Single-image reflection removal : from computational imaging to deep learning

Wan, Renjie

2018

Wan, R. (2018). Single-image reflection removal : from computational imaging to deep learning. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/82986>

<https://doi.org/10.32657/10220/47556>

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

**SINGLE-IMAGE REFLECTION REMOVAL:
FROM COMPUTATIONAL IMAGING TO DEEP LEARNING**

WAN RENJIE
INTERDISCIPLINARY GRADUATE SCHOOL
2019

Single-Image Reflection Removal: From Computational Imaging to Deep Learning

Renjie Wan

Interdisciplinary Graduate School

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2019

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

Date: 01/24/2019

Name: Renjie Wan

Signature:

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

Date: 01/24/2019

Name: Alex C. Kot

Signature:

Authorship Attribution Statement

This thesis contains material from 5 paper(s) published in the following peer-reviewed journal(s) where I was the first and/or corresponding author.

Chapter 2 is published as Renjie Wan, Boxin Shi, Ah-Hwee Tan, and Alex C. Kot. Depth of field guided reflection removal. International Conference on Image Processing, 21-25 (2016). DOI: 10.1109/ICIP.2016.7532311

The contributions of the co-authors are as follows:

- I prepared the manuscript drafts. The manuscript was revised by Dr. Boxin Shi and Prof. Alex Kot.
- I designed the methods and discussed it with Dr. Boxin Shi.
- I designed the experiments and discussed it with Dr. Boxin Shi.

Chapter 3 is published as Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot. Benchmarking Single-Image Reflection Removal Algorithms. IEEE International Conference on Computer Vision (ICCV), 3942-3950 (2017) DOI:10.1109/ICCV.2017.423

The contributions of the co-authors are as follows:

- Prof. Kot suggested the topic of this work.
- I discussed the dataset setup with Dr. Boxin Shi and took all the images in the dataset myself.
- I wrote the drafts of the manuscript. The manuscript was revised together with Dr. Boxin Shi, Prof. Ling-Yu Duan, Prof. Ah-Hwee Tan and Prof. Kot.

Chapter 4 is published as Renjie Wan, Boxin Shi, Ah-Hwee Tan, and Alex C. Kot. Sparsity based reflection removal using external patch search. IEEE International Conference on Multimedia and Expo (ICME), 1500-1505 (2017) DOI: 10.1109/ICME.2017.8019527

The contributions of the co-authors are as follows:

- I prepared the manuscript drafts. The manuscript was revised by Dr. Boxin Shi and Prof. Alex Kot.
- I designed the methods and discussed it with Dr. Boxin Shi.
- I designed the experiments and discussed it with Dr. Boxin Shi.

Chapter 5 is published as Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot. Region-Aware Reflection Removal With Unified Content and Gradient Priors. IEEE Transactions on Image Processing (TIP), 2927-2941 (2018). DOI:10.1109/TIP.2018.2808768

The contributions of the co-authors are as follows:

- I prepared the manuscript drafts. The manuscript was revised by Dr. Boxin Shi Prof. Tan, Prof. Duan, and Prof. Alex Kot.
- I designed the methods and discussed it with Dr. Boxin Shi.
- I designed the experiments and discussed it with Dr. Boxin Shi.

Chapter 6 is published as Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot. CRRN: Multi-scale Guided Concurrent Reflection Removal Network. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4777-4785. (2018) DOI: 10.1109/CVPR.2018.00502

The contributions of the co-authors are as follows:

- I prepared the manuscript drafts. The manuscript was revised by Dr. Boxin Shi Prof. Tan, Prof. Duan, and Prof. Alex Kot.
- I designed the methods and discussed it with Dr. Boxin Shi.
- I designed the experiments and discussed it with Dr. Boxin Shi.

Date: 01/24/2019

Name: Renjie Wan

Signature:

To my parents, my beloved PuPu Cat and motherland

Acknowledgments

Four years ago I made a decision to pursue this Ph.D. in NTU (Nanyang Technological University, Singapore). Although every choice in life means a different path of life, I believe the choice I made was the best. During this four-year journey, there are so many wonderful people surrounding me and supporting me. Herewith, I would like to express my greatest appreciation to all of them who shared with this journey.

Foremost, I am deeply grateful to my supervisor, Prof. Alex C. Kot, who gave me the opportunities to pursue my Ph.D. degree in NTU. This is a key point that changed the orbit of my life. Prof. Kot also initiated the framework and ideas of my Ph.D. topic. His wealth of knowledge, academic excellence, enthusiasm for different research topics, great ideas and patient guidance always benefit and inspire me. The fruitful discussions with him were the major momentum for the continuous progress of my research. I am honor and proud to be Prof. Kot's student.

I gratefully acknowledge my co-supervisor, Prof. Ah-Hwee Tan. His confidence in my abilities to carry out the research work is greatly appreciated. His emphasis on the demonstrability of the ideas has immensely influence my research work. His scrupulous altitude in every detail always reminds me the essence of being a qualified researcher.

I would like to express my deep gratitude to my another co-supervisor, Prof. Boxin Shi from Peking University, for his patience, guidance and support in my research, in particular during the begining of my research. He guided me to grow from a young student to a rigorous researcher. His patience and endeavor in improving every single

word in papers and his criticism and strict attitude towards the scientific research have been a resource and model for me to complete this thesis. Everything he taught me, in both research and life, would be a great treasure in my future career.

I also thanks Dr. Li Yu graduated from NUS for his contribution in this area. Though I do not have opportunities to meet or discuss with him, his works really inspire me a lot when I began this topic. I am also grateful to Wang Yan, Li Sheng, Lu Ze, Liu Jun, Zhang Tianyi, Gu Jiuxiang, Liu Yiding, Yu Tan, Yang Jiong, Wang Qian, and other members in ROSE lab, who as both labmates and friends, were always willing to help and gave their best suggestions. It would be an unforgettable experience to work and play with them all.

Special thanks to my father, Prof. Wan Guogen and my mother, Yang Liqiong. They were always supporting me and encouraging me with their best wishes. I also thanks my PuPu Cat, who always gives me endless love and fortune in my life.

Abstract

Reflection removal aims at enhancing the visibility of the background scene while removing the reflections for images taken through the transparent glass. Though it is of broad application to various computer vision tasks, it is very challenging due to its ill-posed nature and additional priors are needed to make this problem tractable. Traditional reflection removal methods solve this problem by making use of different heuristic observations or assumptions. These assumptions are seldom satisfied in practical scenarios. In this thesis, we generalize the assumptions for the reflection removal problems by using different information or imposing new constraints.

We first propose a method by exploring the blur inconsistency between the background and reflections. Then, we introduce the first benchmark dataset in this area and analyze limitations of existing methods based on this dataset. In the third work, we address this problem by using the sparsity prior and non-local image prior from the external source. Then, with the observation that most reflections only cover a part of the whole image, we propose a method to automatically detect the regions with and without reflections and process them in a heterogeneous manner. At last, we introduce a data-driven method by using the concurrent deep learning framework. Our methods have been evaluated by using the benchmark dataset proposed in our second work. These evaluations cover a diversity of common scenarios in our daily life; hence the experiments prove that our approaches are valid for a broad class of practical scenarios.

The main contributions of this thesis are three folds: We thoroughly study the reflection properties observed in our daily scenarios; we propose the first benchmark evaluation dataset in this area and use it to analyze the limitations of existing methods; we propose various approaches to solve this problem from different angles. The efforts and achievements in this thesis promote the practical capabilities of reflection removal techniques and provide fundamental support for future researches.

Contents

Acknowledgments	i
Abstract	iii
List of Figures	ix
List of Tables	xvii
List of Abbreviations	xviii
List of Notations	xix
1 Introduction	1
1.1 Problem Background	1
1.2 Related Work	4
1.2.1 Single-image method	4
1.2.2 Multiple-image method	11
1.3 Our Contributions	13
1.4 Organization of the Thesis	16
2 Depth of Field guided Reflection Removal	18

2.1	Introduction	19
2.2	Our Method	20
2.2.1	Single-scale inference scheme	21
2.2.2	Multi-scale inference scheme	23
2.2.3	Background Edge Selection	26
2.2.4	Reflection Edge Selection	27
2.2.5	Layer Reconstruction	29
2.3	Experiments	29
2.3.1	Visual quality comparison	30
2.3.2	Analysis for the threshold settings	31
2.4	Conclusion	32
2.4.1	Limitations	33
3	Benchmarking Evaluation Dataset for Reflection Removal Methods	35
3.1	Introduction	35
3.2	Data capture	38
3.2.1	Image alignment	40
3.3	Evaluation	42
3.3.1	Quantitative evaluation	43
3.3.2	Visual quality evaluation	46
3.3.3	Open problems	48
3.4	Conclusion	49
4	Sparsity based Reflection Removal using External Patch Search	50
4.1	Introduction	51

4.2	Sparse Representation in Image Restoration	52
4.2.1	Sparse representation	52
4.2.2	Sparsity in Image Restoration	53
4.2.3	NCSR model	55
4.3	Proposed Method	56
4.3.1	The reflection removal model	56
4.3.2	The selection of the dictionary $\mathbf{D}$	59
4.3.3	The estimation of β_i	60
4.3.4	Optimization	61
4.4	Experiments	63
4.4.1	Data preparation	63
4.4.2	Evaluations	65
4.5	Conclusion	66
5	Region-Aware Reflection Removal with Unified Content and Gradient Priors	68
5.1	Introduction	69
5.2	Our method	72
5.2.1	Detecting regions with and without reflections	73
5.2.2	Content prior	75
5.2.3	Gradient prior	79
5.3	Optimization	81
5.4	Experiment Results	84
5.4.1	Error metrics	86

5.4.2	Comparison with the state-of-the-arts	88
5.4.3	The effect of the reflection dominant region	91
5.4.4	The effect of the gradient prior	91
5.4.5	Comparison with WS17	91
5.4.6	Convergence analysis	92
5.5	Conclusion	93
6	CRRN: Multi-Scale Guided Concurrent Reflection Removal Network	96
6.1	Introduction	97
6.2	Dataset Preparation	99
6.2.1	Real-world refection image dataset for data-driven methods . . .	99
6.2.2	Generating training data	101
6.3	Proposed Method	102
6.3.1	Network architecture	102
6.3.2	Loss function	105
6.3.3	Training strategy	107
6.4	Experiments	107
6.4.1	Comparison with the state-of-the-arts	110
6.4.2	Network analysis	113
6.5	Conclusion	114
7	Conclusions and Future Works	116
7.1	Conclusions	116
7.2	Future Works	118

Author's Publications	121
Bibliography	123

List of Figures

1.1	Four examples captured in front of the glass	2
1.2	Four examples captured in front of the glass	3
1.3	The physical (upper) and mathematical (bottom) image formation models for the type of single-image reflection removal methods, where the background objects and reflections are all in the DoF.	5
1.4	An example with one natural image, its corresponding gradient, and gradient histogram. The gradient histogram of a natural image centers at zero and drops fast which forms a long-tail shape.	5
1.5	The physical (upper) and mathematical (bottom) image formation models for the types of single-image reflection removal methods, where the background objects are in focus but the reflections are not in focus. . . .	6
1.6	Gradient sparsity prior illustration with one natural image, its corresponding gradient, and gradient histogram. The gradient histogram of a natural image centers at zero and drops fast which forms a long-tail shape.	7
1.7	The physical (upper) and mathematical (bottom) image formation models for the types of single-image reflection removal methods, where the thick glass leads to the ghosting effects.	9

1.8	Two examples where the reflections are with ghosting effects. In this situation, the reflections contain two parts and one part is a spatial shift version of another part.	9
1.9	Three failed examples obtained by LB14 [1], SK15 [2], and NR17 [3].	10
1.10	Three samples from the input image sequence of Li <i>et al.</i> 's method [4].	11
1.11	Three examples for the methods using different polarizer angle [5], flash/no flash images [6, 7], and images with different focus [8].	12
2.1	The objects in the depth of field looks sharp in the final captured images, but the objects out of the depth of field looks blurred in the final capture images.	20
2.2	Two images with reflections and the gradient distributions from the regions with reflections (Blur) and without reflections (Clear).	21
2.3	The initial DoF confidence map	22
2.4	DoF confidence map obtained from Scale 1 to Scale 3	23
2.5	The pipeline of our method. For the background edge selection, the input image is first converted to the <i>Lab</i> color space. For each channel, we build one reference pyramids and three blurred pyramids. Then a DoF confidence map for each channel is computed. Finally, E_B are selected based on the confidence map. For the reflection edge selection, we compute the gradient of the input image to get the initial reflection edges. Based on the initial reflection edges and background edges obtained before we can get E_R . With the two sets of edges, B and R can be separated. We multiple R by 10 for better visualization.	24
2.6	The mixture image and its corresponding background edges.	27

2.7	The mixture image and its corresponding initial reflection edges and final reflection edges.	28
2.8	Reflection removal results on three images, compared with LB14 [1]. B^* and R^* are the estimated background and reflection images. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization).	30
2.9	One examples of reflection removal results of our method and LB14 [1].	31
2.10	The initial reflection maps E'_R and final reflection maps E_R obtained by using different thresholds.	32
2.11	Two failed examples obtained by using our method and LB14.	33
3.1	An overview of the SIR^2 dataset: Triplet of images for 50 (selected from 100) wild scenes (left) and 40 controlled scenes (right). Please zoom in the electronic version for better details.	36
3.2	An example of ‘F-variance’ (varying aperture size) and ‘T-variance’ (varying glass thickness) in the controlled scene.	38
3.3	Data capture setup and procedures. The top and bottom row show procedure to take the solid object and postcard dataset, respectively. From left to right: The mixture image I is taken with the glass; the ground truth of reflection R is captured by placing a black sheet of paper behind the glass; the ground truth of background B is captured by removing the glass.	39
3.4	Image alignment of our dataset. The first row and second row are the images before and after registration, respectively.	41
3.5	Examples of visual quality comparison. The top two rows are the results for images taken with F11/T5 and F32/T5, and bottom two rows use images taken with F32/T3 and F32/T10.	45

3.6	Examples of visual quality comparison using the wild scene dataset. The first row shows the results using images from bright scenes and the last two rows are the results using images from the dark scenes.	47
4.1	The framework of our method. Our algorithm runs on RGB channel independently. For simplicity, we only show the process on R chan- nel as an example. We first retrieve similar images from an external database (Step 1); the retrieved images are then registered to the input images (Step 2); similar patches are extracted from the retrieved images based on the exemplar patches (Step 3). In the learning stage, the PCA sub-dictionary is learned from each cluster(Step 4); then the nonlocal information are used to refine the sparse codes of the exemplar patch (Step 5 and Step 6). At last, with the refined sparse codes and the dictio- nary, the patches are refined (Step 7) and the reflection is removed (Step 8).	57
4.2	Reflection removal results comparison using our method, LB14 [1], and SK15 [2] on the postcard data. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM values are displayed below the images.	64
4.3	Reflection removal results comparison using our method, LB14 [1] and SK15 [2] on the solid object data. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualiza- tion), and SSIM values are displayed below the images.	65
5.1	Examples of real-world mixture images and reflection removal results using LB14 [1], SK15 [2], and our method.	69

5.2	The framework of our method. In the patch matching stage, we obtain reference patches from intermediate results of background in the detected reflection dominant regions using internal patch recurrence; then in the removal stage, the information from reference patches are used to refine the sparse codes of the query patches to generate the content prior. With the content prior and long-tail gradient prior, the background image $\mathbf{B}$ is recovered; based on the short-tail gradient prior, the reflection $\mathbf{R}$ is also estimated.	70
5.3	One example of the detected reflection dominant regions (white pixels in the rightmost column) with their corresponding images of background, mixture images, and reference of reflections identified by humans (red pixels in the third column). At the bottom two rows, we show two examples of the patch matching results.	73
5.4	Some sample images of the background $\mathbf{B}$ and reflection $\mathbf{R}$ and their corresponding long-tail and short-tail gradient distributions.	78
5.5	Reflection removal results on two natural images under weak reflections, compared with AY07 [9], LB14 [1], SK15 [2], WS16, and NR17 [3]. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM and sLMSE values are displayed below the images.	85
5.6	Reflection removal results on two natural images under strong reflections, compared with AY07 [9], LB14 [1], SK15 [2], WS16, and NR17 [3]. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM and sLMSE values are displayed below the images.	87

5.7	Results with and without reflection dominant region (the patch brightness $\times 1.3$ for better visualization).	88
5.8	Results with and without the gradient priors (the patch brightness $\times 1.3$ for better visualization).	90
5.9	Comparison between our proposed method and WS17 (the patch brightness $\times 1.3$ for better visualization).	92
5.10	The convergence analysis of our proposed method under different τ values.	93
6.1	Samples of captured reflection images in the ‘RID’ and the corresponding synthetic images using the ‘RID’. From top to bottom rows, we show the diversity of different illumination conditions, focal lengths, and scenes.	97
6.2	Samples of captured reflection images in the ‘RID’ and the corresponding synthetic images using the ‘RID’. From top to bottom rows, we show the diversity of different illumination conditions, focal lengths, and scenes.	99
6.3	The estimated gradient generated by the gradient inference network, compared with the reference gradient.	101

6.4	The framework of CRRN. It consists of two cooperative sub-networks: the gradient inference network (GiN) to estimate the gradients of the background and the image inference network (IiN) to estimate the background and reflection layers. We feed GiN with the mixture image and its corresponding gradient as a 4-channel tensor and IiN with the mixture image containing reflections. The upsampling stage of IiN is closely guided by the associate gradient features from GiN with the same resolution. IiN consists of two feature extraction layers to extract the scale invariant features related with the background. IiN gives the estimated background and reflection images, while GiN gives the estimated gradient of background as output.	103
6.5	Examples of reflection removal results on four wild scenes, compared with FY17[10], NR17 [3], WS16 [11], and LB14 [1]. Corresponding close-up views are shown next to the images (with patch brightness $\times 2$ for better visualization), and SSIM and SSIM _r values are displayed below the images.	108
6.6	The generalization ability comparison with FY17 [10] on their released validation dataset.	109
6.7	The generalization ability comparison with FY17 [10] on their released validation dataset.	109
6.8	The output of IiN and GiN in CRRN against IiN and GiN only. Corresponding close-up views are shown below the images (with patch brightness $\times 1.6$ for better visualization).	111

6.9	The output of IiN and GiN in CRRN against IiN and GiN only. Corresponding close-up views are shown below the images (with patch brightness $\times 1.6$ for better visualization).	112
6.10	Extreme examples with whole-image-dominant reflections, compared with FY17 [10], NR17 [3] WS16 [11] and LB14 [1].	112
6.11	Extreme examples with whole-image-dominant reflections, compared with FY17 [10], NR17 [3] WS16 [11] and LB14 [1].	113

List of Tables

3.1	Benchmark results using controlled scene dataset for four single-image reflection removal algorithms using four error metrics with F-variance and T-variance. The bold numbers indicate the best result among the four methods.	41
3.2	Benchmark results for four single-image reflection removal algorithms for bright and dark scenes in the wild scene dataset. The bold numbers indicate the best result.	43
5.1	Quantitative evaluation results using five different error metrics and compared with AY07 [9], LB14 [1], SK15 [2], WS16 [11], and NR17 [3]. .	89
6.1	Quantitative evaluation results using four different error metrics, and compared with FY17[10], NR17 [3] WS16 [11] and LB14 [1].	110
6.2	Result comparisons of the proposed CRRN against CRRN using $\mathcal{L}_1$ loss in Equation (6.7) only and its sub-networks.	113

List of Abbreviations

DoF	Depth of Field
LB14	The method proposed by Li <i>et al.</i> [1]
SK15	The method proposed by Shih <i>et al.</i> [2]
AY07	The method proposed by Levin <i>et al.</i> [9]
NR17	The method proposed by Nikolaos <i>et al.</i> [3]
FY17	The method proposed by Fan <i>et al.</i> [10]
WS16	The method proposed in Chapter 2
WS17	The method proposed in Chapter 4
WS18	The method proposed in Chapter 5
GMM	Gaussian mixture model
DCT	Discrete cosine transform
SVD	Singular value decomposition
SSIM	Structural similarity index
SI	Structural index
LMSE	Local mean square error
CRRN	Concurrent reflection removal network
SIR ²	Single image reflection removal dataset
RID	Reflection image dataset

List of Notations

I	Input mixture image
B	Background image to be recovered
R	Reflection image to be removed
E_B	Background edges
E_R	Reflection edges
$\mathcal{F}$	The network to be trained
∇B	The gradient of the background image B
∇R	The gradient of the reflection image R

Chapter 1

Introduction

1.1 Problem Background

When we take photos through the transparent glass, the final captured image always contain two parts: the background object behind the glass and the reflections. As shown in Figure 1.1, reflections observed in front of the glass significantly degrade the visibility of the scene behind the glass. Absence of a clear background not only degrades the aesthetic value of an entire image, but also causes difficulties in many computer vision tasks, such as image recognition [2], panorama [12], face recognition [13], and eye detection [14]. However, since the presence of reflections is inevitable in the real world, the natural need is to remove reflections and keep the information of the background scenes as much as possible.

Mathematically, the image formation process for this phenomenon can be directly modeled by the following term:

$$\mathbf{I} = \mathbf{B} + \mathbf{R}, \quad (1.1)$$

where $\mathbf{I}$ is the observed mixture image, $\mathbf{B}$ is the background to be recovered, and $\mathbf{R}$ is the reflection to be removed. Reflection removal aims at enhancing the visibility of the background scene $\mathbf{B}$ while removing the reflections $\mathbf{R}$.

Figure 1.1: Four examples captured in front of the glass¹.

Reflection removal is challenging due to its obviously ill-posed nature – the number of the unknown is twice the number of equations. As shown in Figure 1.2, there is infinite number of possible solutions to this problem in this situation. Besides, different from other layer separation problems (*e.g.*, haze removal and rain streak removal) with significant difference between the layer to be recovered and the layer to be removed, the similarities between the properties of the background and reflections make it more difficult to simultaneously remove the reflections and restore the contents in the background. To reduce the ill-posedness of this problem, additional priors or constraints are needed.

Many previous works have been proposed to address the difficulties existed in the reflection removal problem. Most of them solve this problem by relying on different priors observed under some special circumstances, *e.g.*, the gradient prior for different blur levels between background and reflection [1] or the ghosting effects [2], and so on. Some other methods [12, 15] use multiple images taken from different viewpoints to make this problem less ill-posed. A complete review and analysis of these related work can be found in Section 1.2. In short, while many previous methods have shown very promising results, most of them are still far from the practical use. First of all, the priors

¹Images are from Li *et al.*'s work [4].

Figure 1.2: Multiple solutions for the final estimations in the reflection removal problems².

observed under some special circumstances are often violated in real-world scenarios, since these image priors only describe a limited range of the reflection properties and may project the partial observations as the whole truth. On the other hand, as a special kind of ‘noise’, reflection only occupy a part of the whole image in many situations. However, most existing methods process every part of an image, which downgrades the quality of the regions without reflection. At last, since the previous methods are not evaluated on a benchmark dataset, it is very difficult to evaluate their performances fairly.

²The idea of this figure is borrowed from Figure 1.3 in Li Yu’s thesis [16].

1.2 Related Work

In this section, we will do a concise summary and categorization for the reflection removal problems. Various criteria can be adopted to categorize reflection removal methods. For example, they can be classified by the constraints they rely on (*e.g.*, the sparsity prior [17] and the motion cues [12], *etc.*), by whether special capture setup is employed (*e.g.*, polarizer [18], flash [19], *etc.*), or by the number of images (*e.g.*, single image [1], multiple images [20]). In this chapter, we categorize existing methods in a hierarchical and intuitive manner, by first classifying them according to the number of input images and then by different constraints imposed for solving this problem.

1.2.1 Single-image method

By taking only one image from an ordinary camera as input, the single-image method has the advantage of simplicity in data capture. By reformulating Equation (1.1) to a more specific situation, the image formation process for the single-image method can be expressed as:

$$\mathbf{I}(x) = \mathbf{B}(x) + \mathbf{R}(x) \otimes k, \quad (1.2)$$

where $\otimes$ denotes the convolutional operation, x is the pixel position, k is a convolution kernel, and $\mathbf{I}$, $\mathbf{B}$, and $\mathbf{R}$ is similar to that in Equation (1.1). As discussed in Chapter 1, due to the ill-posed nature of this problem by using only one single image as the input. Different priors or models have to be considered to make this problem tractable.

Type-I: Gradient sparsity prior. For the first type of reflections, as shown in Figure 1.3, objects behind the glass and reflections are approximately in the same focal plane. Thus, $\mathbf{I}(x)$ becomes a linear additive mixture of $\mathbf{B}(x)$ and $\mathbf{R}(x)$ and the kernel k degenerates into a one-pulse kernel δ . It is well known that the image gradient and local features such as edges and corners are sparse according to the statistics of natural im-

Figure 1.3: The physical (upper) and mathematical (bottom) image formation models for the type of single-image reflection removal methods, where the background objects and reflections are all in the DoF.

Figure 1.4: An example with one natural image, its corresponding gradient, and gradient histogram. The gradient histogram of a natural image centers at zero and drops fast which forms a long-tail shape.

ages [21, 22]. Such priors are adopted in earlier works of Levin *et al.* by separating two images with minimal corners and edges [22] or gradients [23]. However, direct optimizing such a problem shows poor convergence when textures become complex, so a more stable solution can be achieved by labeling the gradients for background and reflection with the user assistance [9, 24]. Although natural images vary greatly in their absolute color distributions, their image gradient distributions peak at zero and have heavy tails, as shown in Figure 1.4. Such a long tailed distribution can be modeled by the gradi-

Figure 1.5: The physical (upper) and mathematical (bottom) image formation models for the types of single-image reflection removal methods, where the background objects are in focus but the reflections are not in focus.

ent sparsity prior. For example in [9], a probability distribution is applied to $\mathbf{B}$ and $\mathbf{R}$. Given the user-labeled background edges E_B and reflection edges E_R , $\mathbf{B}$ and $\mathbf{R}$ can be estimated by maximizing:

$$P(\mathbf{B}, \mathbf{R}) = P_1(\mathbf{B}) \cdot P_2(\mathbf{R}), \quad (1.3)$$

where P is the joint probability distribution and P_1 and P_2 are the distributions imposed on $\mathbf{B}$ and $\mathbf{R}$. When Equation (1.3) is expanded, E_B and E_R are imposed on two penalty terms. In [9], P_1 and P_2 are two same narrow Gaussian distributions. For simplicity, the method proposed by Levin *et al.* [9] will be denoted as AY07 [9] in later chapters.

Type-II: Layer smoothness analysis. It is more reasonable to assume that the reflections and objects behind the glass have different distances from the camera, and taking the objects behind the glass in focus is a typical behavior since they are more likely to be the objects we feel interested in. In such a case, as shown in Figure 1.5, the background

Figure 1.6: Gradient sparsity prior illustration with one natural image, its corresponding gradient, and gradient histogram. The gradient histogram of a natural image centers at zero and drops fast which forms a long-tail shape.

$\mathbf{B}$ is still as sharp as that in Type-I method but the reflection $\mathbf{R}$ become a blurred version. Mathematically, the observed image $\mathbf{I}$ becomes an additive mixture of the background and the blurred reflection. The kernel k depends on the point spread function of the camera which is parameterized by a 2D Gaussian function denoted as h .

The differences in the smoothness of the background and reflection provide useful cues to perform the automatic labeling and replace the labor-intensive operation in the Type-I method, *i.e.*, sharp edges are annotated as background ($E_{\mathbf{B}}$) while blurred edges are annotated as reflection ($E_{\mathbf{R}}$). There are methods using the gradient values directly [25] and analyzing gradient profile sharpness [26, 27], and exploring DoF confidence map to perform the edge classification [11].

However, the methods mentioned above all share the same reconstruction step as [9], which means they still impose the same distributions ($P_1 = P_2$) on the gradients of $\mathbf{B}$ and $\hat{\mathbf{R}}$ (a blurred version of $\mathbf{R}$). This is not true for real scenarios, because for two components with different blur levels the sharp component $\mathbf{B}$ usually has more abrupt changes in gradient than the blurred component $\hat{\mathbf{R}}$ as shown in Figure 1.6. To address

this issue, Li *et al.* [1] introduced a more general statistical model by assuming P_1 and P_2 in Equation (1.3) as two narrow distributions as follows:

$$\begin{aligned} P_1(x) &= \frac{1}{z} \max\{e^{-\frac{x^2}{\sigma_1^2}}, \eta\}, \\ P_2(x) &= \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2}{\sigma_2^2}}, \end{aligned} \quad (1.4)$$

where x is the gradient values, z is a normalization factor, σ_1 and σ_2 are both small values making two different narrow Gaussian distributions. By assigning the two probability distributions in Equation (1.4) with different σ values, the distribution P_1 corresponding to the background can drop much faster than P_2 corresponding to the reflection. For simplicity, this method will be denoted as LB14 [1] in later chapters and sections.

However, since the Laplacian data fidelity term used by LB14 [1] is insensitive to any global shift on the pixel values, their approach may make the final estimated result darker. Though they try to compensate for this shift by re-normalizing the output to fall within a range, the color change still cannot be avoided due to the large dimensionality. To solve this problem, Nikolaos *et al.* [3] proposed another method based on a l_0 gradient sparsity prior based on a Laplacian data fidelity term. To overcome the color shift problems in LB14 [1], they use the gradient descent with Adam to optimize their algorithms instead of the Fast Fourier Transform (FFT) in LB14 [1]. For simplicity, this method will be denoted as NR17 [3] in later chapters and sections.

Type-III: Ghosting effect. Both types above assume the refractive effect of glass is negligible, while a more realistic physics model should also consider the thickness of glass. As illustrated in Figure 1.7, light rays from the objects in front of the glass are partially reflected on the outside facet of the glass, and the remaining rays penetrate the glass and are reflected again from the inside facet of the glass.

In this situation, as illustrated in Figure 1.8, the reflections contain two parts and one

Figure 1.7: The physical (upper) and mathematical (bottom) image formation models for the types of single-image reflection removal methods, where the thick glass leads to the ghosting effects.

Figure 1.8: Two examples where the reflections are with ghosting effects. In this situation, the reflections contain two parts and one part is a spatial shift version of another part.

part is a spatial shift version of another part. Such ghosting effects caused by the thick glass make the observed image I a mixture of B and the convolution of R with a two-pulse ghosting kernel $k = \alpha\delta_1 + \beta\delta_2$, where α and β are the combination coefficients and δ_2 is a spatial shift of δ_1 . Shih *et al.* [2] adopted such an image formation model, and they used a GMM model to capture the structure of the reflection. For simplicity, this method will be denoted as SK15 [2] in later chapters and sections.

Though much progress has been made in single-image solutions, the limitations are

Figure 1.9: Three failed examples obtained by LB14 [1], SK15 [2], and NR17 [3].

also obvious due to the challenging nature of this problem: Type-I methods may not work well if the mixture image contains many intersections of edges from both layers; Type-II methods require the smoothness and sharpness of the two layers be clearly distinguishable; Type-III methods need to estimate the ghosting kernel by using the autocorrelation map which may fail on images with strong globally repetitive textures. For example, as illustrated in the lower left part and right part of Figure 1.9, when the background layer appears sharp or the background and reflection looks equally sharp, Type-II methods may also cause damage to the background layers or cannot remove reflections efficiently. As shown in the upper left part of Figure 1.9, when the ghosting effects are not obvious in the final captured images, Type-III method may also have difficulties to handle this kind of scenarios.

Recently, since the deep learning has achieved promising results in both high-level and low-level computer vision problems, its comprehensive modeling ability also benefits the sing-image reflection removal problems. Compared with the methods based on handcraft priors, deep learning can learn the mapping functions from the mixture images to the estimated clean images automatically and also better capture the image properties. For example, Paramanand *et al.* [28] proposed a deep learning approach to learn the edge features of the reflections by using the light field camera. The framework

Figure 1.10: Three samples from the input image sequence of Li *et al.*'s method [4].

introduced by Fan *et al.* [10] exploited the edge information when training the whole network to preserve the image details better. Though the latest deep learning based methods better capture the image properties, the requirement for the large-scale training dataset also limits its practical use in some specific scenarios.

1.2.2 Multiple-image method

Another category of methods adopts multiple images to solve this problem. Compared with the single-image methods that use one single image as the input, the multiple-image methods use multiple images taken with different conditions (*e.g.*, illuminations, viewpoints, different focuses, or varied polarizer angles). Due to the use of multiple images as the input, the limitations existed in the single-image method can be partially suppressed.

The first category of multiple-image methods exploits the motion cues between the background and reflection using at least three images of the same scene from different viewpoints, as shown in Figure 1.10. Assuming the glass is closer to the camera, the projected motion of the background and reflection is different due to the visual parallax. Such different motions existed between each layer can be represented using parametric

Figure 1.11: Three examples for the methods using different polarizer angle [5], flash/no flash images [6, 7], and images with different focus [8].

models, such as the translative motion [29], the affine transformation [15] and the homography [30]. In contrast to the fixed parametric motion, dense motion fields provide more general modeling of layer motions represented by per-pixel motion vectors. For example, as shown in Figure 1.10, the method proposed by Li *et al.* [31] makes use of the SIFT flow [31] to estimate the dense motion fields for each layer by assuming the background dominates in the mixture image and the images are related by a warping as follows:

$$\mathbf{I}_i = \omega_i(\mathbf{R}_i + \mathbf{B}), \quad (1.5)$$

where $\mathbf{I}_i$ is the i th image and $\{\omega_i\}$ is the estimated motion field, which can be used as useful information to remove reflections.

Except for the method proposed by Li *et al.* [4], other existing methods also estimate the dense motion fields for each layer using optical flow [32], SIFT flow [33, 34], and the pixel-wise flow field [12].

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

$$\mathbf{I}_i(x) = \alpha_i \mathbf{B}(x) + \beta_i \mathbf{R}(x), \quad (1.6)$$

where the mixing coefficients α_i and β_i can be estimated by taking a sequence of images using special devices or in different environments. For example, as shown in the left part of Figure 1.11, the methods proposed by [5, 18, 35–37] solves this problem by rotating the polarizers based on the observations that the effect of the reflections can be reduced by placing a polarizer in front of a camera lens to filter out the reflected light being polarized. The method by [38] introduces another method to solve this problem by taking two pictures under different illumination conditions. Sarel *et al.* [39] also propose another method to find the relationship between the background and reflection by using the repetitive dynamic behaviors existed in the captured images.

The third category of multiple-image methods takes a set of images under special conditions and camera settings. For example, as shown in the middle part of Figure 1.11, the method proposed by Agrawal *et al.* [6, 7] solves this problem by using two images with and without flash, since the images taken by using flash and no-flash illustrate different properties. Other methods also use the images with different focus [8] as shown in the right part of Figure 1.11, the light field camera [40], and images taken by the front and back camera of a mobile phone [41] to solve this problem.

Due to the additional information from multiple images, the problem becomes less ill-posed or even well-posed. However, special data capture requirements such as observing different layer motions or the demand for specialized equipment such as the polarizer largely limit such methods for practical use, especially for mobile devices or images downloaded from the Internet.

1.3 Our Contributions

The goal of our work is to design robust and efficient methods to solve the reflection removal problem. To increase the generalization ability of our proposed methods, we focus on solving this problem by using one single image as the input. In this thesis, we

thoroughly study the properties of different reflection and try to solve this problem from different angles. We first introduce three methods on the basis of the non-learning framework and a benchmark evaluation dataset. At last, we introduce a data-driven method based on the deep learning framework. Specifically, our contributions are concluded as follows:

Depth of field guided reflection removal. This method is proposed based on the assumption that the photographers always focus on the background in a particular depth when they take photos. Reflections in different depth layers would be blurred in the final captured images. Thus, DoF, the distance between the nearest and farthest objects in a scene that appears reasonably sharp [42], can be used as an important feature to distinguish background and reflection. Inspired by [42], we propose a DoF confidence map computing strategy to evaluate the blur degrees for all pixels. We also observe that images with different resolution can exhibit different levels of details in the DoF map. Combining the assumption and observation, we develop a multi-scale inference scheme to select background and reflection edges to guide the reflection removal process. With the selected edges, the classical approach in [9] can be directly applied for background reconstruction. Compared with the previous methods (*e.g.* [1, 4]), this proposed method shows better estimation results. This work has been accepted by International Conference on Image Processing 2016 (ICIP 2016) [11]. For simplicity, this method will be denoted as WS16 in this thesis.

Benchmarking Evaluation Dataset for Reflection Removal Methods Due to the lack of suitable benchmark data with ground truth, a quantitative comparison of existing approaches using the same dataset has never been conducted. Even for our method proposed in WS16, we also only conduct the visual quality comparison. To facilitate the quantitative comparisons, we also introduce the first captured single-reflection removal dataset with 40 controlled and 100 wild scenes, ground truth of background and

reflection. For each controlled scene, we further provide ten sets of images under varying aperture settings and glass thicknesses. Extensive experiments on a this benchmark dataset show that the proposed deep learning based method performs favorably against state-of-the-art methods. This dataset has been accepted by International Conference on Computer Vision 2017 (ICCV 2017) [43]. It will be denoted as SIR² in this thesis.

Sparsity based reflection removal by using external patch search. In this work, we propose a method based on the sparse representation model and nonlocal image priors. The sparse representation model is responsible for the background image reconstruction and the nonlocal image prior can learn the correlation existed in the images. To make the final estimated results more robust, we leverage the retrieved image patches from an external database to overcome the limited prior information in the input mixture image. The experimental results show that our proposed model performs better than the existing state-of-the-art reflection removal method for both objective and subjective image qualities. This work has been accepted by International Conference on Multimedia and Expo 2017 (ICME 2017) [17]. For simplicity, this method will be denoted as WS17 in this thesis.

Region-aware reflection removal by using content prior. In this work, we propose a region-aware reflection removal (R3) approach to release the requirement for the external reference images in WS17. As a special kind of ‘noise’, in many real-world scenarios, visually obvious reflections only dominate a part of the whole image plane. Our method first detects the regions with and without reflections automatically. Given the region information, we apply customized strategies to handle them, so that the regional part focuses on removing the reflection with fewer artifacts and the global part keeps the consistency of the color and gradient information. We integrate both the content and gradient priors into a unified framework, with the content priors restoring the missing contents caused by the reflection (regional) and the gradient priors separating

the two images (global). From the experimental results, this method shows fewer reflection residues and more complete image content than previous methods. This work has been accepted by IEEE Transactions on Image Processing [44]. For simplicity, this method will be denoted as WS18 in this thesis.

Deep learning based reflection removal. This method is based on the deep learning framework. The non-learning framework (*e.g.*, the gradient priors and the content priors) adopted by previous methods (*e.g.*, WS16, WS17, and WS18) are often violated in the real-world scenarios since they only describe a limited range of the reflection properties and project the partial observation as the whole truth. To capture the reflection properties more comprehensively, we propose the Concurrent Reflection Removal Network (CRRN) to tackle this problem in a unified framework. Our proposed network integrates image appearance information and multi-scale gradient information with human perception inspired loss functions and is trained on a new dataset with 3250 reflection images taken under diverse real-world scenes. Extensive experiments on a public benchmark dataset show that the proposed method performs favorably against state-of-the-art methods. This work has been accepted by IEEE Conference on Computer Vision and Pattern Recognition 2018 (CVPR 2018) [45]. For simplicity, this method will be denoted as CRRN in this thesis.

1.4 Organization of the Thesis

This thesis is organized as follows:

In Chapter 1, we provide an introduction to the reflection removal problem with the related work, our goals, and contributions.

In Chapter 2, we present the work related to the depth of field guided reflection removal.

In Chapter 3, we introduce the first benchmark dataset in this area and analyze the

limitations of existing methods on the basis of the experimental results on this dataset.

Chapter 4 introduces the sparsity based reflection removal method.

Chapter 5 discusses the region-aware reflection removal with unified content and gradient priors.

Chapter 6 introduces the deep learning based reflection removal network.

Chapter 7 concludes the dissertation by summarizing the proposed methods and discussing potential future research directions.

Chapter 2

Depth of Field guided Reflection Removal

In this chapter, we present a visual depth guided method to remove reflections. Since previous method [9] has already proposed an effective method to reconstruct the background and reflection based on the background and reflection edges, locating reflection and background edges becomes a key step for reflection removal. Different from previous methods that mainly label the edges manually or by multiple images, our idea is to use the Depth of Field (DoF) to label the background and reflection edges automatically with only one single image. We propose a DoF confidence map where pixels with higher DoF values are assumed to belong to the desired background components. Moreover, we observe that images with different resolutions show different properties in the DoF map. Thus, we introduce a multi-scale DoF computing strategy to classify edge pixels more efficiently. Based on the results of edge classification, the background and reflection layers can be reconstructed. Experimental results validate the effectiveness of our method using real-world images.

2.1 Introduction

As described in Chapter 1, it is very difficult to solve the reflection removal problem by using one single image and additional priors are needed to make this problem less ill-posed. In order to solve this problem, AY07 [9] proposed a method on the basis of the Laplacian mixture model by forcing gradient sparsity prior to both of background and reflection layers. However, their method requires additional labor-intensive user-markups to label the background and reflection edges, which is not applicable in the daily scenarios. Though some previous methods [4] are proposed to replace the user-markup steps in Levin *et al.*'s method, the requirement for the images taken from different viewpoints largely limits their practical uses in many different scenarios.

Our proposed method in this chapter is constructed on the basis of AY07 [9], but it removes the requirement for user-markups by using the information from the Depth of Field. Depth of field (DoF) is defined as the distance between the nearest and farthest objects in a scene that appears reasonably sharp [42] in the final captured image. As illustrated in Figure 2.1, the objects in the DoF looks sharp in the final captured image, while the objects not in the DoF are blurred. For a given subject framing and camera position, the DoF is controlled by the ration of lens focal length to aperture diameter and the lens aperture diameter, which is usually specified as the f-number. Mathematically, the DoF value for a camera can be obtained as follows:

$$\text{DoF} = \frac{2Ncf^2s^2}{f^4 - N^2c^2s^2}, \quad (2.1)$$

where N is the aperture size, f is the focal length, s is the subject distance, and c is the circle of confusion.

As a very important property in the photography, different DoF values are choosen to emphasize the region of interest in an image. For most photographers, they always focus on the background objects behind the glass in a particular depth when they take

Figure 2.1: The objects in the depth of field looks sharp in the final captured images, but the objects out of the depth of field looks blurred in the final capture images.

photos, since these objects are more likely to be the objects they feel interested in. In this situation, reflections in different depth would be blurred in the final captured images. Thus, different blur levels brought by the DoF can be used as a very important feature to distinguish the background and reflection.

Inspired by [42], we propose a DoF confidence map computing strategy to estimate the DoF for each pixel in an image by using its blur levels. We also observe that images with different resolution can exhibit different details in the DoF confidence map. Combining the assumption and observation, we develop a multi-scale inference scheme to select background and reflection edges to guide the reflection removal process. With the selected edges, the classical approach in [9] can be directly applied for layer separation. Compared with the state-of-the-art methods (*e.g.* [1, 4]), our method shows better separation results.

The rest of this chapter is organized as follows. We introduce our method in Section 2.2. Experimental results and discussions are presented in Section 2.3. Finally, we conclude this chapter in Section 2.4.

2.2 Our Method

Though Equation (2.1) can compute the DoF for a camera, it cannot be used to measure the DoF value of an image directly. To determine the DoF of an image, our method relies on the fact the the shape of the image horizontal and vertical derivatives histogram is

Figure 2.2: Two images with reflections and the gradient distributions from the regions with reflections (Blur) and without reflections (Clear).

modified after a blurring operation [42]. In this section, we first briefly review the single-scale inference framework to determine the DoF of an image used by previous methods. Then, we introduce our multi-scale inference scheme.

2.2.1 Single-scale inference scheme

Figure 2.2 presents two images with reflections. Since the background objects behind the glass is in the DoF range, it looks sharp in this image while the reflections look blurred. In Figure 2.2, we also plot the log histograms of derivatives of the two images from the regions with reflections (marked with a red rectangle) and without reflections (marked with a blue rectangle). As can be seen, the blurring effect changes the shape of the histograms significantly. It suggests that the distributions of the derivative filter responses can be used to measure the DoF and distinguish the difference between the background and reflection.

Let f_k denote the blurring kernel of size $k \times k$ ($k = \{3, 5, 7\}$). When convolving the image $\mathbf{I}$ with f_k , and computing the horizontal and vertical derivatives from $\mathbf{I} * f_k$, we can compute the distributions of the horizontal and vertical derivatives as follows:

$$\begin{aligned} p_{xk} &\propto \text{hist}(\mathbf{I} * f_k * d_x) \\ p_{yk} &\propto \text{hist}(\mathbf{I} * f_k * d_y), \end{aligned} \tag{2.2}$$

Figure 2.3: The initial DoF confidence map

where $d_x = [1 \ -1]$ and $d_y = [1 \ -1]^\top$.

Then, to measure the difference between the distributions after blurring operations p_{xk} and p_{yk} and the original distributions without blurring operations p_{x1} and p_{y1} , for a pixel (i, j) and a kernel k , we compute the KL divergence between them as follows:

$$D_k(i, j) = \sum_{(n,m) \in W_{i,j}} KL(p_{xk}|p_{x1})(n, m) + KL(p_{yk}|p_{y1})(n, m), \quad (2.3)$$

where $W_{i,j}$ is a window centered on the pixel (i, j) and the KL-divergence for a given pixel located at (i, j) is given by the following formula:

$$KL(p|q)(i, j) = p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right), \quad (2.4)$$

where the two probability density functions p and q both sum to one. The KL-divergence is only defined when p_{ij} and q_{ij} are greater than zero. The quantity $0 \log 0$ is considered as zero.

Then, unlike previous work [42] which only computes a DoF value for a whole image, we measure the DoF values for each pixel by computing the DoF confidence map as follows:

$$\text{DoF}_t = \sum_k D_k(i, j). \quad (2.5)$$

The procedures discussed above are repeated in the L , a , and b channel of the input images, respectively. Then, we can obtain an initial DoF confidence map by summing

Figure 2.4: DoF confidence map obtained from Scale 1 to Scale 3

DoF_t from three channels as follows:

$$\text{DoF} = \sum_{t \in \{L, a, b\}} \text{DoF}_t. \quad (2.6)$$

Two results of the initial DoF confidence maps are shown in Figure 2.3, where the information related to the background are all kept and most information related to the reflections are removed.

2.2.2 Multi-scale inference scheme

The framework presented in Section 2.2.1 have been adopted by many classic methods to explore the property of the blurring effect. Though it has shown promising results, it is still not accurate to know whether a region belongs to the background or not by using only one resolution. As discussed by previous work [46], since the scenes in the world contain objects of many sizes and objects can be at various distances from the photographers, any procedures that are applied only at a single scale may miss information at other scales. From Figure 2.4, by downsampling the image to different scales, the DoF confidence map of Scale 3 exhibits more details than that of Scale 1. Though the scale ambiguity has been studied in many applications (*e.g.*, noise reduction [47], image analysis [48], image enhancement [49], *etc.*), it is seldom discussed in the reflection

Figure 2.5: The pipeline of our method. For the background edge selection, the input image is first converted to the *Lab* color space. For each channel, we build one reference pyramids and three blurred pyramids. Then a DoF confidence map for each channel is computed. Finally, E_B are selected based on the confidence map. For the reflection edge selection, we compute the gradient of the input image to get the initial reflection edges. Based on the initial reflection edges and background edges obtained before we can get E_R . With the two sets of edges, B and R can be separated. We multiple R by 10 for better visualization.

removal problem. To increase the generalization ability of our method, in this section, we extend the classic single-scale framework in Section 2.2.1 to a multi-scale inference scheme to extract the edge information related to the background and reflections.

2.2.2.1 Construction of Image Pyramids

To fuse the information from difference scales, we resort to the image pyramid model by adapting it to our problems. As a classic image processing technique, the image pyramid model can extract the information from different scales by downsampling the images step by step. In our problems, the pyramids are built as follows. First, we build a reference pyramid b_r with 3 layers where each layer is down-sampled by a factor to form the next coarser level. The resolution of the first layer is the same as that of the input

image. Then, based on b_r , we build three pyramids which is denoted as $\{b_3, b_5, b_7\}$. The subscript here is the blurring kernel size of $k \times k$ ($k = \{3, 5, 7\}$) for that pyramid. It means that each layer of the three pyramids is a blurred version of the corresponding layer of the reference pyramid, which can be expressed as follows:

$$b_k = b_r * f_k, \quad (2.7)$$

where $*$ is a 2D convolution operator and f_k is a Gaussian blurring kernel. After this step, as shown in Figure 2.5, four pyramids have been created for the next step which is (b_r, b_3, b_5, b_7) . The pyramids for other channels can be computed in a similar way.

2.2.2.2 Pyramid Fusion

For the four image pyramids obtained before, we use the method in Equation (2.3) to calculate the KL divergence between the corresponding layers of b_k and b_r after the vertical and horizontal derivatives are applied on each layer. The result is denoted as $\{D_k^n\}$ where n is the level number from 1 to 3. Then, similar to Equation (2.5), we compute a DoF confidence map for images with different resolution as

$$D^n(i, j) = \sum_{k=\{3,5,7\}} D_k^n(i, j), n = \{1, 2, 3\}. \quad (2.8)$$

We call the DoF maps with different scales generated by Equation (2.8) as DoF pyramids, which is $\{D^1, D^2, D^3\}$, respectively.

Combining the 3 maps with different scales, we define our final DoF confidence map D as

$$D = (\lambda \cdot D^2 \uparrow + (1 - \lambda) \cdot D^3 \uparrow) \odot (D^1), \quad (2.9)$$

where $\odot$ is the elementary multiplication and $\uparrow$ indicates that D^2 and D^3 are upsampled to the same size of D^1 . The DoF confidence maps in three channels are also shown

Algorithm 1 Multi-scale Background Edge Extraction

Require: Input image I ;
Ensure: Background edge map E_B ;

- 1: **for** $m = 1$ to 3 (corresponding to $\{L, a, b\}$) **do**
- 2: Build one reference image pyramid $\{b_r\}$ with level index $\{1, 2, 3\}$;
- 3: Build three image pyramids $\{b_3, b_5, b_7\}$ with level index $\{1, 2, 3\}$ based on b_r (Eq. (2));
- 4: Compute KL divergence between $\{b_3, b_5, b_7\}$ and b_r using the method in [42];
- 5: Build the DoF pyramid (Equation (2.8));
- 6: Combine multi-scale DoF maps (Equation (2.9) and Equation (2.10));
- 7: **if** $m = 1$ **then**
- 8: Compute τ_s using Equation (2.11);
- 9: **else**
- 10: $\tau_s = \tau_s / 1.5$;
- 11: **end if**
- 12: Select salient edges E_b (Equation (2.12));
- 13: $E_B = \bigvee_{m=1}^3 E_b^m$; **return** E_B ;
- 14: **end for**

in Figure 2.5, which exhibit very large difference in reflection and background areas and can help us to distinguish the two components.

2.2.3 Background Edge Selection

Then, the background edges are determined by removing the pixels belonging to the reflection components in the DoF maps as follows:

$$E_b = H(D - \tau_s), \quad (2.10)$$

where H is the Heaviside step function, generating zeros for negative values and ones for positive values. τ_s is a threshold to determine the background edges.

Similar strategy can be used in computing E_b in three color channels. For different channels of Lab , to infer subtle structures during this process, we decrease the value of τ_s in each iterations to include more details of the background. Instead of using a constant threshold, we use the DoF maps in L channel to determine the initial threshold

Figure 2.6: The mixture image and its corresponding background edges.

value. This is similar to the adaptive threshold proposed by Hou *et al.* [50]. In our case, the adaptive threshold is the mean confidence value shown below,

$$\tau_s = \frac{1}{W \times H} \sum_{i=1}^W \sum_{j=1}^H D(i, j), \quad (2.11)$$

where W and H are the width and height of the confidence map in pixels, respectively.

Finally, the edge map of the background can be generated as follows:

$$E_B = \bigvee_{m=1}^3 E_b^m, \quad (2.12)$$

where $\bigvee$ denotes logical *or* and m is the channel number corresponding to $\{L, a, b\}$, respectively. At last, the background edge maps obtained are illustrated in Figure 2.6. This is a binary mask for the background component.

2.2.4 Reflection Edge Selection

Compared with the background, most reflection layers are related to smaller gradient magnitudes. Thus, in the gradient domain, we can obtain an initial reflection edge map

Figure 2.7: The mixture image and its corresponding initial reflection edges and final reflection edges.

based on the following threshold:

$$E'_R(i) = \begin{cases} 1, & \text{if } \tau_{r1} < g(i) < \tau_{r2} \\ 0, & \text{otherwise} \end{cases}, \quad (2.13)$$

where $g(i)$ is the gradient value of the input mixture image on pixel i . The initial reflection edge map shown in Figure 2.7 includes some misclassified background edges. Having created the map indicating the regions of background components, we now use it to refine E'_R . To remove more small details from the background components, we create a mask by using an appropriate structuring element S to dilate the background edge map as

$$M = E_B \oplus S. \quad (2.14)$$

Then we can reduce the artifacts in the initial reflection edge map as follows:

$$E_R = \overline{M} \odot E'_R, \quad (2.15)$$

where $\overline{M}$ denotes *not* operation over M . Finally, the calculated reflection edge map E_R is shown in Figure 2.7.

2.2.5 Layer Reconstruction

With the background and reflection edge maps generated before, the reflection and background layers can be separated based on the objective function proposed by AY07 [9] in this step. Levin *et al.* [9] have already shown that the long tailed distribution of gradients in natural scenes is an effective prior in this problem. The objective function is defined as follows:

$$\begin{aligned}
 J(\mathbf{B}) = & \sum_{i,k} \rho(f_{i,k} \cdot \mathbf{B}) + \rho(f_{i,k} \cdot (\mathbf{I} - \mathbf{B})) \\
 & + \gamma \sum_{i \in E_{\mathbf{B}},k} \rho(f_{i,k} \cdot \mathbf{B} - f_{i,k} \cdot \mathbf{I}) \\
 & + \gamma \sum_{i \in E_{\mathbf{R}},k} \rho(f_{i,k} \cdot \mathbf{B}),
 \end{aligned} \tag{2.16}$$

where $f_{i,k}$ is the k -th derivative filter. $E_{\mathbf{B}}$ and $E_{\mathbf{R}}$ are two sets of background and reflection edges obtained before, respectively. While the first term in Equation (2.16) ensures the gradients of the two layers as sparse as possible, the last two terms force the gradients of $\mathbf{B}$ at edge positions in $E_{\mathbf{B}}$ to agree with the gradient of input image $\mathbf{I}$ and gradients of $\mathbf{R}$ at edge positions in $E_{\mathbf{R}}$ agree with the gradients of $\mathbf{I}$. More details can be found in [9].

2.3 Experiments

To show the performance of our method, we compare our method with LB14 [1], which also uses one image as input. For the threshold values τ_{r1} and τ_{r2} , we set them to 0.1 and 0.3, respectively. λ in Eq. (4) is set to 0.4 in our experiment. These parameters are set empirically and we will also provide an analysis on the influence of these parameters to the final performances of our method in this section. We evaluate the performance on 10 images from [4, 51] and Internet.

Figure 2.8: Reflection removal results on three images, compared with LB14 [1]. B^* and R^* are the estimated background and reflection images. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization).

2.3.1 Visual quality comparison

We show four examples in Figure 2.8 and one example in Figure 2.9. As we can observe, our method can generate a clear separation with fewer residuals. Considering the regions highlighted by rectangles of Fig. 3, our algorithm can remove the majority of reflections which is better than the ones generated by LB14 [1] where the results still contain visible residual edges. Moreover, in the second row of Fig. 3(d), the reflections are not removed but the details in the non-reflection areas are over-smoothed. On the other hand, the

Figure 2.9: One examples of reflection removal results of our method and LB14 [1].

method in LB14 [1] causes a color change. It is mainly because of the insensitivity of the Laplacian data fidelity term to the spatial shift of the pixel values. From the three results generated by LB14 [1], we can find that the results are darker than the original image and this phenomenon is very obvious in the second and third row of Fig. 3(d).

2.3.2 Analysis for the threshold settings

From previous visual quality comparison, the performances of our method are largely influenced by the selected background edge maps and reflection edge maps. The two edge maps in our method are decided by the thresholds τ_{r1} and τ_{r2} in Equation (2.13) and threshold τ_s in Equation (2.10). In our proposed method, τ_s is set by making it equal to the mean values of the DoF confidence maps and τ_{r1} and τ_{r2} are set empirically. We also conduct another experiment to verify the influence of different thresholds τ_{r1} and τ_{r2} to our proposed method.

Actually, different τ_{r1} and τ_{r2} mainly influences the initial reflection edge maps E'_R in Equation (2.13). As shown in Figure 2.10, when the range of the two thresholds become larger, more misclassified pixels are included in the initial reflection map. To make the estimation more robust to different value settings, Equation (2.15) is proposed to exclude the misclassified pixels in the initial reflection map from the background. From Figure 2.10, E'_R indeed varies with the threshold, *i.e.*, more small edges are included with the increasing threshold. However, due to the introduction of Equation (2.15)

Figure 2.10: The initial reflection maps E'_R and final reflection maps E_R obtained by using different thresholds.

in our method, the isolated misclassified pixels are excluded in the final reflection edge map E_R . Thus, the final estimations remain stable and are not very sensitive to the threshold.

2.4 Conclusion

A method to remove reflections with a single image is proposed in this chapter. Different from previous works based on the single-scale inference scheme, our framework is extended to the multi-scale inference scheme by using the image pyramid model. With the multi-scale inference scheme, we first compute a DoF confidence map. Using this the confidence map, we then generate the background edge map which denotes the edges belonging to the background layers. At last, we introduce a method to generate the reflection edges based on the background edges. With the background and reflection edges, our method can well remove the reflections and keep the background information. The experimental results proves the effectiveness of our method.

Figure 2.11: Two failed examples obtained by using our method and LB14.

2.4.1 Limitations

Though our proposed method have shown promising results when compared with previous method. It still have several limitations as follows:

- Due to the lack of the benchmark dataset, in this chapter, we only compare the visual quality of the final estimated images, which can not be regarded as a fair comparison.
- Our method cannot deal with image containing tiny artifacts or the scenarios where the background and reflection have similar blur levels. Since our method is proposed for the scenarios, where the background and reflections have different blur levels, it may cause damage to the background information, when the background and reflections are similar. For example, as shown in the second row of Figure 2.11, since the reflections and the cloud have similar properties, they are all removed in the final estimated results. For the first row in Figure 2.11, since the background objects contain many small artifacts, in the final estimated results many important background details are all wrongly removed. However, even in

this situations, our method still performs much better than Li *et al.*'s method.

- As a global method which processes every part of the input image, it also causes damage to regions which are not covered by the reflections.
- We believe that reflection removal is an application that would be welcomed on many mobile devices, however, the processing time of our proposed method is still too long for real world use. Exploring ways to speed up the processing pipeline is an area of interest for future work.

In the next chapter, to solve the limitations of the lack of benchmark dataset, we will first propose a benchmark dataset and then analyze the limitations of existing methods on this dataset.

Chapter 3

Benchmarking Evaluation Dataset for Reflection Removal Methods

The single image reflection removal problems have been studied for more than decades. However, almost all existing works including our methods proposed in Chapter 2 evaluate the quality of the final estimated results by checking subjective visual quality; quantitative evaluation is performed only using synthetic data, but seldom on real data due to the lack of appropriate dataset. Though some previous methods also propose a dataset to do some quantitative analysis, the scale and diversity of these datasets are still not enough to evaluate the limitations of each method. To solve this problem, in this chapter, we introduce the first Single-image Reflection Removal dataset (‘**SIR**²’) for the single image reflection removal algorithms. Then, based on this dataset, we also conduct a thorough evaluation to analyze the pros and cons of the existing methods.

3.1 Introduction

Due to the lack of the benchmark dataset with the ground truth, almost all existing works evaluate the quality of the final estimated results by checking subjective visual quality; quantitative evaluation is performed only using synthetic data. Though some

Figure 3.1: An overview of the SIR^2 dataset: Triplet of images for 50 (selected from 100) wild scenes (left) and 40 controlled scenes (right). Please zoom in the electronic version for better details.

existing works release the dataset with their papers [1, 12]. However, their data cannot be used for benchmark purpose due to the lack of ground truth [1] or too limited size (only three scenarios with 45 images [12]). Benchmark datasets have been served as stimuli to future research for various physics-based vision problems such as intrinsic image decomposition [52] and photometric stereo [53]. These factors motivate us to create the SIR^2 benchmark dataset with a large number and a great diversity of mixture images, and ground truth of background and reflection.

In this chapter, to facilitate the benchmarking evaluations and analyze the limitations of existing methods, we propose the first benchmark evaluation dataset SIR^2 for reflection removal problems. Our SIR^2 dataset contains a total of 1,500 images. We capture 40 controlled indoor scenes with complex textures, and each scene contains a triplet of images (mixture image, ground truth of background and reflection) under seven different depth of field and three controlled thickness of glass. We also capture 100 wild scenes with different camera settings, uncontrolled illuminations and thickness of glass. Then, we conduct quantitative evaluations for state-of-the-art single-image reflection removal algorithms [1, 2, 9, 11] using four different error metrics. At last, we analyze the pros and cons per method and per error metric and the consistencies between quantitative results and visual qualities. Our SIR^2 dataset and benchmark are available from the following link: <https://sir2data.github.io/>

An overview of the scenes in SIR^2 dataset is in Figure 3.1 and our dataset has four major characteristics:

With ground truth provided: We treat a triplet of images as one set, which contains the mixture image, and the ground truth of background and reflection.

Diverse scenes: We create three sub-datasets: The first one contains 20 controlled indoor scenes composed by solid objects; the second one uses postcards to compose another set of 20 different controlled scenes; and the third one contains 100 different wild scenes.

Figure 3.2: An example of ‘F-variance’ (varying aperture size) and ‘T-variance’ (varying glass thickness) in the controlled scene.

Varying settings for each controlled scene: For each triplet in the controlled scene dataset, we take images with 7 different DoFs (by changing the aperture size and exposure time) plus 3 different thicknesses of glass.

Large size: In total, our dataset contains $(20 + 20) \times (7 + 3) \times 3 + 100 \times 3 = 1,500$ images.

The rest of this chapter is organized as follows. We introduce our dataset and the capturing setup in Section 3.2. Experimental results and discussions are presented in Section 3.3. Finally, we conclude this chapter in Section 3.4.

3.2 Data capture

All images in our dataset are captured using a Nikon D5300 camera with a 300 mm lens. All images have a resolution 1726×1234 . The camera is set to work at fully manual mode. As shown in Figure 3.3, we use three steps to capture a triplet of images: 1) The mixture image is first captured through the glass; 2) we capture the ground truth of the reflection **R** with a sheet of black paper behind the glass; and 3) finally the ground truth of the background **B** is taken by removing the glass.

Controlled scenes. The controlled scene is composed by a set of solid objects, which uses commonly available daily-life objects (*e.g.* ceramic mugs, plush toys, fruits, *etc.*)

Figure 3.3: Data capture setup and procedures. The top and bottom row show procedure to take the solid object and postcard dataset, respectively. From left to right: The mixture image **I** is taken with the glass; the ground truth of reflection **R** is captured by placing a black sheet of paper behind the glass; the ground truth of background **B** is captured by removing the glass.

for both the background and the reflected scenes. The other scene uses five different postcards and combines them in a pair-wise manner by using each card as background and reflection, respectively (thus we obtain $2 \times C_5^2 = 20$ scenes). We intentionally make the postcard scene more challenging by 1) using postcards with complex textures for both background and reflection and 2) placing a LED desktop lamp near to the objects in front of the glass to make the reflection interference much stronger than under the illumination used in the solid object scenes.

As discussed in Chapter 1 and Chapter 2, the distance between the camera, glass and objects affect the appearance of the captured image: Objects within the DoFs look sharp and vice versa; the glass with different thickness also affects the image appearance by shifting the reflections to a different position. We take the two factors into consideration when building the controlled scene dataset by changing the aperture size and the glass thickness. 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 times $\{1/3\text{ s}, 1/2\text{ s}, 1/1.5\text{ s}, 1\text{ s}, 1.5\text{ s}, 2\text{ s}, 3\text{ s}\}$ corresponding to the seven aperture settings to make the brightness of each picture approximately constant. We denote such variation as ‘F-variance’ for short thereafter, and keep using the same glass with a thickness of 5 mm when varying DoF. To explore how different thickness of glass affects the effectiveness of

existing methods, we place three different glass with 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. As shown in Figure 3.2, for the ‘F-variance’, the reflections taken under F32 are the sharpest, and the reflections taken under F11 have the greatest blur. For the ‘T-variance’, the reflections taken with T10 and T3 shows largest and smallest spatial shift, respectively.

Wild scenes. The controlled scene dataset is purposely designed to include the common priors with varying parameters for a thorough evaluation of state-of-the-art methods (*e.g.* [1, 2, 9, 11]). But real scenarios have much more complicated environments: Most of the objects in the controlled scenes dataset are diffuse, but objects with complex reflectance properties are quite common; real scene contains various depth and distance variation at multiple scales, while the controlled scene contains only flat objects (post-card) or objects with similar scales (solid objects); natural environment illumination also varies greatly, while the controlled scenes are mostly captured in an indoor office environment. To address these limitations of the controlled scene dataset, we bring our setup out of the lab to capture a wild scene dataset with real-world objects of complex reflectance (car, tree leaves, glass windows, *etc.*), various distances and scales (residential halls, gardens, and lecture rooms, *etc.*), and different illuminations (direct sunlight, cloudy sky light and twilight, *etc.*). It is obvious that night scene (or scene with dark background) contains much stronger reflections. So we roughly divide the wild scene dataset into bright and dark scenes since they bring different levels of difficulty to the reflection removal algorithms with each set containing 50 scenes, respectively.

3.2.1 Image alignment

The pixel-wise alignment between the mixture image and the ground truth is necessary to accurately perform quantitative evaluation. During the data capture, we have tried our

Figure 3.4: Image alignment of our dataset. The first row and second row are the images before and after registration, respectively.

best to get avoid of the object and camera motions, by placing the objects on a solid platform and controlling the camera with a remote computer. However, due to the refractive effect of the glass, the spatial shifts still exist between the ground truth of background taken without the glass and the mixture image taken with the glass, especially when the

Table 3.1: Benchmark results using controlled scene dataset for four single-image reflection removal algorithms using four error metrics with F-variance and T-variance. The bold numbers indicate the best result among the four methods.

	F-var.	sLMSE			NCC			SSIM			SI		
		F11	F19	F32	F11	F19	F32	F11	F19	F32	F11	F19	F32
Postcard	AY07	0.959	0.949	0.955	0.892	0.888	0.888	0.854	0.840	0.831	0.877	0.867	0.854
	LB14	0.886	0.900	0.892	0.934	0.930	0.927	0.841	0.826	0.807	0.937	0.919	0.895
	SK15	0.898	0.895	0.900	0.807	0.813	0.809	0.824	0.818	0.789	0.855	0.850	0.819
	WS16	0.968	0.965	0.966	0.938	0.936	0.931	0.888	0.878	0.862	0.908	0.898	0.880
Solid object	AY07	0.969	0.983	0.940	0.985	0.984	0.983	0.868	0.865	0.860	0.934	0.920	0.917
	LB14	0.841	0.848	0.853	0.977	0.979	0.978	0.821	0.825	0.836	0.969	0.967	0.962
	SK15	0.947	0.945	0.950	0.933	0.941	0.937	0.831	0.819	0.808	0.916	0.913	0.912
	WS16	0.966	0.967	0.965	0.976	0.978	0.977	0.879	0.876	0.875	0.947	0.945	0.943
	T-var.	sLMSE			NCC			SSIM			SI		
		T3	T5	T10	T3	T5	T10	T3	T5	T10	T3	T5	T10
Postcard	AY07	0.845	0.844	0.843	0.895	0.894	0.901	0.834	0.834	0.846	0.854	0.856	0.867
	LB14	0.842	0.847	0.840	0.930	0.934	0.930	0.809	0.810	0.808	0.901	0.904	0.903
	SK15	0.951	0.950	0.947	0.820	0.822	0.824	0.800	0.800	0.810	0.830	0.830	0.840
	WS16	0.919	0.918	0.915	0.934	0.935	0.933	0.884	0.882	0.889	0.835	0.833	0.840
Solid object	AY07	0.971	0.974	0.946	0.982	0.984	0.985	0.929	0.933	0.932	0.929	0.933	0.932
	LB14	0.852	0.854	0.852	0.977	0.978	0.977	0.977	0.978	0.977	0.977	0.978	0.977
	SK15	0.949	0.951	0.954	0.934	0.939	0.942	0.911	0.914	0.913	0.911	0.914	0.913
	WS16	0.966	0.967	0.926	0.974	0.977	0.975	0.939	0.943	0.941	0.939	0.943	0.941

glass is thick.

Some existing methods [36, 54] ignore such spatial shift when they perform the quantitative evaluation. But as a benchmark dataset, we need highly accurate alignment. Though the refractive effect introduces complicated motion, we find that a global projective transformation works well for our problem. We first extract SURF feature points [55] from two images, and then estimate the homographic transformation matrix by using the RANSAC algorithm. Finally, the mixture image is aligned to the ground truth of background with the estimated transformation. Figure 3.4 shows an example of a image pair before and after alignment.

3.3 Evaluation

In this section, we use the SIR² dataset to evaluate representative single-image reflection removal algorithms, AY07 [9], LB14 [1], SK15 [2], and WS16 [11], for both quantitative accuracy (w.r.t. our ground truth) and visual quality. We choose these four methods, because they are recent methods belonging to different types according to Chapter 1 with state-of-the-art performance.

For each evaluated method, we use default parameters suggested in their papers or used in their original codes. AY07 [9] requires the user labels of background and reflection edges, so we follow their guidance to do the annotation manually. SK15 [2] requires a pre-defined threshold (set as 70 in their code) to choose some local maxima values. However, such a default threshold shows degenerated results on our dataset, and we manually adjust this threshold for different images to make sure that a similar number of local maxima values to their original demo are generated. To make the image size compatible to all evaluated algorithms, we resize all images to 400×540 .

Table 3.2: Benchmark results for four single-image reflection removal algorithms for bright and dark scenes in the wild scene dataset. The bold numbers indicate the best result.

Bright scenes		sLMSE	NCC	SSIM	SI	Dark scenes		sLMSE	NCC	SSIM	SI
	AY07	0.987	0.959	0.897	0.908		AY07	0.776	0.823	0.795	0.883
	LB14	0.930	0.930	0.866	0.943		LB14	0.751	0.783	0.741	0.897
	SK15	0.951	0.824	0.836	0.873		SK15	0.718	0.752	0.777	0.875
	WS16	0.936	0.982	0.926	0.939		WS16	0.708	0.790	0.803	0.881

3.3.1 Quantitative evaluation

Quantitative evaluation is performed by checking the difference between the ground truth of background $\mathbf{B}$ and the estimated background $\mathbf{B}^*$ from each method.

Error metrics. The most straightforward way to compare two images is to calculate their pixel-wise difference using PSNR or MSE (*e.g.*, in [2]). However, absolute difference such as MSE and PSNR is too strict since a single incorrectly classified edge can often dominant the error value. Therefore, we adopt the local MSE (LMSE) [52] as our first metric: It evaluates the local structure similarity by calculating the similarity of each local patch. To make the monotonicity consistent with other error metric we use, we transform it into a similarity measure:

$$sLMSE(\mathbf{B}, \mathbf{B}^*) = 1 - LMSE(\mathbf{B}, \mathbf{B}^*) \quad (3.1)$$

The $\mathbf{B}$ and $\mathbf{B}^*$ sometimes have different overall intensity, which can be compensated by subtracting their mean values, and the normalized cross correlation (NCC) is such an error metric(*e.g.*, in [12]).

We also adopt the perceptually-motivated measure SSIM [56] which evaluates the similarity of two images from the luminance, contrast, and structure as human eyes do.

The luminance and contrast similarity in the original SSIM definition are sensitive to the intensity variance, so we also use the error metric proposed in [57] by focusing

only on the structural similarity between $\mathbf{B}$ and $\mathbf{B}^*$:

$$\text{SI} = \frac{2\sigma_{\mathbf{B}\mathbf{B}^*} + c}{\sigma_{\mathbf{B}}^2 + \sigma_{\mathbf{B}^*}^2 + c}, \quad (3.2)$$

where $\sigma_{\mathbf{B}}$, $\sigma_{\mathbf{B}^*}$ are the variance of $\mathbf{B}$ and $\mathbf{B}^*$, $\sigma_{\mathbf{B}\mathbf{B}^*}$ is the corresponding covariance, and c is a constant. We call this error metric structure index (SI).

Results. We evaluate the performance of these algorithms using the four error metrics above and show their quantitative performances in Table 4.2 for the controlled scene datasets and Table 4.3 for the wild scene dataset. In Table 4.2, the performances on the solid object dataset are better than those on the other two datasets. This is within our expectation, since the postcard dataset is purposely made more challenging and the wild dataset contains many complex scenes. For AY07 [9], it is hard to tell how F- and T-variance influence its result, because the manual annotated edges are not affected by such different settings. LB14 [1] and WS16 [11] show clear decreasing tendency with F-variance for SSIM and SI, because the reflections with more blur (F11 compared to F32) make it easier for these two methods to classify the reflection edges more accurately thus results in higher-quality reflection removal. LB14 [1] shows the most advantage for NCC and SI, but it does not perform well for sLMSE and SSIM. We notice the results from LB14 [1] are visually darker than the ground truth, so error metric with intensity normalization like NCC and SI reflect their performances more properly. SK15 [2] shows better results for T10 than T3 for most error metrics, because the thicker glass usually shows two overlaid reflections more clearly hence easier for kernel estimation. Though SK15 [2] has relative lower scores in Table 4.2, it does not necessarily mean its performance is worse and we discuss its visual quality advantage in Section 3.3.2.

The wild scene dataset introduces various challenging to methods performing well on the controlled scene datasets, since the priors they rely on may be poorly observed in some of the wild scenes. For example, the different depths and scales of objects cause

Figure 3.5: Examples of visual quality comparison. The top two rows are the results for images taken with F11/T5 and F32/T5, and bottom two rows use images taken with F32/T3 and F32/T10.

unevenly blur levels to degrade the performances of LB14 [1] and WS16 [11]. Some repetitive patterns (*e.g.*, windows on the buildings) also make it difficult for the kernel estimation of SK15 [2]. In general, the performances of the bright scene are much better than those of the dark scene, which indicates that strong reflection on dark background is still challenging for all methods. It is interesting that AY07 [9] performs best among all methods, which means manual labeling with labor and time cost helps indicating useful edges more effectively.

3.3.2 Visual quality evaluation

We show two examples of visual quality comparison of the evaluated algorithms in Figure 3.5 for the controlled scene dataset and Figure 3.6 for the wild scene dataset. In Figure 3.5, through a close check between the estimated results from all images in the controlled scene dataset and the corresponding values from all error metrics, we find SI shows the best consistency with visual quality. The top two rows F11/T5 and F32/T5 show that LB14 [1] and WS16 [11] work more effectively for larger out of focus blur. The last two rows F32/T3 and F32/T10 show SK15 [2] produces cleaner separation with fewer high-frequency artifacts. The edge-based methods like AY07 [9] and WS16 [11] shows better local accuracy, but visible residue edges are more often observed in their results than in SK15 [2].

The examples in the first row of Figure 3.6 show that all methods can successfully remove a certain amount of reflections. However, when the objects behind the glass have uneven blur levels due to the different depths, LB14 [1] wrongly removes the blurred object behind the glass (the grass in the green rectangle). In the second and third row, where the reflection is much stronger, the performance are all degraded. They show over-smoothed results with obvious remaining of the reflection. Only when manual labelings are carefully applied, these artifacts (*e.g.*, the ceiling light in the third example) can be largely suppressed.

Figure 3.6: Examples of visual quality comparison using the wild scene dataset. The first row shows the results using images from bright scenes and the last two rows are the results using images from the dark scenes.

3.3.3 Open problems

From the quantitative evaluations and the visual quality of the four algorithms here, we think the single-image reflection removal algorithm still has great space to be improved. Almost no methods are able to completely remove reflections, and various artifacts are visible in most of the results. From the evaluations above, a straightforward improvement might be achieved by complementing the merits of edge-based methods (Type-I and Type-II of Section 2.1) for achieving higher local accuracy and kernel-based methods (Type-III) for suppressing the edge artifacts. Besides, we summarize two factors that are not well addressed in the four evaluated methods (also for many similar methods mentioned in Chapter 1), and hope to inspire solutions for more advanced methods in future research:

Background vs. reflection: The evaluated methods generally fail on the wild scenes due to that they focus on special properties of reflections for its removal while ignoring the properties of the background. A widely observed prior suitable for the reflection removal may not be suitable for the recovery of the background layer. Future methods may avoid the strong dependence on priors for reflection, which may overly remove information of the background.

Local vs. global: We find that in our dataset, many reflections only occupy a part of the whole images. However, most existing methods still process every part of an image, which downgrades the quality of the regions without reflections. Local reflection regions can only be roughly detected through manually labelling (AY07 [9]). Though our method in Chapter 5 can detect the reflection dominant regions, it still depends on some heuristic observations. Methods that better automatically detect and process the reflection regions may have potential to improve the overall quality.

Partial vs. whole: The evaluated methods in this chapter all belong to the non-learning based methods. Though these methods can solve the problems in some specific scenarios, they are all likely to have poor performances when the scenarios become

different. The reason for this phenomenon is mainly due to their limited description capability to the properties of real-world reflections and project the partial observations as the whole truth. To solve this problem, learning-based methods that can better describe the whole properties of reflections become necessary.

3.4 Conclusion

In this chapter, we introduce SIR² — the first benchmark real image dataset for quantitatively evaluating single-image reflection removal algorithms. Our dataset consists of various scenes with different capturing settings. We evaluated state-of-the-art single-image algorithms using different error metrics and compared their visual quality.

On the other hand, we also analyze the limitations of existing methods based on the experimental results SIR²dataset. In the following chapters, we will propose different methods to solve the open problems raised in Section 3.3.3.

Chapter 4

Sparsity based Reflection Removal using External Patch Search

In this chapter, we propose a reflection removal method benefited from the sparsity and nonlocal image prior. Based on the analysis in Chapter 3, the existing methods mainly focus on special properties of reflections (*e.g.*, the ghosting effects and the blurring effects) while ignoring the properties of the background. However, as we discussed before, a widely observed prior suitable for the reflections removal may not be suitable for the background recovery. On the other hand, even for these special properties used for reflection removal, they all highly rely on very limited scene priors, which are fragile when these special properties are not observed. To solve these problems, our method utilizes the nonlocal image prior to recover the background directly and leverage the nonlocal information from an external database to overcome the limited prior information in the input mixture image. The experimental results show that our proposed model performs better than the existing state-of-the-art reflection removal method for both objective and subjective image qualities.

4.1 Introduction

Though the method proposed in Chapter 2 provides a simple and effective way to remove reflections, it still highly relies on scene priors such as the different sparse gradients brought by distinguishable blur levels between the background and reflections to solve this problem. As discussed in Chapter 2 and Chapter 3, this kind of methods cannot deal with complicated situations where the background and reflection have similar blur levels.

In this chapter, to release the requirement for the distinguishable blur levels between the background and reflections, we propose a novel reflection removal approach by combining the sparsity prior and the nonlocal image prior into a unified framework. The nonlocal image prior mainly makes use of the patch recurrence properties among the image, which is widely adopted in the patch-based image denoising [58] or super-resolution [59] methods to enhance a noisy or blurred patch from the input image by reconstructing this patch with a set of similar ‘clean’ patches. The key assumption of this work is that a set of clean images that share similar contents with the background layer of the input mixture image can be retrieved from an external database and the similar patches can be extracted from the clean images. By using the non-local information from these similar clean patches, we can recover the background information in the input mixture images by regularizing its corresponding sparse codes. Compared with previous methods [1, 2] and the method proposed in Chapter 2, our method does not require special phenomena (such as different levels of blur or ghosting effect) have to be observed on the mixture image so that we can better handle the images with general and complex structures.

The rest of this chapter is organized as follows. In Section 4.2, we briefly review the definitions of sparse representation and nonlocal image priors and their applications in different problems. Then, we introduce our method and its corresponding optimization solutions in Section 4.3. Experimental results and discussions are presented in Sec-

tion 4.4. Finally, we conclude this chapter in Section 4.5.

4.2 Sparse Representation in Image Restoration

In this section, we briefly introduce the definitions of sparse representation and its applications on some related topics (*e.g.*, image denoising and super resolution).

4.2.1 Sparse representation

The general sparse representation method aims at solving a linear representation system by describing signals as a combination of a few atoms from a pre-specified dictionary. Formally, given a dictionary $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n] \in \mathcal{R}^{m \times n}$ and a signal $\mathbf{x} \in \mathcal{R}^m$ where typically $m \leq n$, the sparse representation or sparse approximation $\hat{\alpha}$ for $\mathbf{x}$ can be recovered by:

$$\begin{aligned} & \underset{\alpha}{\operatorname{argmin}} \|\alpha\|_0 \\ & s.t. \|\mathbf{x} - \mathbf{D}\alpha\|_2^2 \leq \epsilon, \end{aligned} \quad (4.1)$$

where $\|\cdot\|_0$ counts the number of non-zero elements in the vector. The model tries to seek the most compact representation for the signal $\mathbf{x}$ given the dictionary $\mathbf{D}$, which can be orthogonal basis ($m = n$), over-complete basis ($m < n$) or dictionary learned from the training data. For orthonormal basis, solution to Equation (4.1) is merely the inner products of the signal with the basis. However, since the optimization for Equation (4.1) is combinatorially NP-hard for general dictionary, the l_0 -norm in Equation (4.1) is always replaced by l_1 norm due to the equality between the solutions obtained by using the two norms. Then, by using the Lagrange multiplier, Equation (4.1) can be relaxed to the l_1 -problem as

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{D}\alpha - \mathbf{x}\|_2^2 + \lambda \|\alpha\|_1. \quad (4.2)$$

Sparsity plays an important or even crucial role in many fields, such as image restoration, compressive sensing, and recognition. In the following, we will make a brief discussion on the role of sparsity in the application of image restoration problems.

4.2.2 Sparsity in Image Restoration

When take a close inspection of the progress made in the field of image processing in the past decades, we can find that much of the progress can be attributed to better modeling of the image content. As the most widely used prior in recent years, sparsity has shown promising results in image restoration including image denoising [60], inpainting [61], super-resolution [62], and deblurring [20]. Among these, we specifically focus the discussion on image denoising due to the similarity between the mathematical formulation of image denoising and reflection removal. Consider a noisy image corrupted by additive noise as follows:

$$\mathbf{y} = \mathbf{x} + \mathbf{n}, \quad (4.3)$$

where $\mathbf{y}$ is the noisy observation, $\mathbf{x}$ is the latent clean image, and $\mathbf{n}$ is the Gaussian noise. The problem of image denoising is to estimate the latent clean image $\mathbf{x}$ without the noise from a noisy observation $\mathbf{y}$. Similar to the reflection removal problem, with more unknowns than knowns, this is also a typical ill-posed inverse problem, thus requiring regularization to stabilize the final solutions.

To solve this problem, Elad *et al.* [60] has shown that a local dictionary can be learned from the noisy image itself to estimate the latent clean image. Since it is difficult for the model in Equation (4.1) to process large images, they use small overlapped patches instead of the whole images to learn such dictionary and reconstruct the whole image. The whole process can be expressed as follows:

$$\underset{\mathbf{D}, \boldsymbol{\alpha}_i, \mathbf{x}}{\operatorname{argmin}} \lambda \|\mathbf{y} - \mathbf{x}\|_2^2 + \sum_i \|\mathbf{D}\boldsymbol{\alpha}_i - \mathbf{M}_i\mathbf{x}\|_2^2 + \sum_{\mathbf{y}} \mu_i \|\boldsymbol{\alpha}_i\|_1, \quad (4.4)$$

where $\mathbf{D}$ is the dictionary with normalized columns, α_i is the corresponding sparse coefficients, $\mathbf{M}_i\mathbf{x}$ is the i th patch extracted from $\mathbf{x}$ and $\mathbf{M}_i$ is binary matrix to extract specific patches, and λ and μ control the noise power and sparsity degree, respectively. The first term in Equation (4.4) demands the similarity between the noised observation $\mathbf{y}$ and its denoised version $\mathbf{x}$. The second term in Equation (4.4) means that each patch from the reconstructed image denoted by $\mathbf{M}_i\mathbf{x}$ can be represented by its corresponding dictionary $\mathbf{D}$ and sparse coefficients α . The final term requires that the number of coefficients used to represent any patches should be as small as possible.

In the algorithm proposed by Elad *et al.* [60], $\mathbf{x}$ and $\mathbf{D}$ are respectively initialized with $\mathbf{y}$ and overcomplete discrete cosine transform (DCT) dictionary, respectively. The minimization of Equation (4.4) starts with extracting and rearranging all the patches of $\mathbf{x}$. The patches are then processed by K-SVD, which estimates sparse coefficients α_i by assuming other parameters are fixed as follows:

$$\alpha_i = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{D}\alpha_i\|_2^2 + \lambda\|\alpha_i\|_1. \quad (4.5)$$

Then, by assuming $\mathbf{D}$ and α_i are fixed, $\mathbf{x}$ is computed as follows

$$\hat{\mathbf{x}} = (\lambda\mathbf{E} + \sum_i \mathbf{M}_i^\top \mathbf{M}_i)^{-1}(\lambda\mathbf{y} + \sum_i \mathbf{M}_i^\top \mathbf{D}\alpha_i), \quad (4.6)$$

where $\mathbf{E}$ is the identity matrix and $\hat{\mathbf{x}}$ is the estimated results without the noise. In this whole process, $\mathbf{D}$ and α_i are updated by using K-SVD. Such conjoined denoising and dictionary adaption is repeated to minimize Equation (4.4). Since $\sum_i \mathbf{M}_i^\top \mathbf{M}_i$ is diagonal, the above expression in Equation (4.6) can be easily calculated in a pixel-wise manner.

4.2.3 NCSR model

Though Equation (4.5) and Equation (4.6) provide a simple way to compute the sparse coefficients α and its corresponding denoised image $\hat{x}$, it may not lead to an accurate enough image reconstruction results by using only the local sparsity constraint $\|\alpha_i\|_1$ without any external prior information. To stabilize the final solutions, Dong *et al.* [58] propose the nonlocally centralized sparse representation (NCSR) model by using the nonlocal correlations existed in the images. It is based on the sparse coding noise (SCN) assumption formulated as follows:

$$v_\alpha = \alpha_y - \alpha_x, \quad (4.7)$$

where α_y is the sparse coefficients obtained by solving Equation (4.5) and α_x is the true sparse coefficients of the original image without noise. From the formulation in Equation (4.7), the sparse coding noise is defined as the difference between α_y and α_x . To better estimate the latent clean image from the noisy observation, the estimated sparse coefficients α_y should be as close as to the true sparse coefficients α_x . Thus, the definition of SCN in Equation (4.7) indicates that the image restoration quality can be improved by suppressing the SCN v_α .

However, due to the difficulties to obtain the true sparse coefficients α_x in most situations, v_α cannot be directly measured. To solve this problem, previous methods [58] mainly solve this problem by using some good estimations of α_x to approximate the true sparse coefficients. When denoting the estimation of α_x as β , Equation (4.7) can be re-expressed as follows:

$$\hat{v}_\alpha = \alpha_y - \beta, \quad (4.8)$$

where $\hat{v}_\alpha$ can be regarded as a good estimation of v_α . Then, Equation (4.5) can be

reformulated as follows:

$$\boldsymbol{\alpha}_i = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1 + \gamma \sum_i \|\boldsymbol{\alpha}_i - \boldsymbol{\beta}_i\|_2, \quad (4.9)$$

where γ is the regularization parameter.

4.3 Proposed Method

Compared with the image denoising mechanism discussed above, reflection removal is a more challenging problem since more than one components are to be estimated from a single input observation. Consider this more challenging situation, we adapt the NCSR model proposed by Dong *et al.* [58] to our problem. In this section, we first present the reflection removal model used in our method and then introduce its corresponding optimization solutions.

4.3.1 The reflection removal model

Let $\mathbf{I}$, $\mathbf{B}$, and $\mathbf{R}$ represent the input mixture image, background, and reflection, respectively. In this work, based on the model proposed in Equation (1.1), we define a new energy function by reformulating Equation (4.4) to model the reflection removal problem as follows:

$$L(\mathbf{B}, \mathbf{R}) = \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \lambda\rho(\mathbf{B}) + \gamma\varrho(\mathbf{R}), \quad (4.10)$$

where $\rho(\mathbf{B})$ and $\varrho(\mathbf{R})$ are the regularization prior terms on the background and reflection layer, respectively. Many previous methods can also be cast into such a framework by using different regularization terms. For example, ρ and ϱ are chosen to be the GMM priors to model the ghosting effects in the method [2]. The morphology separation problems [63, 64] choose the sparsity-based prior as the regularization term. In our proposed model, to stabilize the final estimation results, we adopt the integration of the

Figure 4.1: The framework of our method. Our algorithm runs on RGB channel independently. For simplicity, we only show the process on R channel as an example. We first retrieve similar images from an external database (Step 1); the retrieved images are then registered to the input images (Step 2); similar patches are extracted from the retrieved images based on the exemplar patches (Step 3). In the learning stage, the PCA sub-dictionary is learned from each cluster (Step 4); then the nonlocal information are used to refine the sparse codes of the exemplar patch (Step 5 and Step 6). At last, with the refined sparse codes and the dictionary, the patches are refined (Step 7) and the reflection is removed (Step 8).

sparsity prior and nonlocal image prior to regularize $\mathbf{B}$ and the gradient sparsity prior to regularize $\mathbf{R}$. Formally, given the mixture image $\mathbf{I}$ and the set of clean images retrieved from a dataset, we want to estimate the background $\mathbf{B}$ and reflection $\mathbf{R}$ by

$$\{\hat{\mathbf{B}}, \hat{\mathbf{R}}\} = \underset{\mathbf{B}, \mathbf{R}}{\operatorname{argmin}} L(\mathbf{B}, \mathbf{R}), \quad (4.11)$$

where

$$\begin{aligned} L(\mathbf{B}, \mathbf{R}) = & \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \lambda \sum_i \|\mathbf{M}_i \mathbf{B} - \mathbf{D} \boldsymbol{\alpha}_i\|_2^2 \\ & + \eta \sum_i \|\boldsymbol{\alpha}_i - \boldsymbol{\beta}_i\|_1 + \gamma \sum_{l=1}^L |f_l * \mathbf{R}|^s. \end{aligned} \quad (4.12)$$

We explain each term of the model in detail as follows:

- (i) The first term is the conventional constraint, which means that the mixture image $\mathbf{I}$ should be the summation of estimated background $\mathbf{B}$ and estimated reflection $\mathbf{R}$.
- (ii) The second term means that the estimated background $\mathbf{B}$ can be well represented with respect to its corresponding dictionaries $\mathbf{D}$. $\mathbf{M}_i$ denotes the matrix extracting image patch of size $\sqrt{n} \times \sqrt{n}$. $\mathbf{D}$ denotes the dictionary. $\boldsymbol{\alpha}_i$ is the coefficients corresponding to the dictionary $\mathbf{D}$.
- (iii) The third term is the NCSR model proposed in [58] which enforces that $\boldsymbol{\alpha}_i$ should be as similar as $\boldsymbol{\beta}_i$, where $\boldsymbol{\beta}_i$ is some good estimation of $\boldsymbol{\alpha}_i$.
- (iv) The fourth term is a heavy tailed distribution enforced on the estimated reflection $\mathbf{R}$ to further stabilize the solution, which is widely used in previous methods [9, 51]. Typically, the value of s is set between 0.5 to 0.8. f_l is the Laplacian filters, namely $f_1 = [-1, 1]$, $f_2 = [-1, 1]^\top$, $f_3 = [0, 1, 0; 1, -4, 1; 0, 1, 0]$ as the settings in [1].

Algorithm 2 Sparsity prior based reflection removal**Require:**1: Input mixture image $\mathbf{I}$;**Ensure:**2: Estimated background $\mathbf{B}$ and reflection $\mathbf{R}$;3: Compute the dictionaries $\mathbf{D}$ by k-means and PCA;4: **for** $m = 1$ to M **do**5: **for** $j = 1$ to J **do**6: Update sparse codes α_i^{j+1} by solving Equation (5.18);7: Update the background $\mathbf{B}^{j+1}$ by solving Equation (4.18);8: Update the reflection $\mathbf{R}^{j+1}$ by solving Equation (4.19);9: Set $\mathbf{B}^{m+1} = \mathbf{B}^{j+1}$ and $\mathbf{R}^{m+1} = \mathbf{R}^{j+1}$ if $j = J_{max}$ 10: **end for**11: If $\text{mod}(m, 5) = 0$, update the PCA dictionaries;12: **end for**13: **return** $\mathbf{B}^{m+1}, \mathbf{R}^{m+1}$;**4.3.2 The selection of the dictionary $\mathbf{D}$**

One important issue of sparsity-based method is the selection of dictionary $\mathbf{D}$. Conventional analytically designed dictionaries, such as DCT, wavelet and curvelet dictionaries, are insufficient to characterize the so many complex structures of natural images. The universal dictionaries learned from example image patches by using algorithms such as K-SVD can better adapt to local image structures and also shows very promising results in different applications [60]. In general the learned dictionaries are required to be very redundant such that they can represent various image local structures. However, it has been shown that sparse coding with an overcomplete dictionary is unstable [65], especially in the scenario of image restoration. To make the final estimations more stable, we adopt the PCA dictionary similar to previous work [66]. Different from the K-SVD dictionary, PCA dictionary is built by clustering the training patches extracted from a set of example images into K clusters, and learn a PCA sub-dictionary for each cluster. Then for a given patch, one compact PCA sub-dictionary is adaptively selected to code it, leading to a more stable and sparser representation, and consequently better image restoration results. In this work, we adopt this adaptive sparse domain selection strategy

but learn the sub-dictionaries from the given image itself instead of the example images.

Then as shown in Figure 4.1, the image patches are extracted from image I and clustered into K clusters (typically $K = 70$) by using the K-means clustering method. Since the patches in a cluster are similar to each other, there is no need to learn an over-complete dictionary for each cluster. Therefore, for each cluster we learn a dictionary of PCA bases and use this compact PCA dictionary to code the patches in this cluster. (For the details of PCA dictionary learning, please refer to [58].) These K PCA sub-dictionaries construct a large over-complete dictionary to characterize all the possible local structures of natural images.

4.3.3 The estimation of β_i

As an estimation of α_i , β_i can be estimated from the internal or external sources. In our case, we estimate β_i from an external sources where a set of similar clean images can be found for the input mixture images.

Our framework of similar patch matching process contains three steps, which are similar image retrieval, global image registration and patch match. We adopt the image retrieval method proposed by Philbin *et al.* [67] and retrieve images from an external database. As shown in Figure 4.1, our external database contains different images of landmarks and well-known objects obtained from Internet. Due to the different scales and viewpoints of these retrieved images, for better patch matching, an image registration step is needed. We use a quite standard way to register the images. We first extract SURF feature points from the mixture image and reference images, and then estimate the homographic transformation matrix by using the RANSAC algorithm. Finally, as shown in Figure 4.1, the reference images from the external database are aligned to the mixture image with the estimated transformation.

Let x_i denote the patch from the input mixture image. The nonlocal similar patches z_i that are within the first T closet to the given patch x_i are selected from a large window

centered at pixel i among the registered images. Then, β_i can be computed as the weighted average of those sparse codes associated with the nonlocal similar patches as:

$$\beta_i = \sum_{t=1}^T \omega_{i,t} \alpha_{i,t}, \quad (4.13)$$

where $\alpha_{i,t}$ is the sparse coefficients corresponding to the patch z_i and $\omega_{i,t}$ is the weight and can be obtained as:

$$\omega_{i,t} = \frac{1}{W} \exp(-\|\hat{\mathbf{x}}_i - \hat{\mathbf{z}}_{i,t}\|_2^2/h), \quad (4.14)$$

where $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{z}}_{i,t}$ are the estimates of the patches $\mathbf{x}_i$ and $\mathbf{z}_{i,t}$, h is a pre-determined scalar and W is the normalization factor.

4.3.4 Optimization

The direct minimization of Equation (4.12) is difficult due to the multiple variables involved in the proposed model. Thus, we reduce the original problem into several sub-problems by following the alternating minimization scheme advocated by the previous method in image deblurring and denoising works. In each step, our algorithm reduces the objective function value, and thus will converge to a local minima.

Solving for α_i . For a fixed $\mathbf{B}$ and $\mathbf{R}$, Equation (4.12) reduces to a l_1 minimization problem:

$$\hat{\alpha}_i = \underset{\alpha_i}{\operatorname{argmin}} \lambda \|\mathbf{M}_i \mathbf{B} - \mathbf{D} \alpha_i\|_2^2 + \eta \|\alpha_i - \beta_i\|_1. \quad (4.15)$$

With fixed β_i , Equation (5.18) can be solved iteratively by the surrogate based algorithm [68]:

$$\alpha_i^{(t+1)} = \mathcal{S}_\tau(v_i^{(t)} - \beta_i) + \beta_i, \quad (4.16)$$

where $v_i^{(t)} = \mathbf{D}^\top (\mathbf{M}_i \mathbf{B} - \mathbf{D} \boldsymbol{\alpha}_i^{(t)})/c + \boldsymbol{\alpha}_i^{(t)}$, $\mathcal{S}_\tau(\cdot)$ represents the soft-thresholding operator with threshold $\tau = \eta/\lambda c$, and c is a constant to guarantee the convexity. Due to the orthogonal properties of the local PCA dictionaries $\mathbf{D}$, the sparse coding problem of Equation (5.18) can be solved in just one step [59].

Solving for $\mathbf{B}$. When $\mathbf{R}$ and $\boldsymbol{\alpha}_i$ are fixed, the background $\mathbf{B}$ can be estimated by solving the following optimization problem:

$$\hat{\mathbf{B}} = \underset{\mathbf{B}}{\operatorname{argmin}} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \lambda \sum_i \|\mathbf{M}_i \mathbf{B} - \mathbf{D} \boldsymbol{\alpha}_i\|_2^2, \quad (4.17)$$

where the closed-form solution can be easily obtained as follows:

$$\hat{\mathbf{B}} = (\mathbf{E} + \lambda \sum_i \mathbf{M}_i^\top \mathbf{M}_i)^{-1} (\mathbf{I} + \lambda \sum_i \mathbf{M}_i^\top \mathbf{D} \boldsymbol{\alpha}_i - \mathbf{R}), \quad (4.18)$$

where all elements of matrix $\mathbf{E}$ equal to one.

Solving for $\mathbf{R}$. Given the estimated background $\mathbf{B}$ and sparse representation $\boldsymbol{\alpha}$, the estimation of reflection $\mathbf{R}$ can be updated. The optimization problem (5) becomes

$$\hat{\mathbf{R}} = \underset{\mathbf{R}}{\operatorname{argmin}} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \gamma \sum_{l=1}^L |f_l * \mathbf{R}|^s. \quad (4.19)$$

This problem can be solved efficiently by variable substitution and Fast Fourier Transform (FFT) [69, 70]. Using the new auxiliary variables u_l ($l \in 1, 2, \dots, L$), the Equation (4.19) can be rewritten as:

$$\begin{aligned} \hat{\mathbf{R}} = \underset{\mathbf{R}}{\operatorname{argmin}} & \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \gamma \sum_{l=1}^L |u_l|^s \\ & + \delta \sum_{l=1}^L \|u_l - f_l * \mathbf{R}\|_2^2. \end{aligned} \quad (4.20)$$

It can be divided into two sub-problems: $\mathbf{R}$ -subproblem and u -subproblem. δ is a weight value that varies during the optimization. We follow the setting in [69] to set the value of δ . In the R -subproblem, the Equation (4.20) becomes:

$$\hat{\mathbf{R}} = \underset{\mathbf{R}}{\operatorname{argmin}} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \delta \sum_{l=1}^L \|u_l - f_l * \mathbf{R}\|_2^2, \quad (4.21)$$

which can be solved using FFT as:

$$\hat{\mathbf{R}} = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(\mathbf{I}) + \delta \sum_{l=1}^L \mathcal{F}(e_l)^* \mathcal{F}(u_l) - \mathcal{F}(\mathbf{B})}{\mathbf{E} + \delta \sum_{l=1}^L \mathcal{F}(e_l)^* \mathcal{F}(e_l)} \right), \quad (4.22)$$

where $\mathcal{F}$ denotes FFT, $\mathcal{F}^{-1}$ denotes the inverse FFT and $\star$ is the complex conjugate.

In the u -subproblem, the u_l can be estimated by solving the following equation:

$$\hat{u}_l = \underset{u_l}{\operatorname{argmin}} \gamma \sum_{l=1}^L |u_l|^s + \delta \|u_l - f_l * \mathbf{R}\|_2^2, \quad (4.23)$$

which can be solved efficiently using the method in [69] over each dimension separately.

4.4 Experiments

4.4.1 Data preparation

Since our method in this chapter can only handle images with landmark scenes or famous objects, SIR² dataset in Chapter 3 cannot be used directly in this place. In order to figure out the performance of our results compared with others, we capture the images with the ground truth following a similar way proposed in Chapter 3, where the mixture image is taken through the transparent glass and the ground truth is taken by removing the glass. We prepare two types of data capture setup: one setup uses some landmark postcards as both background and reflection objects; the other setup captures some solid objects (*e.g.*, toys of famous figures) as the background objects.

Figure 4.2: Reflection removal results comparison using our method, LB14 [1], and SK15 [2] on the postcard data. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM values are displayed below the images.

For the external database used in the patch matching stage, we collect approximately 500 images from the Internet, and three images with similar contents (the same landmark or the same toy figure captured in different environment) corresponding to each mixture image are included in the database. We then perform image retrieval [67] to find these three images before the patch matching stage. An example is shown in Figure 6.4, where three images containing the Tower Bridge similar to the input mixture image are

Figure 4.3: Reflection removal results comparison using our method, LB14 [1] and SK15 [2] on the solid object data. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM values are displayed below the images.

retrieved from the external dataset.

4.4.2 Evaluations

We show six example results in Figure 4.2 and Figure 4.3. We compare our method with LB14 [1] and SK15 [2], which all use single image as the input. In all our experiments,

the parameters are fixed as follows: T are set to 7, γ , λ and η is set to 1, 0.5 and 0.85, respectively, M is set to 15 and J is set to 10. The patch size is set to 7×7 . To quantitatively assess the algorithms, the Structural Similarity Index (SSIM) is adopted as the quality measure of the estimated background which is also use by the previous work [1, 2].

Our method shows advantage in all these results over the other two methods in terms of SSIM. Considering the visual quality of three methods, we also provide a more visually pleasing result. Li *et al.*'s method causes some color change so that the estimated background $\mathbf{B}$ are darker than the ground truth. For SK15 [2], the GMM priors bring some patchy artifacts on the estimated background $\mathbf{B}$. Though our method leads to some blurry estimations and over-smooth some highly textured regions due to the limitations from the non-local image priors, it can still generate clearer separation.

4.5 Conclusion

We propose a method to remove reflections based on retrieved external patch by combining the sparsity prior and the nonlocal image prior as a unified optimization. Compared with the previous methods [1, 11], we do not have special requirement for the properties of the background layer and the reflection layer, *e.g.* using different blur levels of the two layers to assist separation. Instead, we refine the sparse coefficients learned from the mixture images with the external patches to generate a more accurate sparse regularization term. Experimental results have already shown that our method outperforms the current state-of-the-art methods both from the quantitative evaluations and visual quality.

Limitations. This method can only handle the landmark scenes or some well-known objects, that can be efficiently retrieved. It is still difficult for our method to deal with the general objects or scenes, for which similar contents cannot be retrieved from the

external database. In the next chapter, we will discuss how to extend this method to more general scenarios.

Chapter 5

Region-Aware Reflection Removal with Unified Content and Gradient Priors

The method proposed in Chapter 4 releases the strict requirements for the differences between the background and the reflections. However, since it needs an external database to retrieve the similar clean patches, it can only handle the landmark scenes or some well-known objects, that can be efficiently retrieved. On the other hand, as we discussed in Chapter 3, many reflections only occupy a part of the whole images. However, most existing methods including our method proposed in Chapter 4 process every part of an image, which downgrades the quality of the regions without reflections. In this chapter, we propose a new region-aware approach based on the method in Chapter 4 to handle the general objects or scenes. Our region-aware method can automatically detect the regions with and without reflections and process them in a heterogeneous manner. On the other hand, instead of the external sources used in previous methods, the new method makes use of the self-similarity existed in the input mixture image itself to obtain the clean patches with similar contents. The experimental results show that our proposed model performs better than the existing state-of-the-art reflection removal method for both objective and subjective image qualities.

Figure 5.1: Examples of real-world mixture images and reflection removal results using LB14 [1], SK15 [2], and our method.

5.1 Introduction

In the real world, when we take photos through the transparent glass, in addition to the reflections, the final captured images always contain additive noises due to the interference from the outside world. Thus, different from previous methods that mainly consider the mixture image as a summation of the background and reflection, we reformulate the mathematical model in Equation (1.1) by taking the additive noise term into considerations as follows:

$$\mathbf{I} = \mathbf{B} + \mathbf{R} + \mathbf{n}, \quad (5.1)$$

where $\mathbf{I}$ is the input mixture image, $\mathbf{B}$ is the background to be recovered, and $\mathbf{R}$ is the reflection to be removed, and $\mathbf{n}$ is the additive noise.

As we have discussed before, different priors are proposed based on Equation (5.1) to make this problem tractable. Some methods makes use of the gradient priors motivated by the fact that natural image gradients have the heavy-tailed distribution. Other methods such as the method proposed in Chapter 4 adopt the content prior based on the patch recurrence properties from an external source to solve this problem. However, no matter what priors are used, existing single-image methods all treat the whole mixture images in a global manner. Since the reflections only occupy a part of the whole image plane like regional ‘noise’, the global mechanism adopted by previous methods has pro-

Figure 5.2: The framework of our method. In the patch matching stage, we obtain reference patches from intermediate results of background in the detected reflection dominant regions using internal patch recurrence; then in the removal stage, the information from reference patches are used to refine the sparse codes of the query patches to generate the content prior. With the content prior and long-tail gradient prior, the background image $\mathbf{B}$ is recovered; based on the short-tail gradient prior, the reflection $\mathbf{R}$ is also estimated.

nounced limitations. For example, as shown in Figure 5.1, either the gradient prior based separation (e.g., LB14 [1]) or the content prior based restoration (e.g., SK15 [2]) shows artifacts in regions with weak reflections, as shown in the example of Figure 5.1. The result of LB14 [1] becomes globally darker than the ground truth background $\mathbf{B}$ and the result of SK15 [2] suffers from the patchy effect where the color becomes non-uniform; both methods are not able to effectively handle locally strong reflections, which results in residue edges on the pillar next to the car.

In this chapter, we propose a Region-aware Reflection Removal (R3) approach to address these limitations. Given regions with and without strong reflections automatically detected, we apply customized strategies to handle them so that the regional part focuses on removing the reflections and the global part keeps the consistency of the color and gradient information. We integrate both the content and gradient priors into a unified framework, with the content priors restoring the missing contents caused by the reflections (regional) and the gradient priors separating the two images (global). As an example, the result of our method shown in Figure 5.1 shows fewer reflection residues and more complete image content than previous methods.

The framework of our method is illustrated in Figure 6.4. Given the input mixture image $\mathbf{I}$, we still consider the reflection removal as image restoration with complementary priors to restore the missing contents, which is similar to the method proposed in Chapter 4 in the patch matching stage, but we utilize the internal patch recurrence from the input mixture image itself instead of relying on external database like the method in Chapter 4, which extends the practicability of our method to more diverse scenes. In the removal stage, we model the gradient distributions of $\mathbf{B}$ and $\mathbf{R}$ with long- and short-tail distributions, respectively, to avoid the direct dependency on commonly assumed image properties (*e.g.*, blur levels [1] or ghosting effect [2]) and hence better suppress artifacts by residual reflections. Our major contributions are summarized as follows:

- We build a R3 framework by automatically detecting regions with and without strong reflections and applying customized processing on different regions for more thorough reflection removal and more complete image content restoration;
- We develop a new content prior based on the internal patch recurrence to effectively restore missing contents covered by reflections;
- We integrate the content prior with newly designed gradient priors that distinctively model the distributions of reflection $\mathbf{R}$ and background $\mathbf{B}$ to achieve robust separation in a jointly optimized manner.

Our method is evaluated on a real-world dataset of 50 scenes with the mixture images and ground truth background and shows superior performance both quantitatively and visually.

The rest of this chapter is organized as follows. We introduce our method in Section 5.2 and its corresponding optimization solutions in Section 5.3. Experimental results and discussions are presented in Section 5.4. Finally, we conclude this chapter in Section 5.5.

5.2 Our method

Based on the mathematical model in Equation (5.1), we formulate the reflection removal as the maximum a posteriori (MAP) estimation problem, which is expressed using the Bayes' theorem as

$$\begin{aligned}
 \{\hat{\mathbf{B}}, \hat{\mathbf{R}}\} &= \operatorname{argmax}_{\mathbf{B}, \mathbf{R}} f(\mathbf{B}, \mathbf{R}, \sigma^2 | \mathbf{I}) \\
 &= \operatorname{argmax}_{\mathbf{B}, \mathbf{R}} f(\mathbf{I} | \mathbf{B}, \mathbf{R}, \sigma^2) f(\mathbf{B}) f(\mathbf{R}) \\
 &= \operatorname{argmin}_{\mathbf{B}, \mathbf{R}} L(\mathbf{I} | \mathbf{B}, \mathbf{R}, \sigma^2) + L(\mathbf{B}) + L(\mathbf{R}),
 \end{aligned} \tag{5.2}$$

where $f(\cdot)$ is the prior distribution and $L(\cdot) = -\log(f(\cdot))$. As commonly adopted by many reflection removal methods [1, 9], we assume the background and reflection distributions are independent, so we have $f(\mathbf{B}, \mathbf{R}) = f(\mathbf{B})f(\mathbf{R})$. The noise term $\mathbf{n}$ in Equation (5.1) is assumed to follow *i.i.d.* Gaussian distribution with the standard deviation as σ , then the likelihood model is represented as

$$L(\mathbf{I} | \mathbf{B}, \mathbf{R}, \sigma^2) = \frac{1}{2\sigma^2} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2. \tag{5.3}$$

$L(\mathbf{B})$ is our unified prior which is formulated as

$$L(\mathbf{B}) = L_c(\mathbf{B}) + L_g(\nabla \mathbf{B}), \tag{5.4}$$

where $L_c(\mathbf{B})$ is the content prior and $L_g(\nabla \mathbf{B})$ is the gradient prior.

In the following, we will first introduce how we determine the regions with and without strong reflections, then we introduce the detailed formulation of content prior based on the region labels, and finally we introduce our gradient priors for background and reflection, respectively.

Figure 5.3: One example of the detected reflection dominant regions (white pixels in the right-most column) with their corresponding images of background, mixture images, and reference of reflections identified by humans (red pixels in the third column). At the bottom two rows, we show two examples of the patch matching results.

5.2.1 Detecting regions with and without reflections

As shown in Figure 5.1, in many real world scenarios, visually obvious reflections only dominate a part of the whole image plane, which we call reflection dominant region. Analogously, for other small regions showing less obvious or no visual artifacts caused by reflections, we call them reflection non-dominant regions. The reflection (non-)dominant regions can be automatically detected by checking the difference be-

tween the input mixture image and the results from single-image reflection removal algorithms [1, 2, 9, 11].

We borrow the idea in [11] which makes use of slightly different blur levels between the background and reflection due to the depth of field to differentiate the two types of regions. Similar to [11], we first calculate the KL divergence between the input mixture image and its blurred version to get a background map denoted as E_B , which indicates the pixels belonging to the background. Then, based on the fact that the reflections are generally with small image gradients [71], the initial reflection map E'_R is obtained by choosing the image gradients below a threshold (set as 0.3). Combining E_B obtained before, the refined reflection map E_R is obtained as

$$E_R = \overline{E}_B \odot E'_R, \quad (5.5)$$

where $\overline{E}_B$ denotes *not* operation over E_B and $\odot$ is the element-wise multiplication. Such an operation enhances E_R with many misclassified pixels in E'_R removed. Finally, we apply a dilation operation S over E_R to further merge isolated pixels and regions in E_R as

$$\mathcal{D} = S(E_R). \quad (5.6)$$

The dialtion operator $S(\cdot)$ we use is a non-flat ball-shaped structuring element with neighborhood and height values all set as 5. $\mathcal{D}(\cdot)$ is a binary matrix, whose element as 1 indicates reflection dominant regions and 0 indicates non-dominant regions.

We show examples of the reflection detection results calculated from Equation (5.5) and Equation (5.6) in the rightmost column of Figure 5.3. Comparing with the manually labelled reference and the mixture image, we observe that pixels with strong reflections and covering large areas are correctly detected as reflection dominant regions. Misclassified pixels covering some sparse regions show little influence on the next stage of operations. The detected reflection dominant regions will be used in two parts of the

following processings: 1) the patch matching step for content prior which will be introduced in the next subsection and 2) the optimization stage which will be introduced later in Section 5.3.

5.2.2 Content prior

The proposed R3 method utilizes the patch recurrence property within the input mixture image itself. Given $\mathbf{q}_i$, an image patch overlaid with reflections and centered at position i , the patch recurrence property aims at using the estimation of $\mathbf{q}_i$ with the L nearest patches $\{\mathbf{p}_{i,l}\}_{l=1}^L$ from its surroundings to restore it. We assume that we have already obtained a set of reference patches $\{\mathbf{p}_{i,l}\}_{l=1}^L$ for now. Then the estimation of $\mathbf{q}_i$, denoted as $\mathbf{u}_i$, can be obtained as the weighted average of $\{\mathbf{p}_{i,l}\}_{l=1}^L$ as follows:

$$\mathbf{u}_i = \sum_{l=1}^L v_{i,l} \mathbf{p}_{i,l}. \quad (5.7)$$

Here, $v_{i,l}$ is the similarity weight expressed as $v_{i,l} = \exp(-\|\mathbf{p}_{i,l} - \mathbf{q}_i\|_2^2 / 2\sigma^2) / c$; c is the normalization constant to guarantee $\sum_l v_{i,l} = 1$ and the parameter σ controls the tolerance to noise due to illumination changes, compression, and so on.

We adopt the NCSR model [58] as the content prior and it can be formulated as follows:

$$L_c(\mathbf{B}) = \sum_i \|\boldsymbol{\alpha}_i - \boldsymbol{\beta}_i\|_1, \text{ s.t. } \mathbf{M}_i \mathbf{B} = \mathbf{D} \boldsymbol{\alpha}_i, \quad (5.8)$$

where $\mathbf{M}_i$ is the matrix extracting an image patch of size $N \times N$ from the background image $\mathbf{B}$; $\mathbf{D}$ denotes the dictionary built from the mixture image $\mathbf{I}$ and $\boldsymbol{\alpha}_i$ is the sparse coefficients corresponding to $\mathbf{q}_i$. Then $\boldsymbol{\beta}_i$ is the nonlocal estimation of $\boldsymbol{\alpha}_i$ in the sparse domain. Equation (5.8) minimizes the difference between $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$, which means that the missing contents in the mixture patch $\mathbf{q}_i$ can be restored by its similar patch $\mathbf{u}_i$. Without losing generality, we choose the K-PCA dictionaries [58, 66] as $\mathbf{D}$. To be specific, the image patches are extracted from the input mixture image $\mathbf{I}$, and clustered

into K clusters using K-means. For each cluster, a dictionary of PCA bases is learned to encode the patches in this cluster. Due to the orthogonal property of the PCA bases, α_i and β_i can be easily computed as $\alpha_i = \mathbf{D}^\top \mathbf{q}_i$ and $\beta_i = \mathbf{D}^\top \mathbf{u}_i$. Please refer to [58, 66] for more details.

Patch matching. Here, we explain how to obtain the reference patches $\{\mathbf{p}_{i,l}\}_{l=1}^L$ in Equation (5.7). If external images with similar contents to the ground truth of background $\mathbf{B}$ are available, patch matching can be accurately performed by searching the whole image and measuring the l_2 distance as shown in Chapter 4. For each $\mathbf{q}_i$, its reference patches $\{\mathbf{p}_{i,l}\}_{l=1}^L$ are searched within $W_H(i)$, a window with size $H \times H$, using l_2 distance:

$$d(\mathbf{q}_i, \mathbf{p}_{i,l}) = \|\mathbf{q}_i - \mathbf{p}_{i,l}\|_2, \forall l \in W_H(i). \quad (5.9)$$

Such a process is illustrated in Figure 6.4 and Figure 5.3. In Figure 5.3, given the mixture patch with reflections, we show its corresponding ground truth without reflections (extracted from $\mathbf{B}$) and the reference patches found using Equation (5.9) and Equation (5.10) (extracted from $\mathbf{I}$, with the dashed box as the searching window), respectively. From the patch matching results shown in Figure 5.3, the reference patches found using Equation (5.10) are more similar to the ground truth than the patches found using Equation (5.9).

Note that such an approach can provide quite clean patches only when the input mixture image contains some landmarks or objects that can be retrieved from an external database. To provide a more broadly applicable solution, the patch matching should be performed within the input mixture image itself. However, we cannot directly apply the simple matching strategy in Equation (5.9), due to that 1) the mixture images include regions with strong reflections (while external patches are all clean) and 2) these strong reflections make the simply l_2 distance measuring rather unreliable. To address these two problems, we develop our patch matching solution as guided by the reflection (non-

)dominant regions detected in Section 5.2.1 with a robust distant function:

$$d(\mathbf{q}_i, \mathbf{p}_{i,l}) = \rho_s(\mathbf{q}_i, \mathbf{p}_{i,l}) + \lambda \rho_r(\bar{\mathbf{q}}_i, \mathbf{p}_{i,l}), \quad (5.10)$$

$$\forall l \in W_H(i), \sum \mathcal{D}(\mathbf{p}_{i,l}) < N/2$$

Some reference patches found using Equation (5.10) may still contain reflections, which affect the accuracy of the patch matching and the subsequent reflection removal. To eliminate these negative effects and make sure that enough reference patches can be found, we add the constraint $\sum \mathcal{D}(\mathbf{p}_{i,l}) < N/2$ in Equation (5.10) to require that fewer than half of all pixels in a patch (note N is total number of pixels of a patch) are labelled as reflection dominant, *i.e.*, we limit the searching of reference patches only within reflection non-dominant regions. $\bar{\mathbf{q}}_i$ denotes as a patch being processed by TV-decomposition and λ is a balancing weight.

Taking the intrinsic image structure into consideration, we define the first robust distance term ρ_s by making use of the image gradient information as a structure-aware criterion:

$$\rho_s(\mathbf{q}_i, \mathbf{p}_{i,l}) = \|\mathbf{q}_i - \mathbf{p}_{i,l}\|_2 + \eta \|\nabla \mathbf{q}_i - \nabla \mathbf{p}_{i,l}\|_2. \quad (5.11)$$

We then define the second robust distance term ρ_r to specifically handle the patches in the reflection dominant regions. Due to the interference of the reflections, the candidate patches may not be truly relevant to the mixture patch. Considering the fact that the reflections are more related with the low-frequency component of images [25], we apply the TV-decomposition [72] to pre-process the input mixture image $\mathbf{I}$, so that structures with large gradient values are retained and the low-frequency components are filtered out. ρ_r is defined as

$$\rho_r(\bar{\mathbf{q}}_i, \mathbf{p}_{i,l}) = \|\bar{\mathbf{q}}_i - \mathbf{p}_{i,l}\|_2 + \eta \|\nabla \bar{\mathbf{q}}_i - \nabla \mathbf{p}_{i,l}\|_2. \quad (5.12)$$

Equation (5.10) is simply a linear combination of ρ_s and ρ_r , which shows a bal-

Figure 5.4: Some sample images of the background $\mathbf{B}$ and reflection $\mathbf{R}$ and their corresponding long-tail and short-tail gradient distributions.

ance between the original mixture patch $\mathbf{q}_i$ and the TV-decomposed $\bar{\mathbf{q}}_i$. In the reflection non-dominant regions, ρ_s can easily find sufficient numbers of patches, thus λ is given a smaller value to decrease the influence of ρ_r ; in contrast, we need a larger λ in the reflection dominant regions. Since the searching of reference patches is limited only within reflection non-dominant regions, we need a larger H for patches from the reflection dominant regions to increase the searching window size for matching sufficient numbers of reference patches. Comparing with the vanilla solution using Equation (5.9), our robust region-aware strategy in Equation (5.10) could find reflection-free patches with much closer appearances to the ground truth, as shown by the examples in the bottom row of Figure 5.3.

5.2.3 Gradient prior

The gradient priors play important roles in the reflection removal stage, as shown in Figure 6.4. A popular choice is fitting the heavy-tailed gradient distribution such as the Laplacian mixtures [9] to both background and reflection. We find such a homogeneous processing cannot take advantages of our R3 framework. Since the regional reflections only cover a part of the whole image, its corresponding gradient distributions should be different from the distributions of the background image, due to its sparser property. Therefore, we design the gradient prior of $\mathbf{B}$ and $\mathbf{R}$ in a heterogeneous manner using different types of distributions.

Assumption verification. To verify the above assumptions, we randomly choose 50 triplets of images from SIR²dataset. As described in Chapter 3, all images are taken by the DSLR camera directly and are not corrected to the linear response. These scenes include substantial real-world objects of complex 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.*).

Nine (out of 50) sample scenes used in our analysis are shown in Figure 5.4 and the corresponding average gradient distributions (over 50 scenes) are plotted next to them. The plotted distributions clearly show that the background and reflection images belong to the long-tail and short-tail distribution, respectively¹. Similar heterogeneous distributions are reported in [1], but their observations are only applicable to images where the background is in focus and reflection is out of focus. Our analysis here shows such heterogeneous distributions also apply to images with the reflection being in focus. We adopt the prior proposed in [20], which regularizes the high frequency part by manually

¹The distributions are only used to clarify our assumption and we do not learn any parameters or distributions from them.

manipulating the image gradients, to fit our gradient distribution for background as

$$L_g(\nabla \mathbf{B}) = \sum_x \phi(\nabla \mathbf{B}(x)), \quad (5.13)$$

where

$$\phi(\nabla \mathbf{B}(x)) = \begin{cases} \frac{1}{\epsilon^2} |\nabla \mathbf{B}(x)|^2, & \text{if } |\nabla \mathbf{B}(x)| < \epsilon, \\ 1, & \text{otherwise,} \end{cases} \quad (5.14)$$

where x is pixel locations. $L_g(\cdot)$ approximates L_0 norm by thresholding a quadratic penalty function parameterized by ϵ to avoid the distribution dropping too fast. Such a prior restores sharper edges belonging to the background image with less noise. Based on the proof in [20], Equation (5.14) is equivalent to

$$\phi(\nabla \mathbf{B}(x)) = \min_{\mathbf{l}_{mx}} \{ |\mathbf{l}_{mx}|^0 + \frac{1}{\epsilon} (\nabla_m \mathbf{B}_x - \mathbf{l}_{mx})^2 \}, \quad (5.15)$$

where $m \in \{h, v\}$ corresponding to the horizontal and vertical directions, respectively; $\mathbf{l}$ is an auxiliary variable and x is the pixel position.

The gradient distribution of $\mathbf{R}$ belongs to short-tail distribution partly due to the higher blur levels of $\mathbf{R}$ [1]. However, as we show in Figure 5.4, the majority of regions in $\mathbf{R}$ have brightness values closing to zero, *i.e.*, its gradient distribution should also have the sparse property when compared with the background. Therefore, we model it using a L_0 -regularized prior as

$$L(\mathbf{R}) = \|\nabla \mathbf{R}\|_0, \quad (5.16)$$

where $\|\cdot\|_0$ counts the number of non-zero values in $\nabla \mathbf{R}$. Such a prior enforces the sparsity property of $\mathbf{R}$ in its gradient domain.

By substituting Equation (5.3), Equation (5.15), Equation (5.8), and Equation (5.16) into Equation (5.2), our complete energy function is represented as

$$\begin{aligned}
 \{\hat{\mathbf{B}}, \hat{\mathbf{R}}\} = \operatorname{argmin}_{\mathbf{B}, \mathbf{R}, \alpha_i} & \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \omega \sum_i \|\mathbf{M}_i \mathbf{B} - \mathbf{D} \alpha_i\|_2^2 \\
 & + \xi \sum_i \|\alpha_i - \beta_i\|_1 + \delta \sum_{m \in \{h, v\}} \|\nabla_m \mathbf{R}\|_0 \\
 & + \gamma \sum_{m \in \{h, v\}} \sum_x \{|\mathbf{l}_{mx}|^0 + \frac{1}{\epsilon} (\nabla_m \mathbf{B}_x - \mathbf{l}_{mx})^2\},
 \end{aligned} \tag{5.17}$$

where i denotes the i -th patch or atoms, x is the pixel position, and $\hat{\mathbf{B}}, \hat{\mathbf{R}}$ are the intermediate results of $\mathbf{B}, \mathbf{R}$ generated at each iteration. It will be optimized in the next subsection.

5.3 Optimization

The direct minimization of Equation (5.17) is difficult due to the multiple variables involved in different terms. Thus, we divide the original problem into several subproblems by following the half-quadratic splitting technique [73] advocated by the previous methods in image deblurring and denoising [70]. The proposed algorithm iteratively updates the variables, reduces the objective function values in each iteration, and finally converges to local minima. We summarize each step of our method as Algorithm 3, and the details are described in the following paragraphs.

Solving for α_i . Given fixed $\mathbf{B}$ and $\mathbf{R}$, Equation (5.17) becomes a l_1 minimization problem:

$$\hat{\alpha}_i = \operatorname{argmin}_{\alpha_i} \omega \|\mathbf{M}_i \mathbf{B} - \mathbf{D} \alpha_i\|_2^2 + \xi \sum_i \|\alpha_i - \beta_i\|_1. \tag{5.18}$$

With fixed β_i , Equation (5.18) can be solved iteratively by the surrogate based algorithm [68]:

$$\alpha_i^{(t+1)} = \mathcal{S}_\tau(v_i^{(t)} - \beta_i) + \beta_i, \tag{5.19}$$

where $v_i^{(t)} = \mathbf{D}^\top (\mathbf{M}_i \mathbf{B} - \mathbf{D} \boldsymbol{\alpha}_i^{(t)}) / c + \boldsymbol{\alpha}_i^{(t)}$, $\mathcal{S}_\tau(\cdot)$ represents the soft-thresholding operator with threshold $\tau = \xi / \omega c$, and c is a constant to guarantee the convexity. Equation (5.19) balances the influence of β_i to α_i , and a larger τ generally allows a quicker convergence. Due to the orthogonal properties of the local PCA dictionaries $\mathbf{D}$, the sparse coding problem of Equation (5.18) can be solved in just one step [59].

Algorithm 3 Region-aware reflection removal algorithm

Require: Mixture image $\mathbf{I}$ and patch size N .

- 1: Estimate the reflection dominant regions $\mathcal{D}(\cdot)$ using Equation (5.5) and Equation (5.6);
- 2: Compute the dictionaries $\mathbf{D}$ by K-means and PCA;
- 3: **for** $m = 1$ to M **do**
- 4: **for** $j = 1$ to J **do**
- 5: Find the reference patches $\{\mathbf{p}_{i,l}\}_{l=1}^L$ corresponding to each patch in $\mathbf{I}$ using Equation (5.10);
- 6: Calculate the weighted average for each mixture patch using Equation (5.7);
- 7: Calculate the sparse codes α_i and β_i ;
- 8: Update sparse codes α_i^{j+1} by solving Equation (5.18);
- 9: Update $\mathbf{B}^{j+1}$ by solving Equation (5.20);
- 10: Update $\mathbf{R}^{j+1}$ by solving Equation (5.16);
- 11: **if** j reaches maximum number of iteration **then**
- 12: Set $\mathbf{B}^{m+1} = \mathbf{B}^{j+1}$ and $\mathbf{R}^{m+1} = \mathbf{R}^{j+1}$;
- 13: **end if**
- 14: **end for**
- 15: **if** $\text{mod}(m, 5) = 0$ **then**
- 16: Update $\mathbf{D}$ and the region labelling $\mathcal{D}(\cdot)$;
- 17: **end if**
- 18: **end for** return $\mathbf{B}^{m+1}$ and $\mathbf{R}^{m+1}$.
- 19:

Ensure: Estimated background $\mathbf{B}^*$ and reflection $\mathbf{R}^*$.

Solving for $\mathbf{B}$. When $\mathbf{R}$ and α_i are fixed, $\mathbf{B}$ can be estimated by solving the following optimization problem:

$$\begin{aligned}
 \hat{\mathbf{B}} = \underset{\mathbf{B}}{\operatorname{argmin}} \quad & \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \omega \sum_i \|\mathbf{M}_i \mathbf{B} - \mathbf{D} \boldsymbol{\alpha}_i\|_2^2 \\
 & + \gamma \sum_{m \in \{h, v\}} \sum_x \{|\mathbf{l}_{mx}|^0 + \frac{1}{\epsilon} (\nabla_m \mathbf{B}_x - \mathbf{l}_{mx})^2\},
 \end{aligned} \tag{5.20}$$

whose closed-form solution can be easily obtained by alternating between updating $\mathbf{l}$ and computing $\mathbf{B}$. Updating $\mathbf{l}$ is calculated as

$$\mathbf{l} = \begin{cases} \nabla \mathbf{B}, & \text{if } |\nabla \mathbf{B}| > \epsilon, \\ 0, & \text{otherwise.} \end{cases} \quad (5.21)$$

With $\mathbf{l}$ being fixed, the closed-form solution for Equation (5.20) is obtained similar to the strategy adopted by previous method [74]:

$$\hat{\mathbf{B}} = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(\mathbf{I}) - \mathcal{F}(\mathbf{R}) + \gamma \mathcal{F}(\sum_i \mathbf{M}_i^\top \mathbf{D} \alpha_i) + \frac{1}{\epsilon^2} \mathcal{F}_L}{\mathbf{E} + \gamma \mathcal{F}(\sum_i \mathbf{M}_i^\top \mathbf{M}_i) + \frac{1}{\epsilon^2} \mathcal{F}_D^2} \right). \quad (5.22)$$

$\mathbf{E}$ is a matrix with all elements being equal to one; $\mathcal{F}(\cdot)$ and $\mathcal{F}(\cdot)^{-1}$ denotes the Fourier transform and its inverse transform, respectively; $\mathcal{F}(\cdot)^*$ is the corresponding complex conjugate operator; and $\mathcal{F}_L = \sum_{m \in \{h,v\}} \mathcal{F}(\nabla_m)^* \mathcal{F}(\mathbf{l}_m)$ and $\mathcal{F}_D = \sum_{m \in \{h,v\}} \mathcal{F}(\nabla_m)^* \mathcal{F}(\nabla_m)$, where ∇_h and ∇_v are the horizontal and vertical differential operators, respectively.

Solving for $\mathbf{R}$. With all variables unrelated to $\mathbf{R}$ being fixed, the optimization problem for $\mathbf{R}$ becomes

$$\hat{\mathbf{R}} = \underset{\mathbf{R}}{\operatorname{argmin}} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \delta \|\nabla \mathbf{R}\|_0. \quad (5.23)$$

Equation (5.23) can be solved by introducing the auxiliary variables $\mathbf{g} = (\mathbf{g}_h, \mathbf{g}_v)$ w.r.t. the image gradients of $\nabla \mathbf{R}$ in horizontal and vertical directions, which is also adopted by [75]. Equation (5.23) can be expressed as

$$\hat{\mathbf{R}} = \underset{\mathbf{R}}{\operatorname{argmin}} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \mu \|\nabla \mathbf{R} - \mathbf{g}\|_2^2 + \delta \|\mathbf{g}\|_0. \quad (5.24)$$

The values of $\mathbf{g}$ are initialized to be zeros. In each iteration, the solution of $\mathbf{R}$ is obtained by solving

$$\min_{\mathbf{R}} \|\mathbf{I} - \mathbf{B} - \mathbf{R}\|_2^2 + \mu \|\nabla \mathbf{R} - \mathbf{g}\|_2^2 \quad (5.25)$$

The closed-form solution for the least squares problem above can be easily obtained as

$$\mathbf{R} = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(\mathbf{I}) - \mathcal{F}(\mathbf{B}) + \mu \mathcal{F}_G}{1 + \mu \sum_{m \in \{h,v\}} \mathcal{F}(\nabla_m)^* \mathcal{F}(\nabla_m)} \right), \quad (5.26)$$

where $\mathcal{F}_G = \mathcal{F}(\nabla_h)^* F(\mathbf{g}_h) + \mathcal{F}(\nabla_v)^* F(\mathbf{g}_v)$. Finally, given $\mathbf{R}$, we compute $\mathbf{g}$ by

$$\min_{\mathbf{g}} \mu \|\nabla \mathbf{R} - \mathbf{g}\|_2^2 + \delta \|\mathbf{g}\|_0. \quad (5.27)$$

Equation (5.27) is a pixel-wise minimization problem, whose solution is calculated as

$$\mathbf{g} = \begin{cases} \nabla \mathbf{R}, & |\nabla \mathbf{R}|^2 > \frac{\delta}{\mu}, \\ 0, & \text{otherwise.} \end{cases} \quad (5.28)$$

5.4 Experiment Results

To evaluate the performance of reflection removal, the majority of existing methods compare the visual quality of the estimated background images on 3 to 5 sets of real data [9, 11], or perform the quantitative evaluations using the synthetic images [1, 3]. Due to the lack of real-world dataset with ground truth, quantitative comparison using real data has seldom been done. Thanks to the dataset introduced in Chapter 3, we can compare our R3 method with state-of-the-art methods for both quantitative accuracies (w.r.t. its corresponding ground truth) and visual quality. Our experiments are conducted on the 50 sets of real data randomly selected from SIR² dataset, as described in Section 5.2.3. Though these images are all taken by DSLR camera with high resolution, considering the computation time and to make the image size compatible to all evaluated algorithms, all images are resized to 400×500 . Since the computations in our methods all belong to the per-pixel computation, such operations do not influence the final results, which are also adopted by previous methods [1, 2].

The main parameters in our method are set as follows: δ , ω , and γ in Equation (5.17)

Figure 5.5: Reflection removal results on two natural images under weak reflections, compared with AY07 [9], LB14 [1], SK15 [2], WS16, and NR17 [3]. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM and sLMSE values are displayed below the images.

are set to 0.004, 1.5, and 1, respectively. Empirically, for the patches from the reflection non-dominant regions, ξ in Equation (5.17) and Equation (5.18) is set to 3.2 (with $\tau = 2$); λ and the initial value of H in Equation (5.10) are set to 1 and 30, respectively. For the patches from the reflection dominant regions, ξ is set to 22.5 (with $\tau = 15$); λ and

the initial value of H are set to 0.01 and 10, respectively. μ in Equation (5.27) is set to 0.008. The patch size is set to 7×7 . L in Equation (5.7) is set to 8. The initial value of ϵ in Equation (5.14) is set to 0.05 and is divided by 2 in each iteration. H is added by 10 automatically if the number of reference patches found within current window is less than L .

5.4.1 Error metrics

We adopt the structural similarity index (SSIM) and local mean square error (LMSE), which are widely used by previous methods [1, 3, 52], as error metrics for quantitative evaluation. To make the value of LMSE consistent with SSIM, we convert it to a similarity measure as follows:

$$\text{sLMSE}(\mathbf{B}, \mathbf{B}^*) = 1 - \text{LMSE}(\mathbf{B}, \mathbf{B}^*), \quad (5.29)$$

where $\mathbf{B}$ is the ground truth and $\mathbf{B}^*$ is the estimated background image.

The luminance and contrast similarity in the original SSIM definition are sensitive to the intensity variance, so we define the structure index (SI) to focus only on the structural similarity between $\mathbf{B}$ and $\mathbf{B}^*$. SI shares similar format as the error metric proposed in [57], but it omits the luminance and contrast part in its original form as

$$\text{SI} = \frac{2\sigma_{\mathbf{B}}\sigma_{\mathbf{B}^*} + c}{\sigma_{\mathbf{B}}^2 + \sigma_{\mathbf{B}^*}^2 + c}, \quad (5.30)$$

where $\sigma_{\mathbf{B}}$ and $\sigma_{\mathbf{B}^*}$ are the variances of $\mathbf{B}$ and $\mathbf{B}^*$, respectively, and $\sigma_{\mathbf{B}}\sigma_{\mathbf{B}^*}$ is the corresponding covariance.

SSIM, sLMSE, and SI are error metrics evaluating the global similarity between $\mathbf{B}$ and $\mathbf{B}^*$. In our region-aware context, the reflections only dominate a part of the whole image. Based on our observations, though some methods [1, 11] downgrade the quality of the whole images, they can remove the local reflections quite effectively. We define

Figure 5.6: Reflection removal results on two natural images under strong reflections, compared with AY07 [9], LB14 [1], SK15 [2], WS16, and NR17 [3]. Corresponding close-up views are shown next to the images (the patch brightness $\times 2$ for better visualization), and SSIM and sLMSE values are displayed below the images.

the regional SSIM and SI, denoted as $SSIM_r$ and SI_r , to complement the limitations of global error metrics. We manually label the reflection dominant regions (*e.g.*, like the third column of Figure 5.3) and evaluate the SSIM and SI values at these regions similar to the evaluation method proposed in [76].

Figure 5.7: Results with and without reflection dominant region (the patch brightness $\times 1.3$ for better visualization).

5.4.2 Comparison with the state-of-the-arts

We compare our method with state-of-the-art single image reflection removal methods, including AY07 [9], LB14 [1], SK15 [2], NR17 [3], and our methods proposed in Chapter 2 and Chapter 4. For simplicity, we denote the method proposed in Chapter 2 as WS16 and the method proposed in Chapter 4 as WS17. We use the codes provided by their authors and set the parameters as suggested in their original papers. Except that for SK15 [2] we adjust its pre-defined threshold (set as 70 in their code) that chooses some local maxima values, since we find the default value shows degenerated results on our data and we manually adjust it for different images to make sure that a similar number of local maxima values to their original demo are generated. AY07 [9] requires the user annotations of background and reflection edges, and we follow their guidance to do the annotation manually.

Quantitative evaluations. The quantitative evaluation results using five different error metrics and compared with five state-of-the-art methods are summarized in Table 6.1, where the errors between the input mixture images and the corresponding ground truth

Table 5.1: Quantitative evaluation results using five different error metrics and compared with AY07 [9], LB14 [1], SK15 [2], WS16 [11], and NR17 [3].

	Baseline	Ours	AY07	LB14	SK15	WS16	NR17
sLMSE	0.969	0.980	0.927	0.939	0.830	0.963	0.969
SSIM	0.940	0.944	0.906	0.862	0.870	0.937	0.930
SI	0.965	0.958	0.943	0.958	0.913	0.955	0.950
SSIM _r	0.857	0.936	0.880	0.847	0.858	0.921	0.908
SI _r	0.886	0.942	0.905	0.906	0.886	0.921	0.925

are used as the baseline comparison. The numbers displayed are the mean values over all 50 images in our dataset. As shown in Table 6.1, the proposed algorithm consistently outperforms other methods for all five error metrics. The higher SSIM and sLMSE values indicate that our method recovers the whole background image with better quality, whose global appearance is closer to the ground truth. For SI values, all methods are lower than the baseline, which is partly because all methods impair the global structures of the input images. However, due to the regional strategy in our R3 method, it still beats the other five methods and achieves the second best result. The higher SI values tell that our method preserves the structural information more accurately. The higher SSIM_r and SI_r values mean that our method can remove strong reflections more efficiently in the reflection dominated regions than other methods. LB14 [1] shows the second best result on SI; the most recent method NR17 [3] shows the second best results on SSIM, sLMSE, SSIM_r and SI_r.

Visual quality comparison. We then show examples of estimated background images by our method and five other methods in Figure 5.5 (three examples with weak reflections) and Figure 5.6 (three examples with strong reflections) to check their visual quality. In these examples, our method removes the reflections more effectively and recovers the details of the background image more clearly. NR17 [3] and LB14 [1] remove the reflections to some extent, but from the results shown in the third example of Fig-

Figure 5.8: Results with and without the gradient priors (the patch brightness $\times 1.3$ for better visualization).

ure 5.5 and Figure 5.6, some residue edges remain visible for the reflections that are not out of focus. LB14 [1] also causes a color change of the input mixture image, where the results are much darker than the ground truth. Both LB14 [1] and WS16 [11] show some over-smooth artifacts, when they are not able to differentiate the background and reflection clearly. When the edges can be correctly labelled, AY07 [9] shows acceptable results in some examples (*e.g.*, the third example in Figure 5.5), but the performance is poor when the edges cannot be clearly differentiated by human labelling (*e.g.*, the first example in Figure 5.6). The performance of SK15 [2] is a bit degenerated with these examples, and it shows some patchy artifacts. When the reflection is strong (*e.g.*, the first example in Figure 5.6), our method not only removes the undesired reflections but also restores the missing contents of the background caused by the reflection, thanks to the region-aware content prior.

5.4.3 The effect of the reflection dominant region

Comparing with existing methods, region-aware processing is unique in the proposed whole framework. To evaluate whether it effectively recovers the details in reflection dominant regions and avoid artifacts in reflection non-dominant regions, we show recovered background image with and without the reflection (non-)dominant region labelling. Two examples are shown in Figure 5.7. In both examples, the methods without reflection dominant regions only attenuate the reflections but fail to remove them, but the region-aware approach successfully removes the reflections; in the top example, the image details (*e.g.*, the patch in the red box) of the method without the reflection dominant regions are rather blurred.

5.4.4 The effect of the gradient prior

We conduct another experiment to show the effectiveness of the gradient priors in Figure 5.8. Although for the image patches in the red and green boxes, both reflections are removed regardless of whether gradient priors are considered, the image patches in the blue boxes clearly show that the gradient prior helps to keep the sharpness of the edges so that the structural information is better recovered in the background image.

5.4.5 Comparison with WS17

Our proposed method WS17 in Chapter 4 also makes use of the patch recurrence from several similar images and content priors, by assuming that reflection-free images with similar content are available from an external database. To make their assumptions satisfied, we use images containing objects which can be easily retrieved from an external database, and provide both the input mixture image and external database to WS17. The comparison between our method and WS17 are illustrated in Figure 5.9. With the help of an external database, WS17 shows superior performance in some parts (the blue box

Figure 5.9: Comparison between our proposed method and WS17 (the patch brightness $\times 1.3$ for better visualization).

in Figure 5.9). But our method still provides comparable results to WS17 with only internal image recurrence, thanks to the robust patch matching in reflection dominant regions. Note our method can be applied to much broader categories of images.

5.4.6 Convergence analysis

The last experiment shows the convergence of our algorithm. As we have claimed in Section 5.3, a larger τ in Equation (5.19) generally allows a quicker convergence of our R3 method. In our settings, the patches from the reflection dominant regions are given a larger τ values defined as τ_1 here and the patches from the reflection non-dominant regions are assigned a smaller τ defined as τ_2 here. We set $\tau_1 = 15$ and $\tau_2 = 2$ in our experiments. To validate the settings, we test different values by fixing one and changing another one. The performances with different values are illustrated in Figure 5.10. By fixing $\tau_2 = 2$, τ_1 is set to 10, 15 (the values used in our experiments), 100 and 200. A larger τ_1 can achieve better results in the first iteration and converge faster under a larger value. By fixing $\tau_1 = 15$, τ_2 is set to 5.5, 2 (the values used in our experiments), 12.5, 15 and 20. A larger τ_2 decrease SSIM values after approximate six

Figure 5.10: The convergence analysis of our proposed method under different τ values.

iterations, which indicates that the image structure are impaired. It is partly due to the over-smooth effect of the non-local image prior we adopt, which is explained in [77]. A smaller τ_2 achieves similar performances when compared with the value used in our experiment. Considering the performance variation with different τ , the parameters in our experiments ($\tau_1 = 15$ and $\tau_2 = 2$) can achieve good results and keep stable after six iterations

5.5 Conclusion

We introduce reflection dominant regions to single image reflection removal problem to efficiently remove reflections and avoid artifacts caused by incompletely removed reflections in an adaptive manner. We integrate the content prior and gradient prior into a unified R3 framework to take account of both content restoration and reflection suppression. By refining the sparse coefficients learned from the mixture images with

the reference patches, our method can generate a more accurate sparse regularization term to reconstruct the background images. We show better performances than state-of-the-art methods for both the quantitative and visual qualities.

Limitations. In spite of the effectiveness of our R3 method, it also has several limitations:

- The patch selection step is computationally expensive. Its complexity increases linearly with the window size. However, our current implementation is an unoptimized Matlab implementation, which takes three minutes for the patch matching and fewer than 30 seconds for other steps on a modern PC. Based on the experience in denoising [78] with the similar formulation, the computation can be sped up by using more efficient programming languages (*e.g.*, C++) and parallel implementations;
- Our method adopts the non-local image prior as the content prior. As mentioned in [77], it is prone for the non-local image priors to over-smooth highly textured regions, especially in the case of strong artifacts. The performance drops when the background is textured;
- Our method is based on the observation that reflection only dominates a part of an image. However, in real scenes, it is possible that the whole image is overlaid with strong reflections; in such a case our method may fail due to the ‘rare patch effect’ [79].
- Since our method utilizes the reference patches around the mixture patch to remove the reflections, the information of the background must be kept more or less. If very strong reflections exist in a scene, the reference patches cannot be found since very few details of the background are kept. In this situation, the reflection removal problems degrade to an image inpainting problem;

- Though our method does not explicitly rely on image priors (*e.g.*, the blur levels [1] or ghosting effects [2]), the reflection dominant region detection is based on the depth of field of the input mixture image. When the depth of field is not uniform, the detection may be less accurate. In such a situation, our performance is similar to that in Figure 5.7 where sharp edge information cannot be clearly recovered.

To address these limitations, in the next chapter, we will present another work with better generalization ability to deal with complicated scenarios by using the deep learning techniques.

Chapter 6

CRRN: Multi-Scale Guided Concurrent Reflection Removal Network

As we discussed in Chapter 3, previous methods utilize different non-learning based priors such as the separable sparse gradients caused by different blur levels and content priors based on the nonlocal correlations in the input images. Though these methods achieve promising results in some specific situations, they often fail due to their limited description capability to the properties of real-world reflections. In this chapter, we propose the Concurrent Reflection Removal Network (CRRN) with better description capability to tackle this problem in a unified framework. Our proposed network integrates image appearance information and multi-scale gradient information with human perception inspired loss function, and is trained on a new dataset with 3250 reflection images taken under diverse real-world scenes. Extensive experiments on the SIR² dataset show that the proposed method performs favorably against state-of-the-art methods.

Figure 6.1: Samples of captured reflection images in the ‘RID’ and the corresponding synthetic images using the ‘RID’. From top to bottom rows, we show the diversity of different illumination conditions, focal lengths, and scenes.

6.1 Introduction

Though the reflection removal problem has been discussed for more than decades, most methods can be cast into the two-stage framework proposed by Levin *et al.* in their pioneer work [9], where they first locate the reflection regions (*e.g.*, by classifying the background and reflection edges) and then restore the background layers based on the edge information as shown in Figure 6.1. The only difference for previous methods is how to locate the reflection regions. Most of the existing reflection removal methods remove reflections by using some heuristic observations, *e.g.*, the gradient priors on the basis of the different blur levels between background and reflection like our method proposed in Chapter 2 and the method proposed in [4]. These non-learning based methods can show promising results in some specific situations. However, they are often violated in real-world scenarios, since the low-level image priors they adopt only describe a limited range of the reflection properties and project the partial observation as the whole truth. When the structures and patterns of the background are similar to those of the reflections, the non-learning based methods have difficulty in simultaneously removing reflections and recovering the background [80].

To capture the reflection properties more comprehensively, recent methods have adopted the deep learning to solve this problem [10, 28]. Existing deep learning based methods [10, 28] show improved modeling ability that captures a variety of reflection

image characteristics [80, 81]. However, they still adopt a two-stage framework for gradient inference and image inference as many non-learning based methods [9, 11], which do not fully explore the multi-scale information for background recovery. Moreover, they mainly rely on the pixel-wise loss ($\mathcal{L}_2$ and $\mathcal{L}_1$ loss), that may generate blurry predictions [82, 83]. Last but not least, existing methods are mainly trained with synthetic images which can never capture the comprehensive information in real world image formation process completely.

To address these drawbacks, we propose the Concurrent Reflection Removal Network (CRRN) to remove reflections observed in the wild scenes, as illustrated in Figure 6.4. Our major contributions are summarized as follows:

- In contrast to the conventional two-stage framework that classifies the gradients, and then recovers the background [9, 11, 25, 34], we combine the two separate stages (gradient inference and image inference) in one unified mechanism to remove reflections concurrently.
- We propose a multi-scale guided learning network to better preserve the background details, where the background reconstruction in the image inference network is closely guided by the associated gradient features in the gradient inference network.
- We design a perceptually motivated loss function, which helps suppress the blurry artifacts introduced by the pixel-wise loss functions, and generate better results.
- To facilitate the training of CRNN for general compatibility on real data, we capture a large-scale reflection image dataset to generate training data, which has proved to improve the performance and generality of our method. To the best of our knowledge, this is the first reflection image dataset for data-driven methods.

The remainder of this paper is organized as follows. Section 6.2 gives a brief overview on the preparation for the training dataset. Section 6.3 is devoted to our new proposed

Figure 6.2: Samples of captured reflection images in the ‘RID’ and the corresponding synthetic images using the ‘RID’. From top to bottom rows, we show the diversity of different illumination conditions, focal lengths, and scenes.

reflection removal model. Experiments are presented in Section 6.4. The conclusions and discussions are presented in Section 6.5.

6.2 Dataset Preparation

6.2.1 Real-world reflection image dataset for data-driven methods

Real-world image datasets play important roles in studying physics-based computer vision [53] and face anti-spoofing [84] problems. Although the reflection removal problem has been studied for more than decades, publicly available datasets are rather limited. The data-driven methods need a large-scale dataset to learn the reflection image properties. As far as we know, ‘SIR²’ [43] is the largest reflection removal image datasets, which provides approximately 500 image triplets composed of mixture, background, and reflection images, but its scale is still not sufficient for training a complicated neural network. Considering the difficulty in obtaining the image triplet like ‘SIR²’, an alternative solution to the data size bottleneck is to use the synthetic image dataset. The recent deep learning based method [10] provides a reasonable way to generate the reflection images by taking the regional properties and blurring effects of the reflections

into consideration to make their data similar to the images taken in the wild. However, the ignorance of other reflection image properties (*e.g.*, ghosting effects, various types of noise in the imaging pipeline) may degrade the training and thus limits its wide applicability to real-world scenes.

To facilitate the training of CRRN for general compatibility on real data, we have constructed a large-scale **Reflection Image Dataset** called ‘**RID**’, which contains 3250 images in total. We can then use the captured reflection images from the ‘**RID**’ to synthesize the input mixture images.

To collect reflection images, we use a NIKON D5300 camera configured with varying exposure parameters and aperture sizes under a fully manual mode to capture images in different scenes. The reflection images are taken by putting a black piece of paper behind the glass while moving the camera and the glass around, which is similar to what have been done in [12, 43].

The ‘**RID**’ has the following two major characteristics, with example scenes demonstrated in Figure 6.2:

- **Diversity.** We consider three aspects to enrich the diversity of the ‘**RID**’: 1) We take the reflection images at different illumination conditions to include both strong and weak reflections (the first row in Figure 6.2 left); 2) we adjust the focal lengths randomly to create different blur levels of reflection. (the second row in Figure 6.2 left); 3) the reflection images are taken from a great diversity of both indoor and outdoor scenes, *e.g.*, streets, parks, inside of office buildings, and so on (the third row in Figure 6.2 left).
- **Scale.** The ‘**RID**’ has 3250 images in total with approximately 2000 reflection images from the bright scenes and other reflection images are from the relatively dark scenes to meet the request of data-driven methods.

Figure 6.3: The estimated gradient generated by the gradient inference network, compared with the reference gradient.

6.2.2 Generating training data

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

$$\mathbf{I} = \alpha \mathbf{B} + \beta \mathbf{R}, \quad (6.1)$$

where $\mathbf{I}$ is the mixture image, $\mathbf{B}$ is the background to be recovered, and $\mathbf{R}$ is the reflection to be removed. In Equation (6.1), the mixture image $\mathbf{I}$ is a linearly weighted additive of the background $\mathbf{B}$ and the reflection $\mathbf{R}$. Our partially synthetic and partially real training image $\mathbf{I}$ is generated by adding the reflection images from the ‘RID’ as reflection $\mathbf{R}$ and other natural images (*e.g.*, we use the COCO dataset [85] and the PASCAL VOC dataset [86]) as the background $\mathbf{B}$ with different weighting factors.

To ensure a sufficient amount of training data, α and β are randomly sampled from 0.8 to 1 and 0.1 to 0.5, respectively, and we further augment the generated image with two different operations: image rotation and flipping. In total, our training dataset includes 14754 images.

6.3 Proposed Method

In this section, we describe the design methodology of the proposed reflection removal network, the optimization using human perception inspired loss function, and the details for network training.

6.3.1 Network architecture

According to Equation (6.1), given the observed images with reflections $\mathbf{I}$, our task here is to estimate $\mathbf{B}$. Since the estimation of $\mathbf{B}$ and $\mathbf{R}$ are intrinsically correlated and the gradient information $\nabla\mathbf{B}$ has been proved to be a useful cue that guides the reflection removal process [9, 11, 12], we develop the Concurrent Reflection Removal Network (CRRN) with a multi-task learning strategy, which concurrently estimates $\mathbf{B}$ and $\mathbf{R}$ under the guidance of $\nabla\mathbf{B}$. CRRN can be trained using multiple loss functions based on the ground truth of $\mathbf{B}$, $\mathbf{R}$, and $\nabla\mathbf{B}$, as shown in Figure 6.4. Given the input image $\mathbf{I}$, we denote the dense prediction of $\mathbf{B}$, $\mathbf{R}$ and $\nabla\mathbf{B}$ as follows:

$$(\mathbf{B}^*, \mathbf{R}^*, \nabla\mathbf{B}^*) = \mathcal{F}(\mathbf{I}, \theta), \quad (6.2)$$

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

CRRN is implemented by designing two cooperative sub-networks. Different from the conventional two-stage framework, we combine the gradient inference and the image inference into one unified mechanism to do the two parts concurrently. For the gradient inference network (GiN), the input is a 4-channel tensor, which is the combination of the input mixture image and its corresponding gradients; it estimates $\nabla\mathbf{B}$ to extract the image gradient information from multiple scales and guide the whole image reconstruction process. The image inference network (IiN), takes the mixture image as the input and

Figure 6.4: The framework of CRRN. It consists of two cooperative sub-networks: the gradient inference network (GiN) to estimate the gradients of the background and the image inference network (IiN) to estimate the background and reflection layers. We feed GiN with the mixture image and its corresponding gradient as a 4-channel tensor and IiN with the mixture image containing reflections. The upsampling stage of IiN is closely guided by the associate gradient features from GiN with the same resolution. IiN consists of two feature extraction layers to extract the scale invariant features related with the background. IiN gives the estimated background and reflection images, while GiN gives the estimated gradient of background as output.

extracts background feature representations which describe the global structures and the high-level semantic information to estimate $\mathbf{B}$ and $\mathbf{R}$. To allow the multiple estimation tasks to leverage information from each other, IiN shares the convolutional layers from GiN. The detailed architecture of GiN and IiN is introduced as follows:

Gradient inference Network (GiN): GiN is designed to learn a mapping from $\mathbf{I}$ to $\nabla\mathbf{B}$. As shown in Figure 6.4, the structure of GiN is a mirror-link framework with the encoder-decoder CNN architecture. The encoder part consists of five convolutional layers with stride equal to 1 and five convolutional layers with stride equal to 2. Each layer with stride 1 is followed by the layer with stride 2, which can progressively extract and down-sample features. In the decoder part, the features are upsampled and combined to reconstruct the output gradient without the reflection interference. In order to preserve the sharp details and avoid losing gradient information, the early encoder features are linked to its corresponding decoder layers with the same spatial resolution. An example result is shown in Figure 6.3, which demonstrates GiN successfully removes the gradients from reflection and remains the gradient belonging to the background.

Image inference Network (IiN): IiN is a multi-task learning network constructed on the basis of VGG16 network [81]. Recent works show that VGG16 network trained with large amount of data on high-level computer vision tasks can be well generalized to inverse imaging tasks such as shadow removal [76] and saliency detection [87]. To make the feature representations from the pre-trained VGG16 model suitable for our problem, we first replace the fully-connected layers in VGG16 model by a 3×3 convolutional layer [76] and then fine tune them for the reflection removal task.

After feature extractions with VGG16 net, we design a joint filtering network to predict $\mathbf{B}$ with multi-context features. It consists of two feature extraction layers and five transposed convolutional layers. We adopt the ‘Reduction-A/B layers’ from Inception-ResNet-v2 [32] as the ‘Feature extraction layers A/B’ in CRRN. Such a model is able to

extract the scale invariant features by using multi-size kernels [88], but it is seldom used in image-to-image problems due to its decimated features caused by pooling layers. To make it fit our problem, we make two modifications: First, the pooling layers in the original model are replaced by two convolutional layers with 1×1 and 7×7 filter sizes, respectively; second, the stride of all convolutions are decreased to 1. The transposed convolutional layers in this part have a parallel framework which are composed of three sub-layers, as shown in Figure 6.4. We also adopt the residual network to help learn the mapping due to the narrow intensity range of the residual ($\mathbf{I} - \mathbf{B}$) [80].

Multi-scale guided inference. Multi-scale representations have shown to be effective in the extraction of image details for reflection removal [11] and other inverse imaging problems [82, 89]. To make full use of the multi-scale information of the decoder part in GiN, the output of each transposed convolutional layers of GiN is concatenated with the output of transposed convolutional layers in IiN at the same level, which is illustrated in Figure 6.4.

6.3.2 Loss function

Previous methods mainly adopt the pixel-wise loss function [10]. It is simple to calculate, but produces blurry predictions due to its inconsistency with human visual perception for natural images. To provide more visually pleasing results, we take the human perception into considerations when design our loss function.

In IiN, we adopt the perceptually motivated Structural similarity index (SSIM) [90] to measure the similarity between the estimated $\mathbf{B}^*$ and $\mathbf{R}^*$ and their corresponding ground truth. SSIM is defined as

$$\text{SSIM}(x, x^*) = \frac{(2\mu_x\mu_{x^*} + C_1)(2\sigma_{xx^*} + C_2)}{(\mu_x^2 + \mu_{x^*}^2 + C_1)(\sigma_x^2 + \sigma_{x^*}^2 + C_2)}, \quad (6.3)$$

where μ_x and μ_{x^*} are the means of x and x^* , σ_x and σ_{x^*} are the variances of x and x^* , and σ_{xx^*} is their corresponding covariances. SSIM measures the similarity between two images from the luminance, the contrast, and the structure. To make the values compatible with the common settings of the loss function in deep learning, we define our loss function for LiN as

$$\mathcal{L}^{\text{SSIM}}(x, x^*) = 1 - \text{SSIM}(x, x^*), \quad (6.4)$$

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

Despite its perceptual contribution, SSIM may cause changes of brightness and shifts of colors which makes the final results become dull [91], due to its insensitiveness to uniform bias. To solve this problem, we also introduce the $\mathcal{L}_1$ loss for the background layer to better balance brightness and color.

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

$$\mathcal{L}^{\text{SI}}(x, x^*) = 1 - \text{SI}(x, x^*). \quad (6.5)$$

SI is used to measure the structural similarity between two images as demonstrated in [43], which is defined as

$$\text{SI} = \frac{2\sigma_{xx^*} + c}{\sigma_x^2 + \sigma_{x^*}^2 + c}, \quad (6.6)$$

where all parameters share similar definitions as Equation (6.4).

Combining the above terms, our complete loss function becomes

$$\begin{aligned} \mathcal{L} = & \gamma \mathcal{L}^{\text{SSIM}}(\mathbf{B}, \mathbf{B}^*) + \mathcal{L}_1(\mathbf{B}, \mathbf{B}^*) + \\ & \mathcal{L}^{\text{SSIM}}(\mathbf{R}, \mathbf{R}^*) + \mathcal{L}^{\text{SI}}(\nabla \mathbf{B}, \nabla \mathbf{B}^*), \end{aligned} \quad (6.7)$$

where the weighting coefficient γ is empirically set as 0.8 in our experiments.

6.3.3 Training strategy

We have implemented CRRN using PyTorch¹. To prevent overfitting, our network employs the multi-stage training strategy: GiN is first trained independently for 40 epochs with learning rate 10^{-4} , then it is connected with IiN, and the entire network is fine-tuned end-to-end, which grants the two sub-networks more opportunities to cooperate accordingly. The learning rate for the whole network training is initially set to 10^{-4} for the first 50 epochs and then decreases to 10^{-5} for the next 30 epochs.

Prior works that use deep learning to solve the inverse imaging problems [92, 93] or layer separation problems [94] mainly optimize the whole network on patches with resolution $n \times n$ cropped from the whole images. However, many real-world reflections only occupy some regions in an image like the regional ‘noise’ [43], we call it regional properties of reflections. Training with the patches without obvious reflections could potentially degrade the final performance. To get avoid of such negative effects, CRRN is trained using whole images with different sizes. We adopt a multi-size training strategy by feeding images of two sizes: coarse scales 96×160 and fine scale 224×288 , to make the network scale-invariant.

6.4 Experiments

To evaluate the performance of CRRN, we first compare with state-of-the-art reflection removal algorithms for both quantitative benchmark scores and visual qualities on the SIR² dataset [43]. We then conduct a self-comparison experiment to justify the necessity of the key components in CRRN. The SIR² dataset contains image triplets from controlled indoor setup and wild data. The indoor data are mainly designed to explore

¹<http://pytorch.org/>

Figure 6.5: Examples of reflection removal results on four wild scenes, compared with FY17[10], NR17 [3], WS16 [11], and LB14 [1]. Corresponding close-up views are shown next to the images (with patch brightness $\times 2$ for better visualization), and SSIM and SSIM_r values are displayed below the images.

the influence of different parameters [43]. Since our method aims at removing reflections appeared in the wild scenes, we only evaluate on their wild dataset.

We adopt SSIM [90] and SI [43] as error metrics for our quantitative evaluation,

Figure 6.6: The generalization ability comparison with FY17 [10] on their released validation dataset.

Figure 6.7: The generalization ability comparison with FY17 [10] on their released validation dataset.

which are widely used by previous reflection removal methods [1, 12, 43]. Due to the regional properties of reflections, we experimentally observe that many existing reflection removal methods [1, 3, 11] may downgrade the quality of whole images, although they can remove the local reflections cleanly. The original definitions of SSIM and SI, which evaluate the similarity between B and B^* in the whole image plane, may not reflect the performance of reflection removal unbiasedly. We therefore define the regional SSIM and SI, denoted as $SSIM_r$ and SI_r , to complement the limitations of global error

Table 6.1: Quantitative evaluation results using four different error metrics, and compared with FY17[10], NR17 [3] WS16 [11] and LB14 [1].

	SSIM	SI	SSIM _r	SI _r
Ours	0.895	0.925	0.861	0.890
FY17 [10]	0.867	0.902	0.812	0.847
NR17 [3]	0.884	0.903	0.850	0.880
WS16 [11]	0.876	0.910	0.843	0.881
LB14 [1]	0.833	0.920	0.801	0.861

metrics. We manually label the reflection dominant regions and evaluate the SSIM and SI values at these regions similar to the evaluation method proposed in [44, 76].

6.4.1 Comparison with the state-of-the-arts

We compare our method with state-of-the-art single-image reflection removal methods, including FY17 [10], NR17 [3], WS16 [11], and LB14 [1]. For a fair comparison, we use the codes provided by their authors and set the parameters as suggested in their original papers. For FY17 [10], we follow the same training protocol introduced in their paper to train their network using our training dataset.

Quantitative comparison. The quantitative evaluation results using four different error metrics and compared with four state-of-the-art methods are summarized in Table 6.1. The numbers displayed are the mean values over all 100 sets of wild images in the SIR² dataset. As shown in Table 6.1, CRNN consistently outperforms other methods for all four error metrics. The higher SSIM values indicate that our method recovers the whole background image with better quality, whose global appearance is closer to the ground truth. The higher SI values indicate that our method preserves the structural information more accurately. The higher SSIM_r and SI_r values mean that our method can remove strong reflections more efficiently in the regions overlaid with reflections than other methods. NR17 [3] shows the second best average performance with all error metrics.

Figure 6.8: The output of liN and GiN in CRRN against liN and GiN only. Corresponding close-up views are shown below the images (with patch brightness $\times 1.6$ for better visualization).

Visual quality comparison. We then show examples of estimated background images by our method and four state-of-the-art methods in Figure 6.5 to check their visual quality. In these examples, our method removes reflections more effectively and recovers the details of the background images more clearly. All the non-learning based methods (NR17 [3], WS16 [11], and LB14 [1]) remove the reflections to some extent, but residual edges remain visible for the reflections that are not out of focus, and they also show some over-smooth artifacts when they are not able to differentiate the background and reflection clearly (*e.g.*, the result generated by WS16 [11] in the second column). LB14 [1] causes some color change in the estimated result (*e.g.*, the fourth column) partly due to the insensitivity of the Laplacian data fidelity term to the spatial shift of the pixel values [3]. NR17 [3] and LB14 [1] sometimes achieve similarly good quantitative values in SSIM (*e.g.*, the first column), but their estimated results still show obviously visible residual edges (the red box of LB14 [1] in the first column). The deep learning based method FY17 [10] is also good at preserving the image details and it does not cause the over-smooth artifacts as non-learning based method. However, the network in FY17 [10] is less effective in cleaning the residual edges comparing to CRRN. The SSIM and SSIM_r values below each image also prove the advantage of our method.

Comparing generality with FY17 [10]. The applicability to general unseen data of deep learning based methods is important yet challenging. To show the generalization

Figure 6.9: The output of IiN and GiN in CRRN against IiN and GiN only. Corresponding close-up views are shown below the images (with patch brightness $\times 1.6$ for better visualization).

Figure 6.10: Extreme examples with whole-image-dominant reflections, compared with FY17 [10], NR17 [3] WS16 [11] and LB14 [1].

ability of our method, we show results using released validation dataset from the project website of FY17 [10]². In this experiment, CRRN is still trained with our dataset described in Section 6.2.2 and strategy in Section 6.3.3, but for FY17 [10] we use the model released in their website (trained with their own data). Due to the lack of ground truth, only the visual quality is compared here. From the result shown in Figure 6.7, it is not surprised that FY17 [10] performs well using their trained model on their validation dataset, but CRRN also achieves reasonably good results and performs even

²<https://github.com/fqnchina/CEILNet>

Figure 6.11: Extreme examples with whole-image-dominant reflections, compared with FY17 [10], NR17 [3] WS16 [11] and LB14 [1].

Table 6.2: Result comparisons of the proposed CRRN against CRRN using $\mathcal{L}_1$ loss in Equation (6.7) only and its sub-networks.

	SSIM	SI	SSIM _r	SI _r
IiN in CRRN	0.895	0.925	0.861	0.890
IiN in CRRN ($\mathcal{L}_1$)	0.883	0.910	0.849	0.865
IiN only	0.867	0.892	0.843	0.859

better in some regions (*e.g.*, the red box in the left part of Figure 6.7). Recall that when FY17 [10] is trained with our data and tested on the SIR² dataset, its quantitative and qualitative performances are below our method as shown in previous experiments.

6.4.2 Network analysis

CRRN consists of two sub-networks, *i.e.*, GiN and IiN. To further analyze the contribution of GiN and the perceptually motivated losses, we have trained three variant networks, one using $\mathcal{L}_1$ loss only, one using IiN only without the gradient feature layers and the other one using GiN only.

Table 6.2 shows the values using four error metrics of the two variant networks and

the complete CRRN model. The comparisons between the results obtained by GiN in CRRN and GiN alone are shown in Figure 6.9. We can see that none of the three models perform better than the concurrent model using the perceptually motivated losses. When only using the pixel-wise losses, the performance of CRRN become worse. When removing GiN, LiN alone has relatively poor performance and the SSIM errors on the global and regional scales decreased to 0.867, compared with 0.895 by the concurrent model. From Figure 6.9, GiN in the CRRN model also outperforms GiN alone. The output of LiN only and GiN only remains more visible residual edges than that in CRRN as shown in the green and blue boxes in Figure 6.9. This demonstrates the effectiveness of the embedding mechanism in our network, where the two sub-network benefits each other in the whole estimation process.

6.5 Conclusion

We present a concurrent deep learning based framework to effectively remove reflection from a single image. Unlike the conventional pipeline that regards the gradient inference and image inference as two separate processes, our network unifies them as a concurrent framework, which integrates high-level image appearance information and multi-scale low-level features. Thanks to the newly collected real-world reflection image dataset and the corresponding training strategy, our method shows better performance than state-of-the-art methods for both the quantitative values and visual qualities and it is verified to be effectively generalized to other unseen data.

Limitations. The performance of CRRN may drop when the whole images are dominated by reflections. We show two examples on such extreme cases in Figure 6.11. In these examples, CRRN cannot remove the reflection completely and the estimated background still remains visible residual edges. However, even in this challenging examples, CRRN still removes the majority of reflections and restores the background details,

which performs better than all other state-of-the-art methods. On the other hand, training a deep learning network directly on the images may suffer from gradient vanishing problem and the CNN may also introduce the color shift to the estimated image [80]. In the future, we will continue working on these parts to improve the generalization ability for dealing with challenging scenes.

Chapter 7

Conclusions and Future Works

This chapter provides a summary of the works presented in the previous chapters in this thesis. While each previously mentioned chapter has a self-contained conclusion and discussion, this chapter aims at reviewing these chapter in a unified and global manner. Meanwhile, we also describe the potential directions for future work.

7.1 Conclusions

This thesis has developed four distinct but also related works for reflection removal problems. Improvements over previous method and comparisons between the four works have been demonstrated through experiments. Specifically we note the following findings:

- In Chapter 2, we present a method to automatically remove reflections on the basis of the Depth of Field (DoF). Our approach works based on the observation that most people focus on the background behind the glass when taking photos. By making using of different blur levels brought by this phenomenon, we propose a DoF confidence map to find the background and reflection edges. Based on these edge information, our approach can reconstruct the background images automat-

ically. Due to the lack of benchmark dataset, the performances are evaluated by comparing the visual quality only. However, when compared with previous methods, our method still shows better results and does not need any user-assistances or multiple images to label the edges.

- In Chapter 3, to address the limitations from the lack of benchmark dataset existed in Chapter 2 and previous methods, we propose SIR² — the first benchmark real image dataset for quantitatively evaluating single-image reflection removal algorithms. Our dataset consists of various scenes with different capturing settings. We evaluate state-of-the-art single-image algorithms using different error metrics and compared their visual quality. Then, we thoroughly analyze the limitations of existing methods and propose possible ways to solve these problems.
- In Chapter 4, to solve the limitations discussed in Chapter 3, we propose a method to remove reflections based on retrieved external patches by combining the sparsity prior and the nonlocal image prior into a unified framework. In our framework, the sparsity prior is responsible for the background image reconstruction and the non-local prior can learn the correlations in the images. Compared with previous method, due to the introduction of the non-local prior information, our method do not have special requirement for the properties of the background layer and the reflection layer, *e.g.* the different blur levels of the two layers. In this method, we refine the sparse coefficients learned from the mixture images with the external patches to generate a more accurate sparse regularization term. Experimental results have already shown that our method outperforms the current state-of-the-art methods both from the quantitative evaluations and visual quality.
- In Chapter 5, we revise the method proposed in Chapter 4 by replacing the external sources with the internal sources to find the non-local image prior information from the input mixture image itself. On the other hand, different from previous

methods that process every part of the input images, we introduce reflection dominant regions to efficiently remove reflections in some specific regions and avoid artifacts in the reflection non-dominant regions. We integrate the content prior and gradient prior into a unified region-aware framework to take account of both content restoration and reflection suppression. By refining the sparse coefficients learned from the mixture images with the reference patches, our method can generate a more accurate sparse regularization term to reconstruct the background images. We show better performances than state-of-the-art methods for both the quantitative and visual qualities.

- Though the methods proposed in previous chapters show some promising results, they can only solve this problem in some specific scenarios due to non-learning priors they use. In Chapter 6, to increase the generation ability, we present a concurrent deep learning based framework to effectively remove reflection from a single image. Unlike the conventional pipeline that regards the gradient inference and image inference as two separate processes, our network unifies them as a concurrent framework, which integrates high-level image appearance information and multi-scale low-level features. Thanks to the newly collected real-world reflection image dataset and the corresponding training strategy, our method shows better performance than state-of-the-art methods for both the quantitative values and visual qualities and it is verified to be effectively generalized to other unseen data.

7.2 Future Works

In this section, we explain some future directions stemming from our existing researches. Though the experiments in Chapter 6 have proved the success of deep learning to solve this problem, deep learning methods require large number of training data to

optimize the whole network. On the other hand, the deep learning technique is quite sensitive to very small nuances of the training data. Any discrepancies between the data for training and for inference can lead to very large errors in the final results. Thus, the non-learning based methods can still play key roles in some situations where the training data is difficult to obtain. In this section, we first present some suggestions for the non-learning based method and then we discuss the future directions for the deep learning based methods.

Non-learning based methods. For non-learning based methods, it is very important to find suitable priors for the specific problems. Existing priors mainly focus on the low-level properties of an image (*e.g.*, sparsity and GMM prior), while ignoring the context of an image. It leads to some unrealistic results. Even for our method proposed in Chapter 5 and Chapter 4, though they have considered the context information of the input image, they still simply make use of the non-local correlations in the images. Future non-learning based methods may investigate high-level "semantic" priors in addition to low-level priors to produce more naturally looking images.

Deep learning based method The deep learning framework has become the de facto standard for different computer vision tasks. However, based on the experiments in Chapter 6, we find that the generalization ability of the training dataset plays a key role in the performances of the deep learning based methods. Though our method in Chapter 6 has proposed 'RID' dataset to circumvent the limitations existed in previous method [10], it still has several limitations since we do not consider the spatially-varying coefficients when we generate the synthetic images. Future methods can improve the performances by taking the spatially-varying coefficients into considerations to generate better training dataset.

On the other hand, future methods should circumvent the limitations from the two-stage framework brought by the non-learning based methods. Though we have proposed

a concurrent framework to address the limitations existed in the two-stage framework, it is still just a reformulation of the two-stage framework and adopts low-level gradient priors used in the traditional methods. Based on the modeling ability of deep learning shown in other computer vision tasks, it is reasonable to believe that more high level semantic information brought by deep learning can largely improve the final performances.

At last, the deep learning based method should pay more attention to the regional properties of reflections. As we discussed in previous chapters, different from other ‘noise’ (*e.g.*, rain and haze), the reflections only cover some isolated parts of an image. Though our method in Chapter 5 has proposed a method to locate the reflection dominant regions, it is still based on some heuristic observations, which are not applicable in many scenarios. Future deep learning based methods can propose better reflection localization methods based on the generalization ability of deep learning.

We believe that the waves push forward waves. The younger generalizations who devote themselves to this problem can propose better methods by using more advanced techniques.

Author's Publications

Journal Papers

1. **Renjie Wan**, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, Wen Gao and Alex C. Kot, “Region-aware reflection removal with unified content and gradient priors”, IEEE Transactions on Image Processing.
2. **Renjie Wan**, Boxin Shi, Haoliang Li, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot, “CoRRN: Multi-Scale guided Cooperative Reflection Removal Network”, submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (under major revision).

Conference Papers

1. Haoliang Li, Sinno Jialin Pan, **Renjie Wan**, and Alex C. Kot, “Heterogeneous Transfer Learning via Deep Matrix Completion with Adversarial Kernel Embedding ”, To appear in Proceedings of 33rd AAAI Conference on Artificial Intelligence (AAAI-19), 2019.
2. **Renjie Wan**, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot, “CRRN: Multi-Scale Guided Concurrent Reflection Removal Network”, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
3. **Renjie Wan**, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot, “Benchmarking Single-Image Reflection Removal Algorithms”, in Proceedings of the Inter-

-
- national Conference on Computer Vision (ICCV), 2017.
4. **Renjie Wan**, Boxin Shi, Ah-Hwee Tan, and Alex C. Kot, "Sparsity based reflection removal using external patch search", in Proceedings of IEEE International Conference on Multi and Expo (ICME), 2017. (Oral)
 5. **Renjie Wan**, Boxin Shi, Ah-Hwee Tan, and Alex C. Kot, "Depth of field guided reflection removal", in Proceedings of IEEE International Conference on Image Processing (ICIP), 2016. (Oral)

Bibliography

- [1] Y. Li and M. S. Brown, “Single image layer separation using relative smoothness,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [2] Y. Shih *et al.*, “Reflection removal using ghosting cues,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3193–3201.
- [3] N. Arvanitopoulos, R. Achanta, and S. Susstrunk, “Single image reflection suppression,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [4] Y. Li and M. S. Brown, “Exploiting reflection change for automatic reflection removal,” in *Proc. International Conference on Computer Vision (ICCV)*, 2013.
- [5] H. Farid and E. H. Adelson, “Separating reflections and lighting using independent components analysis,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 1999.
- [6] A. Agrawal *et al.*, “Removing photography artifacts using gradient projection and flash-exposure sampling,” *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 24, no. 3, pp. 828–835, 2015.
- [7] A. Agrawal, R. Raskar, and R. Chellappa, “Edge suppression by gradient field transformation using cross-projection tensors,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2006.

- [8] Y. Y. Schechner, N. Kiryati, and R. Basri, "Separation of transparent layers using focus," *Springer International Journal of Computer Vision*, 2000.
- [9] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, 2007.
- [10] Q. Fan *et al.*, "A generic deep architecture for single image reflection removal and image smoothing," *arXiv preprint arXiv:1708.03474*, 2017.
- [11] R. Wan *et al.*, "Depth of field guided reflection removal," in *Proc. International Conference on Image Processing (ICIP)*, 2016.
- [12] T. Xue *et al.*, "A computational approach for obstruction-free photography," *ACM Transactions on Graphics*, vol. 34, no. 4, p. 79, 2015.
- [13] J.-S. Park *et al.*, "Glasses removal from facial image using recursive error compensation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 805–811, 2005.
- [14] T. Sandhan and J. Y. Choi, "Anti-glare: Tightly constrained optimization for eyeglass reflection removal," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1241–1250.
- [15] 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.
- [16] L. YU, "Separating layers in images and its applications," Ph.D. dissertation, 2015.
- [17] R. Wan *et al.*, "Sparsity based reflection removal using external patch search," in *Proc. International Conference on Multimedia and expo (ICME)*, 2017.

- [18] 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.
- [19] A. Agrawal *et al.*, “Removing photography artifacts using gradient projection and flash-exposure sampling,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 828–835, 2005.
- [20] L. Xu, S. Zheng, and J. Jia, “Unnatural l0 sparse representation for natural image deblurring,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 1107–1114.
- [21] R. Fergus *et al.*, “Removing camera shake from a single photograph,” *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 25, no. 3, pp. 787–794, 2006.
- [22] A. Levin, A. Zomet, and Y. Weiss, “Separating reflections from a single image using local features,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2004.
- [23] A. Levin, A. Zomet, and Y. Weiss, “Learning to perceive transparency from the statistics of natural scenes,” in *Proc. Conference on Neural Information Processing Systems (NIPS)*, 2002.
- [24] A. Levin and Y. Weiss, “User assisted separation of reflections from a single image using a sparsity prior,” in *Proc. European Conference on Computer Vision (ECCV)*, 2004.
- [25] Y.-C. Chung *et al.*, “Interference reflection separation from a single image,” in *Proc. IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2009.

- [26] Q. Yan, Y. Xu, and X. Yang, "Separation of weak reflection from a single superimposed image using gradient profile sharpness," in *Proc. International Symposium on Circuits and Systems (ISCAS)*, 2013.
- [27] Q. Yan *et al.*, "Separation of weak reflection from a single superimposed image," *IEEE Signal Processing Letter*, vol. 21, no. 21, pp. 1173–1176, 2014.
- [28] P. Chandramouli, M. Noroozi, and P. Favaro, "Convnet-based depth estimation, reflection separation and deblurring of plenoptic images," in *Proc. Asian Conference on Computer Vision (ACCV)*. Springer, 2016, pp. 129–144.
- [29] 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.
- [30] X. Guo, X. Cao, and Y. Ma, "Robust separation of reflection from multiple images," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [31] Y. Li and M. S. Brown, "Exploiting reflection change for automatic reflection removal," in *Proc. International Conference on Computer Vision (ICCV)*, 2013.
- [32] C. Szegedy *et al.*, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *AAAI*, 2017.
- [33] C. Sun *et al.*, "Automatic reflection removal using gradient intensity and motion cues," in *Proc. of ACM Multimedia*, 2016.
- [34] 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.

- [35] 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.
- [36] Y. Diamant and Y. Y. Schechner, “Overcoming visual reverberations,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2008.
- [37] B. Sarel and M. Irani, “Separating transparent layers through layer information exchange,” in *Proc. European Conference on Computer Vision (ECCV)*, 2004.
- [38] 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.
- [39] B. Sarel and M. Irani, “Separating transparent layers of repetitive dynamic behaviors,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [40] Q. Wang *et al.*, “Automatic layer separation using light field imaging,” *arXiv preprint arXiv:1506.04721*, 2015.
- [41] P. Kalwad *et al.*, “Reflection removal in smart devices using a prior assisted independent components analysis,” in *Electronic Imaging*. SPIE, 2015, pp. 940 405–940 405.
- [42] O. Le Meur, T. Baccino, and A. Roumy, “Prediction of the inter-observer visual congruency (iovc) and application to image ranking,” in *Proceedings of the 19th ACM International conference on Multimedia, 2011*, pp. 373–382.
- [43] R. Wan *et al.*, “Benchmarking single-image reflection removal algorithms,” in *Proc. International Conference on Computer Vision (ICCV)*, 2017.
- [44] R. Wan *et al.*, “Region-aware reflection removal with unified content and gradient priors,” *IEEE Transactions on Image Processing*, 2018.

- [45] R. Wan *et al.*, “CRRN: Concurrent multi-scale guided reflection removal network.” *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [46] E. H. Adelson *et al.*, “Pyramid methods in image processing,” *RCA engineer*, vol. 29, no. 6, pp. 33–41, 1984.
- [47] U. Rajashekar and E. P. Simoncelli, “Multiscale denoising of photographic images,” in *The Essential Guide to Image Processing*. Elsevier, 2009, pp. 241–261.
- [48] J. Shi, L. Xu, and J. Jia, “Discriminative blur detection features,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [49] Y.-M. Baek *et al.*, “Color image enhancement using the laplacian pyramid,” in *Pacific-Rim Conference on Multimedia*. Springer, 2006, pp. 760–769.
- [50] X. Hou and L. Zhang, “Saliency detection: A spectral residual approach,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8.
- [51] T. Xue *et al.*, “A computational approach for obstruction-free photography,” *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, p. 79, 2015.
- [52] R. Grosse *et al.*, “Ground truth dataset and baseline evaluations for intrinsic image algorithms,” in *Proc. International Conference on Computer Vision (ICCV)*, 2009.
- [53] B. Shi *et al.*, “A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3707–3716.
- [54] Y. Y. Schechner, J. Shamir, and N. Kiryati, “Polarization and statistical analysis of scenes containing a semireflector,” *JOSA A*, vol. 17, no. 2, pp. 276–284, 2000.
- [55] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *Proc. European Conference on Computer Vision (ECCV)*, 2006.

- [56] A. Ninassi *et al.*, “On the performance of human visual system based image quality assessment metric using wavelet domain,” in *SPIE Conference Human Vision and Electronic Imaging XIII*, 2008.
- [57] S.-H. Sun, S.-P. Fan, and Y.-C. F. Wang, “Exploiting image structural similarity for single image rain removal,” in *Proc. International Conference on Image Processing (ICIP)*, 2014.
- [58] W. Dong *et al.*, “Nonlocally centralized sparse representation for image restoration,” *IEEE Transactions on Image Processing*, 2013.
- [59] Y. Li *et al.*, “Learning parametric distributions for image super-resolution: Where patch matching meets sparse coding,” in *Proc. International Conference on Computer Vision (ICCV)*, 2015.
- [60] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image processing*, vol. 15, no. 12, pp. 3736–3745, 2006.
- [61] B. Shen *et al.*, “Image inpainting via sparse representation,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 697–700.
- [62] J. Yang *et al.*, “Image super-resolution via sparse representation,” *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [63] V. Abolghasemi, S. Ferdowsi, and S. Sanei, “Blind separation of image sources via adaptive dictionary learning,” *IEEE Transactions on Image Processing*, 2012.
- [64] G. Peng and W. Hwang, “Reweighted and adaptive morphology separation,” *SIAM SIIMS*, 2014.

- [65] M. Elad and I. Yavneh, “A plurality of sparse representations is better than the sparsest one alone,” *IEEE Transacation on Information Theory*, 2009.
- [66] W. Dong *et al.*, “Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization,” *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp. 1838–1857, 2011.
- [67] J. Philbin *et al.*, “Object retrieval with large vocabularies and fast spatial matching,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2007.
- [68] X. Zhang, “Matrix analysis and applications,” *Tsinghua and Springer Publishing house, Beijing*, 2004.
- [69] D. Krishnan and R. Fergus, “Fast image deconvolution using hyper-laplacian priors,” in *Proc. Conference on Neural Information Processing Systems (NIPS)*, 2009.
- [70] H. Zhang *et al.*, “Close the loop: Joint blind image restoration and recognition with sparse representation prior,” in *Proc. International Conference on Computer Vision (ICCV)*, 2011.
- [71] Q. Yan *et al.*, “Separation of weak reflection from a single superimposed image,” *IEEE Signal Processing Letters*, vol. 10, no. 21, pp. 1173–1176, 2014.
- [72] T. Chen *et al.*, “Total variation models for variable lighting face recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1519–1524, 2006.
- [73] D. Geman and C. Yang, “Nonlinear image recovery with half-quadratic regularization,” *IEEE Transactions on Image Processing*, vol. 4, no. 7, pp. 932–946, 1995.
- [74] J. Pan *et al.*, “ l_0 -regularized intensity and gradient prior for deblurring text images and beyond,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 342–355, 2017.

- [75] J. Pan *et al.*, “Deblurring text images via l0-regularized intensity and gradient prior,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [76] L. Qu *et al.*, “Deshadownet: A multi-context embedding deep network for shadow removal,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [77] P. Qiao *et al.*, “Learning non-local image diffusion for image denoising,” *arXiv preprint arXiv:1702.07472*, 2017.
- [78] E. Luo, S. H. Chan, and T. Q. Nguyen, “Adaptive image denoising by targeted databases,” *IEEE Transactions on Image Processing*, 2015.
- [79] C. Deledalle, V. Duval, and J. Salmon, “Non-local methods with shape-adaptive patches (nlm-sap),” *Journal of Mathematical Imaging and Vision*, vol. 43, no. 2, pp. 103–120, 2012.
- [80] X. Fu *et al.*, “Removing rain from single images via a deep detail network,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [81] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [82] W.-S. Lai *et al.*, “Deep laplacian pyramid networks for fast and accurate super-resolution,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [83] J. Snell *et al.*, “Learning to generate images with perceptual similarity metrics,” in *Proc. International Conference on Image Processing (ICIP)*.
- [84] H. Li *et al.*, “Unsupervised domain adaptation for face anti-spoofing,” *IEEE Transactions on Information Forensics and Security*, 2018.
- [85] T.-Y. Lin *et al.*, “Microsoft coco: Common objects in context,” in *Proc. European Conference on Computer Vision (ECCV)*, 2014.

- [86] M. Everingham *et al.*, “The pascal visual object classes (voc) challenge,” *Springer International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.
- [87] N. Liu and J. Han, “Dhsnet: Deep hierarchical saliency network for salient object detection,” in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [88] Y. Kim, I. Hwang, and N. I. Cho, “A new convolutional network-in-network structure and its applications in skin detection, semantic segmentation, and artifact reduction,” *arXiv preprint arXiv:1701.06190*, 2017.
- [89] T.-W. Hui, C. C. Loy, and X. Tang, “Depth map super-resolution by deep multi-scale guidance,” in *Proc. European Conference on Computer Vision (ECCV)*, 2016.
- [90] Z. Wang *et al.*, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [91] H. Zhao *et al.*, “Loss functions for image restoration with neural networks,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 47–57, 2017.
- [92] K. Zhang *et al.*, “Learning deep CNN denoiser prior for image restoration,” *arXiv preprint arXiv:1704.03264*, 2017.
- [93] C. Dong, C. C. Loy, and X. Tang, “Accelerating the super-resolution convolutional neural network,” in *Proc. European Conference on Computer Vision (ECCV)*, 2016.
- [94] W. Yang *et al.*, “Joint rain detection and removal from a single image,” *arXiv preprint arXiv:1609.07769*, 2016.