Single Image Reflection Separation via Component Synergy

Qiming Hu Xiaojie Guo*
College of Intelligence and Computing, Tianjin University, Tianjin, China
huqiming@tju.edu.cn, xj.max.guo@gmail.com

Abstract

The reflection superposition phenomenon is complex and widely distributed in the real world, which derives various simplified linear and nonlinear formulations of the problem. In this paper, based on the investigation of the weaknesses of existing models, we propose a more general form of the superposition model by introducing a learnable residue term, which can effectively capture residual information during decomposition, guiding the separated layers to be complete. In order to fully capitalize on its advantages, we further design the network structure elaborately, including a novel dual-stream interaction mechanism and a powerful decomposition network with a semantic pyramid encoder. Extensive experiments and ablation studies are conducted to verify our superiority over state-of-the-art approaches on multiple real-world benchmark datasets. Our code is publicly available at https://github.com/mingcv/DSRNet.

1. Introduction

As a typical layer superimposition scenario, pictures captured through glass-like surfaces may be blended with undesired reflection, which not only impairs the aesthetic value but also hinders the downstream tasks [31]. In the meantime, scenes of interest are possibly concealed in reflection [35]. Therefore, both the parts transmitted through a surface (transmission layer, $\mathbf{T}$) and those reflected (reflection layer, $\mathbf{R}$) are desired to be reconstructed from a superimposed image $\mathbf{I}$ to fulfill the practical demands.

Following a common assumption [17], $\mathbf{I}$ is linearly composed by $\mathbf{T}$ and $\mathbf{R}$ expressed as $\mathbf{I} = \mathbf{T} + \mathbf{R}$, which has been popular due to its simplicity. However, in real-world scenarios, the reflection and transmission layers are likely to be weakened due to diffusion and other problems during the superimposition [35]. Therefore, several methods [34, 39] introduce scalars α and β to the two components, respectively, obtaining $\mathbf{I} = \alpha\mathbf{T} + \beta\mathbf{R}$. However, the linear superimposition model is often violated due to over-exposure and other phenomena [37]. Thus, an alpha-matting map $\mathbf{W}$ is introduced to the model as $\mathbf{I} = \mathbf{W} \circ \mathbf{T} + \bar{\mathbf{W}} \circ \mathbf{R}$ with $\bar{\mathbf{W}} = \mathbf{1} - \mathbf{W}$, which, however, increases the degree of freedom and makes this ill-posed problem much harder. In view of all these limitations, it is not a trivial task to represent different kinds of reflection scenarios with a single model. In this work, we take a step forward in advancing the solution and deliver a more general form of the superimposition procedure by introducing a residue term as follows:

$$\mathbf{I} = \tilde{\mathbf{T}} + \tilde{\mathbf{R}} = \mathbf{T} + \mathbf{R} + \Phi(\mathbf{T}, \mathbf{R}), \quad (1)$$

where $\mathbf{T}$ and $\mathbf{R}$ denote the groundtruth scenes of the trans-

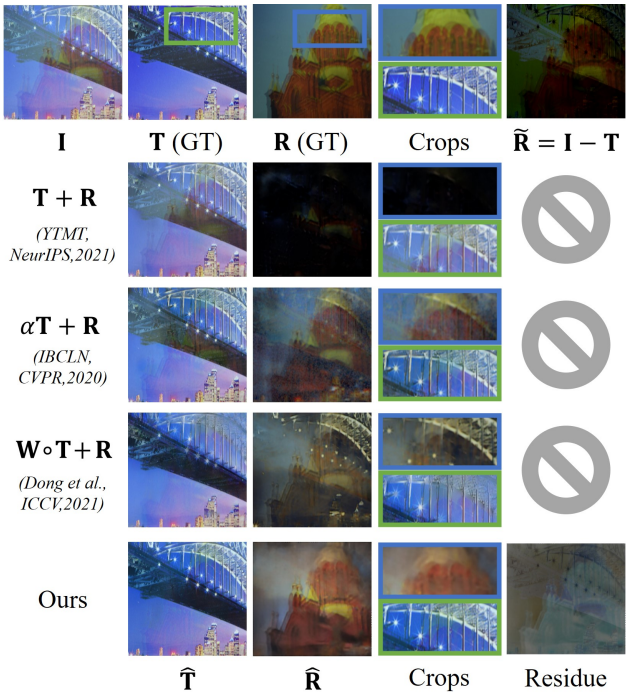

Figure 1: A visual example drawn from the SIR^2 dataset. The first row displays a real input $\mathbf{I}$ with ground-truth $\mathbf{T}$ and $\mathbf{R}$. In rows 2-5, the separations based on different physical models are compared, where $\hat{\mathbf{T}}$ and $\hat{\mathbf{R}}$ represent the predictions. Our method is able to produce a "Residue" term, separating the nonlinearity from the layer reconstructions.

*Corresponding author. This work was supported by National Natural Science Foundation of China under Grant no. 62072327.

mission and reflection layers, while their respective information contained in $\mathbf{I}$ after superimposition and other degradations and finally reach the camera sensors is represented by $\tilde{\mathbf{T}}$ and $\tilde{\mathbf{R}}$. $\Phi(\mathbf{T},\mathbf{R}) = \mathbf{I} - \mathbf{T} - \mathbf{R}$ is the residue of the reconstruction, which can be caused by attenuation, overexposure, etc. $\Phi(\cdot,\cdot)$ represents a group of functions that can be used to model the residue in specific situations. For example, $\Phi(\mathbf{T},\mathbf{R}) = 0$ in [23, 5, 17, 41, 36, 12], $\Phi(\mathbf{T},\mathbf{R}) = (\alpha-1)\mathbf{T}+(\beta-1)\mathbf{R}$ in [34, 39], $\Phi(\mathbf{T},\mathbf{R}) = -(\bar{\mathbf{W}}\circ\mathbf{T}+\mathbf{W}\circ\mathbf{R})$ in [37, 42]. In an effort to rule these different cases, a learnable module is desired to be introduced to model the residue $\Phi(\mathbf{T},\mathbf{R})$. To clarify it, we provide a visual example in Fig. 1, where the first row shows a superimposed image $\mathbf{I}$, its groundtruth transmission $\mathbf{T}$ (GT) and reflection $\mathbf{R}$ (GT) layers collected by SIR^2 dataset. According to the linear model, the guidance for reflection restoration is $\tilde{\mathbf{R}} = \mathbf{I} - \mathbf{T}$ with incomplete signals rather than $\mathbf{R}$ (GT) that contains full information, which not only impairs the training of networks but also hinders the downstream usages. Therefore, as we point out in Eq. (1), a promising way out of this dilemma is to utilize ground-truth layers $\mathbf{T}$ and $\mathbf{R}$ as guiding signals and employ an extra residual term to handle the non-linearity ("Residue" in Fig. 1). As shown, the components that violate the linear assumption are separated from the layer predictions $\hat{\mathbf{T}}$ and $\hat{\mathbf{R}}$, and thus more complete and precious layer separations than previous methods are obtained.

To further exploit the synergy between the components $\mathbf{T}$ and $\mathbf{R}$, we deliberate upon the facilitation of inter-component feature interaction. The efficiency of a dual-stream interactive network has been verified in SIRS problem by [12]. Building upon their analysis, we subsequently advance a more effective scheme. Inspired by the gated mechanisms [2, 32], we present a simple yet effective mutually-gated interaction diagram (MuGI) serving as a better feature interaction candidate, by means of which we design the MuGI block to build our decomposition network. Moreover, the hypercolumn [41], as a commonly used semantic information encoder in the SIRS task, tends to aggregate features in a lossy way. Therefore, we propose a dual-stream pyramid fusion network (DSFNet) to replace it, which hierarchically decomposes and fuses the multi-scale semantic information leveraging our proposed MuGI blocks and dual-stream fusion blocks (DSF Block). The roughly decomposed features of layers are further delivered into the dual-stream fine-grained decomposition phase (DS-DNet). The two sub-networks constitute the main branch of our **D**ual-stream **S**emantic-aware network with **R**esidual correction, namely DSRNet.

In summary, our main contributions are as follows:

- We build a general form for SIRS by introducing a learnable residue term, which is more flexible to different scenarios and boosts the separation of layers;
- We exploit the synergy of features via mutually-gated interaction block within a dual-stream semantic-aware network, which facilitates information usage;
- Extensive experiments are conducted to demonstrate the effectiveness of our design. Overall, it achieves state-of-the-art performance against the alternatives on multiple real-world benchmarks for SIRS.

2. Related Work

Plenty of efforts have been devoted to handling the image reflection separation problem in the past decades. They either leverage multiple images to acquire more cues for the layer decomposition or exploit extra priors to manage the problem through a single image. In what follows, we organize the previous methods based on the images they require.

Multiple Image Reflection Separation. In general, the transmission component tends to be unpolarized, yet the reflection component varies when rotating the polarization filter mounted in front of a camera sensor. Inspired by this phenomenon, a variety of methods [28, 14, 13, 27, 21, 16] resort to the physical solution that separates different components through a sequence of images with varied polarization orientations. Besides the polarization cues, the layer separations can also be indicated by different focuses [6], stereo information [31], flash on/off [1, 15], relative motions [29, 8, 22, 38, 40, 25, 26] and scene consistency [9, 30, 10]. Although involving multiple images mitigates the ill-posedness of the separation problem, *this pipeline requires professional devices (such as polarizers) and manual operations, which limits its application.*

Single Image Reflection Separation. Compared with multiple-image solutions, single-image-based methods show more merits, being more flexible and requiring less manual operations. However, as there is no such thing as a free lunch, single-image schemes call for additional priors to cope with its ill-posedness nature, such as gradient-based constraints [19, 18, 17, 23] and manual annotations [17]. Specifically, Levin *et al.* [19, 18] impose sparse constraints on gradients to acquire the separations that have fewer edges. Levin and Weiss [17] demand several manual annotations of respective layers to obtain favorable decompositions, which involves additional human labor. Li and Brown [23] suppose that one layer is smoother than the other, thus applying unbalanced penalization on the gradients of the two layers to decompose them. *Although these methods work fine in restricted scenarios, the complicated real-world conditions can considerably surpass their assumptions, leading to unsatisfactory results.*

The advent of deep learning technology mitigates the incongruity between the volume of data and modeling capacity. Certain underlying assumptions can be subtly embedded in the process of data synthesis, and subsequently

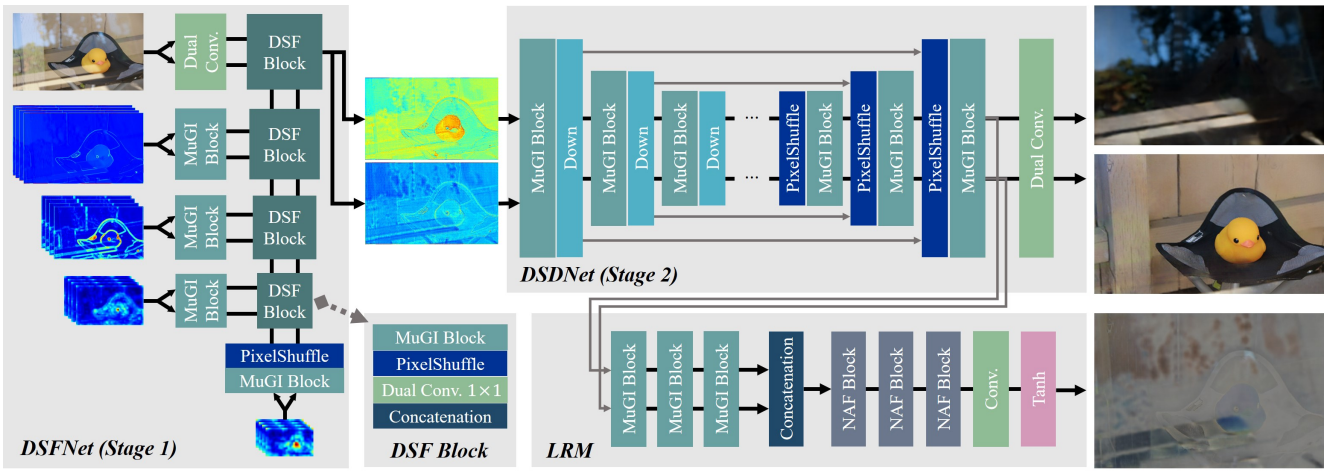

Figure 2: The architecture of our proposed DSRNet, which consists of two cascaded stages and a learnable residue module (LRM). In Stage 1, DSFNet aggregates and separates hierarchical semantic information with two interactive feature streams. The roughly separated features are further fed into the DSDNet for fine-grained decompositions in Stage 2, where the LRM takes decomposed features, separating out the components that violate the linear assumption.

learned by deep models. Concretely, CEILNet [5] imposes the relative smoothness prior to the synthesis of reflection layers and combines them with transmission layers by addition. An edge-aware network is developed to capture the transmission components, which yet ignores high-level semantics that are likely to aid the SIRS task. Zhang *et al.* [41] hence introduce HyperColumn features [11] using a pre-trained VGG-19 network to acquire semantic awareness, besides the perceptual and adversarial losses. Moreover, the exclusivity loss is developed to penalize the intersected gradients. ERRNet [36] goes a step further by leveraging an additional set of misaligned pairs. However, it appears to overlook the estimation of the reflection layer, which is essentially an image component and potentially important to distinguish the transmission parts. Li *et al.* [24] thereby presents a two-stage network (RAGNet) to first estimate reflection components and predict transmission ones guided by them, whereas the reflection estimation is isolated from the transmission in this way. Further, Hu and Guo [12] come up with the YTMT strategy, which pays equal attention to the two components, and a dual-stream interactive network is developed to restore the two layers simultaneously. However, their linear assumption makes the predicted reflection components tend to be weak in many cases. Other than them, BDN [39] and IBCLN [20] make use of reflection models weighted by scalars, and iteratively estimate both the components. This scheme prevents the reflection from being too weak, but its restorations often struggle to be free from the transmission parts. Furthermore, Wen *et al.* [37] simulate the nonlinear superimposition phenomenon by predicting a three-channel alpha blending weight map with the adversarial guidance of collected unpaired images. Dong *et al.* [3] develops an iterative network and estimates a probabilistic reflection confidence map in each iteration. *However, it is not an easier task to estimate an alpha blending map than to determine each of the blended components, while the residual estimation only needs to retain the redundant information during decomposition, which is more practical than the former.* Considering the drawbacks of the previous reflection models, we propose a general form of them with a learnable residue term, which is shown to be more effective and flexible.

3. Methodology

As shown in Fig. 2, our proposed DSRNet comprises two cascaded stages and a learnable residue module (LRM). In what follows, we first clarify the motivation of LRM and then introduce the MuGI block and the architecture of the DSRNet followed by its training loss functions.

3.1. Residue of the Linear Combination

It often occurs that a superimposed image $\mathbf{I}$ cannot be perfectly represented by the linear combination of $\mathbf{T}$ and $\mathbf{R}$, leading to a residue term, as discussed in Section 1. Under its disturbance, the error of the additive reconstruction criterion $\varepsilon = \|\mathbf{I} - (\hat{\mathbf{T}} + \hat{\mathbf{R}})\|$ remains large even though the predicted layers can well reconstruct the ground truths. Further minimizing ε will put redundant information to either $\hat{\mathbf{T}}$ or $\hat{\mathbf{R}}$, which induces the deviation of predictions from their ground truths instead. Moreover, there are many kinds of simplified physical models proposed by previous work with their own shortcomings. Naturally, a unified model is de-

sired to be put forward. Therefore, we introduce an extra residue term to offset the error in additive reconstruction, which, at the same time, unifies the different physical models. As depicted in Fig. 2, we leverage a learnable residue module (LRM) after Stage 2 to estimate the residual information during the decomposition by analyzing the features of $\hat{\mathbf{T}}$ and $\hat{\mathbf{R}}$ before the final layer. LRM consists of an interactive and a fusion part to collect and merge residual information from two branches. The dual-stream signals are concatenated before entering the fusion part. To constrain the space of residue, the $\tanh(\cdot)$ function is utilized as the final activation. With the participation of LRM, we define the following reconstruction loss with residual rectification (R^3 Loss):

$$\mathcal{L}_{rec} := \|\mathbf{I} - (\hat{\mathbf{T}} + \hat{\mathbf{R}}) - \Phi(\hat{\mathbf{T}}, \hat{\mathbf{R}})\|_1, \tag{2}$$

where Φ denotes the LRM and $\|\cdot\|_1$ means the ℓ_1 norm. Guided by this objective, the information beyond the additive model will flow to the residue term. Notably, this term can be totally discarded during the testing phase, avoiding extra computational costs.

3.2. Mutually-gated Interactive Block

Now that we employ the LRM to model the residual components during the additive reconstruction, the rest part can be seen as a linear model again as $\tilde{\mathbf{I}} = \mathbf{I} - \Phi(\hat{\mathbf{T}}, \hat{\mathbf{R}}) = \hat{\mathbf{T}} + \hat{\mathbf{R}}$. As displayed in Fig. 3 (a), we exploit the synergy between $\hat{\mathbf{T}}$ and $\hat{\mathbf{R}}$ by developing the **Mu**tually-**G**ated **I**nteractive mechanism, namely MuGI:

$$\begin{cases} \hat{\mathbf{F}}_{\mathbf{T}} = \mathcal{G}_1(\mathbf{F}_{\mathbf{T}}) \circ \mathcal{G}_2(\mathbf{F}_{\mathbf{R}}); \\ \hat{\mathbf{F}}_{\mathbf{R}} = \mathcal{G}_1(\mathbf{F}_{\mathbf{R}}) \circ \mathcal{G}_2(\mathbf{F}_{\mathbf{T}}), \end{cases} \tag{3}$$

where $\mathbf{F}_{\mathbf{T}}, \mathbf{F}_{\mathbf{R}} \in \mathbb{R}^{H_1 \times W_1 \times C_1}$ denote intermediate features of streams for predicting $\hat{\mathbf{T}}$ and $\hat{\mathbf{R}}$, respectively, and $\hat{\mathbf{F}}_{\mathbf{T}}, \hat{\mathbf{F}}_{\mathbf{R}} \in \mathbb{R}^{H_2 \times W_2 \times C_2}$ represent the outputs of the mutual gate. $\mathcal{G}_1$ and $\mathcal{G}_2$ are functions selecting which part of features to engage the interaction. $\circ$ indicates the element-wise multiplication. Here we use a simple implementation of MuGI to evaluate its capability of solving SIRS, which divides a feature map in half at the channel dimension, and $\mathcal{G}_1(\cdot)$ always selects the former half, while $\mathcal{G}_2(\cdot)$ chooses the latter. We have $H_1 = H_2, W_1 = W_2$ and $C_2 = C_1/2$ in this way. Such a gated mechanism captures the mutual dependency of the two components in the feature space, say if either stream contains signals the other one desires, they tend to be split into the latter half (selected by $\mathcal{G}_2(\cdot)$) and transferred into the sibling stream by the gate.

Our simple implementation of MuGI makes it plug-and-play. Therefore, in general, most of the single-path blocks can be converted into a dual-stream one by parallel setting two of them and relating them with MuGI. In this paper, we follow a state-of-the-art block design presented by [2]

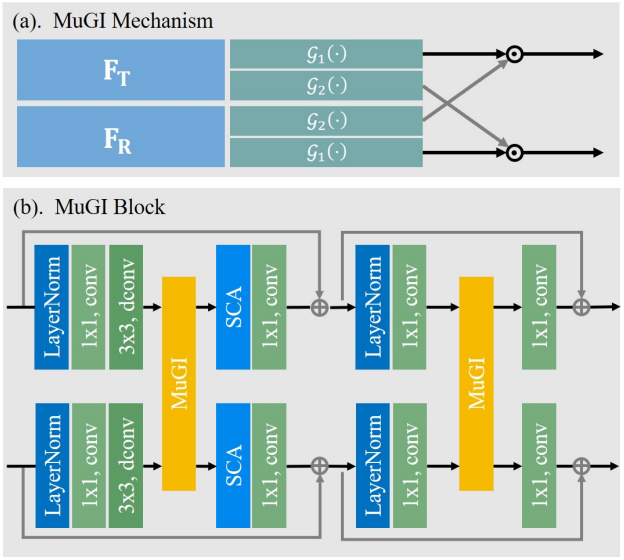

Figure 3: MuGI mechanism and MuGI Block.

and convert its single-path NAF block into a dual-stream mutually-gated interactive block (MuGI block). Its diagram is depicted in Fig. 3 (b), in which the simple gates are replaced by MuGI gates. Specifically, the layer normalization makes dual-stream features comparable, 1×1 convolution doubles the channel dimensions in order to conduct MuGI without information loss which reduces dimensions by half during the interaction, channel attention and the following 1×1 convolution serve as fusion/reweighting roles.

3.3. Dual-stream Semantic-aware Network

The semantic information provided by pre-trained models is usually introduced to alleviate the impact of ill-posedness in SIRS. The HyperColumn [41, 36, 12], as a prevalent multi-scale semantic feature extractor in SIRS, aggregates semantics through interpolating the features extracted by a pre-trained model into the same scale as the input images. It then concatenates them together and employs a 1×1 convolution to rapidly reduce the channel dimensions (typically from 1475 to 64 or 256), before feeding them into the decomposition networks. This strategy omits the inner relationship of multi-scale features and may discard informative signals. To overcome the drawbacks mentioned above, we present a dual-stream pyramid fusion network (DSFNet), which hierarchically aggregates the extracted features by jointly upsampling and interacting. As shown in Fig. 2, given the hierarchical features extracted by a pre-trained deep network (e.g., VGGNet-19), DSFNet gathers them in a bottom-up manner with MuGI blocks and dual-stream fusion blocks (DSF Block), the structure of which is shown in Fig. 2. "Dual Conv" means two parallel convolutional layers. The features extracted by the pre-

Methods	Real20 (20)		Objects (200)		Postcard (199)		Wild (55)		Average	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Zhang *et al.* [41]	22.55	0.788	22.68	0.879	16.81	0.797	21.52	0.832	20.08	0.835
BDN [39]	18.41	0.726	22.72	0.856	20.71	0.859	22.36	0.830	21.65	0.849
ERRNet [36]	22.89	0.803	24.87	0.896	22.04	0.876	24.25	0.853	23.53	0.879
IBCLN [20]	21.86	0.762	24.87	0.893	23.39	0.875	24.71	0.886	24.10	0.879
RAGNet [24]	22.95	0.793	26.15	0.903	23.67	0.879	25.53	0.880	24.90	0.886
DMGN [7]	20.71	0.770	24.98	0.899	22.92	0.877	23.81	0.835	23.80	0.877
Zheng *et al.* [43]	20.17	0.755	25.20	0.880	23.26	0.905	25.39	0.878	24.19	0.885
YTMT [12]	23.26	0.806	24.87	0.896	22.91	0.884	25.48	0.890	24.05	0.886
Ours	**24.23**	**0.820**	**26.28**	**0.914**	**24.56**	**0.908**	**25.68**	**0.896**	**25.40**	**0.905**
Dong *et al.*† [3]	23.34	0.812	24.36	0.898	23.72	0.903	25.73	0.902	24.21	0.897
Ours†	**23.91**	**0.818**	**26.74**	**0.920**	**24.83**	**0.911**	**26.11**	**0.906**	**25.75**	**0.910**

Table 1: Quantitative results on four real-world benchmark datasets of methods. The best results are indicated in **bold**. † indicates extra training data that are involved as in [3].

trained network at each level first interact through a MuGI block, thereby features in the same scale are related and transformed into dual-stream features ($\mathbf{F_T} = \mathbf{F_R}$ for these blocks). After the interaction, the deepest features extracted by VGGNet are upscaled and then fused with shallower features through the DSF block, in which the dual-stream features come from the two adjacent scales are concatenated at channel dimension, fused by a 1x1 convolutional layer and then upscaled at each stream. The MuGI blocks are further employed to promote cross-scale interactions. Note that, shallow features extracted from RGB inputs are fused at the top of DSFNet, which preserves fine-grained details to restore both layers. The aggregated features provide rough separations of layers and are further refined by the second stage. The following stage (DSDNet) is constructed by the proposed MuGI blocks in a U-shaped manner. More details can be found in the supplementary materials.

3.4. Loss Function

Besides the R^3 loss demonstrated in Section 3.1, we further introduce pixel, perceptual, and exclusion loss for pixel-wise and semantic fidelity as well as the gradient independence of layers, which are described as follows:

Pixel Loss. The pixel loss is used to constrain the consistency of the reconstructed layer and groundtruth in both the natural image domain and gradient domain, minimizing their errors as follows:

$$\begin{aligned}\mathcal{L}_{pix} := &\|\hat{\mathbf{T}} - \mathbf{T}\|_2^2 + \|\hat{\mathbf{R}} - \mathbf{R}\|_2^2 \\ &+ \alpha(\|\nabla\hat{\mathbf{T}} - \nabla\mathbf{T}\|_1 + \|\nabla\hat{\mathbf{R}} - \nabla\mathbf{R}\|_1),\end{aligned} \tag{4}$$

where $\|\cdot\|_2$ denotes the ℓ_2 norm. ∇ stands for the gradient operator, which extracts gradients in both vertical and horizontal directions. We express one of them in the formulation for clarity and average them in practice. α is set as 2 in our experiments.

Perceptual Loss. The pixel-wise losses cannot assure the multi-scale consistency between predictions and their groundtruths, which may result in overwhelming punishment for the whole image due to small differences in local brightness. Therefore, the perceptual loss is further introduced as follows:

$$\mathcal{L}_{per} := \sum_i \omega_i \|\phi_i(\hat{\mathbf{T}}) - \phi_i(\mathbf{T})\|_1, \tag{5}$$

where $\phi_i(\cdot)$ designates the features drawn by layer $i \in \{2, 7, 12, 21, 30\}$ of a VGG-19 model. ω_is are combining weights of terms at different layers. We follow the setting of hyper-parameters in [36].

Exclusion Loss. We introduce the exclusion loss to strengthen the gradient independence prior and reduce the structural coupling in estimated separations as below:

$$\begin{aligned}\mathcal{L}_{exc} &:= \frac{1}{N}\sum_{n=0}^{N-1} \|\Psi(\hat{\mathbf{T}}^{\downarrow n}, \hat{\mathbf{R}}^{\downarrow n})\|_2^2, \\ \Psi(\hat{\mathbf{T}}, \hat{\mathbf{R}}) &:= \tanh\left(\eta_1|\nabla\hat{\mathbf{T}}|\right) \circ \tanh\left(\eta_2|\nabla\hat{\mathbf{R}}|\right),\end{aligned} \tag{6}$$

where $\hat{\mathbf{T}}^{\downarrow n}$ and $\hat{\mathbf{R}}^{\downarrow n}$ represent taking down-sampling by 2^n times (2^N at most) of $\hat{\mathbf{T}}$ and $\hat{\mathbf{R}}$. η_1 and η_2 are normalization factors, which are identical to [41].

Gathering all the loss terms yields the final objective as:

$$\mathcal{L}_{all} := \mathcal{L}_{pix} + \beta_1\mathcal{L}_{per} + \beta_2\mathcal{L}_{exc} + \beta_3\mathcal{L}_{rec}, \tag{7}$$

where $\beta_1 = 0.01$, $\beta_2 = 1$, and $\beta_3 = 0.2$ are set empirically.

4. Experimental Validation

4.1. Implementation Details

Implemented in PyTorch, our models are optimized with the Adam optimizer on a single RTX 3090 GPU for 20

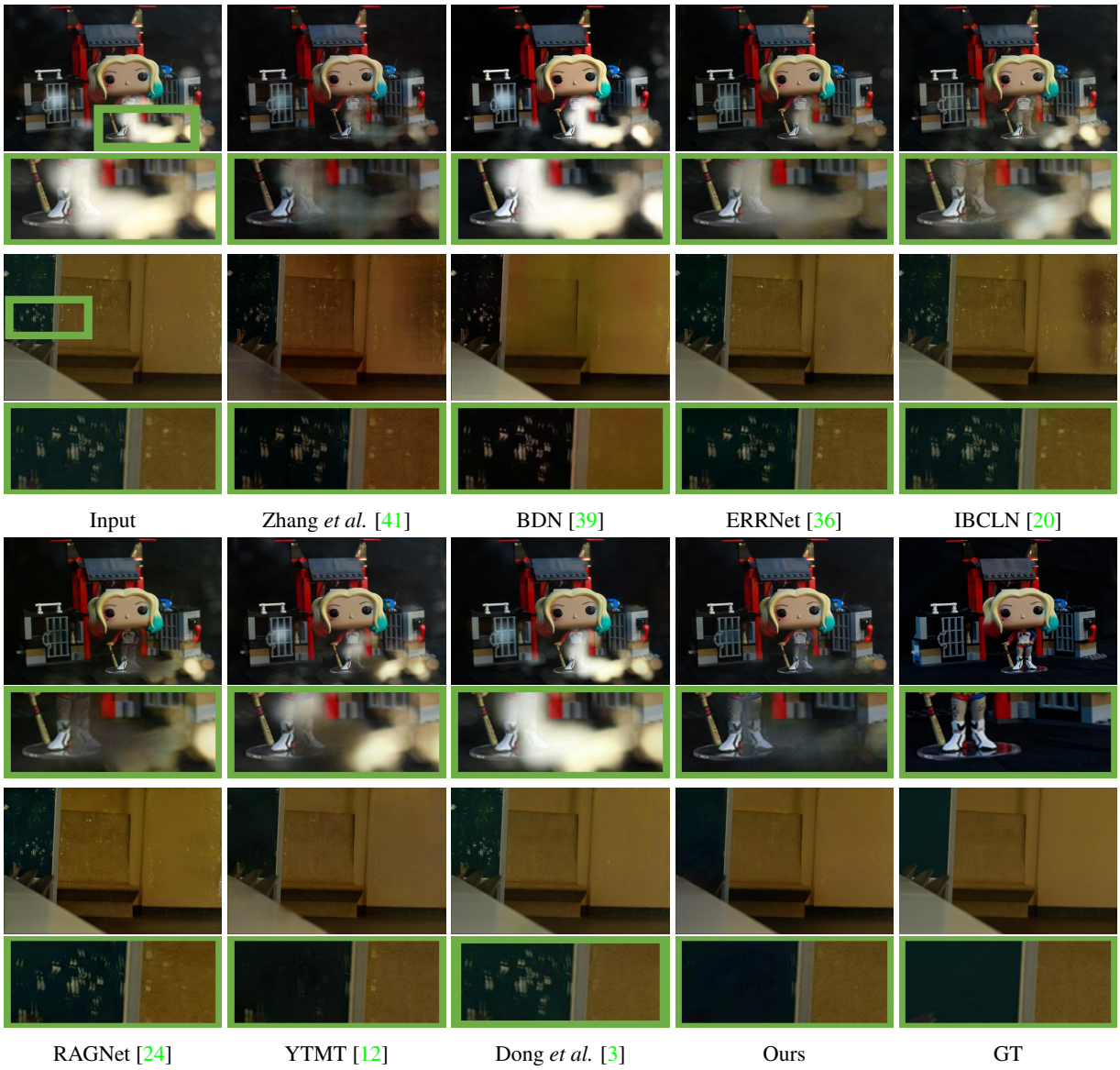

Figure 4: Visual comparison of estimated transmission layers between state-of-the-arts and ours on real-world samples.

epochs at most to reach favorable overall performance. The learning rate is initialized as 10^{-4} and fixed during the training phase with a batch size of 1.

Dataset. Our training dataset embraces both real and synthesized images. Following [5, 12], the training dataset are composed of 90 real pairs from [41] and 7,643 synthesized pairs from the PASCAL VOC dataset [4]. For synthesized data, transmission and reflection layers are weakened by coefficients $\gamma_1 \in [0.8, 1.0]$ and $\gamma_2 \in [0.4, 1.0]$ during blending them with the following model:

$$\mathbf{I}_{syn} = \gamma_1 \mathbf{T}_{syn} + \gamma_2 \mathbf{R}_{syn} - \gamma_1 \gamma_2 \mathbf{T}_{syn} \circ \mathbf{R}_{syn}, \quad (8)$$

where $\mathbf{T}_{syn}, \mathbf{R}_{syn}$ and $\mathbf{I}_{syn}$ represent the transmission, reflection, and superimposed layers during synthesis, respectively. This formulation is inspired by the "screen" blending mode in digital image processing, which always reserves lighter colors for the blending layers.

4.2. Performance Evaluation

Quantitative comparison. A comparison is made between state-of-the-art methods as shown in Table 1, involving Zhang *et al.* [41], BDN [39], ERRNet [36], IBCLN [20], RAGNet [24], DMGN [7], Zheng *et al.* [43], YTMT [12] and ours on four real-world dataset, including Real20 [41] and three subsets of the SIR^2 Dataset [33]. Besides, to meet the configuration of [3], we train our DSRNet ad-

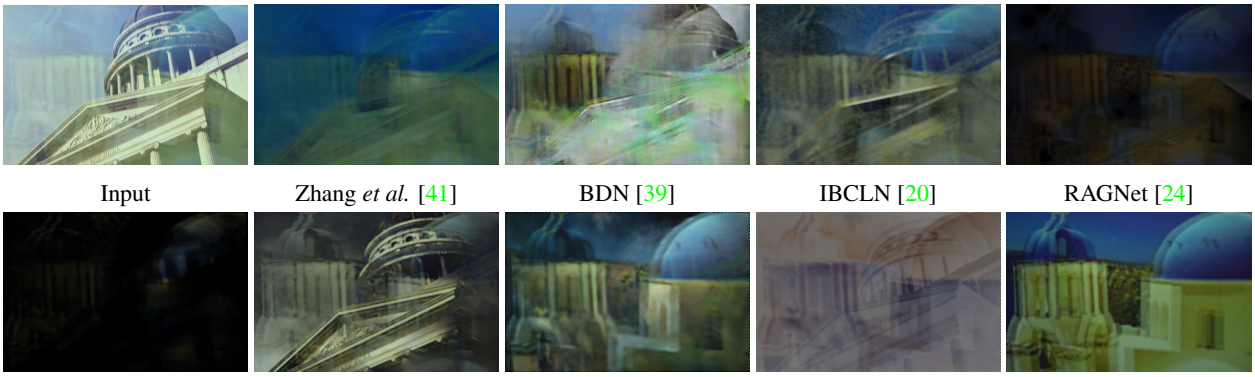

Figure 5: Visual comparison of estimated reflections between state-of-the-arts and ours on a sample drawn from the SIR^2 dataset. Ours has a significant advantage over other alternatives by separating the residue component from the estimations.

ditionally under their training settings, which includes 200 extra real pairs provided by the "Nature" dataset [20], and 13,700 synthesized image pairs provided by [41] instead.

It turns out that our method shows its superiority over other competitors on all the testing datasets in both settings, gaining $0.50dB$ and $1.54dB$ in terms of average PSNR in the two settings, respectively. Given the four real-world datasets contains a variety of scenes, illumination conditions, and glass thickness, it is not a trivial task to achieve the best performance of all metrics on these datasets simultaneously. The experimental results demonstrate that our proposed SIRS scheme has significantly higher performance and stronger generalization ability, which explains our main contributions.

Qualitative comparisons. To further explain our performance advantages, we provide visual comparisons of transmission layers in Fig. 4 against state-of-the-art methods, including Zhang *et al.* [41], BDN [39], ERRNet [36], IBCLN [20], RAGNet [24], YTMT [12], and Dong *et al.* [3]. As can be observed in Fig. 4, the method proposed by Zhang *et al.* fails to handle the case containing scattered reflection components in the second row. BDN has trouble dealing with the images with highlights like in the first row and even aggravates the reflections. The problem is likely to be caused by the linear reflection model, which lacks the ability to model specular highlights. For the results of ERRNet, which contains only a single branch to estimate transmission layers, it lacks the direct modeling of the reflections, therefore only removing parts of them. IBCLN also has trouble removing highlights and scattered reflections. The reflection layers in RAGNet are estimated without the participation of the transmission, hence showing inferior performance. YTMT can better cope with the highlights in the last row owing to its dual-stream design but is still limited by the linear assumption and cannot remove strong reflections. The method proposed by Dong *et al.* shows inferior performance in both cases due to the difficulty of estimating the blending weight. Overall, our results are more visually favorable and contain fewer residual reflection components, which further reinforces our claims.

We further deliver a comparison of reflection layers in Fig. 5 to illustrate that our method can better reproduce the reflection scenes. It can be seen that the methods (Zhang *et al.*, RAGNet, and YTMT) based on the additive model provide weak reflection maps. In the meantime, the methods based on the linear model with scalars (BDN and IBCLN) and the one containing an alpha blending map can hardly avoid mixing the transmission components in reflection estimations. This phenomenon reflects the weakness of a fixed simplified model, requiring the reconstruction of more than two components. As shown by our results, introducing a residue term can significantly enhance the restoration quality of the reflection scene, and separates out the components beyond the additive relationship.

4.3. Ablation Study

To better analyze our improvements in the physical model and network structure, we carried out a detailed ablation study from the perspective of the reconstruction losses, feature interaction mechanisms, and feature encoders. We gather the results (PSNR and SSIM averaged on Real20 and SIR^2 datasets) in Table 2 and deliver a visual comparison in Fig. 6. In what follows we detail every point of them.

Reconstruction Loss. In view of the problems existing in the linear combination model, we use a learnable residue term to modify it, enhancing its flexibility so that it can cover more reflection scenes. In order to promote the learning of this residue term, we propose the reconstruction loss of residual correction. To illustrate the necessity of this setting, we perform an ablation study to compare the settings without reconstruction loss (w/o Recons. Loss) and using the linear reflection model (w/ Linear Recons.). It shows

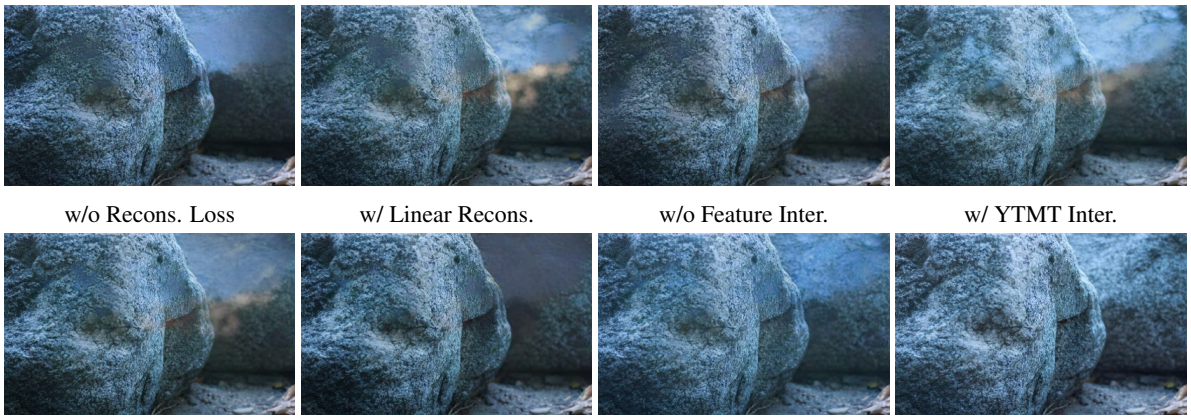

Figure 6: The visual comparison of different settings involved in the ablation study.

Models	PSNR	SSIM
w/o Recons. Loss	24.84	0.897
w/ Linear Recons.	22.21	0.881
w/o Feature Inter.	24.81	0.895
w/ YTMT Inter.	24.89	0.901
w/o Feature Enc.	24.75	0.903
w/ HyperColumn	23.99	0.894
Ours	**25.40**	**0.905**

Table 2: Ablation study on different configurations. The initial two rows compare the configuration devoid of any reconstruction loss and that integrating a linear reconstruction loss. Rows 3-4 offer a comparison between the settings lacking feature interaction and incorporating the YTMT feature interaction mechanism. Rows 5-6 shed light on configurations that exclude the feature encoder and utilize HyperColumn as the feature encoder. The last row shows the results of our full version.

that the linear model is inferior to the setting without any reconstruction criterion because of the existence of ε. Meanwhile, without a reconstruction term, the model is likely to have trouble determining the portion of the components, excessively separating content to the reflection layer, resulting in textureless regions in the transmission layer as shown in Fig. 6.

Feature Interaction Mechanism. The feature interaction mechanism connects the two information streams in our proposed DSRNet, both exchanging and conditioning the dual-stream information, which facilitates the layer reconstruction. As shown in Table 2, the model performance appears to degrade after removing the feature interaction (w/o Feature Inter.), which tells us that the network can hardly achieve state-of-the-art performance via a two-branch network without feature interaction. Further, we include the results of the YTMT feature interaction mechanism (w/ YTMT Inter.) by replacing MuGI with the YTMT mechanism. It can be seen that the model with the YTMT mechanism is only slightly better than the scheme without interaction, which demonstrates the superiority of the proposed MuGI mechanism.

Feature Encoder. In comparison to the hypercolumn framework introduced by the previous method (referred to as "w/ HyperColumn") and the approach that without using any feature encoder ("w/o Feature Enc."), our method demonstrates noteworthy performance advantages. These advantages stem from two main factors. Firstly, our approach employs a hierarchical and gradual reduction of channel dimensions, followed by fusion operations. Secondly, DSFNet leverages a feature interaction mechanism, allowing high-level semantic features to interact before being fed into the subsequent stage.

5. Conclusion

In this paper, we modified the commonly-used linear model in the single image reflection separation task and proposed the reconstruction loss with residual correction. The proposed model is more flexible and effective compared with the previous methods. Meanwhile, we further improved the feature interaction mechanism in dual-stream networks and the usage of hierarchical semantic information, proposed the MuGI as a novel interaction paradigm, and a dual-stream semantic-aware network, namely DSRNet. Extensive experiments revealed that our proposed method has achieved state-of-the-art performance on all real-world benchmark datasets and verified our contributions. In the future, further constraints are desired to be applied to the residue term to reduce its solution space, and more reflection models are hopefully to be covered by it to fit a wider range of reflection phenomena in the real world.

References

[1] Amit K. Agrawal, Ramesh Raskar, Shree K. Nayar, and Yuanzhen Li. Removing photography artifacts using gradient projection and flash-exposure sampling. *TOG*, 24(3):828–835, 2005. 2

[2] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *ECCV*, pages 17–33, 2022. 2, 4

[3] Zheng Dong, Ke Xu, Yin Yang, Hujun Bao, Weiwei Xu, and Rynson W. H. Lau. Location-aware single image reflection removal. In *ICCV*, pages 4997–5006, 2021. 3, 5, 6, 7

[4] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 6

[5] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David P. Wipf. A generic deep architecture for single image reflection removal and image smoothing. In *ICCV*, pages 3258–3267, 2017. 2, 3, 6

[6] Hany Farid and Edward H. Adelson. Separating reflections and lighting using independent components analysis. In *CVPR*, pages 1262–1267, 1999. 2

[7] Xin Feng, Wenjie Pei, Zihui Jia, Fanglin Chen, David Zhang, and Guangming Lu. Deep-masking generative network: A unified framework for background restoration from superimposed images. *TIP*, 30:4867–4882, 2021. 5, 6

[8] Kun Gai, Zhenwei Shi, and Changshui Zhang. Blind separation of superimposed moving images using image statistics. *TPAMI*, 34(1):19–32, 2012. 2

[9] Xiaojie Guo, Xiaochun Cao, and Yi Ma. Robust separation of reflection from multiple images. In *CVPR*, pages 2195–2202, 2014. 2

[10] Byeong-Ju Han and Jae-Young Sim. Reflection removal using low-rank matrix completion. In *CVPR*, pages 3872–3880, 2017. 2

[11] Bharath Hariharan, Pablo Andrés Arbeláez, Ross B. Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, pages 447–456, 2015. 3

[12] Qiming Hu and Xiaojie Guo. Trash or treasure? an interactive dual-stream strategy for single image reflection separation. In *NeurIPS*, pages 24683–24694, 2021. 2, 3, 4, 5, 6, 7

[13] Naejin Kong, Yu-Wing Tai, and Joseph S. Shin. A physically-based approach to reflection separation: From physical modeling to constrained optimization. *TPAMI*, 36(2):209–221, 2014. 2

[14] Naejin Kong, Yu-Wing Tai, and Sung Yong Shin. High-quality reflection separation using polarized images. *TIP*, 20(12):3393–3405, 2011. 2

[15] Chenyang Lei and Qifeng Chen. Robust reflection removal with reflection-free flash-only cues. In *CVPR*, pages 14811–14820, 2021. 2

[16] Chenyang Lei, Xuhua Huang, Mengdi Zhang, Qiong Yan, Wenxiu Sun, and Qifeng Chen. Polarized reflection removal with perfect alignment in the wild. In *CVPR*, pages 1747–1755, 2020. 2

[17] Anat Levin and Yair Weiss. User assisted separation of reflections from a single image using a sparsity prior. *TPAMI*, 29(9):1647–1654, 2007. 1, 2

[18] Anat Levin, Assaf Zomet, and Yair Weiss. Learning to perceive transparency from the statistics of natural scenes. In *NeurIPS*, pages 1247–1254, 2002. 2

[19] Anat Levin, Assaf Zomet, and Yair Weiss. Separating reflections from a single image using local features. In *CVPR*, pages 306–313, 2004. 2

[20] Chao Li, Yixiao Yang, Kun He, Stephen Lin, and John E. Hopcroft. Single image reflection removal through cascaded refinement. In *CVPR*, pages 3562–3571, 2020. 3, 5, 6, 7

[21] Rui Li, Simeng Qiu, Guangming Zang, and Wolfgang Heidrich. Reflection separation via multi-bounce polarization state tracing. In *ECCV*, pages 781–796, 2020. 2

[22] Yu Li and Michael S. Brown. Exploiting reflection change for automatic reflection removal. In *ICCV*, pages 2432–2439, 2013. 2

[23] Yu Li and Michael S. Brown. Single image layer separation using relative smoothness. In *CVPR*, pages 2752–2759, 2014. 2

[24] Yu Li, Ming Liu, Yaling Yi, Qince Li, Dongwei Ren, and Wangmeng Zuo. Two-stage single image reflection removal with reflection-aware guidance. *CoRR*, abs/2012.00945, 2020. 3, 5, 6, 7

[25] Yu-Lun Liu, Wei-Sheng Lai, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Learning to see through obstructions. In *CVPR*, pages 14203–14212, 2020. 2

[26] Yu-Lun Liu, Wei-Sheng Lai, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Learning to see through obstructions with layered decomposition. *TPAMI*, 44(11):8387–8402, 2022. 2

[27] Youwei Lyu, Zhaopeng Cui, Si Li, Marc Pollefeys, and Boxin Shi. Reflection separation using a pair of unpolarized and polarized images. In *NeurIPS*, pages 14532–14542, 2019. 2

[28] Shree K. Nayar, Xi-Sheng Fang, and Terrance E. Boult. Separation of reflection components using color and polarization. *IJCV*, 21(3):163–186, 1997. 2

[29] Bernard Sarel and Michal Irani. Separating transparent layers of repetitive dynamic behaviors. In *ICCV*, pages 26–32, 2005. 2

[30] Christian Simon and In Kyu Park. Reflection removal for in-vehicle black box videos. In *CVPR*, pages 4231–4239, 2015. 2

[31] Sudipta N. Sinha, Johannes Kopf, Michael Goesele, Daniel Scharstein, and Richard Szeliski. Image-based rendering for scenes with reflections. *TOG*, 31(4):100:1–100:10, 2012. 1, 2

[32] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxim: Multi-axis mlp for image processing. In *CVPR*, pages 5769–5780, 2022. 2

[33] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot. Benchmarking single-image reflection removal algorithms. In *ICCV*, pages 3942–3950, 2017. 6

[34] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot. Crrn: Multi-scale guided concurrent reflection removal network. In *CVPR*, pages 4777–4785, 2018. 1, 2

[35] Renjie Wan, Boxin Shi, Haoliang Li, Ling-Yu Duan, and Alex C. Kot. Reflection scene separation from a single image. In *CVPR*, pages 2395–2403, 2020. 1

[36] Kaixuan Wei, Jiaolong Yang, Ying Fu, David P. Wipf, and Hua Huang. Single image reflection removal exploiting misaligned training data and network enhancements. In *CVPR*, pages 8178–8187, 2019. 2, 3, 4, 5, 6, 7

[37] Qiang Wen, Yinjie Tan, Jing Qin, Wenxi Liu, Guoqiang Han, and Shengfeng He. Single image reflection removal beyond linearity. In *CVPR*, pages 3771–3779, 2019. 1, 2, 3

[38] Tianfan Xue, Michael Rubinstein, Ce Liu, and William T. Freeman. A computational approach for obstruction-free photography. *TOG*, 34(4):79:1–79:11, 2015. 2

[39] Jie Yang, Dong Gong, Lingqiao Liu, and Qinfeng Shi. Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal. In *ECCV*, pages 675–691, 2018. 1, 2, 3, 5, 6, 7

[40] Jiaolong Yang, Hongdong Li, Yuchao Dai, and Robby T. Tan. Robust optical flow estimation of double-layer images under transparency or reflection. In *CVPR*, pages 1410–1419, 2016. 2

[41] Xuaner Cecilia Zhang, Ren Ng, and Qifeng Chen. Single image reflection separation with perceptual losses. In *CVPR*, pages 4786–4794, 2018. 2, 3, 4, 5, 6, 7

[42] Qian Zheng, Jinnan Chen, Zhan Lu, Boxin Shi, Xudong Jiang, Kim-Hui Yap, Ling-Yu Duan, and Alex C Kot. What does plate glass reveal about camera calibration? In *CVPR*, pages 3022–3032, 2020. 2

[43] Qian Zheng, Boxin Shi, Jinnan Chen, Xudong Jiang, Ling-Yu Duan, and Alex C. Kot. Single image reflection removal with absorption effect. In *CVPR*, pages 13395–13404, 2021. 5, 6