Single Image Reflection Removal with Edge Guidance, Reflection Classifier, and Recurrent Decomposition

Ya-Chu Chang* Chia-Ni Lu* Chia-Chi Cheng Wei-Chen Chiu
National Chiao Tung University, Taiwan

Abstract

Removing undesired reflection from an image captured through a glass window is a notable task in computer vision. In this paper, we propose a novel model with auxiliary techniques to tackle the problem of single image reflection removal. Our model takes a reflection contaminated image as input, and decomposes it into the reflection layer and the transmission layer. In order to ensure quality of the transmission layer, we introduce three auxiliary techniques into our architecture, including the edge guidance, a reflection classifier, and the recurrent decomposition. The contributions and the efficacy of these techniques are investigated and verified in the ablation study. Furthermore, in comparison to the state-of-the-art baselines of reflection removal, both quantitative and qualitative results demonstrate that our proposed method is able to deal with different kinds of images, achieving the best results in average.

Figure 1: Reflection Removal methods attempt to recover the transmission layer from a reflection contaminated image. Given a reflection contaminated input image (the first column), our method aims to decompose the reflection layer (the last column) and generate the reflection-free transmission layer (the third column), which must be quite similar to its corresponding groundtruth (the second column).

1. Introduction

When taking a picture through a transparent medium such as a glass window, reflection often appears and ruins the photo. For instance, we may have attempted to shoot the landscapes outside the window when traveling on train, but fail to capture the picturesque scene due to the undesired obstruction of the reflection. Such circumstances can be alleviated via the image reflection removal (IRR) process, as the examples shown in Figure 1. IRR attempts to recover the transmission layer from a reflection contaminated image, which has attracted research attention from the computer vision community and becomes an active research area [2, 4, 27, 28, 33].

In the problem of reflection removal, the reflection contaminated image I is often modeled as a linear combination of the transmission layer T and the reflection layer R, i.e., $I = T + R$ [4, 27]. Our goal is then to decompose the reflection contaminated image into a clean, reflection-free image, i.e. the transmission layer, and the reflection layer. This is extremely challenging since IRR is an ill-posed problem, there apparently exists infinite ways of decomposing I into $\{T, R\}$ if I is provided without additional constraints or priors. Traditional methods thus use multiple images with variations as input (e.g. taking images with slightly different viewing angles) [6, 7, 8, 13, 23, 29, 30], or employ a variety of hand-crafted priors to tackle this problem [11, 12, 16]. However, multiple images are often hard to collect and not suitable for practical use, and hand-crafted priors are not always generalizable to all cases of images with reflection. Recently, as the deep-learning-based approaches has begun to flourish and the single image reflection removal (SIRR) task has attracted more and more attention owing to its simplicity for the practical use, a number of methods were proposed to develop end-to-end deep models for addressing single image reflection removal. Nevertheless, although these methods have reached state-of-the-art performance on several benchmark datasets [25, 33], SIRR remains unresolved across various imaging conditions and diverse content of scenes.

In order to resolve the aforementioned problems, we propose to use several carefully-designed objectives as well as the guidance in this paper for alleviating the difficulty

*Both authors contributed equally to the paper

of this ill-posed problem from different perspectives. We introduce a deep convolution network with three auxiliary extensions: *Edge Guidance*, *Reflection Classifier*, and *Recurrent Decomposition*. First, the edge guidance provides supplementary edge information to benefit our reflection removal method, exploiting the observation that the edges on transmission layer and the ones on reflection layer often present distinct distributions. Second, based on the assumption $I = T + R$ and the exploration of the shared structures/patterns between the reflection contaminated image I and the reflection layer R, we train a reflection classifier to provide novel objectives for benefiting our model learning. Third, by building upon the idea of sequential decomposition proposed in [32] but with our novel modifications, we introduce the recurrent mechanism to achieve better performance in reflection removal without extra memory consumption. Finally, our full model aggregates all the aforementioned extensions, and decomposes a reflection contaminated image into the reflection and transmission layers. In brief, our model is with holistic design to not only carefully integrate the pros of prior arts but also introduce the novel components, thus leading to superior performance with respect to the state-of-the-art baselines.

2. Related Works

Prior research works of reflection removal can typically be categorized into two types according to their data requirement: multiple-image approaches and single-image approaches. We organize and discuss related approaches as follows, with focus on the recent deep-learning-based ones:

Multiple-image approaches. Owing to the harsh challenge of addressing ill-posed problem on single-image reflection removal, many prior works start with multiple input images. They often take two or more pictures shot in the same scene but from slightly different camera positions [6, 15, 7, 8, 23, 24, 30] or various polarization angles [5, 19, 29]. Guo *et al*. [7] exploit the relative motion cues between transmission layer and reflection layer, and use homography to represent the motion of each layer in order to estimate the transmission layer. A deep-learning approach proposed by Wieschollek *et al*. [29] uses the polarization properties of light to separate the reflection and transmission components of the recorded irradiance. There are also some methods which require special conditions and camera settings, such as input pair of images taken with flash and without flash [1], different focuses [20], or two images taken simultaneously by the front and back cameras of a smart device [10]. However, most of these methods are based on strict assumptions, and the multiple images are not that easy to capture under those constraints.

Single-image approaches. Contrary to multiple-image approaches, single-image ones are more suitable for practical use, and can be directly applied to any photographs. However, removing reflection from single image is an ill-posed problem. Thus some traditional methods rely on additional priors, such as the gradient sparsity prior [11, 12, 13], ghosting prior [21], or relative smoothness prior [16, 31]. However, the priors leveraged in these methods are usually heuristic and limited to specific scenarios. Deep learning approaches have been proposed in recent years [2, 4, 14, 17, 26, 27, 28, 32, 33]. CEILNet [4] is the first to tackle single image reflection removal using deep-learning techniques. They propose a two-stage framework which first predicts an edge map of the transmission layer, then exploits edge information to assist the CNN networks in decomposition process. Zhang *et al*. [33] combine a fully convolutional network with perceptual loss, adversarial loss, and exclusion loss, which consider both pixel-level (low-level) and feature-level (high-level) image information for driving reflection removal. BDN [32] presents a cascade network structure that the transmission and reflection layers are alternately estimated between sub-networks. Wen *et al*. [28] propose two networks, a synthesis network and a removal network. The synthesis network predicts three non-linear blending masks for three different types of reflection. Then the removal network will be trained jointly with the synthesized data. Wei *et al*. [27] propose a network (i.e. ERRNet) which can be applied to misaligned training data based on the properties on the feature maps extracted by VGG [22]. They also enhance their network with global information by applying pyramid pooling module, and using hypercolumn features as the network input. IBCLN [14] utilizes a cascaded refinement approach, where two convolutional LSTM networks learn to predict the transmission and the residual reflection simultaneously. Moreover, they propose residual reconstruction loss to balance the error from the two sub-networks while encouraging the model training. However, as the ill-posed property of the single-image reflection removal, all the aforementioned approaches have their pros and cons across different scenarios and conditions. There still has not existed one-size-fits-all solution towards this challenging problem. In this paper, we revisit several ideas in the prior works as well as propose novel extensions to push the research on this direction forward.

Benchmark datasets. Here we also briefly review the datasets that are widely used for evaluation nowadays on the problem of reflection removal. A benchmarking real-world dataset for single image reflection removal, SIR^2 [25] was proposed few years ago. It is a dataset with ground truth pairs of the transmission and reflection layers, which provides the users to evaluate their methods thoroughly. The SIR^2 dataset consists of three sub-datasets: postcard, solid objects, and wild scenes. Zhang *et al*. [33] propose another real-world dataset, which consists of 90 images as the training data, and 20 images for testing. Please note that due to the difficulty of collecting a large amount of data with

groundtruth for the transmission and reflection layers, most deep learning methods use synthetic data to train their models, where the process of synthetic data generation is mostly built upon the assumption of $I = T + R$ [4, 33].

3. Proposed Method

As motivated in the previous sections, our model aims to learn a reflection removal network which performs sequential and recurrent decomposition on a reflection contaminated image under the edge guidance of the transmission image, where a novel reflection classifier is adopted to benefit the overall model learning. In the following we will sequentially detail the base model used for decomposition, the reflection classifier, edge guidance, as well as the extension of decomposition via our proposed recurrent mechanism.

3.1. Base Model of Decomposition

We adopt a state-of-the-art network of reflection removal, bidirection network (BDN) proposed by Yang *et al*. [32], as the basis for building up our proposed model. We choose BDN as our base model not only because of its superior performance but also because of the unique architecture, in which BDN explicitly utilizes the assumption between the reflection contaminated image I, reflection layer R and the transmission/background layer T (i.e. $I = T+R$) into its model design. Basically, BDN is a cascaded deep neural network composed of three sub-networks: $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$, where they are all built based on U-Net [18].

The first sub-network $\mathcal{G}_1$ straightforwardly takes a reflection contaminated image I as input to predict the transmission layer $\tilde{T}_1$; the second sub-network $\mathcal{G}_2$ then estimates the reflection layer $\tilde{R}_1$ based on the input of I and $\tilde{T}_1$; eventually, the third sub-network $\mathcal{G}_3$ takes I and $\tilde{R}_1$ to output the final estimation of transmission layer, which is denoted as $\tilde{T}_2$. The overall computation of BDN can be written as:

$$\tilde{T}_2 = \mathcal{G}_3(\mathcal{G}_2(\mathcal{G}_1(I), I), I). \tag{1}$$

The main idea behind $\mathcal{G}_2$ and $\mathcal{G}_3$ stems from the assumption $I = T + R$, such that it is easier to estimate the transmission layer T when the information of the reflection layer R is provided in addition to I, and vice versa. Originally in [32], BDN is trained by using the objectives based on the L_2 distance between $\{\tilde{T}_1, \tilde{R}_1, \tilde{T}_2\}$ and their corresponding groundtruths, as well as the adversarial loss applied on $\tilde{T}_2$. While in our implementation, we add another L_1 loss in addition to the L_2 for improving the sharpness of the resultant estimation of transmission layer, and remove the adversarial loss in order to alleviate the complexity for model training. Furthermore, we step forward to apply the perceptual loss proposed in [9], penalizing the Euclidean distance between the deep features extracted from $\tilde{T}_2$ and the ones from its corresponding groundtruth. By adopting the perceptual loss, we consider the error of the predicted results not only in the low-level/pixel-level but also in the high-level/feature-level, and thus our objective function in learning the base BDN model becomes:

$$\mathcal{L}_{rec}^{base} = \lambda_1\mathcal{L}_1 + \lambda_2\mathcal{L}_2 + \lambda_{feat}\mathcal{L}_{feat}, \tag{2}$$

where $\{\lambda_1, \lambda_2, \lambda_{feat}\}$ are used to balance $\{\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_{feat}\}$ and we have $\lambda_1 = 1$, $\lambda_2 = 4$, $\lambda_{feat} = 150$ in our experiments. $\{\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_{feat}\}$ are defined as:

$$\mathcal{L}_1 = \sum \|\tilde{T}_1 - T\|_1 + \|\tilde{R}_1 - R\|_1 + \|\tilde{T}_2 - T\|_1, \tag{3}$$

$$\mathcal{L}_2 = \sum \|\tilde{T}_1 - T\|_2 + \|\tilde{R}_1 - R\|_2 + \|\tilde{T}_2 - T\|_2, \tag{4}$$

$$\mathcal{L}_{feat} = \sum \sum_l \lambda_l \| \Phi^l(\tilde{T}_2) - \Phi^l(T)\|_2, \tag{5}$$

where λ_l are the balancing weights. T and R denote the groundtruth for the transmission and reflection layers respectively, and $\Phi^l(\cdot)$ denotes the features obtained from the l-th layer of a pretrained VGG network, basically `conv1_1`, `conv1_2`, `conv2_1`, `conv2_2`, and `conv3_1` layers are used in our implementation. Please note that summation $\sum$ used in this paper is performed over all the training data, unless otherwise specified. With such a simple modification on the objectives, the performance of our base model already has the improvement in comparison to the original BDN [32], as what will be shown in Section 4.1.

3.2. Extensions for Improving Decomposition

As the reflection removal problem (i.e. decomposition of both transmission and reflection layers from a single input image) is ill-posed, in order to ease the complexity for network to learn such a difficult task, we propose three extensions, i.e. edge guidance, reflection classifier, and recurrent decomposition, which are equipped to the base model and benefit the learning as well as the final performance.

3.2.1 Edge Guidance

The first extension we propose is to have the edge guidance, where the base model now takes not only the reflection contaminated image I as the single input, but also the edge map $\tilde{T}_{edge}$ of the transmission layer T estimated by an edge estimator $\mathcal{E}$. In other words, now the decomposition procedure is guided by the additional modality $\tilde{T}_{edge}$ which helps to reduce the difficulty for the model to predict the transmission and/or reflection layers. The motivation to leverage the information of image edges/gradients comes from the empirical observation that the gradients of transmission and reflection layers usually exhibit different distributions, where the reflection layer often tends to be more blurry and unclear. The idea of having edge guidance to benefit reflection removal is actually not new, several research works [4, 17, 23] have explored the similar idea in

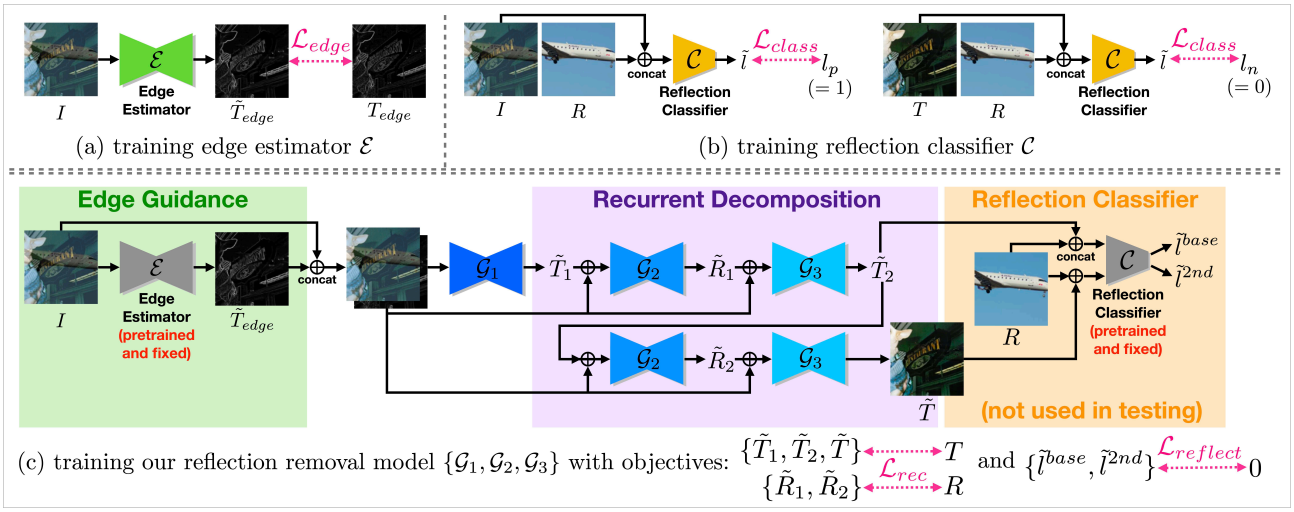

Figure 2: Illustration of our proposed method together with the training objectives. (a) The training procedure of edge estimator $\mathcal{E}$, which takes reflection contaminated image I as input and predicts edge map $\tilde{T}_{edge}$ of the transmission layer T. (b) The training procedure of reflection classifier $\mathcal{C}$, which is used to distinguish whether the reflection exists. $\mathcal{C}$ takes reflection layer R concatenated with transmission layer T or with the reflection contaminated image I as an input pair, and then outputs the corresponding label of the pair. (c) The full model of our proposed method. We shade each sub-network in different colors, where the gray-shaded ones are pretrained and fixed, while others are learnable in our full model training procedure. Our full model takes I as input, first produce the estimated transmission layer $\tilde{T}_2$, and then perform recurrent decomposition to obtain the final estimation $\tilde{T}$ of the transmission layer. Note that $\mathcal{L}_{reflect}$ is computed on both $\tilde{T}_2$ and $\tilde{T}$.

different ways. For instance, [17] estimates the gradient map of transmission layer T with two sub-aperture views available on a dual-pixel sensor, then use the gradients in their objective functions of reflection removal. However, most of the camera APIs nowadays do not access to sub-aperture view images, we would need specific cameras to obtain the sub-aperture views, thus making this approach slightly impractical. Instead, the work from [4] uses an estimation network which takes a reflection contaminated image I and its corresponding map of image gradients as the input for predicting the edge map of transmission layer.

Our approach to derive and use $\tilde{T}_{edge}$ in the decomposition procedure is similar to the one in [4] but still with few differences. Firstly, the groundtruth edge map T_{edge} for a given transmission layer T used in our learning edge estimator $\mathcal{E}$ is the normalized map of gradient magnitude:

$$T_{edge} = \text{normalize}(\sqrt{\nabla_x T^2 + \nabla_y T^2}), \tag{6}$$

where normalize function performs min-max feature scaling to bring all values into the range $[0, 1]$, and $\{\nabla_x, \nabla_y\}$ compute the image gradients along horizontal and vertical directions respectively. For [4] they directly apply filtering on T by using a 3×3 Laplacian operator to obtain T_{edge}. In other words, our groundtruth represents the probability map for each pixel being the edge, while [4] considers the value of edges in the absolute scale. In which the latter could be problematic and confusing for the edge estimator $\mathcal{E}$ to learn, when the strength of reflection has large variation in the training dataset. Secondly, the edge estimator used in our proposed method simply takes I as input, while [4] requires the input pair of I and its corresponding edge map. Our edge estimator $\mathcal{E}$ outputs $\tilde{T}_{edge} = \mathcal{E}(I) \in [0, 1]$, i.e. the edge map estimation of the transmission layer T. The objective for learning $\mathcal{E}$ is defined as:

$$\mathcal{L}_{edge} = \sum \|\tilde{T}_{edge} - T_{edge}\|_1 + \|\tilde{T}_{edge} - T_{edge}\|_2. \tag{7}$$

Lastly, owing to the special design of our base model, which performs decomposition in multiple stages, we not only use the edge guidance in predicting $\tilde{T}_1$ from I (i.e. our $\mathcal{G}_1$), but also in estimating $\tilde{R}_1$/$\tilde{T}_2$ from $\{I, \tilde{T}_1\}$/$\{I, \tilde{R}_2\}$ (i.e. our $\mathcal{G}_2$ and $\mathcal{G}_3$ respectively).

The network architecture of our edge estimator $\mathcal{E}$ is similar to U-Net [18], which has five downsampling and upsampling blocks of convolution. The channel size of the last downsample block is modified to 512, and we replace the ReLU activation functions with LeakyReLU ones. Besides, we add batch normalization after every convolutional layer and before activation functions.

3.2.2 Reflection Classifier

Different from the edge guidance which provides more information as the input to the decomposition model, here we propose an auxiliary component, i.e. reflection classifier $\mathcal{C}$,

for providing more constraints and objectives to further improve the model learning. The basic intuition behind the design of our reflection classifier is that: as I is an image contaminated by the reflection R, there should exist some structures or local patterns in I which are similar to the ones shown in R; while the transmission layer T and reflection R are typically distinct from each other, there should be no repeated patterns shared across T and R. Therefore, we can train the reflection classifier based on the positive pairs composed of $\{I, R\}$ and the negative ones composed of $\{T, R\}$. To be detailed, the network structure of the classifier $\mathcal{C}$ is based on VGG19 [22] framework, with modifications on the first convolution layer and the FC layer to support the input size of $\{I, R\}/\{T, R\}$, and a sigmoid activation function after the last layer to make the output label between $[0, 1]$. Its training objective is defined as:

$$\mathcal{L}_{class} = \sum -[l_p \log \mathcal{C}(I, R) + (1 - l_n) \log(1 - \mathcal{C}(T, R))], \tag{8}$$

where l_p and l_n are the groundtruth labels of reflection, $l_p = 1$ for the input pair of $\{I, R\}$ while $l_n = 0$ for the one of $\{T, R\}$. After training $\mathcal{C}$, we can use it to distinguish whether an estimated transmission layer $\tilde{T}$ still contains the reflection R by checking the output of $\mathcal{C}(\tilde{T}, R)$. In other words, if the decomposition is perfect, then $\mathcal{C}(\tilde{T}, R)$ should be quite close to 0 as the reflection is no longer observable in the estimated transmission layer by $\mathcal{C}$, otherwise it would be closer to 1. The reflection classifier $\mathcal{C}$ then can be used to define an objective function for training our decomposition network, which will be explained in the Section 3.3.

3.2.3 Recurrent Decomposition

As described in Section 3.1, the BDN network uses three cascade sub-networks $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ to sequentially decompose a reflection contaminated image I into the transmission layer T and reflection layer R. This sequential decomposition process can actually be further extended by adding more cascade sub-networks, as mentioned in the BDN paper [32]. However, adding more sub-networks will definitely lead to more expenses in computation and memory usage. Therefore, here we propose the idea of recurrent decomposition in order to resolve this issue. Particularly, instead of increasing the number of sub-networks with their weights non-shared, we reuse the $\mathcal{G}_2$ and $\mathcal{G}_3$ networks again, namely the second recurrence: $\mathcal{G}_2$ of the second recurrence again takes the reflection contaminated I together with the estimated transmission layer $\tilde{T}_2$ from $\mathcal{G}_3$ in our base model as the input, and outputs a reflection estimation $\tilde{R}_2$; afterwards, $\mathcal{G}_3$ of second recurrence takes the input of $\{I, \tilde{R}_2\}$ to estimate the transmission layer, denoted as $\tilde{T}_3$. Design of such recurrent decomposition is then as:

$$\tilde{T} = \tilde{T}_3 = \mathcal{G}_3(\mathcal{G}_2(\mathcal{G}_3(\mathcal{G}_2(\mathcal{G}_1(I), I), I), I), I), \tag{9}$$

where $\tilde{T} = \tilde{T}_3$ is the final estimation of the transmission layer by our recurrent model. The objective function $\mathcal{L}_{rec}^{\text{2nd}}$ for training the second recurrence is quite similar to the first recurrence (i.e. the base model), which is defined as:

$$\mathcal{L}_{rec}^{\text{2nd}} = \lambda_1 \mathcal{L}_1^{\text{2nd}} + \lambda_2 \mathcal{L}_2^{\text{2nd}} + \lambda_{feat} \mathcal{L}_{feat}^{\text{2nd}}, \tag{10}$$

where $\mathcal{L}_1^{\text{2nd}}$, $\mathcal{L}_2^{\text{2nd}}$ and $\mathcal{L}_{feat}^{\text{2nd}}$ are defined as

$$\mathcal{L}_1^{\text{2nd}} = \sum \|\tilde{R}_2 - R\|_1 + \|\tilde{T}_3 - T\|_1, \tag{11}$$

$$\mathcal{L}_2^{\text{2nd}} = \sum \|\tilde{R}_2 - R\|_2 + \|\tilde{T}_3 - T\|_2. \tag{12}$$

$$\mathcal{L}_{feat}^{\text{2nd}} = \sum \sum_l \lambda_l \| \Phi^l(\tilde{T}_3) - \Phi^l(T) \|_2. \tag{13}$$

By using the mechanism of our recurrent decomposition, the gradients of $\mathcal{L}_{rec}^{\text{2nd}}$ can contribute to update both $\mathcal{G}_2$ and $\mathcal{G}_3$ twice along the backpropagation, thus improving the efficacy of our decomposition network to achieve better reflection removal, without requiring extra memory to handle the additional sub-networks. Please note here the second recurrence does not include the $\mathcal{G}_1$ network, since $\mathcal{G}_1$ considers solely the input image I, having it in the second recurrence would imply to discard the intermediate results obtained from the first recurrence (i.e. the base model).

3.3. Full Model

We now include all the aforementioned extensions into the base model, and build up our **Full Model**: (1) we use the edge extractor $\mathcal{E}$ to predict the edge map of transmission layer, and exploit it as an additional input for the base model; (2) we use the novel reflection classifier $\mathcal{C}$ to "score" the estimated transmission layer obtained during the procedure of decomposition, and derive an objective based on $\mathcal{C}$ to improve the model learning; (3) we extend the base model to have the recurrent decomposition, by reusing sub-networks to achieve better results for reflection removal.

3.3.1 Network Structure

The architecture of our full model is illustrated in Figure 2. The sub-networks $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ used in our full model are almost identical to the ones in BDN [32], but with modifications on their first convolution layer to support the additional input channel from the edge guidance $\tilde{T}_{edge} = \mathcal{E}(I)$. To be detailed, in the first recurrence of our full model, the computations below are sequentially performed:

$$\begin{aligned} \tilde{T}_1 &= \mathcal{G}_1(I, \tilde{T}_{edge}) \\ \tilde{R}_1 &= \mathcal{G}_2(I, \tilde{T}_{edge}, \tilde{T}_1) \\ \tilde{T}_2 &= \mathcal{G}_3(I, \tilde{T}_{edge}, \tilde{R}_1) \end{aligned} \tag{14}$$

then the second recurrence follows to apply:

$$\begin{aligned}\tilde{R}_2 &= \mathcal{G}_2(I, \tilde{T}_{edge}, \tilde{T}_2) \\ \tilde{T}_3 &= \mathcal{G}_3(I, \tilde{T}_{edge}, \tilde{R}_2)\end{aligned} \tag{15}$$

and we take $\tilde{T} = \tilde{T}_3$ as the final result of transmission layer estimation produced by our full model.

3.3.2 Training Objectives

Our full model is trained in a stage-wise manner: first, $\mathcal{L}_{edge}$ and $\mathcal{L}_{class}$ are used to learn the edge estimator $\mathcal{E}$ and the reflection classifier $\mathcal{C}$ respectively, as shown in Figure 2(a) and Figure 2(b); then, we keep both $\mathcal{E}$ and $\mathcal{C}$ fixed, and train the sub-networks $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ for the decomposition, as shown in Figure 2(c). The objectives used for training $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ can be summarized into two kinds: **Reconstruction loss** and **Reflective loss**, as detailed below.

Reconstruction loss. We penalize the reconstruction error in both pixel- and feature-levels between the estimated transmission layers, i.e. $\{\tilde{T}_1, \tilde{T}_2, \tilde{T}_3\}$, with respect to the groundtruth of T, as well as the one between $\{\tilde{R}_1, \tilde{R}_2\}$ and the groundtruth of R. Therefore, the reconstruction loss $\mathcal{L}_{rec}$ for learning the sub-networks is defined as:

$$\mathcal{L}_{rec} = \mathcal{L}_{rec}^{base} + \lambda^{\text{2nd}} \mathcal{L}_{rec}^{\text{2nd}}, \tag{16}$$

where λ^{2nd} is used to balance the importance between the losses in the first and the second recurrence. Since we expect that the results obtained by the second recurrence should be better than the ones from the first recurrence, λ^{2nd} is set to 2 in our experiments.

Reflective loss. For leveraging the reflection classifier $\mathcal{C}$ into our training procedure, we feed the transmission layers estimated by $\mathcal{G}_3$ (i.e. $\tilde{T}_2$ or $\tilde{T}$) together with the groundtruth reflection layer R into the reflection classifier, and the corresponding outputs (denoted as $\tilde{l}^{base} = \mathcal{C}(\tilde{T}_2, R)$ and $\tilde{l}^{\text{2nd}} = \mathcal{C}(\tilde{T}, R)$) can be considered as the indicator which shows the degree of having reflection remained in the transmission layer estimation. Thus, by minimizing $\tilde{l}^{base}$ and $\tilde{l}^{\text{2nd}}$, we are able to update the sub-networks in order to make estimated transmission layer as free from the reflection as possible. The reflective loss $\mathcal{L}_{reflect}$ can then be written as:

$$\begin{aligned}\mathcal{L}_{reflect} &= \sum \tilde{l}^{base} + \lambda^{\text{2nd}} \sum \tilde{l}^{\text{2nd}} \\ &= \sum \mathcal{C}(\tilde{T}_2, R) + \lambda^{\text{2nd}} \sum \mathcal{C}(\tilde{T}, R).\end{aligned} \tag{17}$$

The full objective to train $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ is defined as:

$$\mathcal{L} = \lambda_{rec}\mathcal{L}_{rec} + \lambda_{reflect}\mathcal{L}_{reflect}, \tag{18}$$

where λ_{rec} and $\lambda_{reflect}$ are set to 1 and 0.5 respectively in our experiments. We will make all our source code, models, and the dataset publicly available for reproduction upon paper acceptance.

Method	SIR^2 [25]		Zhang [33]	
	PSNR	SSIM	PSNR	SSIM
Base	22.62	0.873	21.32	0.782
Base + E	24.10	**0.887**	22.22	0.798
Base + C	23.05	0.874	21.83	0.791
Base + R	22.95	0.867	21.4	0.788
Full	**24.33**	0.884	**22.86**	**0.810**

E: Edge Guidance
C: Reflection Classifier
R: Recurrent Decomposition

Table 1: Ablation study.

4. Experiments

Training data. For fair comparison, we follow the similar training data preparation as [4, 27, 33]. To be detailed, we use both synthetic and real data in our training dataset. For synthetic data, we synthesize 17k images from 2012 PASCAL VOC training images [3] with the generation method from CEILNet [4], and 2k images from images on Flickr with the generation method from [33]. For real data, we use 90 real-world training images obtained from [33], and apply typical data augmentation techniques (flipping and random cropping). During training process, we randomly choose 6k images for use in each epoch.

4.1. Ablation Study

To investigate the contribution of each component in our model, we perform ablation study by using different model variants: starting from the base model (built upon BDN [32]), we sequentially add different extensions onto it. The experiments are conducted on 453 images from SIR^2 [25] and 20 real-world testing images from Zhang *et al*. [33]. The quantitative evaluation is based on PSNR and SSIM metrics to assess the quality between the estimated transmission layer and the corresponding groundtruth.

From the quantitative results in Table 1, we can see that our base model already achieves better performance than the original BDN (which is shown in Table 2), due to our modifications on the objectives, where L_1 loss contributes to improve the sharpness of the results and perceptual loss enhances the results with more realistic textures. Also, the variants of equipping the base model with each of our proposed extensions are able to provide boost with respect to the base model, and finally our full model adequately integrates these extensions, where the extensions benefit each other and reaches the best result together. Several qualitative examples are provided in Figure 3, where we can observe that each of our proposed extension contributes in different manners. Firstly, edge extractor tends to help eliminate the blur edges from the reflection layer, and emphasize the contour of the transmission layer; secondly, training with reflection classifier lets our model better learn how

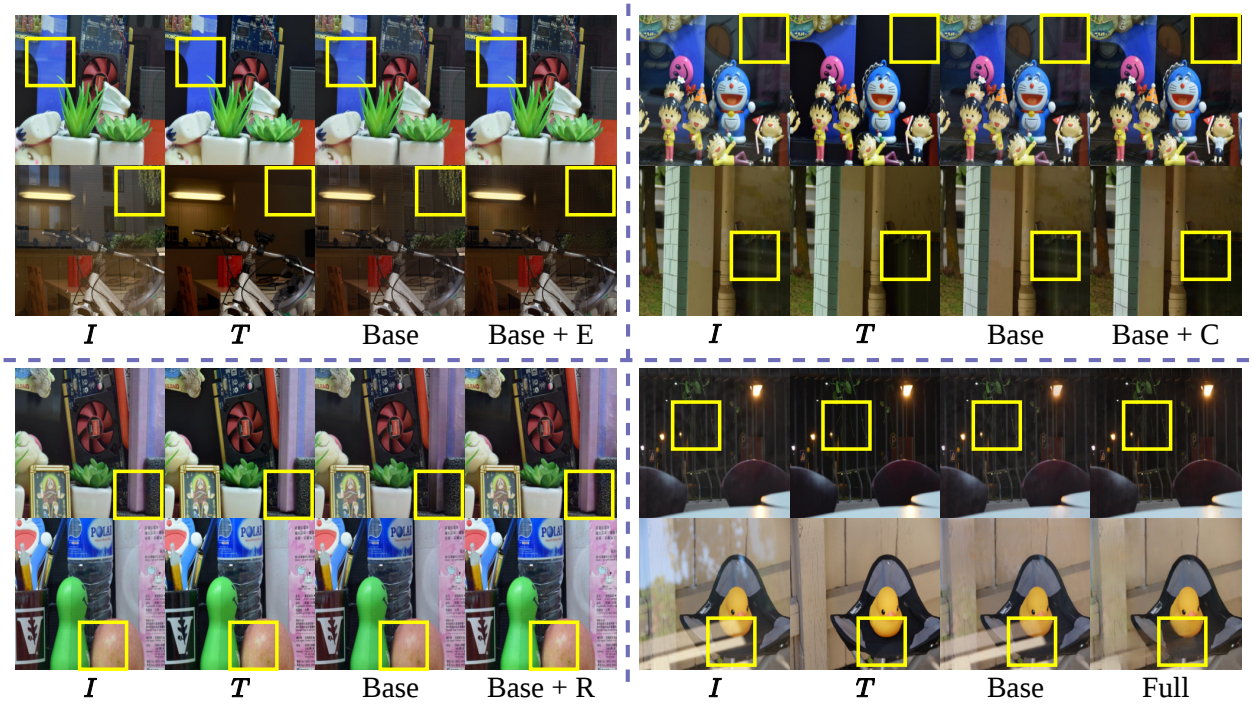

Figure 3: Example results for ablation study (cf. Section 4.1). For each block, input images, groundtruth transmission layers, results generated by base model, and results produced by base model with extensions are sequentially provided from left to right columns. "E", "C", "R" stand for edge guidance, reflection classifier, and recurrent decomposition respectively. We can observe that these extensions contribute differently to the results, and our full model is able to aggregate the advantages of each extension and produce favorable results.

Method	SIR^2 [25]						Zhang [33]		Average	
	Postcard		Solid Objects		Wild Scenes					
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
CEILNet [4]	21.08	0.829	23.53	0.884	22.06	0.826	18.45	0.690	22.12	0.8463
Zhang *et al*. [33]	16.85	0.799	22.72	0.879	21.56	0.836	21.30	0.821	20.07	0.8381
BDN [32]	20.41	0.855	22.71	0.863	22.11	0.833	18.14	0.726	21.48	0.8501
Wen *et al*. [28]	19.28	0.803	19.48	0.772	23.71	0.855	21.28	0.818	19.96	0.7967
ERRNet [27]	22.04	0.876	24.87	0.896	24.25	0.853	22.89	0.803	23.53	0.8787
IBCLN [14]	23.39	0.875	24.87	0.893	24.71	0.886	21.86	0.762	24.10	0.8791
ours	22.73	0.860	25.61	0.905	25.41	0.892	22.86	0.810	24.26	0.8806

Table 2: Quantitative results on two real-world benchmark datasets. The best and the second best results are colored in red and blue respectively. The "Average" scores are obtained by taking the average of all images from these two datasets.

to distinguish reflection from transmission layer, and therefore our model can further eliminate the remnants of the reflection layer; lastly, recurrent decomposition creates more exquisite results, which preserve the details and colors of transmission layers. Our full model gathers all advantages of three extensions and reaches the greatest improvement.

4.2. Quantitative Results

Here we compare our method against previous works which are based on deep-learning models. The experiments are conducted on two real-world benchmark datasets: 20 real-world testing images from Zhang *et al*. [33], and SIR^2 [25] which has three sub-datasets (i.e. Postcard, Solid objects and Wild scenes). The evaluation on the predicted transmission layer is based on both PSNR and SSIM metrics, which are widely used in the related works. The quantitative results are shown in Table 2. Our proposed network outperforms other methods on SIR^2 dataset except on Postcard sub-dataset, and reaches the second best on 20 real-world images from Zhang *et al*. in terms of PSNR. Overall, our method achieves the best performance in average, in comparison to other state-of-the-art methods.

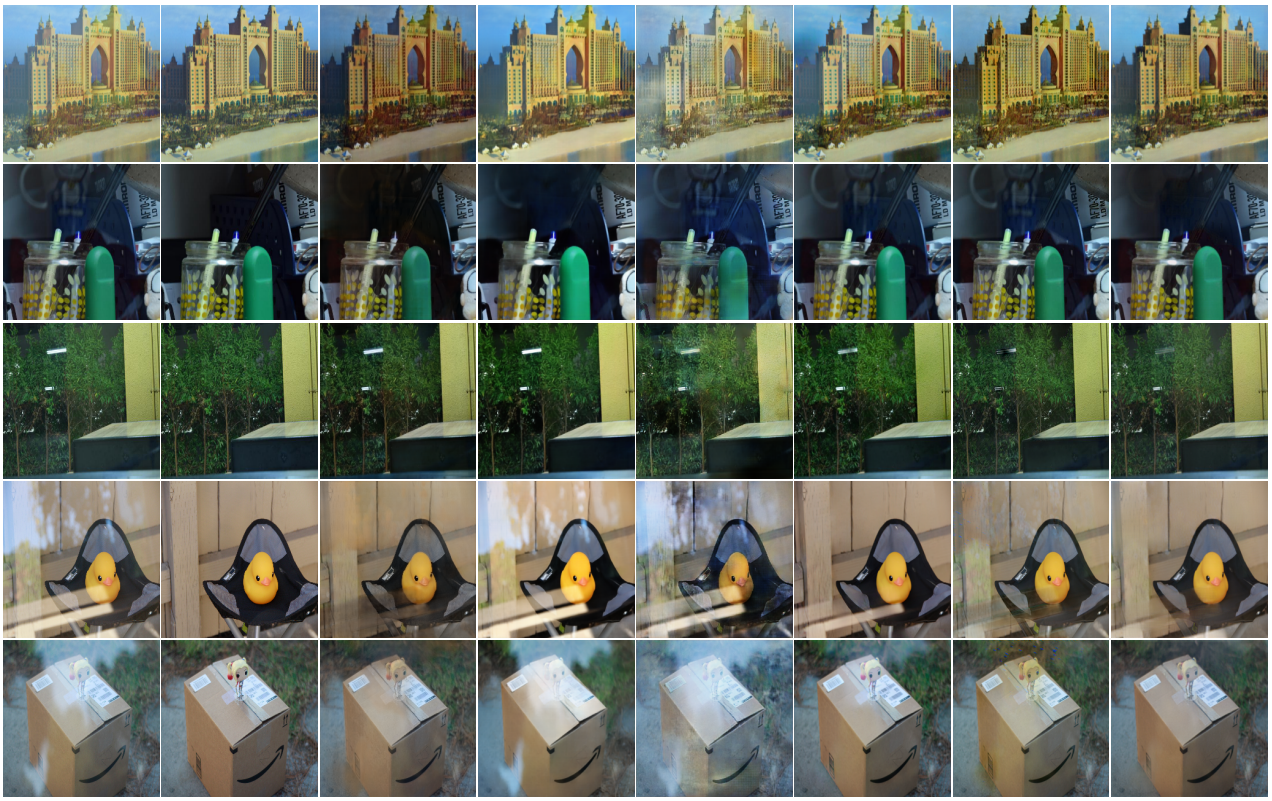

Figure 4: Qualitative examples on real-world images. The images are obtained from SIR^2 [25] (rows 1-3) and Zhang *et al*. [33] (rows 4-5). More qualitative examples are provided in the supplementary materials.

4.3. Qualitative Results

Figure 4 shows the example results obtained from Zhang *et al*. [33], BDN [32], Wen *et al*. [28], ERRNet [27], IBCLN [14], and our full model. Zhang *et al*. fail to produce adequate results, showing deviation on the image color, and could not well maintain the structure and details. BDN fails to handle reflections with high intensity. Wen *et al*. overly smooth the results and create spotty artifacts. This phenomenon could be caused by the property of their synthetic training data, which is likely to contain white patterns. ERRNet shows decent results for most of the images, however, as the estimation of reflection layer can provide complementary information when recovering the transmission layer, it couldn't well predict the transmission layer in every case potentially because it does not estimate reflection layer. On the other hand, IBCLN removes most of the undesirable reflections as it considers the reflection layer in estimation, but it tends to overly remove the reflections with high intensity, and there exists slight color shifts in some results as well. Since our method considers both the edge information and the remaining reflection in estimation, it can handle the reflection of high brightness and get clearer structure simultaneously. Moreover, by using the recurrent decomposition, the efficacy of our model is boosted with the aforementioned information twice, thus leading to the results with better preserved details and correct color. More qualitative examples are provided in the supplement.

5. Conclusion

We propose to improve the single-image reflection removal via three auxiliary techniques, including edge guidance, reflection classifier, and recurrent decomposition. The proposed method adequately decomposes the reflection contaminated input and generates reasonable estimation of the transmission layer. Ablation study shows the contribution of each auxiliary technique. In comparison to the state-of-the-art baselines, our method can remove undesired reflections as well as preserve details and color of the transmission layer, producing favorable reflection-free image.

Acknowledgement This project is supported by Ministry of Science and Technology of Taiwan (under grants MOST-109-2634-F-009-015, MOST-109-2634-F-009-020, and MOST-109-2636-E-009-018) and MediaTek (under MediaTek-NCTU Research Center). We are also grateful to the National Center for High-performance Computing of Taiwan for computer time and facilities.

References

[1] Amit Agrawal, Ramesh Raskar, Shree K Nayar, and Yuanzhen Li. Removing photography artifacts using gradient projection and flash-exposure sampling. In *ACM Transactions on Graphics (TOG)*, 2005.

[2] Zhixiang Chi, Xiaolin Wu, Xiao Shu, and Jinjin Gu. Single image reflection removal using deep encoder-decoder network. *ArXiv:1802.00094*, 2018.

[3] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, 2010.

[4] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David Wipf. A generic deep architecture for single image reflection removal and image smoothing. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[5] Hany Farid and Edward H Adelson. Separating reflections and lighting using independent components analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999.

[6] Kun Gai, Zhenwei Shi, and Changshui Zhang. Blind separation of superimposed moving images using image statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2011.

[7] Xiaojie Guo, Xiaochun Cao, and Yi Ma. Robust separation of reflection from multiple images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[8] Byeong-Ju Han and Jae-Young Sim. Reflection removal using low-rank matrix completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 2016.

[10] Pramati Kalwad, Divya Prakash, Venkat Peddigari, and Phanish Srinivasa. Reflection removal in smart devices using a prior assisted independent components analysis. In *Digital Photography XI*, 2015.

[11] Anat Levin and Yair Weiss. User assisted separation of reflections from a single image using a sparsity prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2007.

[12] Anat Levin, Assaf Zomet, and Yair Weiss. Learning to perceive transparency from the statistics of natural scenes. In *Advances in Neural Information Processing Systems (NIPS)*, 2003.

[13] Anat Levin, Assaf Zomet, and Yair Weiss. Separating reflections from a single image using local features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[14] Chao Li, Yixiao Yang, Kun He, Stephen Lin, and John E Hopcroft. Single image reflection removal through cascaded refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3565–3574, 2020.

[15] Yu Li and Michael S Brown. Exploiting reflection change for automatic reflection removal. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[16] Yu Li and Michael S Brown. Single image layer separation using relative smoothness. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[17] Abhijith Punnappurath and Michael S Brown. Reflection removal using a dual-pixel sensor. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[19] Bernard Sarel and Michal Irani. Separating transparent layers through layer information exchange. In *European Conference on Computer Vision (ECCV)*, 2004.

[20] Yoav Y Schechner, Nahum Kiryati, and Ronen Basri. Separation of transparent layers using focus. *International Journal of Computer Vision (IJCV)*, 2000.

[21] YiChang Shih, Dilip Krishnan, Fredo Durand, and William T Freeman. Reflection removal using ghosting cues. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ArXiv:1409.1556*, 2014.

[23] Chao Sun, Shuaicheng Liu, Taotao Yang, Bing Zeng, Zhengning Wang, and Guanghui Liu. Automatic reflection removal using gradient intensity and motion cues. In *ACM Conference on Multimedia (MM)*, 2016.

[24] Richard Szeliski, Shai Avidan, and P Anandan. Layer extraction from multiple images containing reflections and transparency. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000.

[25] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot. Benchmarking single-image reflection removal algorithms. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[26] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot. Crrn: Multi-scale guided concurrent reflection removal network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[27] Kaixuan Wei, Jiaolong Yang, Ying Fu, Wipf David, and Hua Huang. Single image reflection removal exploiting misaligned training data and network enhancements. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[28] Qiang Wen, Yinjie Tan, Jing Qin, Wenxi Liu, Guoqiang Han, and Shengfeng He. Single image reflection removal beyond linearity. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[29] Patrick Wieschollek, Orazio Gallo, Jinwei Gu, and Jan Kautz. Separating reflection and transmission images in the wild. In *European Conference on Computer Vision (ECCV)*, 2018.

[30] Tianfan Xue, Michael Rubinstein, Ce Liu, and William T Freeman. A computational approach for obstruction-free photography. *ACM Transactions on Graphics (TOG)*, 2015.

[31] Qing Yan, Yi Xu, Xiaokang Yang, and Truong Nguyen. Separation of weak reflection from a single superimposed image. *IEEE Signal Processing Letters*, 2014.

[32] Jie Yang, Dong Gong, Lingqiao Liu, and Qinfeng Shi. Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal. In *European Conference on Computer Vision (ECCV)*, 2018.

[33] Xuaner Zhang, Ren Ng, and Qifeng Chen. Single image reflection separation with perceptual losses. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.