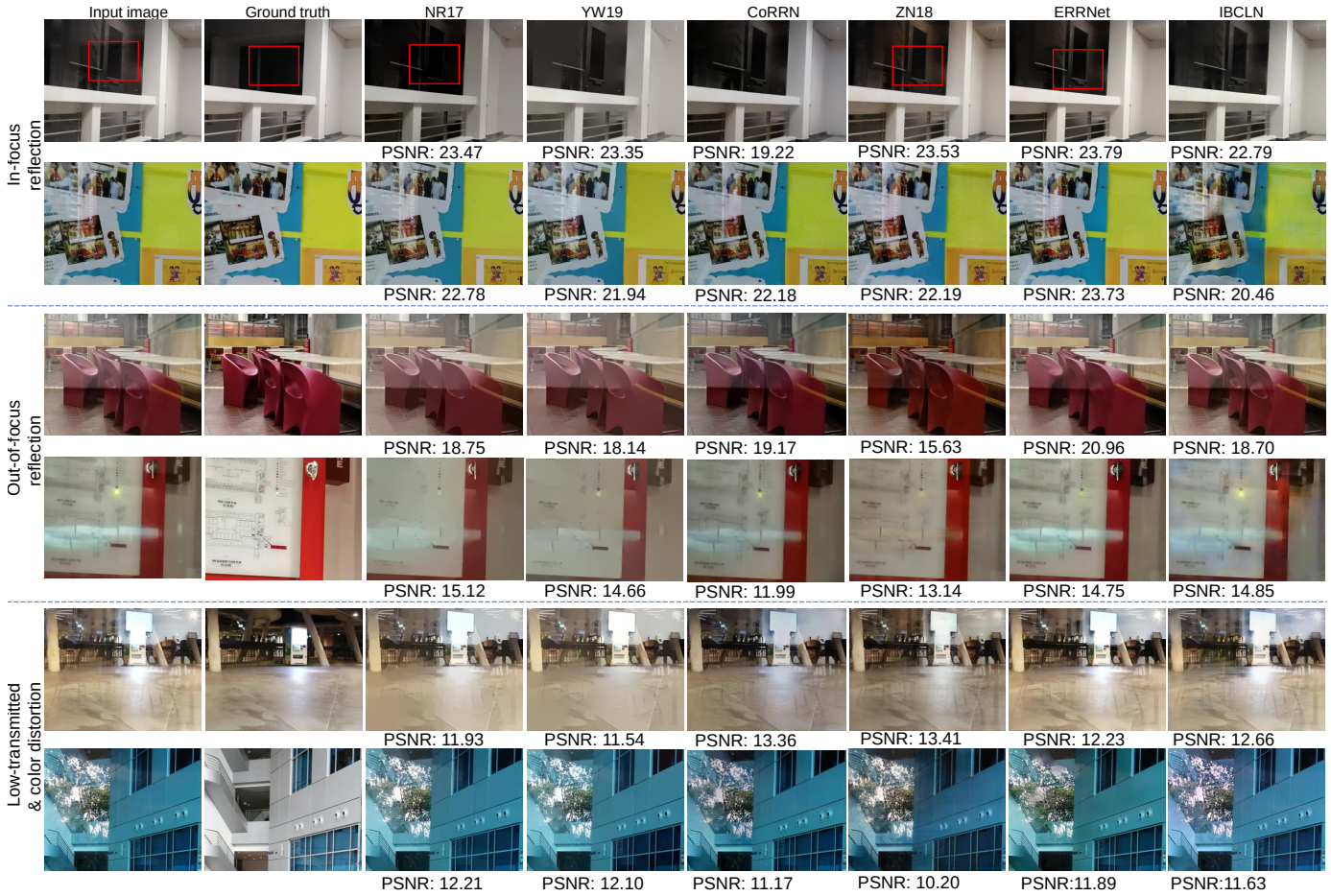



Fig. 5: Examples with four types of reflections from the in-the-wild dataset. We show the results obtained by NR17 [30], YW19 [33], CoRRN [13], ZN18 [15], ERRNet [14], and IBCLN [19]. The last example is with the low-transmitted reflection and color distortion.

TABLE 10: The results obtained by CoRRN [13], ZN18 [15], ERRNet [14], and IBCLN [19] based on by the synthetic strategy, mixed strategy with images captured through tinted glass (Mixed strategy with t-image), mixed strategy without images captured through tinted glass (Mixed strategy w/o t-image), and learning-based strategy.

| | Synthetic strategy | | | | | | | Mixed strategy with t-image | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | SI↑ | LMSE↓ | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | SI↑ | LMSE↓ |
| CoRRN [13] | 0.821 | 0.807 | 0.121 | 21.04 | 0.871 | 0.873 | 0.296 | 0.829 | 0.837 | 0.111 | 24.07 | 0.883 | 0.882 | 0.286 |
| ZN18 [15] | 0.835 | 0.793 | 0.138 | 21.48 | 0.869 | 0.874 | 0.162 | 0.877 | 0.847 | 0.111 | 25.64 | 0.921 | 0.896 | 0.150 |
| ERRNet [14] | 0.827 | 0.797 | 0.132 | 20.75 | 0.871 | 0.869 | 0.167 | 0.885 | 0.869 | 0.077 | 24.65 | 0.925 | 0.911 | 0.154 |
| IBCLN [19] | 0.829 | 0.770 | 0.150 | 20.63 | 0.872 | 0.871 | 0.287 | 0.841 | 0.819 | 0.134 | 22.14 | 0.885 | 0.881 | 0.265 |
| | Mixed strategy w/o t-image | | | | | | | Learning-based strategy | | | | | | |
| | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | SI↑ | LMSE↓ | SSIM↑ | SSIM$_r$↑ | LPIPS↓ | PSNR↑ | NCC↑ | SI↑ | LMSE↓ |
| CoRRN [13] | 0.814 | 0.811 | 0.118 | 22.36 | 0.874 | 0.869 | 0.290 | 0.712 | 0.709 | 0.136 | 19.72 | 0.837 | 0.838 | 0.411 |
| ZN18 [15] | 0.841 | 0.826 | 0.125 | 22.56 | 9.876 | 0.875 | 0.158 | 0.721 | 0.712 | 0.132 | 19.94 | 0.857 | 0.845 | 0.341 |
| ERRNet [14] | 0.858 | 0.829 | 0.113 | 21.26 | 0.873 | 0.872 | 0.156 | 0.728 | 0.718 | 0.139 | 18.54 | 0.814 | 0.800 | 0.496 |
| IBCLN [19] | 0.823 | 0.805 | 0.143 | 20.16 | 0.870 | 0.870 | 0.271 | 0.738 | 0.729 | 0.152 | 19.80 | 0.841 | 0.867 | 0.386 |

thicknesses. Besides, the second row of Figure 4 also illustrates that the reflection with ghosting effect can also be removed if it is blurred.

The low-transmitted reflection is another challenge for almost all methods since the image formation model they rely on does not explicitly consider this type. More discussions related to the low-transmitted reflection can be found in Section 6.1.

Though deep-learning-based methods are proposed to address the limited description ability of handcrafted priors, the dependence on training images leads to their poorer generalization ability to unseen examples. Since the out-of-focus reflection is employed as the backbone for synthesizing training data, almost all deep-learning-based methods show promising results on examples with the out-of-focus reflection and degraded performance for other types. This problem can be alleviated by employing real images for training. From the results shown in Table 8, the methods (ERRNet [14] and ZN18 [15]) trained based on the mixed strategy show more consistent results than others.

### 5.4 Evaluations for training data strategy

In this section, we conduct several experiments to evaluate the strategy for training data generation. For fairness, we train each

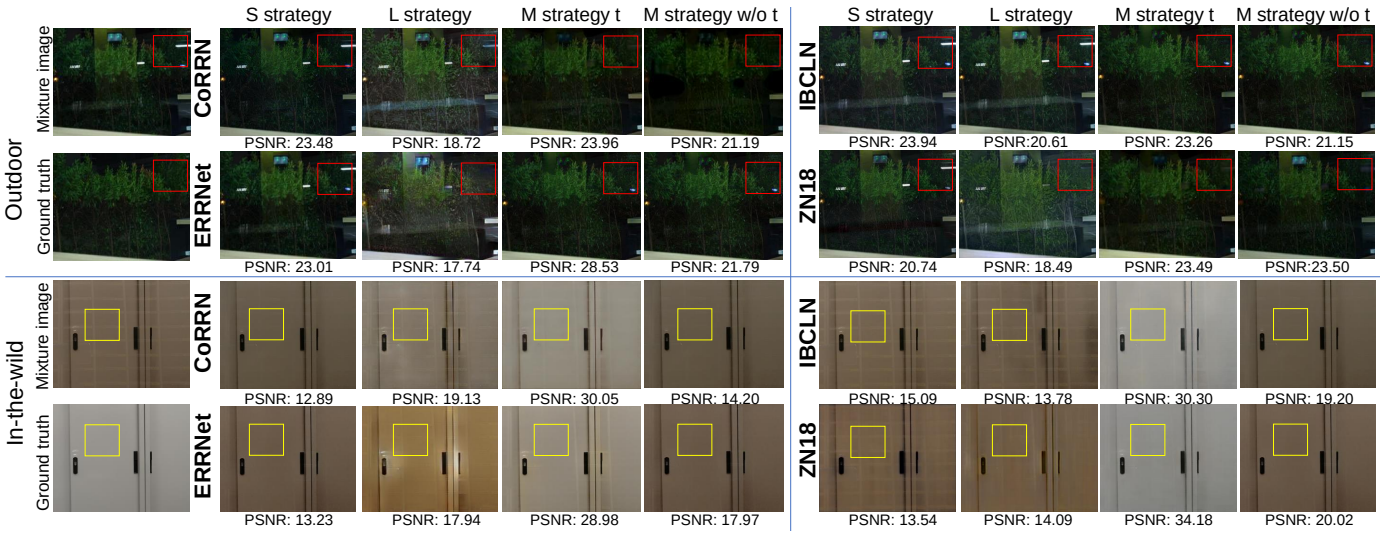



Fig. 6: Examples from the outdoor dataset and the in-the-wild dataset, respectively. We show the results obtained by NR17 [30], YW19 [33], CoRRN [13], ZN18 [15], and ERRNet [14]. The red boxes in the first part indicate the regions with low-transmitted reflections. The yellow boxes in the second part indicate the regions with out-of-focus reflections. "M strategy t" denotes the mixture strategy with tinted images. "M strategy w/o t" denotes the mixture strategy without tinted images.

method with different image sizes ($96\times128$, $128\times196$, and $224\times288$) to avoid potential over-fitting issues. We choose four representative methods with available training codes, CoRRN [13], ZN18 [15], ERRNet [14], IBCLN [19] to evaluate the effectiveness of each strategy. Each method is trained with its default batch size and $400$ epochs, a number larger their default epoch sizes, to ensure the final convergence.

#### 5.4.1 Synthetic strategy

We first train each network by only using the synthetic data generated by CLIP [12] and the half-synthetic strategy. From the results shown in Figure 6, since CLIP [12] mainly considers out-of-focus reflections, the networks trained on this strategy show acceptable results on the examples with out-of-focus reflections (*e.g.*, the blurred reflections labeled by yellow boxes in the in-the-wild scenes of Figure 6). However, it shows degraded performance for other types (*e.g.*, the low-transmitted reflections labeled by the red boxes and the color distortion in the in-the-wild scenes of Figure 6). Besides, since existing data synthesis methods cannot generate images with the color distortion effect, the models trained on the synthetic strategy cannot generalize to examples captured through tinted glass.

#### 5.4.2 Mixed strategy

Then, we train the network by combing the synthetic dataset with real images from Real20 [15] and Nature dataset [19]. Besides, 200 real image pairs are captured using the three-step way similar to SIR$^{2+}$ for the training purpose. Among these captured images, about 150 image pairs are captured through tinted glass. From the results shown in Table 10 (Mixed strategy with t-image), the models trained using real images achieve better performance than the models without real images. From Figure 6, the models trained by the mixed strategy with t-image show better removal performance for the low-transmitted reflection and in-focus reflection. For example, the low-transmitted reflections labelled by red boxes in Figure 6 are successfully removed, and the in-focus reflections are also attenuated. For the examples captured through tinted glass, the mixed strategy with t-image also shows more promising results. From the examples shown in the last row of Figure 6, the models trained with the mixed strategy with t-image remove reflections and recovers color information.

We further train a network by only using synthetic images and the real images from Real20 [15] and Nature dataset [19]. From the results in Table 10 (Mixed strategy w/o t-image), the models trained on this strategy cannot outperform the models trained by the mixed strategy with t-image though they are still better than the models trained by the synthetic and learning-based strategies. This is mainly due to the poor ability to handle the background color distortion caused by tinted glass. From Figure 6, for the images captured through tinted glass, the reflections can be suppressed, but the color distortion cannot be corrected. The experiments also further prove that our SIR$^{2+}$ contains more diversified scenarios that existing datasets cannot cover.

#### 5.4.3 Learning-based strategy

We further evaluate the learning-based strategy proposed in [40]. The synthetic images are generated by the method proposed in [40]. From the results shown in Figure 6, the models trained by this learning-based strategy also show some promising results (*e.g.*, the results in the second row for ERRNet [14] Figure 6). However, since the reflections always distribute unevenly across the whole image plane, the deep network may not learn reflection properties properly. It leads to the additional artifacts shown in Figure 6, which worsen the error metric values.

## 6 Open Problems for Existing Methods

Based on our benchmark dataset and evaluation, we list several open problems in existing methods to inspire future research on single-image reflection removal.

### 6.1 Transmitted vs. low-transmitted

Since existing reflection removal methods mainly focus on the transmitted reflection, it leads to failure cases with the low-

transmitted reflection in Figure 3 and Figure 5. For the low-transmitted reflection, since light rays from background objects are almost occluded, the classical reflection removal strategy inherited from layer separation methods may not successfully recover the background layer from a mixture image.

Instead of solely regarding reflection removal as an image separation problem, future methods may leverage advantages from inpainting problems to handle the low-transmitted reflection. For example, by using the contextual attention [94] designed for the image inpainting problem, the reflection removal methods can utilize surrounding information to restore low-transmitted regions. The obstacle is that the regions to be recovered for image inpainting are assumed to be known, while they are unknown for reflection removal. Some recent methods [46], [47] have been proposed to localize reflection regions, which show the dawn for this problem.

### 6.2 The dependence for training data

From discussions in Section 5.4, existing deep-learning-based methods show degraded performance for unseen examples. It may be due to the low generalization ability of their synthetic training data. The synthetic images mainly consider the blurring effect and the regional property [12]. The two properties are not enough to cover the various phenomena in the real world. Though the mixed strategy ameliorates the performance by including real images for training, it is still difficult to obtain real images with a sufficiently large diversity.

One possible direction is to consider the domain adaptation/generalization [95] in the data synthesis stage. An example is the learning-based strategy [40] discussed in Section 5.4.3, which utilizes the generative framework to synthesize mixture images. However, since reflections always distribute unevenly and sparsely across the whole image plane, the deep network may not learn reflection properties properly. It also makes the generalization for the reflection removal problem more difficult than that for other tasks (*e.g.*, image deraining and image dehazing) with dense degradation distribution. How to extract features properly from unevenly distributed reflections should be considered by future methods.

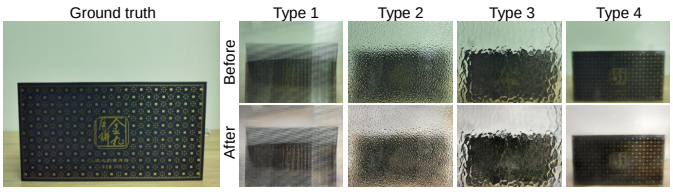



Fig. 7: One example captured through four different obscure glass. The first row denotes the image before processing and the second row contains the images after processing.

### 6.3 General vs. specific

Most methods are designed for general scenes with irregular and diverse properties. The difficulty in handling the general scenarios is related to the dependency for training data discussed in Section 6.2. Since the synthetic training images can only cover limited properties, this dilemma makes it difficult for existing methods to generalize to unseen general scenarios. Besides, the changing background scenes under general scenarios further increase the difficulty to estimate the missing details occluded by reflections, especially by low-transmitted reflections.

Instead of solely focusing on general scenes, future methods may address the reflection removal problem on specific scenes (*e.g.*, the face images [2] or text images [41] taken through glass). A more stable environment may facilitate the solution of this problem under specific scenarios. Besides, it is also easier to analyze the reflection property in specific scenes.

### 6.4 Background degradation

Reflection removal aims at seeing through glass by improving the background scene visibility. From this aspect, it should be able to handle the interference caused by $g(\cdot)$ and $f(\mathbf{R})$ in Equation (1). The examples captured through tinted glass in SIR$^{2+}$ have shown their limitations in handling the background color distortion caused by $g(\cdot)$. A more challenging example can be found in Figure 7, where obscure glass significantly changes the background appearance and the existing method fails on it.

The background degradation caused by $g(\cdot)$ introduces new difficulties for reflection removal. To handle more complicated background degradation, reflection removal should adopt new strategies. Our experiments have shown that existing models can perform better for examples with color distortion by using training images with the similar distortion. For the more challenging example in Figure 7, future methods may leverage advantages from NLOS problem [96], [97] by using electromagnetic radiation to recover the appearance.

We capture 30 examples through obscure glass when collecting data. Since these images are beyond the settings of existing methods, we do not use them for benchmarking purposes. Though one related method [98] has been proposed for this problem, the 30 examples are still the first dataset for obscure glass. We provide it for future researchers.

## 7 Conclusions

As an extension of our previous work [17], we aiming at providing a more comprehensive survey by exploring the reflection properties and discussing the recently proposed deep learning methods. Then, to investigate the performance of the mainstream methods surveyed in our paper, we expand and rearrange our previous SIR$^2$ dataset [17] to a SIR$^{2+}$ dataset with additional images from the wild environment. Finally, we discuss the performance of existing methods from the results on the SIR$^{2+}$ dataset and propose several open problems for future researchers.

## References

[1] Y. Shih, D. Krishnan, F. Durand, and W. T. Freeman, "Reflection removal using ghosting cues," *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2015.

[2] R. Wan, B. Shi, H. Li, L.-Y. Duan, and A. C. Kot, "Face image reflection removal," *International Journal of Computer Vision*, pp. 1–15, 2020.

[3] H. Zhang, J. Yang, Y. Zhang, N. M. Nasrabadi, and T. S. Huang, "Close the loop: Joint blind image restoration and recognition with sparse representation prior," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 770–777.

[4] C. Liu, H.-Y. Shum, and W. T. Freeman, "Face hallucination: Theory and practice," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 115–134, 2007.

[5] H. Farid and E. H. Adelson, "Separating reflections and lighting using independent components analysis," in *Proc. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 1999.

[6] Y. Y. Schechner, J. Shamir, and N. Kiryati, "Polarization-based decorrelation of transparent layers: The inclination angle of an invisible surface," in *Proc. International Conference on Computer Vision (ICCV)*, 1999.

[7] Y. Lyu, Z. Cui, S. Li, M. Pollefeys, and B. Shi, "Reflection separation using a pair of unpolarized and polarized images," in *Advances in Neural Information Processing Systems*, 2019, pp. 14 532–14 542.

[8] P. Wieschollek, O. Gallo, J. Gu, and J. Kautz, "Separating reflection and transmission images in the wild," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 89–104.

[9] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007.

[10] Y. Li and M. S. Brown, "Single image layer separation using relative smoothness," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2014.

[11] R. Wan, B. Shi, A. H. Tan, and A. C. Kot, "Depth of field guided reflection removal," in *Proc. International Conference on Image Processing (ICIP)*, 2016.

[12] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "A generic deep architecture for single image reflection removal and image smoothing," in *Proc. International Conference on Computer Vision (ICCV)*, 2017.

[13] R. Wan, B. Shi, H. Li, L.-Y. Duan, A.-H. Tan, and A. K. Chichung, "CoRRN: Cooperative reflection removal network," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[14] K. Wei, J. Yang, Y. Fu, D. Wipf, and H. Huang, "Single image reflection removal exploiting misaligned training data and network enhancements," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8178–8187.

[15] X. Zhang, R. Ng, and Q. Chen, "Single image reflection separation with perceptual losses," *arXiv preprint arXiv:1806.05376*, 2018.

[16] 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.

[17] 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.

[18] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2018.

[19] C. Li, Y. Yang, K. He, S. Lin, and J. E. Hopcroft, "Single image reflection removal through cascaded refinement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3565–3574.

[20] H. Ritland, "Relation between refractive index and density of a glass at constant temperature," *Journal of the American Ceramic Society*, vol. 38, no. 2, pp. 86–88, 1955.

[21] R. Wan, B. Shi, H. Li, L.-Y. Duan, and A. C. Kot, "Reflection scene separation from a single image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2398–2406.

[22] H. Bach and N. Neuroth, *The properties of optical glass*. Springer Science & Business Media, 1998.

[23] C. Tuchinda, S. Srivannaboon, and H. W. Lim, "Photoprotection by window glass, automobile glass, and sunglasses," *Journal of the American Academy of Dermatology*, vol. 54, no. 5, pp. 845–854, 2006.

[24] A. P. Pentland, "A new sense for depth of field," *IEEE transactions on pattern analysis and machine intelligence*, no. 4, pp. 523–531, 1987.

[25] S. G. Angulo, J. R. Alonso, M. Strojnik, A. Fernández, G. García-Torales, J. Flores, and J. A. Ferrari, "All-in-focus image reconstruction robust to ghosting effect," in *Applications of Digital Image Processing XLI*, vol. 10752. International Society for Optics and Photonics, 2018, p. 1075229.

[26] S. Stallinga and B. Rieger, "Accuracy of the gaussian point spread function model in 2d localization microscopy," *Optics express*, vol. 18, no. 24, pp. 24 461–24 476, 2010.

[27] B. Buttery and G. Davison, "The ghost artifact." *Journal of Ultrasound in Medicine*, vol. 3, no. 2, pp. 49–52, 1984.

[28] Y. Huang, Y. Quan, Y. Xu, R. Xu, and H. Ji, "Removing reflection from a single image with ghosting effect," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 34–45, 2019.

[29] Y.-C. Chung, S.-L. Chang, J.-M. Wang, and S.-W. Chen, "Interference reflection separation from a single image," in *Proc. WACV*, 2009.

[30] N. Arvanitopoulos, R. Achanta, and S. Susstrunk, "Single image reflection suppression," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2017.

[31] 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.

[32] 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.

[33] Y. Yang, W. Ma, Y. Zheng, J.-F. Cai, and W. Xu, "Fast single image reflection suppression via convex optimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8141–8149.

[34] P. Chandramouli, M. Noroozi, and P. Favaro, "Convnet-based depth estimation, reflection separation and deblurring of plenoptic images," in *Proc. ACCV*, 2016.

[35] D. Lee, M.-H. Yang, and S. Oh, "Generative single image reflection separation," *arXiv preprint arXiv:1801.04102*, 2018.

[36] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[37] 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 Pattern Recognition (CVPR)*, 2018.

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

[39] D. Ma, R. Wan, B. Shi, A. C. Kot, and L.-Y. Duan, "Learning to jointly generate and separate reflections," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2444–2452.

[40] Q. Wen, Y. Tan, J. Qin, W. Liu, G. Han, and S. He, "Single image reflection removal beyond linearity," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3771–3779.

[41] C. Wang, R. Wan, F. Gao, B. Shi, and L.-Y. Duan, "Learning to remove reflections for text images," in *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, 2019.

[42] Q. Fan, Y. Yin, D. Chen, Y. Wang, A. Aviles-Rivero, R. Li, C.-B. Schnlieb, D. Lischinski, and B. Chen, "Deep reflection prior," *arXiv preprint arXiv:1912.03623*, 2020.

[43] S. Kim, Y. Huo, and S.-E. Yoon, "Single image reflection removal with physically-based rendering," *arXiv preprint arXiv:1904.11934*, 2019.

[44] Y. Liu, Y. Li, S. You, and F. Lu, "Semantic guided single image reflection removal," *arXiv preprint arXiv:1907.11912*, 2019.

[45] Y.-C. Chang, C.-N. Lu, C.-C. Cheng, and W.-C. Chiu, "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*, 2021, pp. 2033–2042.

[46] Y. Li, M. Liu, Y. Yi, Q. Li, D. Ren, and W. Zuo, "Two-stage single image reflection removal with reflection-aware guidance," *arXiv preprint arXiv:2012.00945*, 2020.

[47] Z. Dong, K. Xu, Y. Yang, H. Bao, W. Xu, and R. W. Lau, "Location-aware single image reflection removal," *arXiv preprint arXiv:2012.07131*, 2020.

[48] Q. Zheng, B. Shi, J. Chen, X. Jiang, L.-Y. Duan, and A. C. Kot, "Single image reflection removal with absorption effect," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 395–13 404.

[49] Y. Hong, Q. Zheng, L. Zhao, X. Jiang, A. C. Kot, and B. Shi, "Panoramic image reflection removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7762–7771.

[50] Y.-L. Liu, W.-S. Lai, M.-H. Yang, Y.-Y. Chuang, and J.-B. Huang, "Learning to see through obstructions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 215–14 224.

[51] C. Lei and Q. Chen, "Robust reflection removal with reflection-free flash-only cues," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 811–14 820.

[52] Q. Yan, Y. Xu, and X. Yang, "Separation of weak reflection from a single superimposed image using gradient profile sharpness," in *Proc. ISCAS*, 2013.

[53] Q. Yan, Y. Xu, X. Yang, and T. Nguyen, "Separation of weak reflection from a single superimposed image," *IEEE Signal Processing Letter*, vol. 21, no. 10, pp. 1173–1176, 2014.

[54] C. Sun, S. Liu, T. Yang, B. Zeng, Z. Wang, and G. Liu, "Automatic reflection removal using gradient intensity and motion cues," in *Proc. of ACM MM*, 2016.

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

[56] D. Heydecker, G. Maierhofer, A. I. Aviles-Rivero, Q. Fan, C.-B. Schönlieb, and S. Süsstrunk, "Mirror, mirror, on the wall, who's got the clearest image of them all?-a tailored approach to single image reflection removal," *arXiv preprint arXiv:1805.11589*, 2018.

[57] 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 Pattern Recognition (CVPR)*, 2017.

[58] 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.

[59] T. Sirinukulwattana, G. Choe, and I. S. Kweon, "Reflection removal using disparity and gradient-sparsity via smoothing algorithm," in *Proc. International Conference on Image Processing (ICIP)*, 2015.

[60] X. Guo, X. Cao, and Y. Ma, "Robust separation of reflection from multiple images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2187–2194.

[61] J. Yang, H. Li, Y. Dai, and R. T. Tan, "Robust optical flow estimation of double-layer images under transparency or reflection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1410–1419.

[62] B.-J. Han and J.-Y. Sim, "Glass reflection removal using co-saliency-based image alignment and low-rank matrix completion in gradient domain," *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 4873–4888, 2018.

[63] A. Punnappurath and M. S. Brown, "Reflection removal using a dual-pixel sensor," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1556–1565.

[64] J.-B. Alayrac, J. Carreira, and A. Zisserman, "The visual centrifuge: Model-free layered video representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2457–2466.

[65] Q. Wang, H. Lin, Y. Ma, S. Kang, and J. Yu, "Automatic layer separation using light field imaging," *arXiv preprint arXiv:1506.04721*, 2015.

[66] Y. Ni, J. Chen, and L.-P. Chau, "Reflection removal based on single light field capture," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2017, pp. 1–4.

[67] N. Yun, J. Chen, and L.-P. Chau, "Reflection removal on single light field capture using focus manipulation," *IEEE Transactions on Computational Imaging*, 2018.

[68] T. Li, D. P. Lun, Y.-H. Chan *et al.*, "Robust reflection removal based on light field imaging," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1798–1812, 2018.

[69] H. Farid and E. H. Adelson, "Separating reflections and lighting using independent components analysis," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 1999.

[70] Y. Diamant and Y. Y. Schechner, "Overcoming visual reverberations," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2008.

[71] N. Kong, Y. Tai, and J. S. Shin, "A physically-based approach to reflection separation: from physical modeling to constrained optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[72] B. Sarel and M. Irani, "Separating transparent layers through layer information exchange," in *Proc. European Conference on Computer Vision (ECCV)*, 2004.

[73] C. Lei, X. Huang, M. Zhang, Q. Yan, W. Sun, and Q. Chen, "Polarized reflection removal with perfect alignment in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1750–1758.

[74] B. Sarel and Irani, "Separating transparent layers of repetitive dynamic behaviors," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2005.

[75] K. I. Diamantaras and T. Papadimitriou, "Blind separation of reflections using the image mixtures ratio," in *Proc. International Conference on Image Processing (ICIP)*, vol. 2. IEEE, 2005, pp. II–1034.

[76] A. Agrawal, R. Raskar, and R. Chellappa, "Edge suppression by gradient field transformation using cross-projection tensors," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 2301–2308.

[77] A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 828–835, 2005.

[78] Y. Chang, C. Jung, J. Sun, and F. Wang, "Siamese dense network for reflection removal with flash and no-flash image pairs," *Springer International Journal of Computer Vision*, 2020.

[79] Y. Y. Schechner, N. Kiryati, and R. Basri, "Separation of transparent layers using focus," *Springer International Journal of Computer Vision*, 2000.

[80] P. Kalwad, D. Prakash, V. Peddigari, and P. Srinivasa, "Reflection removal in smart devices using a prior assisted independent components analysis," in *Digital Photography XI*, vol. 9404. International Society for Optics and Photonics, 2015, p. 940405.

[81] R. Abiko and M. Ikehara, "Single image reflection removal based on gan with gradient constraint," *IEEE Access*, vol. 7, pp. 148 790–148 799, 2019.

[82] Z. Chi, X. Wu, X. Shu, and J. Gu, "Single image reflection removal using deep encoder-decoder network," *arXiv preprint arXiv:1802.00094*, 2018.

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

[84] J. Yang, D. Gong, L. Liu, and Q. Shi, "Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 654–669.

[85] 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 *European conference on computer vision*. Springer, 2014, pp. 740–755.

[86] 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.

[87] R. Szeliski, S. Avidan, and P. Anandan, "Layer extraction from multiple images containing reflections and transparency," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2000.

[88] P. Fischer, A. Dosovitskiy, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. Van der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," *arXiv preprint arXiv:1504.06852*, 2015.

[89] N. Kong, Y.-W. Tai, and J. S. Shin, "A physically-based approach to reflection separation: from physical modeling to constrained optimization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 2, pp. 209–221, 2013.

[90] C. Lei, X. Huang, C. Qi, Y. Zhao, W. Sun, Q. Yan, and Q. Chen, "A categorized reflection removal dataset with diverse real-world scenes," *arXiv preprint arXiv:2108.03380*, 2021.

[91] 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.

[92] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE signal processing magazine*, vol. 26, no. 1, pp. 98–117, 2009.

[93] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on image processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[94] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514.

[95] H. Li, S. Wang, R. Wan, and A. K. Chichung, "Gmfad: Towards generalized visual recognition via multi-layer feature alignment and disentanglement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[96] A. Turpin, V. Kapitany, J. Radford, D. Rovelli, K. Mitchell, A. Lyons, I. Starshynov, and D. Faccio, "3d imaging from multipath temporal echoes," *Physical Review Letters*, vol. 126, no. 17, p. 174301, 2021.

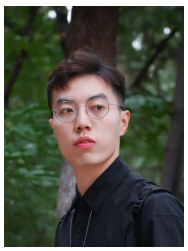
[97] N. Scheiner, F. Kraus, F. Wei, Phan *et al.*, "Seeing around street corners: Non-line-of-sight detection and tracking in-the-wild using doppler radar," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2068–2077.

[98] Q. Shan, B. Curless, and T. Kohno, "Seeing through obscure glass," in *European Conference on Computer Vision*. Springer, 2010, pp. 364–378.

**Renjie Wan** received his BEng degree from the University of Electronic Science and Technology of China in 2012 and the Ph.D. degree from Nanyang Technological University, Singapore, in 2019. He is currently an Assistant Professor of Hong Kong Baptist University, Hong Kong. He is the outstanding reviewer of ICCV 2019 and the recipient of the Microsoft CRSF Award, VCIP 2020 Best Paper Award, and the Wallenberg-NTU Presidential Postdoctoral Fellowship.

**Boxin Shi** 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 a Boya Young Fellow Assistant Professor and Research Professor at Peking University, where he leads the Camera Intelligence Lab. 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. His papers were awarded as Best Paper Runner-Up at International Conference on Computational Photography 2015 and selected as Best Papers from ICCV 2015 for IJCV Special Issue. He has served as an editorial board member of IJCV and an area chair of CVPR/ICCV. He is a senior member of IEEE.

**Haoliang Li** received his B.S. degree from the University of Electronic Science and Technology of China in 2013 and the Ph.D. degree from Nanyang Technological University, Singapore in 2018. He is currently an Assistant Professor at City University of Hong Kong. He is the recipient of the Wallenberg-NTU Presidential Postdoctoral Fellowship and VCIP 2020 Best Paper Award. His research interest is multimedia forensics and transfer learning.



**Yuchen Hong** is a Ph.D. candidate in the Electronics Engineering and Computer Science School of Peking University. He received the B.E. degree from Beijing University of Posts and Telecommuications in 2020. His research interests include computational photography and computer vision.

**Ling-Yu Duan** 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 is also with Peng Cheng Laboratory, Shenzhen, China, since 2019. He received 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 has published about 200 research papers. He received the IEEE ICME Best Paper Award in 2019/2020, the IEEE VCIP Best Paper Award in 2019, and 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 MPEG Compact Descriptor for Video Analytics (CDVA) standard (ISO/IEC 15938-15). Currently he is an Associate Editor of IEEE Transactions on Multimedia, ACM Transactions on Intelligent Systems and Technology and ACM Transactions on Multimedia Computing, Communications, and Applications, and serves as the area chairs of ACM MM and IEEE ICME. He is a member of the MSA Technical Committee in IEEE-CAS Society.

**Alex C. Kot (S'85-M'89-SM'98-F'06)** has been with the Nanyang Technological University, Singapore since 1991. He was Head of the Division of Information Engineering and Vice Dean Research at the School of Electrical and Electronic Engineering. Subsequently, he served as Associate Dean for College of Engineering for eight years. He is currently Professor and Director of Rapid-Rich Object SEarch (ROSE) Lab and NTU-PKU Joint Research Institute. He has published extensively in the areas of signal processing, biometrics, image forensics and security, and computer vision and machine learning. Dr. Kot served as Associate Editor for more than ten journals, mostly for IEEE transactions. He served the IEEE SP Society in various capacities such as the General Co-Chair for the 2004 IEEE International Conference on Image Processing and the Vice-President for the IEEE Signal Processing Society. He received the Best Teacher of the Year Award and is a co-author for several Best Paper Awards including ICPR, IEEE WIFS and IWDW, CVPR Precognition Workshop and VCIP. He was elected as the IEEE Distinguished Lecturer for the Signal Processing Society and the Circuits and Systems Society. He is a Fellow of IEEE, and a Fellow of Academy of Engineering, Singapore.