IMAGE REFLECTION REMOVAL BASED ON IMAGE GRADIENT REGENERATION

TINGTIAN LI

PhD

The Hong Kong Polytechnic University

2020

The Hong Kong Polytechnic University

Department of Electronic and Information Engineering

IMAGE REFLECTION REMOVAL BASED ON IMAGE GRADIENT REGENERATION

TINGTIAN LI

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

July 2019

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____

Tingtian Li

_____

Abstract

In daily photography, it is common that we take pictures through a semi-reflective material (such as glass) and obtain images with a reflection of another unwanted scene. Reflection does not only degrade the visual quality of the captured images but also affects the subsequent applications of the images, such as recognition. Therefore, the methods for removing the reflection in images has attracted much attention from hobbyists to professionals in photography.

However, reflection removal is a challenging and severely ill-posed problem. It is because we need to solve two unknowns (background and reflection) from only one observation (the captured image with reflection). Due to the ill-posedness of the problem, traditional reflection removal methods often introduce different priors of the background and reflection for constraining the problem. Since the background and reflection images have very similar morphological properties, those priors are only valid in some specific situations. Whenever the prior is not valid, the residues of the reflection will appear in the resulting image and degrade the image quality. In this thesis, we propose a novel strategy for reflection removal. Rather than following the existing approaches in searching for a perfect prior that can accurately distinguish the background and reflection in all situations, we believe it is more realistic and effective to look for a remedial strategy in case the separation is unsuccessful. The proposed background gradient regeneration strategy suggests to firstly remove the reflection components in an aggressive manner even in the

expense of losing some of the background components. The missing background components are then regenerated based on the remaining ones using different estimation methods. As shown in our experiments, such a strategy can lead to fewer reflection residues in the reconstructed background image and the resulting algorithms are more robust in general imaging environments.

Based on this strategy, three reflection removal algorithms are proposed in this thesis. The first algorithm is for the situation that the light field (LF) images are available. It first estimates the depths along the image edges using the LF epipolar plane image (EPI). Based on the edge depths, we identify the background edges in the condition that they have a distinct depth difference from the reflection edges. For those edges that cannot be confidently classified, they will be ignored and iteratively regenerated using a Markov Random Field (MRF) method. The final background image is reconstructed using another iterative optimization process when all the background edges are regenerated.

Although this method is effective, the required iterative optimization processes are time-consuming. For improving the computation speed, we propose the second deep neural network (DNN) based method using multi-view images. The second proposed algorithms have a similar framework as the first one. The major difference is their implementation backbone. The proposed DNN-based method firstly estimates the edge depths using a convolutional neural network (CNN). The background edges are identified following a similar approach as the first method.

Then, a generative adversarial network (GAN) is used to regenerate the missing background edges. Finally, the background image is reconstructed based on the estimated background edges using another CNN. Comparing with the first approach, the deep learning-based method can increase the speed by over 1,000 times when running with a Graphics Processing Unit (GPU) without sacrificing the image quality.

In practice, we often need to deal with the reflection removal problem given only a single image of the scene. Therefore, we also propose the third method that only requires a single input image. With a single image, it is more difficult to achieve an accurate estimation of the edge depths. To solve the problem, we make use of a prior that many traditional approaches have used, that is, the reflection images are often blurry. Such prior is valid in many practical situations since background and reflection components often reside in different depth ranges. A camera focuses on the background is likely to have the reflection out-of-focused and leads to the blurry reflection image. Following the background gradient regeneration strategy, we firstly train a CNN to aggressively remove the blurry components in the image, which are likely the reflection components. Such aggressive strategy will also remove some background edges as well. Then, based on the resulting image, we derive a reflection edge confidence map. We use the map to obtain the background edges with high confidence and regenerate the missing ones using a GAN. The background image is also reconstructed at the same time. The proposed algorithm

gives state-of-the-art performance compared with the existing single-image DNN-based approaches. Similar to the second proposed approach, the proposed algorithm just needs a couple of seconds to complete the task of reflection removal when implementing with GPUs. The algorithm is particularly suitable to those images with blurry reflection, which is not uncommon in practice.

Overall, we show in this thesis that the proposed reflection removal methods using the background gradients regeneration strategy can achieve more robust and better performance compared to the traditional reflection removal methods. In particular, the proposed deep learning-based algorithms have provided real-time performance due to their high computational efficiencies when implementing with GPUs. We believe that the research results of this work have significantly contributed to the field of study and will arouse great interests from the digital imaging industry.

LIST OF PUBLICATIONS

1. **Tingtian Li**, and Daniel P.K. Lun, "Single-Image Reflection Removal via a Two-Stage Background Recovery Process," *IEEE Signal Processing Letters (IEEE SPL)*, 2019.
2. **Tingtian Li**, Yuk-Hee Chan, and Daniel P.K. Lun, "Improved multiple-image based reflection removal algorithm using deep neural networks," *IEEE Transactions on Image processing (IEEE TIP)*, 2019. (under review)
3. **Tingtian Li**, and Daniel P.K. Lun, "Image reflection removal using the Wasserstein generative adversarial network", *IEEE International Conference on Acoustics, Speech, and Signal Processing (IEEE ICASSP)*, 2019.
4. **Tingtian Li**, Daniel P.K. Lun, Yuk-Hee Chan, and Budianto, "Robust reflection removal based on light field imaging", *IEEE Transactions on Image processing (IEEE TIP)*, vol. 28, pp. 1798-1812, 2019.
5. **Tingtian Li**, and Daniel P.K. Lun, "A novel reflection removal algorithm using the light field camera*," IEEE International Symposium on Circuits and Systems (IEEE ISCAS)*, 2018. (one of the best student paper award top ten finalists)
6. **Tingtian Li**, and Daniel P.K. Lun, "Salient object detection using array images," *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2017.
7. **Tingtian Li**, and Daniel P.K. Lun, "Super-resolution imaging with occlusion removal using a camera array*," IEEE International Symposium on Circuits and Systems (IEEE ISCAS)*, 2016.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my great gratitude to my supervisor Dr. Daniel P.K. Lun for his continuous guidance, suggestions, generosity, and patience through my PhD study. Through talks and discussions with him, I learned the skills of conducting research, making a presentation and writing a research paper. His suggestions always help me identify my research weaknesses and improve my research qualities. His wisdom will also influence my entire rest of life.

Besides, I am also very grateful to Prof. Wan-Chi Siu for sharing his views and comments on presentations of research students in weekly group meetings and dinners. In the group meetings, I practiced my presentation skills and broadened my research visions.

I am also grateful to my supportive lab members at the Centre of Multimedia Signal Processing and peers during my PhD study: Wai Lam Hui, Budianto, Siu Chak Cheung, Wei Kuang, Zhi-Song Liu, Li-Wen Wang, Chu Tat Li, Sik Ho Tsang, Meng Yao, Xue-Fei Yang, Huan Dou, Chao Shi, Tsz-Kwan Lee, Huiling Zhou, Qiuliang Ye, Jun Xiao, Xiuyuan Wang, Yikun Pan. Thanks for sharing their knowledge and making my PhD study memorable.

I also would like to thank my family for inspiring me to dedicate myself to my study. Their support and love are important for the successful finish of my PhD study.

Table of Contents

List of Figures

List of Tables

Chapter 1.

Introduction

In daily photography activities, it is common to image through a semi-reflecting material such as glass. In this scenario, the reflection of an unwanted scene is often found in the captured image, which degrades the image quality as well as the subsequent image analysis. Traditionally, photographers may install a polarizer before the camera lens for reducing the reflection. However, a polarizer can only remove the reflection components with an incident angle equal to the Brewster angle [1]. Real-life reflections can come from different sources and different angles, hence cannot be totally removed by a polarizer.

Alternatively, the reflection can be removed using image processing methods. The reflection removal problem is a typical blind image separation (BIS) problem. The problem can be illstruated as in Fig 1.1. The scenes behind and in front of a glass are captured by the camera as a superimposed image. The target is to sperate

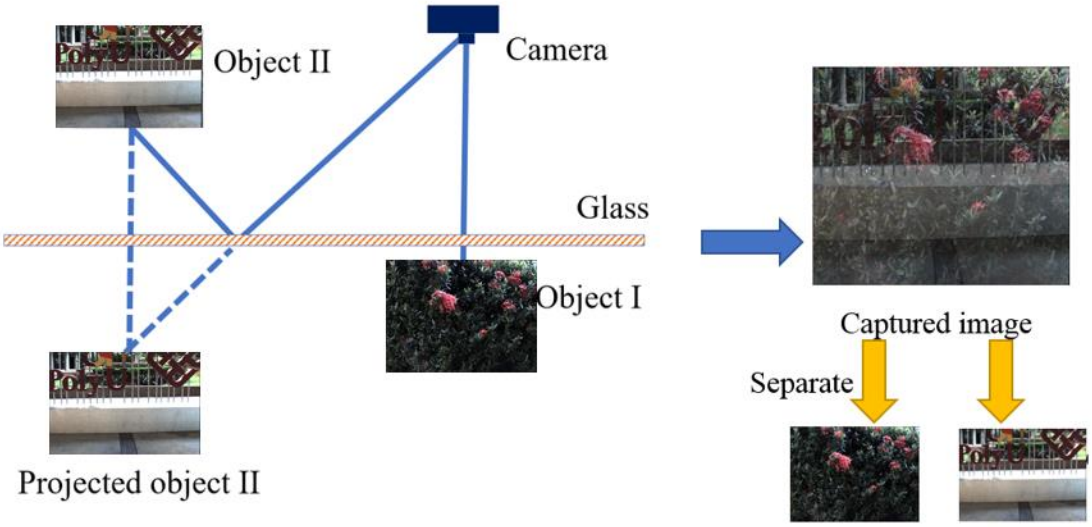


Fig. 1.1. An illustration of the formation of images with reflections. Our target is to separate the images of object I and II from the captured image.

the background and reflection images from the captured image. Mathematically, an image I with a reflection scene I_R superimposed on the background scene I_B can be modeled as

$$I = I_B + I_R. \tag{1.1}$$

Decomposing I_B and I_R from I is a severely ill-posed problem because we need to obtain two unknowns from only one equation. In the last two decades, some approaches were proposed using image statistical priors or deep learning approaches to solve this underdetermined problem. However, due to the close morphological properties between the background and reflection, current methods still cannot robustly achieve the task. For better solving this problem, we consider in this thesis using a new strategy namely background gradient regeneration. That is, we first

aggressively remove the reflection components irrespective of the possibility that some of the background components will also be removed. Then we regenerate the lost background gradients for restoring the background image. We propose three different algorithms based on this gradient regeneration strategy. We consider both the cases that multi-view images and only a single image of the scene are available. The backbones of these approaches include conventional optimization methods and the recently popular convolutional neural networks (CNN) [2-6].

1.1. The Background Gradient Regeneration Strategy

A glass erected between a camera and the target scene acts as a semi-transparent mirror. It reflects the scene in front of the glass while transmitting the background scene behind it. The image thus captured will contain both the background and reflection scenes superimposed to each other. Directly separating them is very difficult. For this reason, current methods often impose different priors to help constraint the problem. Since both the background and reflected scenes are natural, only some weak assumptions on the properties of the background and reflection images can be made when constructing these priors. For example, methods [7, 8] assume only the background is focused and identify those defocused and blurry components as reflection components. Other assumptions include the different responses to motions [9-11], differences in depth ranges [12], etc. of the background and reflection images. Even with these constraints, we can often find in the results

of the current approaches that many reflection residues still remain in the image. It is due to the huge variety of daily captured images that no assumption can be valid in all situations. Rather than continuously searching for a perfect prior as in the previous approaches, we propose using a novel gradient regeneration strategy to remove the reflection. In this strategy, we first aggressively remove the reflection components to ensure that no reflection residue will remain in the image. However, it may mistakenly remove the background components. Therefore, we carry out the second step to recover the lost background components and finally reconstruct the background image.

The regeneration step obviously is the most difficult part of the strategy. As shown in (1.1), the ill-posedness of the problem renders directly regenerating the lost background pixels very difficult, even with the initially underestimated background as a hint. However, due to the sparsity of image edges [13-16], the edges of two uncorrelated images are seldom overlapped. It means that the background and reflection images, which are often uncorrelated, will have their edges at different positions. It greatly simplifies the regeneration process. Therefore, we carry out the background component regeneration in the gradient domain instead of the spatial domain. With the hint of the initially underestimated background edges, we can identify the remaining background edges by utilizing their spatial and statistical relationships. Lastly, the background image can be extracted from the original image guided by the identified background edges.

The background gradient regeneration strategy has the merit that it can give more robust performance even when the background and reflection images have similar statistical distributions. In this thesis, we propose three approaches based on this gradient regeneration strategy. The first and second ones are multiple image-based while the third one is single-image based. The difference between the first and second approaches is that the first one utilizes the traditional optimization methods while the second one utilizes the deep neural networks (DNN) for all estimation processes. In all these proposed methods, the background gradient regeneration strategy contributes significantly to the effective and robust removal of the reflection components in the image.

1.2. Contributions of this thesis

1.2.1. A Novel Method Using Light Field Images for Robust Reflection Removal

Traditional multiple-image reflection removal methods can only work well under stringent scenarios, such as restrictive environments [9], weak reflection intensities [10] or with guided initializations [11]. Another problem of the traditional multiple-image methods is that they need to take pictures in different views sequentially. These methods cannot deal with dynamic scenes where the objects are moving. To

solve this problem, it was suggested in [12] using a light field (LF) camera [17, 18] to capture the multiple images simultaneously. However, the method has some stringent requirements on the imaging environment such as the background and reflection must have absolutely different disparity ranges and the camera orientation must be perpendicular to the reflecting surface. They introduce much difficulty in actually using the algorithm. In the first part of this thesis, we propose a novel reflection removal method based on the background gradient regeneration strategy using light field images. This method has no requirement on the disparity ranges of the background and reflection images or the camera orientation. To summarize, the main contributions of this method, which will be further described in Chapter 3, are as follows:

1. We explore the theoretical support of using LF epipolar plane images (EPI) to estimate the disparities of different layers of an LF image with reflection. We verify that if an LF image is formed by the superimposition of two LF image layers of different disparities, the EPI strong gradient points of both images will be at different positions of the combined EPI and the gradient values will be preserved. We can use them to identify the positions of the background and reflection strong gradients as well as their depths with no ambiguity.

2. We propose a general sandwich model to describe the depth ranges of the background and reflection images. The model allows a shared depth range for both images which is more realistic in practical situations. Following this model,

the proposed method does not require the background and reflection images to have absolutely different depth ranges as in the existing approaches. An aggressive approach is implemented to separate the background and reflection gradients based on their depths. As a result, we obtain some background gradients with high confidence, although the ones with less confidence will be ignored and removed.

3. To detect and regenerate the background gradients which are removed due to the aggressive process as mentioned above, a new algorithm is developed based on an observation that these gradients can be found in the initial background estimate and its residue.

1.2.2. Deep Learning Based Robust Reflection Removal Using Multiple Images

Although the above-proposed method can show robust performance compared to the previous approaches, the use of the traditional optimization processes on large matrices is rather time-consuming. Also, the light field images can require a large memory space [19]. Considering the recent successes of deep neural network in solving inverse problems [20-24] and its fast speed owing to the parallel feedforward structure, some recent methods [7, 25] try to remove reflection using deep learning approaches. However, those methods can only deal with the reflection that is blurry, which is not the case in many practical situations. For better solving

the problem, we propose a novel multiple-image reflection removal method using DNN. It also uses the background gradient regeneration strategy and has a similar framework as the method described in Section 1.2.1. However, this method has a much faster speed and uses fewer input images. In addition, since DNN can reconstruct images without handcrafted priors, it shows better performance compared to conventional optimization methods. To summarize, the main contributions of this method, which will be further described in Chapter 4, are as follows:

1. We propose a novel deep learning-based framework to solve the ill-posed reflection removal problem. Unlike the traditional DNN-based methods, this approach has no requirement on the properties of the reflection image such as blurry [7, 8, 25], weak intensity [10] or double-reflected [26].

2. Rather than using the LF EPI for estimating the depths of strong gradients, a CNN is trained to achieve the task. As different from the traditional deep learning depth estimation methods, the proposed CNN directly generates the depths of the image edges based on their disparities using an unsupervised training approach. It is not affected by the depth ambiguity due to the superimposition of the background and reflection images with different depths. In addition, it does not require the ground truth depth maps which can be difficult to obtain in practice.

3. We use a Wasserstein Generative Adversarial Network (WGAN) to regenerate the lost background edges due to the initial aggressive reflection removal process. Benefited from the jointly trained adversarial term, WGAN can regenerate the background edges which closely follow the distribution of the ground truth.

4. Instead of using the traditional optimization method, which is time-consuming, a CNN is trained to extract the background image from the original one guided by its edges. By having the three major functional blocks, edge depth estimation, background edge regeneration, and background image extraction, implemented using the deep learning approaches, the proposed algorithm can achieve more than 1,000 times improvement in terms of computation speed over the traditional optimization approaches when implementing with GPUs.

1.2.3. Deep Learning Based Single-Image Reflection Removal Using A Two-Stage Background Recovery Process

In practice, we often need to deal with the reflection removal problem given only a single image of the scene. Therefore, in the last part of this thesis, we propose a deep learning-based reflection removal method using a single input image. The proposed method is also based on the background gradient regeneration strategy. With only one image, the problem becomes far more unconstrained. Similar to other single-image reflection removal methods [7, 25], we also only consider the situations that the reflection is defocused and blurry. The proposed method has two

stages. At the first stage, it aggressively removes the reflection components for improving the reflection suppression ability. At the second stage, we regenerate those background gradients suppressed at the first stage. The experimental results show that this method can better remove the sharp reflection components compared to other single-image DNN-based methods. To summarize, the main contributions of this method, which will be further described in Chapter 5, are as follows:

1. We propose a novel two-stage single-image based reflection removal method using deep learning approaches. We investigate the perceptual feature difference between normal images and those with reflection. Then we propose to include a feature reduction term in the training of the network to aggressively remove the reflection components at the first stage.

2. We use the initially underestimated background to infer a reflection edge confidence map and use it to regenerate the background gradients suppressed at the first stage.

3. We propose a network trained with an adversarial term to extract the background image from the original one (with reflection) guided by its edges.

1.3. Organization of The Thesis

This thesis consists of six chapters. After the introduction in Chapter 1, Chapter 2 gives a literature review related to this thesis. Chapter 3 to 5 present three novel reflection removal algorithms based on the background gradient regeneration strategy as mentioned above. More specifically, Chapter 3 and Chapter 4 present the proposed multiple-image reflection removal algorithms based on the traditional optimization and deep learning approaches, respectively. Chapter 5 presents the proposed single-image reflection removal algorithm using a two-stage process. Chapter 6 draws the conclusion of this thesis and some possible future works along with this thesis are also presented.

Chapter 2.

Literature Review

In this chapter, we review and discuss several existing reflection removal methods ranging from using the traditional polarizing filters [27-29] to state-of-the-art image processing approaches [8-12, 30, 31]. We also discuss the popular deep neural networks and their applications to reflection removal [7, 25]. The inputs of these methods are from a single image [7, 8, 25, 31] to multiple-view images [9-11] (including light field images [12]), or even polarized images [27-29] and flashed images [30].

2.1. Polarizers for Reflection Removal

2.1.1. Reducing Reflection Using Polarizers

Placing a polarizing filter in front of the camera lens is the most traditional way to reduce reflection. The reflection exists when a light beam hits the boundary of

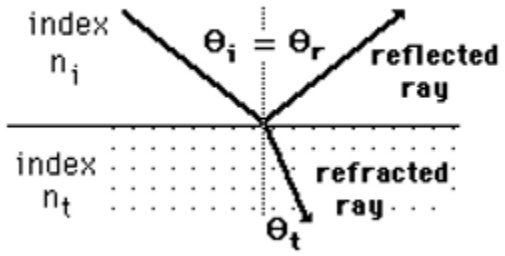


Fig. 2.1. An illustration of how the reflection is generated when a light beam hits the boundary of two media with different refractive indices.

two media with different refractive indices as shown in Fig. 2.1. Some of the light is often reflected while the other penetrates through the boundary. When the light incident angle is equal to the Brewster angle [1], the reflected light will be linearly polarized. If a camera has a polarizer set at an angle perpendicular to the polarized reflected light, the light can be filtered out before reaching the sensor of the camera. Although many photographers make use of this approach, the resulting images often still contain many reflection residues. It is because in a practical situation there can be many light sources in a reflection scene; they can get to the camera at different incident angles. Thus, those not in the Brewster angle will not be polarized and filtered out by the polarizer.

2.1.2. Signal Processing Approaches Using Multiple Polarized Images

While a single polarizer often cannot totally solve the reflection problem, some approaches use multiple polarizers [27-29] to create constrained environments for

solving this problem. For instance, [27] presents an approach that can recover the background layer by classifying background and reflection components based on polarized images captured at two different angles. The classification of different layers is obtained by considering the weighted pixel-wise differences of these polarized images. An inversion process is then performed to reconstruct the final image. In [28], it is suggested that the contribution of reflection can be smoothly reduced when we gradually rotate the angle of a polarizer for planar surface reflection. They use a variable matte to describe this spatially varying contribution of reflection and use it to separate the background and reflection gradients. But for further improving the reflection removal performance, they still need to incorporate an interactive user guide to their optimization framework. On the other hand, an approach that only uses three polarized images is proposed in [29]. It exploits the physical property of polarization applied to a double-surfaced transparent medium and proposes a multiscale scheme to automatically separate background and reflection. Although [27-29] can remove reflection to some certain extents, they impose stringent requirements on the position and orientation of the camera when taking images, which is difficult to achieve in practice. Besides, the requirement of having a static background for completing the imaging process further limits their applications in practical scenarios.

2.2. Reflection Removal Using Flashlights

In [30], an interesting approach which uses a flash and non-flash image pair for reflection removal is proposed. The approach uses a novel gradient coherence model to relate the gradient components in the flash and non-flash images. Based on this gradient coherence model, the reflection components can be removed using a gradient projection method. However, this method requires the hot spot and reflection components in both the flash and non-flash images to be located at different positions. It means that the flashlight must overwhelm all the reflection components, otherwise, some reflection components will stay at the same locations in both the flash and non-flash images. The hot spot in the flash image should also be small for avoiding overlapping with some reflection components. Such stringent requirements render this method not so practical.

2.3. Single−Image Optimization Based Reflection Removal Methods

In the past 20 years, much effort has been made in using image processing methods for reflection removal. These methods try to remove the reflection in an image by solving a blind image separation problem. It is well-known that the blind image separation problem is severely ill-posed since we need to solve two unknowns (background and reflection) based on one observation (the observed

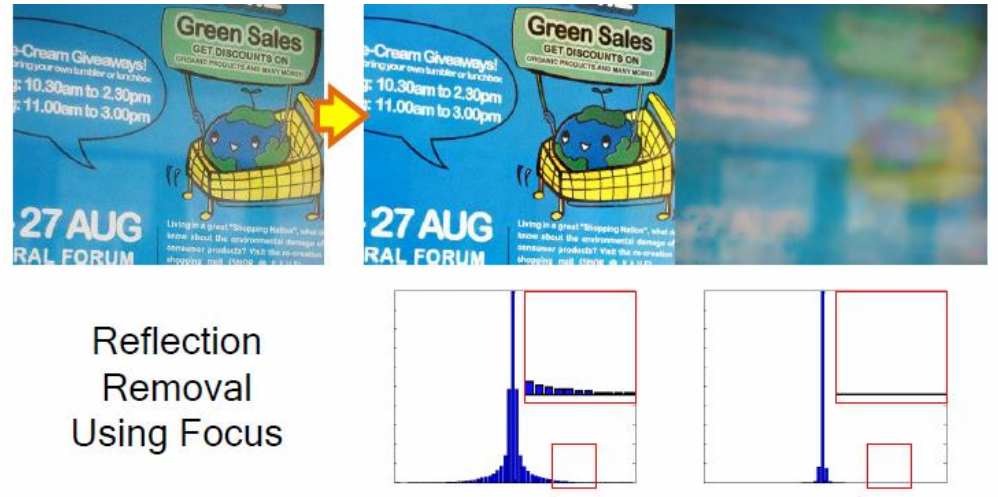
(a) (b) (c) (d) (f)

Fig. 2.2. One example to illustrate the gradient independence property [9]. (a) is obtained by superimposing (d) on (b). (c) and (f) are the strong gradients of (b) and (d) respectively.

image). For constraining this ill-posed problem, these methods usually incorporate various priors such as gradient sparsity, distribution and gradient independence in the optimization process [8, 26, 31].

2.3.1. Gradient Independence Property

It is well-known that strong gradients of natural images are sparse. It gives rise to the gradient independence property which indicates that the strong gradients of two natural images seldom overlap each other. Fig. 2.2 shows an example of the gradient independence property. We can see that, although the image in (a) is obtained by superimposing (d) on (b), the strong gradients of (b) and (d) are independent and at different positions, as shown in (c) and (e). Therefore, it provides us an important prior for solving the ill-posed reflection removal problem in the gradient domain. We can identify the background through its gradients instead of the other ambiguous background pixels. Combined with other priors such as the



Input background image reflection image

Fig. 2.3. The histograms of the background and reflection image gradients [8].

known distributions of the gradients, many single-image based reflection removal methods [8, 26, 31] are developed.

2.3.2. Gradient Distribution Assumptions

[31] assumes the gradients of the background and reflection layers are sparse and follow a mixture of Laplacian distributions. Based on the gradient independence property, they manually label the gradients belonging to different layers for guiding the optimization process to converge. This method can reduce some reflection components. However, the required manually labeling process is time-consuming and the simple gradient distribution model is difficult to fit all natural images.

Due to the observation that background images are often focused and reflection images are often defocused, [8] makes use of two different distributions to describe

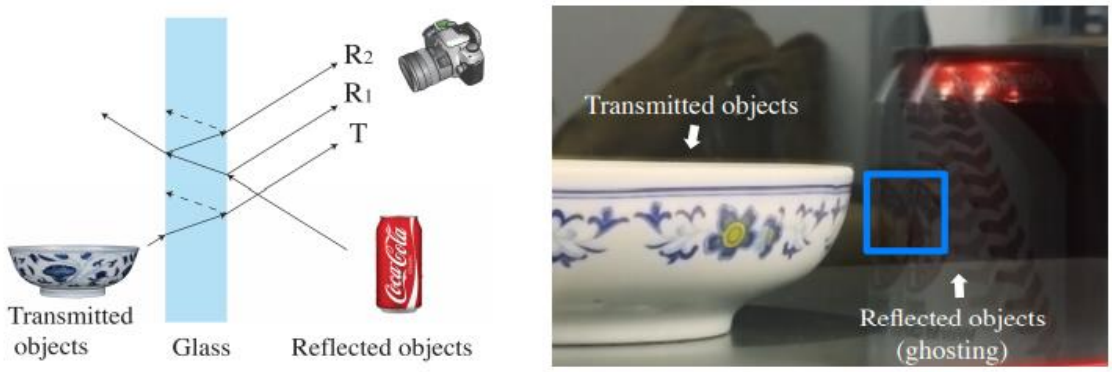
their gradients. Fig. 2.3 shows the histograms of the gradients of a pair of background and reflection images. We can see that the histogram of the focused background gradients is long-tailed, while the one for the blurry reflection is short-tailed. It is because the sharp background has more large-valued gradients. Therefore, [8] defines a long-tailed distribution model below to describe the distribution of the background gradients,

$$P_1(x) = \frac{1}{z} max\{e^{-x^2/\sigma_1^2}, \epsilon\}, \tag{2.1}$$

where x represents the gradient value, z is a normalization factor and σ_1 is a small constant. A short-tailed distribution defined below is used to model the distribution of the reflection gradients,

$$P_2(x) = \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2}{\sigma_2^2}}, \tag{2.2}$$

where σ_1 is a small constant. Compared to (2.2), (2.1) has a minimum boundary ϵ so that $P_1(x)$ will not drop so fast as $P_2(x)$ in (2.2), which leads the long tail in $P_1(x)$. The different gradient distributions provide us an important hint to separate the background and reflection. Similar to [31], these simple distributions are difficult to fit the huge variety of natural scenes, but they inspire the recent single-image deep-learning based approaches [7, 25] to distinguish the background and reflection components using DNN.



(a) (b)

Fig. 2.4. The formation of the ghosting effect [26]. (a) Light rays from the reflection object are partially reflected both by the inner-side and far-side of the glass, which results in two reflection R_1 and R_2. R_2 is the shifted and attenuated version of R_1. (b) The captured image with the ghosting effect.

2.3.3. Reflection Ghosting Cues

Many reflection problems are generated when imaging through a glass. In [26], it is observed that light rays may be partially reflected by both the inner-side and far-side of thick glass. It proposes a ghost cue that uses a double-impulse convolutional kernel to model this double-reflection effect. An example of the ghosting effect is shown in Fig. 2.4. We can see in the figure that a light ray of the reflection object is reflected twice by both the inner-side and far-side of the glass. They form two reflections R_1 and R_2 respectively, where R_2 is the shifted and attenuated version of R_1. This ghosting effect can be mathematically obtained by a double impulse convolution process on R_1. Based on the ghosting effect of the reflection image, [26] proposes an optimization method to separate the background

and reflection images. However, as indicated in [26], not all glasses are so thick to produce an obvious ghosting effect. Furthermore, when the photographer is far from the glass, the ghost effect can be very weak and invisible.

2.4. Multiple-Image Optimization-Based Reflection Removal Methods

As it is very difficult to solve the severely ill-posed reflection removal problem using only one image, researchers proposed to capture more images of the scene at different angles or different times to provide more information for reflection removal [9-12, 32]. These methods assume the background and reflection have distinct properties in these images and utilize these properties for their separation.

2.4.1. Using the 2D Homography

It is often the case that the background and reflection scenes are at different distances from the camera. If we have multiple views of the scene, we can register the background in different views using a homography (assume the background is planar) while the reflection will be misaligned. Based on this idea, [9] uses the differences in 2D homographies of the background and reflection to achieve their separation. One example is shown in Fig. 2.5. When combining the vectorized registered images into a matrix, we can find that the background components appear

to be smooth (since all views are similar). The misaligned reflection components appear to have relatively large variations in pixel values across views. Then when applying a low-rank decomposition to the matrix, the background part which is smooth will reside in the low-rank part. The misaligned reflection components will be separated from the background components and reside in the residual part. However, this method is only suitable for planar backgrounds. For a non-planar background, some background components will also be misaligned and removed since they cannot be registered by a single homography. Another problem of this method is that the reflection features covered on the background image can negatively influence the estimation of the background homography. The accuracy of the estimated background homography is always in doubt.

(a) (b) (c)

Fig. 2.5. A reflection misalignment example [9]. (a) A planar scene with reflection in two views. (b) The aligned background after background planar transformation. (c) The misaligned reflection.

2.4.2. Using SIFT Flow

The subtle differences of the image components in different views can be also described by the SIFT flow [33]. In [10], the SIFT flow is used to register the dominant background components. The authors assume that the background is dominant while the weak reflection edges may not be found in every view. Therefore, the SIFT flow can show high registration accuracy for the background components but low registration accuracy for the reflection components. The pipeline of a reflection removal example [10] is shown in Fig. 2.6. We can see in the figure that the SIFT flow can well register the strong background edges. On the contrary, the SIFT flow fails to register the weak reflection edges. Based on this observation, the background and reflection edges can be distinguished according to

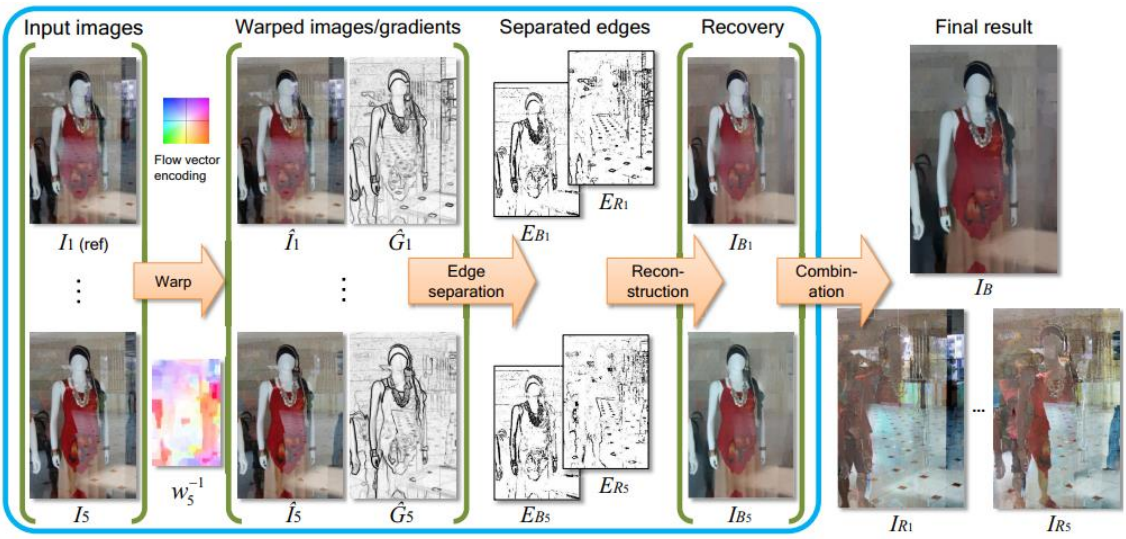


Fig. 2.6. The flow chart of the method in [10].

their alignment extents. After obtaining the background edges in different views, this method reconstructs the background images in different views by several regularization processes. However, this method requires reflection edges to be weak and not appear in every view. If the reflection is also strong, this method tends to leave reflection residuals in the background result.

2.4.3. Using Optical Flow

When taking an image sequence by a camera with a slight motion, the background and reflection components often appear different attributes in the image sequence which can be made use of to facilitate their separation. To register the changes in the background and reflection components across images, optical flows are adopted in [11]. However, as mentioned in [10], the measured optical flow of the background can have very poor accuracy due to the interference of the reflection. Therefore, this method requires a very good initialization to guide the optimization

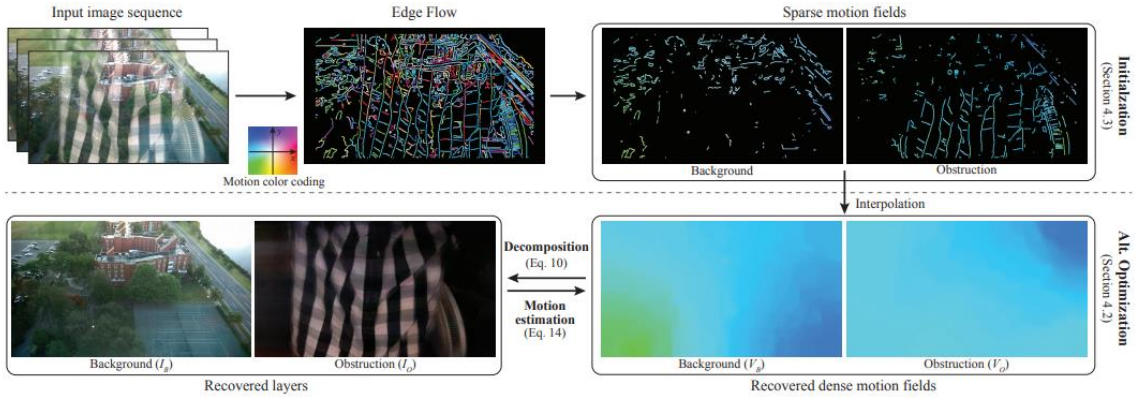


Fig. 2.7. The flow chart of the method in [11]. The method mainly consists of two steps: the initialization and iterative reconstruction.

process to correctly converge. Fig. 2.7 shows the pipeline of this method. In [11], the optical flows are initialized with the two most dominant homographies estimated from the image edges of different views (due to the motion of the camera). However, as mentioned before, a homography can only well register a planar scene. It will lead to huge mistakes when applying to a non-planar scene. Moreover, because there are many variables need to be simultaneously regularized during the optimization process, a wrongly initialized variable can let the estimations of other variables to converge to wrong local minima.

2.4.4. Separating Background Components Using Light Field Images

Recently, light field (LF) images are also used for reflection removal [12]. Using a light field camera, we can capture multiple images of the scene in a single shot. Therefore, this method is also suitable for dynamic scenes. The method in [12] assumes the background and reflection layers have distinct depth ranges. Therefore, a fixed threshold is used to separate the background and reflection components based on their depths. As the method is designed based on the Lytro Illum LF camera, one of the image layers must be within 1.5 meters to the camera and the other is not. Another strict requirement of this method is that the reflecting surface must be perpendicular to the camera. As the depth ranges of the background and reflection are different for various natural scenes, one pre-determined and fixed depth threshold is not possible to correctly separate the background and reflection

in all situations. The requirement on the camera orientation is also not practical since it introduces great limitation to the photography style.

2.5. Deep Convolutional Neural Networks and Its Application for Reflection Removal

2.5.1. Deep Convolutional Neural Networks

The concept of the convolutional neural network was firstly proposed to recognize the handwritten ZIP codes [34] and later for classifying other various objects such as hand-written digits in MNIST [35]. However, at that stage, the performance of CNN still fell behind other classification methods such as Support Vector Machine (SVM) [36]. The main reasons are that the size of the training dataset was not large enough, and the computational power was also not sufficient to train a deep CNN. The situation however changed dramatically in the first decade of the 21st century. The availability of very large training datasets and advanced GPUs with powerful parallel computational ability made possible the training of deep CNNs. Finally, the capability of CNN was widely recognized at the ILSVRC 2012 competition where the AlexNet [3] was proposed. The network was trained on the huge dataset ImageNet with 1.2 million images and won the competition. Other layer architectures such as ReLU [37] and training strategy Dropout [38] were also used for further improving the network performance. After this milestone, deep

CNN was adopted in many other areas, such as object detection [39-45], object segmentation [46-50], motion estimation [51-53], etc.

Recently, CNN is also used in solving inverse problems such as super-resolution [54, 55] and denoising [20, 21]. It is because CNN has a strong ability to learn the mapping from the input images to the ground truth images. Furthermore, advanced CNN structures which are originally used for other applications, such as the skip connections for object classification [4] and image segmentation [48], can also be modified and used for solving inverse problems [20, 21].

2.5.2. Generative Adversarial Networks

Generative adversarial networks (GANs) recently attract much attention and are intensively studied by researchers, although the first GAN (DCGAN [56]) for producing novel image samples was only proposed in 2016. DCGAN can generate novel images following specific distributions. It contains a generator G and a discriminator D. The generator tries to produce fake samples by minimizing a cross-entropy loss function (2.3) as follows:

$$\mathbb{E}_{x\sim P_r}[logD(x)] + \mathbb{E}_{x\sim P_g}\big[log\big(1 - D(x)\big)\big] \tag{2.3}$$

where x is the input, P_r represents the distribution of real samples and P_g represents the distribution of fake samples produced by the generator. The discriminator is jointly trained to distinguish the fake samples from the real samples by maximizing

the loss function (2.3). When the loss function is minimized, the generator can produce fake samples following the distribution of the real samples which cannot be distinguished by the discriminator.

However, the training of DCGAN with the loss function (2.3) is unstable and sometimes even does not converge. It is because when the probability density functions P_r and P_g overlap very slightly or even do not overlap, the gradients of the loss function (2.3) will be close to zero. Such a situation often happens at the beginning of the training when the distribution of the fake samples deviates a lot from the real samples [57]. Therefore, [57] proposes using the Wasserstein distance conditioned by the infimum of the joint distribution of P_r and P_g in their loss function as follows:

$$\mathbb{E}_{x \in P_r}[D(x)] - \mathbb{E}_{x \in P_g}\big[D\big(G(z)\big)\big] \tag{2.4}$$

It also requires the clip of the gradients of the discriminator for fulfilling the requirement of Lipschitz continuity [57] or adding a gradient regularization term [58]. The advantage of using the Wasserstein distance is that even when P_r and P_g have no overlap region, the loss function shows smooth changes which can still provide valid gradients. Therefore, the network can be easier to be trained. Also for conquering the gradient vanishing problem of DCGAN, [59] proposes the Boundary Equilibrium GAN (BEGAN) which also uses the Wasserstein distance but is conditioned by the infimum of the joint distribution of the discriminator responses

of real and fake samples. It further includes an equilibrium to control the trade-off between the reality and variation of the generated data. In [60], the Least Square GAN (LSGAN) is proposed. It uses the least square function instead of the cross-entropy loss function for avoiding the gradient saturation problem. Such modifications in WGAN, BEGAN and LSGAN largely improve the training speed and stability over the original DCGAN, which further promote the use of GAN in different applications. For instance, GAN is also applied for solving the inverse problems due to its ability to promote the perceptual quality of the inferred results [23, 61-63]. It is because the adversarial term of GAN can act as a trainable prior that enforces the results to follow the distribution of ground truth samples. The results can usually show sharper edges and higher perceptual quality.

2.5.3. Reflection Removal Based on Deep Neural Networks

Recently, deep neural networks are also used for solving the reflection removal problem [7, 25, 64]. These methods claim to be able to remove reflection using a single image. The basic idea of these methods is inspired by [8], which assumes the background is focused and the reflection is defocused. Therefore, the reflection becomes blurry and has a different distribution from the background image. The

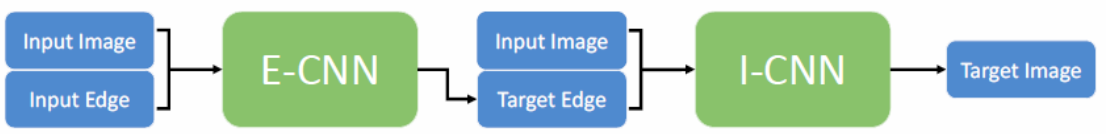


Fig. 2.8. The pipeline of the method [7]. This method first estimates the sharp background edges, then reconstructs the background images from its edges.

sharp background can be identified by a CNN. The pipeline of the method in [7] is shown in Fig. 2.8. The method firstly uses a CNN to identify the sharp background edges and subsequently uses another CNN to reconstruct the background. Similar to [7], [25] tries to distinguish the sharp background components via minimizing the VGG perceptual feature [65] distance between the reconstructed background and sharp natural images. Because the perceptual features of sharp and blurry images are different, this process can drop the blurry components and only keep the sharp components in the image. It also includes an adversarial term inspired by GAN to further improve the fidelity of the results. Another attempt of reflection removal using DNN is reported in [64]. The approach first removes the blurry reflection components using a CNN, then it sends the estimated background image to another network for obtaining a better reflection image. Lastly, this better reflection image is fed to the third network to produce the final background image. However, all these methods are only suitable for the situations that the reflection is defocused. When taking pictures with a small aperture, we can find both the background and

reflection are focused and show sharp edges. Thus, this blurry reflection constraint is not general for all the photography settings. Moreover, even the reflection is out-of-focus, some reflection edges can still have high gradient values. Those high gradient edges can be mistakenly recognized as the background components and produce reflection residuals in the final results.

2.5.4. Training Dataset Synthesization

To train and test the DNN-based algorithms, it is required to have a large set of images with reflections and their background ground truths. Although it is possible to obtain such images and labels using some optical approaches (such as imaging with and without a glass) [25, 66] , it is difficult to obtain many image pairs since the imaging process is very labor-intensive and the scenes have to be static. Therefore, current methods use synthesized images to train their networks [7, 25, 64]. Their synthetization approaches are very similar. In general, they firstly randomly pick two images from a clean image dataset as background and reflection images, and add them together to synthesize an image with reflection. Before the addition, the reflection image is blurred by a Gaussian kernel for simulating the defocused effect and then its intensity will be attenuated by reducing its mean value. Finally, the pixel values which are above the image range after the addition will be clipped. Using such a synthetization process, researchers can produce sufficient data to train their deep neural networks.

2.6. Summary

In this chapter, we reviewed and discussed the principles as well as the shortcomings of the traditional approaches using the polarizing filter and state-of-the-art image processing methods for reflection removal. We also reviewed some basic features of CNN and GAN, which are used in our proposed DNN-based reflection removal methods in Chapter 4 and 5.

The main shortcomings of the methods using polarizing filters are they have some stringent requirements on the camera position, orientation and the background environment, which are difficult to achieve in practice. For those single-image optimization-based image processing methods, they remove the reflection using handcrafted priors, such as gradient sparsity, gradient independence, and gradient distributions. Although they may work in some specific situations such as when the reflection is defocused or the glass is thick, their results can be erroneous in other situations. It is because the handcrafted features cannot well fit the huge variety of natural images and photography situations. For the multiple-image optimization-based methods, they exploit the differences between the background and reflection in homographies and motion which can be obtained when multiple images of the scene are available. However, the models adopted usually can only work in some specific scenarios. They can be erroneous due to the ambiguity introduced by the

superimposition of the background and reflection. They are also not suitable for dynamic scenes since the multiple images are often captured sequentially. For solving this problem, LF based method was recently proposed since multiple views of the scene can be captured in one shot. However, the existing LF approach requires the background and reflection to be in specific depth ranges which limits its application. As to the DNN based reflection removal methods, the current approaches can only work for the images with a focused background and a defocused reflection. In fact, even the reflection is defocused, some reflection edges can still have high gradient values. They will be mistakenly recognized as the background components and kept in the final results.

For better solving the reflection removal problem, we propose three novel algorithms which will be described in detail in the following chapters.

Chapter 3.

Robust Reflection Removal Based on Light Field Imaging

(This chapter is extracted from my paper [67]: Tingtian Li, Daniel P.K. Lun, Yuk-Hee Chan, and Budianto, “Robust reflection removal based on light field imaging”, IEEE Transactions on Image processing, vol. 28, pp. 1798-1812, 2019.)

In this chapter, we propose a novel reflection removal method based on the background gradient regeneration strategy using light field (LF) images. For the proposed algorithm, we first identify the depth of the strong gradient points of the background and reflection using the epipolar plane image (EPI) extracted from the input LF image. Following the background gradient regeneration strategy, only those strong gradient points with distinct depth values will be kept and those of which the depth values are difficult to classify will be removed. They are then regenerated using an iterative estimation process based on their relationship with those strong gradient points that have been classified. The initial estimated background image is then refined using the estimated background gradients.

Experimental results show that the proposed reflection removal algorithm achieves superior performance over the traditional approaches both qualitatively and quantitatively. They verify the robustness of the proposed algorithm when working with images captured from real-life scenes.

3.1. Introduction

It is important to remove the unwanted reflection of an image since it does not only affect the visibility of the background but also introduces ambiguity that perturbs the subsequent analysis on the image. As mentioned before, many optimization-based approaches have been developed and various priors are adopted for solving this unconstrained problem. Most priors that the previous methods adopted are gradient based, such as gradient sparsity and gradient independence [9-11, 31]. The former one is a well-known property of natural images and the latter one is based on the observation that the strong gradients of the background image I_B and reflection image I_R are normally non-overlapped. However, the effect of just adding these priors in the optimization is limited due to the huge variety of natural images. Researchers tend to utilize multiple images of the scene to acquire more information for removing the reflections. The multiple-image based methods, in general, have better performance than the single-image ones. However, these multiple-image methods all have strong assumptions on the property of the

reflection image and/or the imaging environment as discussed in Chapter 2. Besides, all of them require multiple shots of the target scene hence are not suitable for dynamic applications where either the background or reflection objects are moving.

Different from traditional cameras, LF cameras can capture multiple views of a scene in one exposure. Hence, they can be used in dynamic applications. Thanks to the commercialization efforts of Lytro and Raytrix, nowadays people can easily acquire an LF camera with a reasonable cost. Four-dimensional (4D) LF imaging [17] has demonstrated its power in solving various problems like refocusing [18, 68, 69], depth estimation [70-73] and super-resolution imaging [73, 74] in the computer vision area. Quite recently, LF cameras are also used to solve the reflection removal problem [12]. By assuming the background and reflection are at two absolutely different distances from the camera, the method in [12] applies a fixed threshold to separate the background and reflection pixels with respect to their depths. Such assumption, however, is not valid in many practical situations, since the background and reflection can share the same depth range. In this chapter, we first explore the LF EPI [73, 75] and show that its strong gradient points will be preserved after adding to the EPI of another LF image. Such property lets us easily identify the strong gradient points of the background and reflection images, and we can further use them to give a rough estimation of each image layer by a sparse regularization process. To solve the problem that the background and reflection edges can share the same depth range, we propose a sandwich layer model that allows the

background and reflection images to have components sharing the same depth range. Since the strong gradient points in this depth range are difficult to be classified as belonging to the background or reflection, they are removed such that the initial estimates of the background and reflection will have some components missing. We then propose a method which gradually refines the initial background estimate by detecting and recovering the gradients in the shared depth range. It is achieved based on an observation that the initial background estimate and its residue can provide information on the positions of the missing gradients. It gives us the clue to recover these gradients for refining the initial background estimate.

3.2. Using EPI Gradient in Separating Background and Reflection

In this section, we first make a brief review of LF EPI and explain how its gradients can be used in the estimation of the disparity map for the problem of reflection removal.

3.2.1. Layer Classification Based on The EPI Strong Gradient Points

Although there are several approaches to represent the light field, the 4D one which uses two planes to represent the viewpoints and image plane, as shown in Fig. 3.1, is the most popular [17]. In the figure, the planes Π and Ω are the viewpoint plane and image plane respectively. Here we use the coordinate systems (s, t) for Π and (x, y) for Ω. Therefore, we can describe each light ray by a 4D coordinate system (s, t, x, y). If we fix t and y as t^* and y^*, and let s and x vary, we will get the so-called EPI slice Σ_{y^*,t^*}. The slope reciprocal $\Delta x/\Delta s$ at $\Sigma_{y^*,t^*}(x, s)$ can represent the disparity at point (x, y^*) for the view (s, t^*) [73, 76]. Hence the EPI slope is often used to evaluate the disparity, and in turn, the depth of the scene. The slope directions can be obtained using the structure tensor [73, 77, 78], which

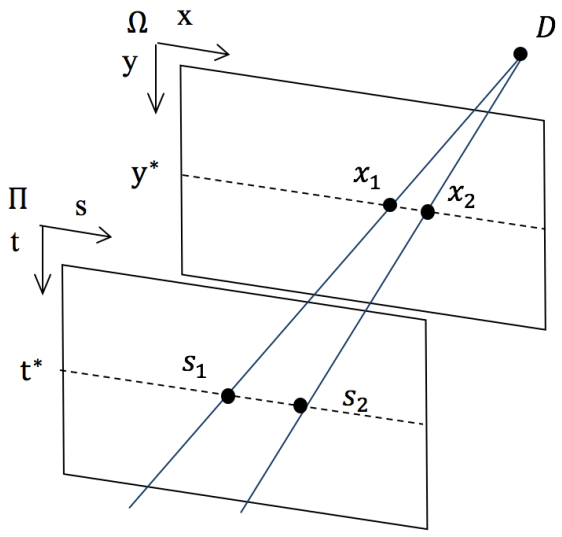


Fig. 3.1. The 4D LF model. A light ray can be described using the 4D coordinates (s, t, x, y).

determines the gradient direction by finding the eigenvectors where the direction the magnitude changes most rapidly or most slowly. The structure tensor for EPI Σ_{y^*,t^*} can be described as

$$J_{\Sigma_{y^*,t^*}}(x,s) = \begin{bmatrix} G_\sigma * \big(\partial(x)\partial(x)\big) & G_\sigma * \big(\partial(x)\partial(s)\big) \\ G_\sigma * \big(\partial(x)\partial(s)\big) & G_\sigma * \big(\partial(s)\partial(s)\big) \end{bmatrix} = \begin{bmatrix} J_{xx} & J_{xs} \\ J_{xs} & J_{ss} \end{bmatrix}, \quad (3.1)$$

where $\partial(x)$ and $\partial(s)$ represent the gradient components in x and s directions respectively at point (x,s) in EPI Σ_{y^*,t^*}. G_σ is a Gaussian kernel with variance σ and the operation symbol '$*$' denotes convolution. The disparity values for all x can be generated by [77],

$$P_{\Sigma_{y^*,t^*}}(x) = \frac{\Delta x}{\Delta s} = \tan\theta, \quad (3.2)$$

where

$$\theta = \frac{1}{2}\arctan\left(\frac{J_{ss}-J_{xx}}{2J_{xs}}\right). \quad (3.3)$$

A reliability measure can also be generated as follows:

$$r_{\Sigma_{y^*,t^*}}(x) = \frac{(J_{ss}-J_{xx})^2+4(J_{sx})^2}{(J_{ss}+J_{xx})^2}. \quad (3.4)$$

A disparity map $P_{\Sigma_{y,t^*}}(x)$ and reliability map $r_{\Sigma_{y,t^*}}(x)$ based on the EPIs in the horizontal direction can then be obtained by repeating the above for all y. We can

also obtain a disparity map $P_{\Sigma_{x,s^*}}(y)$ and reliability map $r_{\Sigma_{x,s^*}}(y)$ based on the EPIs in the vertical direction using a similar approach. Then the final disparity map is generated by selecting the disparity value with higher reliability. That is,

$$P(x,y) = \begin{cases} P_{\Sigma_{y,t^*}}(x) & if\ r_{\Sigma_{y,t^*}}(x) > r_{\Sigma_{x,s^*}}(y) \\ P_{\Sigma_{x,s^*}}(y) & otherwise \end{cases}. \tag{3.5}$$

In practice, if the reliability value is too small, $P(x,y)$ can be inaccurate and will

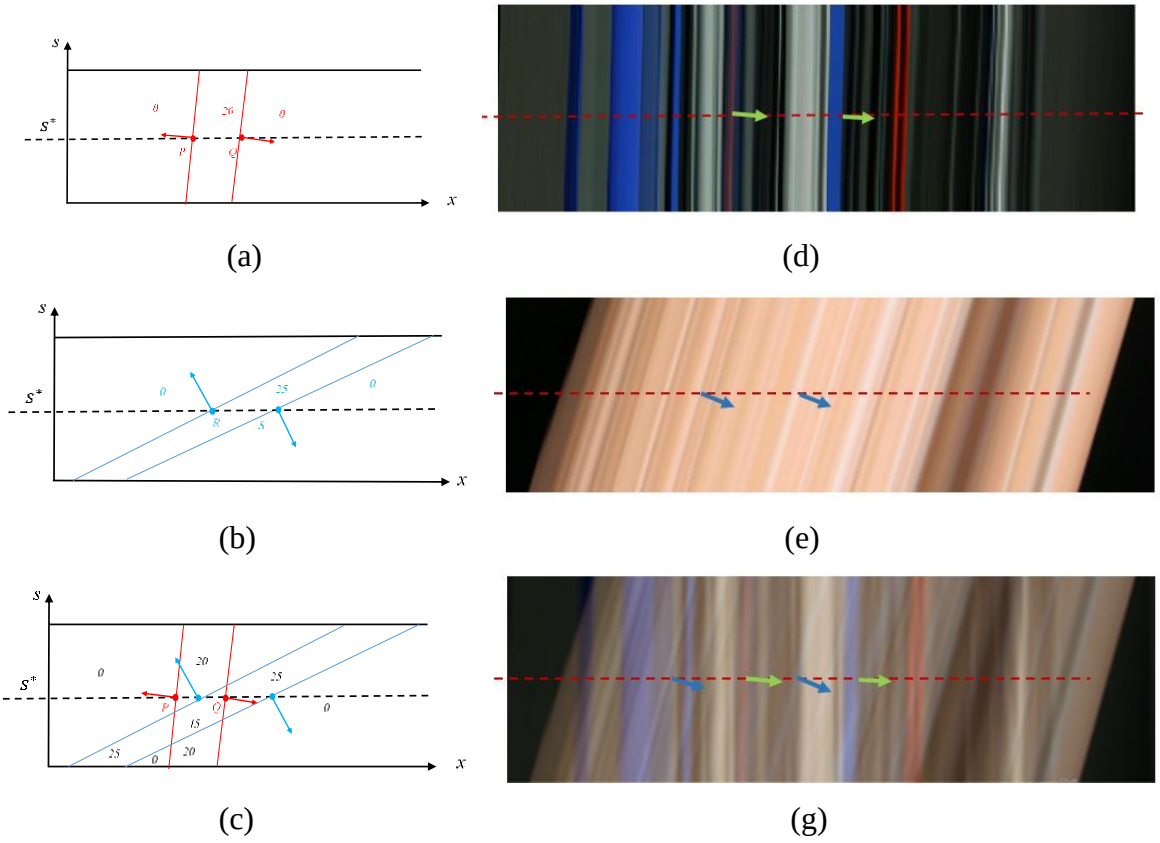


Fig. 3.2. An illustration of the relationship of the strong gradient points in the original and combined EPIs. (a) An EPI of an LF image. Two strong gradient points P and Q are noted. (b) An EPI of another LF image. Two strong gradient points R and S are noted. (c) The combined EPI. The numbers represent the pixel magnitudes after combination. It can be seen that all strong gradient points in (a) and (b) locate at different positions with the same values as before. (d), (e) and (g) are real cases for (a), (b) and (g) respectively.

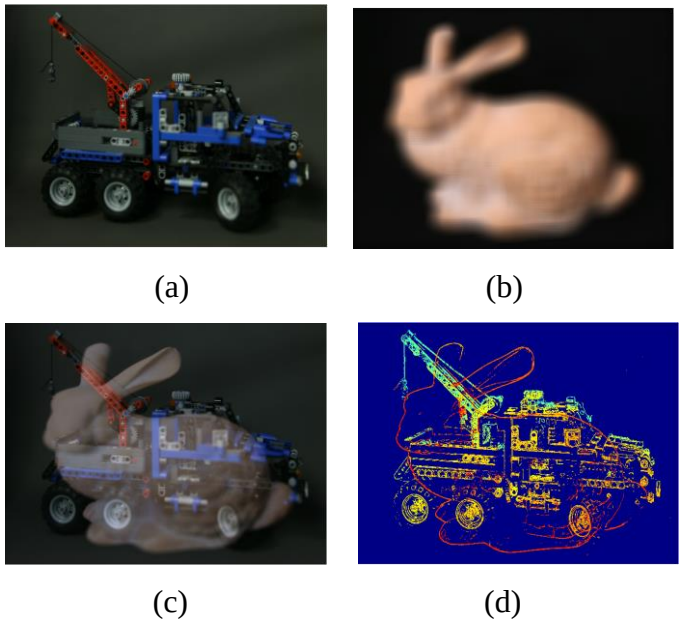
simply be set as invalid. One of the situations that it will happen is when the pixel (x, y) has no or very weak gradient. Hence, $P(x, y)$ can also be considered as the disparity map at the strong gradient points.

For the problem of reflection removal, a reflection image is superimposed on the background image. When the scene is captured by an LF camera, the resulting EPIs will also be a superimposition of the EPIs of both images. Since these images can have different depths, we can find the resulting EPI also has slopes of different angles, and they cross each other randomly in the EPI. Particularly in the regions where they cross each other, it is difficult to determine the slope of the EPI pixels and further classify them into the background or reflection layer. To deal with the problem, we consider again the gradient of the EPIs, of which the disparity map is derived in (3.1) to (3.5). In particular, we investigate the behavior of the strong and weak gradient points of the background and reflection as follows:

Case 1: Strong gradient points of both layers

This case is illustrated in Fig. 3.2(a) to (c). In the figure, both EPIs have two strong gradient points. When the EPIs are added up, the strong gradient points do not overlap each other and preserve the same values as shown in Fig. 3.2(c). Such a phenomenon is not coincident. It is known that the strong gradient points of an EPI correspond to the strong gradient points of the image. Due to the gradient independence assumption [9, 31], it is rare to have strong gradient points of two

uncorrelated images overlapped each other. Consequently, we can also assume that the EPI strong gradient points of two uncorrelated images will be at different positions in the combined EPI. Besides, as shown in Fig. 3.2(c), the gradient value will remain the same wherever a strong gradient point locates in the combined EPI. A real such case is also illustrated in Fig. 3.2(e) to (g). We can find the edge directions are barely changed when two EPI are overlapped. Consequently, we can easily estimate the disparities at these strong EPI gradient points. An example is shown in Fig. 3.3. In the example, two LF images are added together with the weightings of 0.6 and 0.4 respectively. The central view of the resulting LF image



(a) (b)

(c) (d)

Fig. 3.3. An example of a disparity map generated from the strong gradient points in the combined EPI. (a) An LF image with all views overlapped. (b) Another LF image with all views overlapped. The extent of blurring represents the amount of disparity. We can see the disparity of (b) is larger than (a). (c) The central view of an LF image generated by combining (a) and (b) with the weightings of 0.6 and 0.4 respectively. (d) The estimated disparity map based on the strong gradient points of the EPI of (c). The red and blue color means large and small disparities respectively. Since in most cases they are not overlapped, they can be easily identified.

is shown in Fig. 3.3(c). The EPIs of the resulting LF image is then generated. Based on the EPIs, we first estimate the disparity map of the image in Fig. 3.3(c) using the structure tensor method in (3.1) to (3.5) and keep only those at the strong gradient points. It can be seen in Fig. 3.3(d) that the disparities of the two layers at the strong gradient points can be easily identified as they are at different positions.

Case 2: Weak gradient points of both layers

For the weak EPI gradient points of both layers, they may or may not overlap with the EPI gradient points of the other layer. For those that do not overlap with another EPI gradient point, the disparity at those points can still be estimated as usual. In case they overlap with another EPI gradient point, their correct gradient value can no longer be recovered. The estimated disparity value will appear as noises in the disparity map and will be regulated in the later optimization process.

To summarize, as the first step of our proposed algorithm, we make use of the structure tensor method in (3.1) to (3.5) to generate a disparity map in the EPI domain. Since the gradient points in the EPI domain have a close relationship with the gradient points in the spatial domain, the resulting disparity map will contain accurate disparity values at the strong gradient points of both the background and

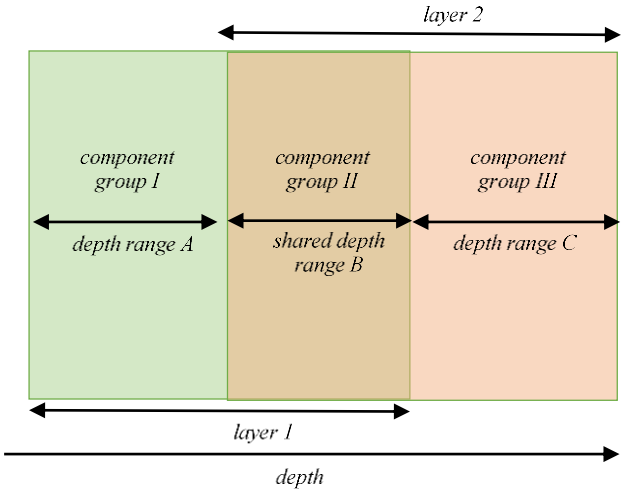


Fig. 3.4. The new sandwich model. In this model. Component group I only belongs to layer 1 and component group III only belongs to layer 2 (layer 1 is assumed to be relatively closer to the camera). Both layers share component group II.

reflection images and at the weak gradient points in case they do not overlap with other gradient points. We also expect that there will be noises caused by the overlapped weak gradient points of both images.

3.2.2. The Sandwich Model and Initial Image Reconstruction

If the background and reflection have absolutely different depth ranges, the disparity map generated in Section 3.2.1 should be sufficient to classify most of the strong gradient points; and we can use these gradients to reconstruct the background and reflection images. Unfortunately, it is not uncommon in many practical situations that some components of the background and reflection share a common depth range. It means that their disparities can be very similar. For this reason, we

propose a new sandwich model, as shown in Fig. 3.4, to take care of such situation. As shown in the figure, the model has one shared depth range for both layers. Assume that we can find two thresholds, K_1 and K_2, which are at the boundaries of component groups I and II, as well as groups II and III, respectively. Then, all strong gradient points with disparities smaller than K_1 will belong to layer 1, and those greater than K_2 will belong to layer 2. For those that are greater than K_1 but smaller than K_2, it is difficult to classify them by only their disparities due to the reasons mentioned above. We will discuss in the next section how these components can be classified by exploring their relationships with the components in groups I and III.

To find the thresholds K_1 and K_2, we apply the K-means clustering method [79] (where K=2 in this case) on the estimated disparity values at all edges. We denote the centers of the two clusters as C_1 and C_2 ($C_1 < C_2$). Then, we set the two thresholds as

$$\begin{aligned} K_1 &= C_1 + \sigma \cdot (C_2 - C_1); \\ K_2 &= C_2 - \sigma \cdot (C_2 - C_1), \end{aligned} \tag{3.6}$$

where σ is a parameter to control the purity of the classification result. In our experiment, we set $\sigma = 0.2$, which is a conservative choice to ensure that the classification has a high true positive rate. Then, we only need to take care of those

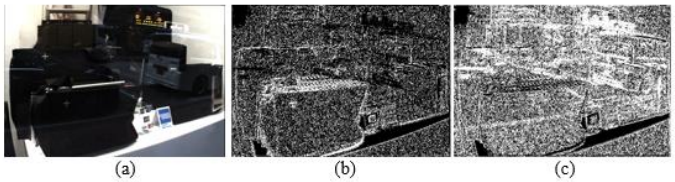


Fig. 3.5(i). An example of E_B^0 and E_R^0. We can see that E_B^0 and E_R^0 can roughly separate the background and reflection gradient components.

Fig. 3.5(ii). The initial separation results. All the results are normalized by (3.10) for the ease of visualization. (a) The original image I. (b) The initial estimate of the background of the background layer I_B^0 . (c) The residue of the initial background estimate $I_{\overline{B}}^0 = I - I_B^0$. (d) The initial estimate of the reflection layer I_R^0. (e) The residue of initial reflection estimate $I_{\overline{R}}^0 = I - I_R^0$. We can see that the components of I_B^0 almost only belong to the background layer and its residue $I_{\overline{B}}^0$ does not only contain the reflection components but also the missing background components. And similarly, I_R^0 loses some reflection components which can be found in its residue $I_{\overline{R}}^0$.

misclassified gradients. Based on K_1 and K_2, two initial gradient masks are obtained as follows:

$$E_B^0 = \{P(x,y) > K_2\ , \forall x, \forall y\ \};$$
$$E_R^0 = \{P(x,y) < K_1\ , \forall x, \forall y\ \}, \tag{3.7}$$

where $P(x,y)$ is defined in (3.5). $E_B^0, E_R^0 \in \{0,1\}$ are the two initial gradient masks for the background and reflection layers, respectively. Without loss of generality, we assume that the background layer is the closer layer (otherwise, only a change of symbols is required). Fig. 3.5(i) shows an example of the initial gradient masks. It can be seen in Fig. 3.5(i)(b) that the locations of some of the background gradients

are correctly indicated in E_B^0. However, we can also find that some background gradients are missed out in E_B^0. Based on the masks, we can reconstruct the background and reflection images in the gradient domain as follows:

$$I_B^0 = \arg\min_{I_B^0} J = \|D * I_B^0\|_1 + \left\|D * I_{\overline{B}}^0\right\|_1 + \lambda\left\|E_B^0 \cdot D * I_{\overline{B}}^0\right\|_1 + \lambda\left\|E_{\overline{B}}^0 \cdot D * I_B^0\right\|_1; \quad \text{s.t. } I_{\overline{B}}^0 = I - I_B^0;\ \ E_{\overline{B}}^0 = \mathbf{1} - E_B^0, \tag{3.8}$$

$$I_R^0 = \arg\min_{I_R^0} J = \|D * I_R^0\|_1 + \left\|D * I_{\overline{R}}^0\right\|_1 + \lambda\|E_R^0 \cdot D * I_R^0\|_1 + \lambda\left\|E_{\overline{R}}^0 \cdot D * I_R^0\right\|_1; \quad \text{s.t. } I_{\overline{R}}^0 = I - I_R^0;\ \ E_{\overline{R}}^0 = \mathbf{1} - E_R^0, \tag{3.9}$$

where $\mathbf{1}$ refers to an all '1' matrix, and λ is a constant. In (3.8) and (3.9), the initial estimates of the background and reflection image, i.e., I_B^0 and I_R^0, are obtained by minimizing the sum of a few sparsity priors in the gradient domain. This approach is based on the gradient sparsity assumption that the total gradient of the background (or reflection) should be sparser than that of the original image, which contains the sum of the background and reflection. Thus, when the estimate I_B^0 (or I_R^0) approaches the true background (or reflection), its total gradient should approach the minimum. The same is applied to their residues $I_{\overline{B}}^0 = I - I_B^0$ and $I_{\overline{R}}^0 = I - I_R^0$. In addition, based on the gradient independence assumption [9, 31], the total gradient of the background after multiplying with the gradient mask of the reflection should be small since their strong gradient points will not overlap. Thus, when the estimate

I_B^0 (or I_R^0) approaches the true background (or reflection), its total gradient after multiplying with the mask of its residue, which approaches the true reflection (or background), should approach the minimum. In (3.8) and (3.9), $D \equiv D_{i=1,\dots,5}$ represents a set of derivative filter kernels such that $D_1 = D_2^T = [1, -1]$ are the first-order derivative filters in the horizontal and vertical directions, respectively; $D_3 = D_4^T = [1, -2, 1]$ and $D_5 = D_2 * D_1$ are the second-order derivative filters in the horizontal, vertical and diagonal directions, respectively. The use of the second-order filters is for rectifying the discontinuities in the gradient domain due to the rare situations in which the strong gradient points overlap each other. Here, (3.8) and (3.9) can be solved by the iteratively reweighted least squares (IRSL) method. Fig. 3.5(ii) shows an example of the initial separation results. For the ease of visualization, the biases of the resulting images are adjusted to the original biases as follows:

$$I_{display} = I_{result} - mean(I_{result}) + mean(I) \tag{3.10}$$

As shown in Fig. 3.5(ii), almost all components of the initial background estimate belong to the background layer. However, many components are missing and can be found in its residue. The same is applied to the initial reflection estimate. To enhance the separation results, we develop a new method to detect and recover the missing components from the residues, which will be described in the next section.

3.3. Detect and Regenerate the Missing Background Gradients

As mentioned above, there can be components of both the background and reflection layers sharing the same disparity range (i.e. component group II in Fig. 3.4). These components are supposed to be removed from I_B^0 and I_R^0 since they cannot be accurately classified as belonging to the background or reflection. It can be seen in the initial estimation result (Fig. 3.5(ii)(b) and (d)) that large parts of I_B^0 and I_R^0 are darkened. They are the parts which have been removed. To retrieve back these missing components, we have another observation about the gradients in the initial estimation. By comparing between I_B^0 and its residue $I_{\overline{B}}^0$ (such as Fig. 3.5(ii)(b) and (c)), we observe that although the background components in the shared depth range are supposed to be removed due to the conservative thresholds used in (3.7), the strong gradient points of the missing background components can still be visualized in I_B^0 (circled in Fig. 3.5(ii)(b)). It is due to the first two terms in (3.8) which promote the sparsity in the image. However, their magnitudes are rather small such that directly detecting them based on their magnitude can be erroneous. Note that both I_B^0 and $I_{\overline{B}}^0$ contain the strong gradients of the background's missing components, although the ones in $I_{\overline{B}}^0$ are much clear. On the other hand, the strong gradients of the reflection image are less visualized in I_B^0. It is because the magnitude of the reflection is often much lower than the background as most semi-reflective

materials such as glass can only partially reflect the light projected onto it. So, for a particular spatial position (x, y), if the gradients $I_B^0(x, y)$ and $I_{\bar{B}}^0(x, y)$ are the same, they likely belong to the background. Based on the same argument, if the gradients $I_R^0(x, y)$ and $I_{\bar{B}}^0(x, y)$ are the same, they likely belong to the reflection. We will make use of this property to detect and recover the missing components in I_B^0.

As mentioned above, directly detecting the gradients of the missing components in I_B^0 based on their weak magnitudes can be erroneous. Therefore, we suggest considering also the gradient directions. While there are several ways to detect the directions of gradients, we suggest considering the Histogram of Oriented Gradients (HOG) method [80]. HOG is a feature descriptor for object detection. It contains the weighted (according to the magnitude) distribution (histograms) of directions of gradients (oriented gradients) of an image cell normalized with the nearby cells within a block. It is suitable in this problem because HOG is invariant to the local illumination of the image and can measure the direction of gradients of small magnitude. The procedure is as follows. First, to avoid the disturbance from the very weak gradients whose orientations are very unstable, we only consider the strong gradients at some spatial position set $\varphi^t = \{(x, y)||\partial_B^t(x, y)| > \epsilon\}$, where $|\partial_B^t(x, y)|$ is the magnitude of the gradient of $I_B^t(x, y)$ at iteration t; and ϵ is a very small constant. Then, we compute the HOG feature vectors H_B^t, $H_{\bar{B}}^t$ and H_R^0 at each spatial positon in the set $\varphi^t$of I_B^t, $I_{\bar{B}}^t = I - I_B^t$ and I_R^0, respectively. For keeping the spatial resolution, we use a relatively small cell size of 3x3, and the block size is

2x2 as usual. Here, we use the UoCTTI variant [81, 82] of HOG of which the feature vector length for every pixel is 31. So, the size of every feature vector is $h \times w \times 31$, where h and w are the height and width of the image. Then we measure the Euclidean distances of H_B^t and $H_{\bar{B}}^t$ as well as H_R^0 and $H_{\bar{B}}^t$ as follows:

$$U_B^t(x,y) = \left\| H_B^t(x,y) - H_{\bar{B}}^t(x,y) \right\|_2 ; \tag{3.11}$$

$$U_R^t(x,y) = \left\| H_R^0(x,y) - H_{\bar{B}}^t(x,y) \right\|_2 ,$$

for all $(x,y) \in \varphi^t$. U_B^t measures the similarity between the HOG in I_B^t and its residue $I_{\bar{B}}^t$. If $U_B^t(x,y)$ is small, the gradient at (x,y) of I_B^t and $I_{\bar{B}}^t$ should belong to the background as discussed above. $U_R^t(x,y)$ measures the similarity between the gradients in I_R^0 and $I_{\bar{B}}^t$. Due to the conservative thresholds used in (3.6), I_R^0 contains mainly the components of the reflection layer. And $I_{\bar{B}}^t$ also has the components of the reflection layer. So, if $U_R^t(x,y)$ is small, it indicates that the gradient at (x,y) of $I_{\bar{B}}^t$ should belong to the reflection. Then, U_B^t at the same point (x,y) should be large, since I_B^t should not have reflection components. Thus, U_R^t can be used to validate U_B^t in the classification process.

To perform the classification, we borrow the ideas of the Markov Random Field (MRF) [83] and the K-nearest neighbors (KNN) matting [84] to formulate the following optimization function:

$$L^t = \arg\min_L F(L) = \sum_{p \epsilon \varphi^t} \left(U_p(L_p) + \lambda \sum_{q \epsilon SKNN(p)} V_{p,q}(L_p, L_q) \right), \tag{3.12}$$

$$U_p(L_p) = U_R^t(p)(1 - L_p) + U_B^t(p) L_p, \tag{3.13}$$

$$V_{p,q}(L_p, L_q) = \left(1 - N\left(\left\|\partial_{\bar{B}}^t(p) - \partial_{\bar{B}}^t(q)\right\|_1\right)\right) \cdot [L_p \neq L_q], \tag{3.14}$$

where λ is a constant and the function $N\{x\}$ normalizes x to between 0 to 1. The proposed energy function F in (3.12) is defined so that its minimum corresponds to a good classification of the gradients in $I_{\bar{B}}^t$. L represents the label set. L_p denotes the label of the gradient at position p in set φ^t. It is set to 1 for the background gradient and 0 for the reflection gradient. The data term $U_p(L_p)$ penalizes the cost function if a wrong classification of L_p is made. More specifically, if the gradient of $I_{\bar{B}}^t$ at p belongs to the background but is incorrectly classified as a reflection (i.e. L_p is set to 0), $U_p(L_p)$ will have a large value since $U_R^t(p)$ is large in this case. On the other hand, if the gradient of $I_{\bar{B}}^t$ at p belongs to the reflection but is incorrectly classified as background (i.e. L_p is set to 1), $U_p(L_p)$ will also have a large value since $U_B^t(p)$ is large in this case.

Similar to MRF, the data term U_p is supplemented with a smoothness term $V_{p,q}$ in (3.12), which measures the smoothness of the gradients in $I_{\bar{B}}^t$. It is observed that strong gradients, such as the edges of an object, are smooth in some orientation.

Adjacent gradients with similar orientation likely belong to the same object in the same layer. Thus, the smoothness term in (3.12) is designed such that it will be large and penalizes the cost function F if neighboring gradients in $I_{\bar{B}}^{t}$ with similar orientations are assigned with different labels. In (3.14), the function $[L_p \neq L_q] =$ 1 if $L_p \neq L_q$; and 0 otherwise. Thus, the term $V_{p,q}$ of two pixels p and q in $I_{\bar{B}}^{t}$ will be zero if they have the same label. Otherwise, $V_{p,q}$ is evaluated based on the 1-norm difference of the gradients $\partial_{\bar{B}}^{t}$. Note that $F(L)$ in (3.12) is evaluated by accumulating $V_{p,q}$ for all pixel pairs $\{p, q\}$ within the similarity-based KNN (SKNN) set of p, which is defined as the set of K nearest neighboring pixels (where K is chosen as 7) of p measured by the similarity in gradient value and distance. Normally, all pixels within the SKNN set should have the same label due to the smoothness of object gradients. If a pixel q within the set is wrongly classified, the classification of p will still follow the majority in the set since $V_{p,q}$ is small. In the situation that p is wrongly classified such that it is different from most others in the set, a large sum of $V_{p,q}$ will be generated. It penalizes the cost function and forces the label of p to change.

The optimization problem in (3.12) can be solved by the max-flow/min-cut method [85]. Finally, a mask based on L is generated as follows:

$$S^{t+1} = \rho\{L^t = 1\}, \tag{3.15}$$

where $\rho\{x\}$ represents a morphological dilation operator with size 2x2 within the set φ^t. It is used since we assume the neighboring gradients around the classified labels also likely belong to the same layer. Note that S^t can be considered as a mask of the gradients that appear at the same positions of both I_B^t and its residue. It thus has included the gradients of the missing background components based on the argument discussed earlier. So, using S^t, we update the initial gradient masks as follows:

$$E_B^t = E_B^{t-1} \cup S^t \cup E_B^0 \cap (\sim E_R^0); \tag{3.16}$$

$$E_R^t = E_{\bar{B}}^{t-1} \cap (\sim S^t) \cap (\sim E_B^0) \cup E_R^0 \,,$$

for $t > 0$. $E_{\bar{B}}^t$ is defined in (3.8). Recall that E_B^0 is estimated with a conservatively selected disparity threshold. Most of the gradient points it covers belong to the background, although a lot of the background's gradient points can be missing. To enhance E_B^0 , we firstly exclude those also covered by the reflection gradient mask E_R^0. Then we add back those covered by S^t to E_B^{t-1} in each iteration as shown in (3.16). With the improved estimation of I_B^t in each iteration, the estimation of S^t will also improve and in turn enhance the estimation of E_B^t. The design of E_R^t follows a similar philosophy. The new gradient masks now include the information of the missing components. They can be used to refine the background estimate as follows:

$$I_B^t = \arg\min_{I_B^t} J = \|D * I_B^t\|_1 + \|D * I_R^t\|_1 + \lambda_1 \|E_B^t \cdot D * I_R^t\|_1 + \lambda_1 \|E_R^t \cdot D * I_B^t\|_1; \quad \text{s.t. } I_R^t = I_{\bar{B}}^t = I - I_B^t; \text{ for } t > 0. \tag{3.17}$$

Note that unlike the existing approach which requires the optimization of a number of parameters simultaneously, there is only one optimization parameter I_B^t in (3.17) (we can find I_R^t by $I_R^t = I_{\bar{B}}^t = I - I_B^t$ for $t > 0$). It reduces the possibility that the optimization process falls into the wrong local minimum. Similar to (3.8), we use IRSL to minimize (3.17). We iteratively update the background layer until converged (e.g. the change of the recovered I_B^t becomes very small). An illustration

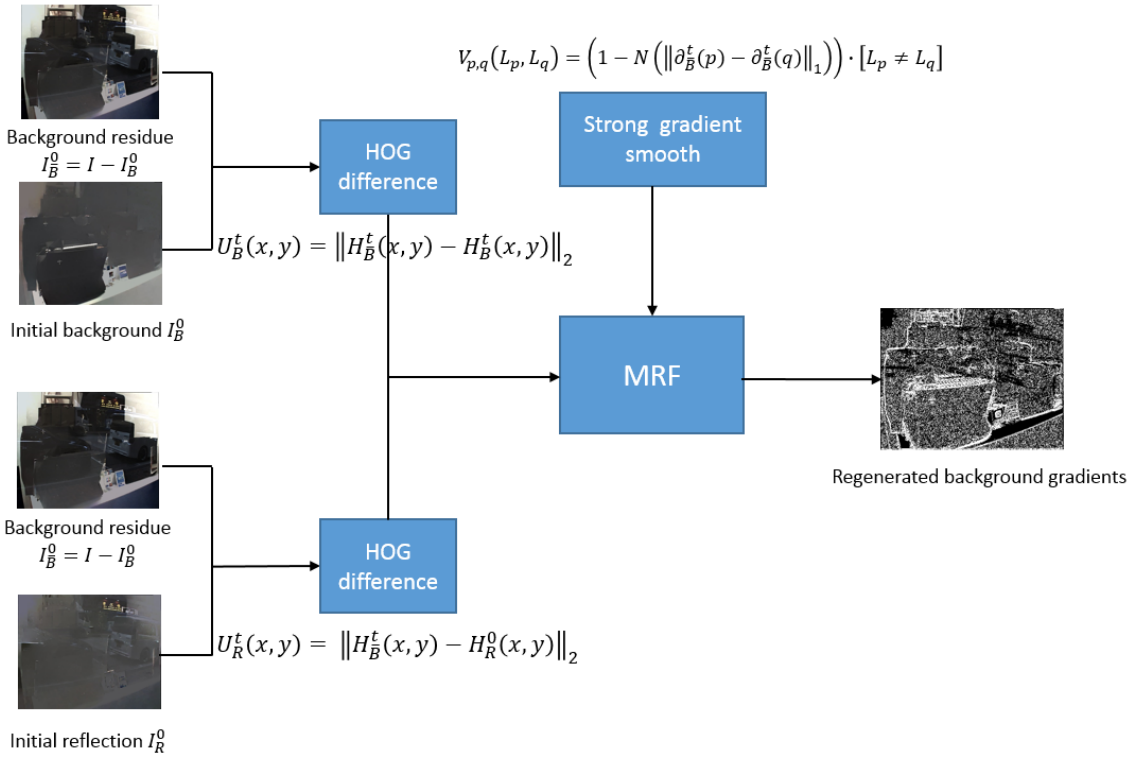


Fig. 3.6. An illustration of the whole process of background gradient regeneration.

of the proposed background gradient regeneration method is shown in Fig. 3.6. The whole algorithm is summarized below:

Algorithm:

Generate the disparity map using (3.1) to (3.5).

Obtain the initial estimates E_B^0, E_R^0, I_B^0 and I_R^0 using (3.6) to (3.9).

Set $t = 0$.

While not converge

$t = t + 1$;

Compute S^t using (3.11) to (3.15);

Compute E_B^t, E_R^t using (3.16);

Obtain I_B^t, I_R^t using (18);

End

Fig. 3.7 shows an example of the proposed algorithm at different stages of operations. It can be seen in Fig. 3.7(g) that the initially estimated background image has many components missing. It is because the initial gradient mask E_B^0 misses out many strong gradients as shown in Fig. 3.7(b). With the help of S^1as shown in Fig. 3.7(e), the updated gradient mask E_B^1 (Fig. 3.7(c)) starts to restore some of the missing components. It, in turn, improves the estimation of S^2 (Fig. 3.7(f)) and then E_B^2 (Fig. 3.7(d)), as can be seen in the circled regions. Note that while more and more missing background components are recovered in S^2 (see the upper two

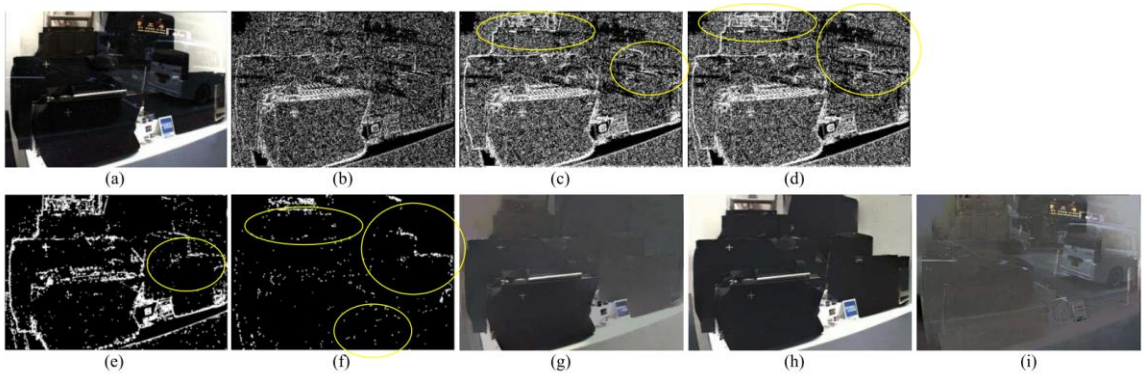


Fig. 3.7. The intermediate results. (a) The original image I. (b) The estimated initial gradient mask E_B^0. (c) The improved gradient mask E_B^1. (d) The improved gradient mask E_B^2. See the improved estimation (circled). (e) Mask S^1. (f) Mask S^2. (g) The estimated initial background layer I_B^0. (h) The resulting background layer I_B. (i) The resulting reflection layer I_R.

circles in Fig. 3.7(f)), we also notice the mask covers less background gradient points (such as the ones in the lower circle in Fig. 3.7(f)). It is because with the improved estimation of I_B^1, there are less common gradient points with the residue of I_B^1, which means that they have been correctly recovered in I_B^1 thus S^2 does not need to include them. The final background image generated by the proposed algorithm is shown in Fig. 3.7(h). It shows a significant improvement over the initial guess. The resulting reflection image is also shown in Fig. 3.7(i).

3.4. Comparisons and Evaluation

To evaluate the performance of the proposed algorithm, we make a comparison with four other multiple-image reflection removal methods both qualitatively and quantitatively. These methods include superimposed image decomposition (SID)

[9], layer separation using SIFT flow (LS-SIFTF) [10], layer separation using motion flow (LS-MF) [11] and layer separation using disparity signs (LS-DS) [12]. All these approaches make use of the depth information of the scene to separate the background and reflection. Since [9, 11] and [10] capture the multiple views of a scene using a sequential approach, they can only be used in static scenes. [12] makes use of the LF camera to capture the multiple views of a scene in one shot. Hence it can be applied to dynamic scenes as the proposed method. However, it has a stringent requirement about the depth of the background and reflection layers, as well as the orientation of the camera. We will show in the later comparisons how these restrictions affect the separation performance.

3.4.1. Qualitative Evaluation

For qualitative evaluations, we compare visually the quality of the background and reflection images separated by different approaches. For testing the proposed algorithm and the method in [12], we make use of the Lytro Illum LF camera to obtain the LF images of a number real-life scenes in which the background is superimposed by reflection. For the same set of real-life scenes, we use the same LF camera to capture the scenes from 5 different angles. And then the central view of each LF image is collected to form the multiple-view images required by the methods [9, 11] and [10]. We show a few sets of comparison results in Fig. 3.8 and

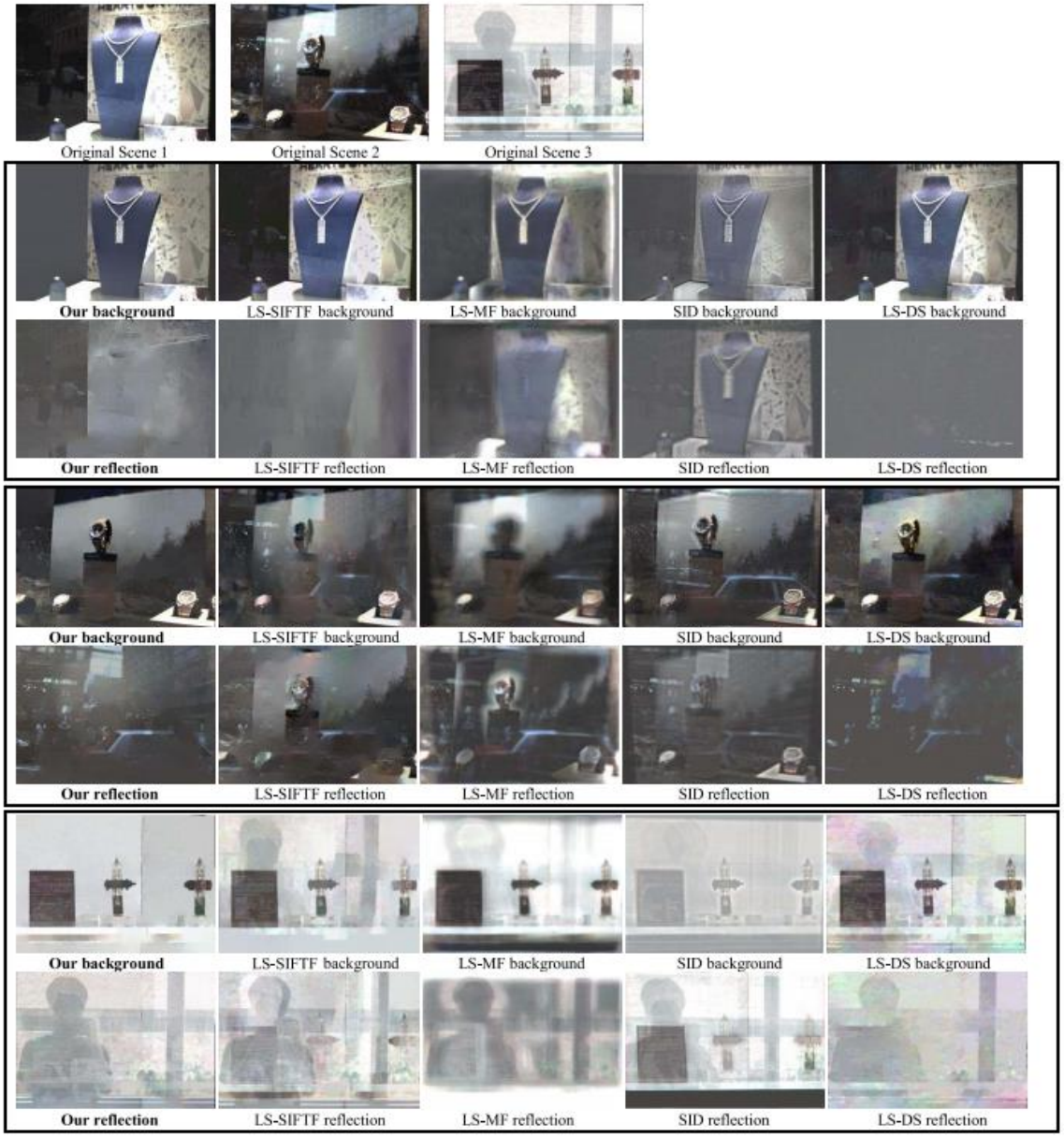


Fig. 3.8. Comparison results of scene 1 to 3. For the ease of visualization, the images are normalized by (3.10). So for some images, the background plus reflection may not be equal to the original images. We can see that the proposed method shows robust and better results compared to other methods.

Fig. 3.9. Since they are all real-life scenes, there is no ground truth in all cases. But

from the contents in the separated background and reflection images, we can easily

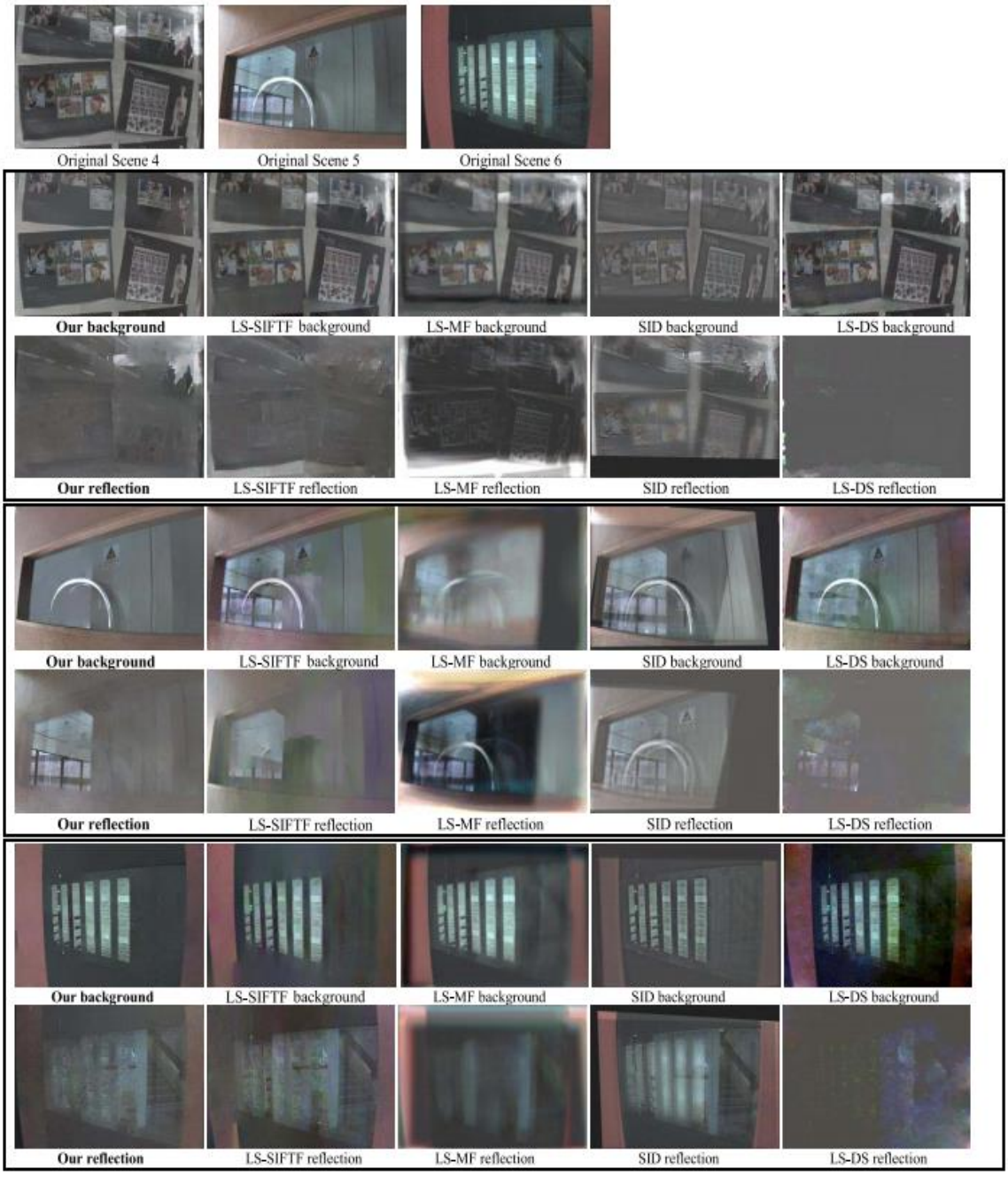


Fig. 3.9. Comparison results of scene 4 to 6. For the ease of visualization, the images are normalized by (3.10). So for some images, the background plus reflection may not be equal to the original images. We can see that the proposed method shows robust and better results compared to other methods.

identify which approach performs the best.

As described above, traditional methods all have their own limitations to the input

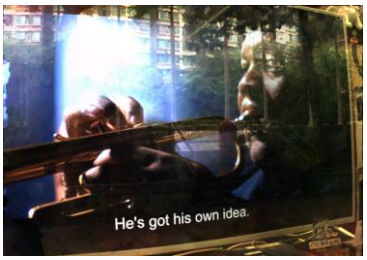
images, it is difficult to ensure that they perform well for all images, particularly those taken from real-life scenes since it is difficult to control the scene environment. As shown in Fig. 3.8 and Fig. 3.9, the recovered backgrounds tend to retain some residual reflection components, and their reflection images also often contain background components. For method LS-SIFTF, it is noticed that it cannot separate those reflections with strong gradients. It is because SIFT flow will also register those reflection gradients as background. We can see many regions with strong reflection gradients are wrongly separated. For LS-MS, the optimization process can easily fall into the wrong local minimum. We can find that the reconstructed background layers, which are constructed by the combination of all views, may be blurred due to the inaccurate motion flows. For SID method, it shows poor performance for scenes with non-planar background since it uses 2D homography to register images. Moreover, the results of SID tend to be over-smooth because of the use of low-rank decomposition with inaccurate registrations. For LS-DS, it has a stringent requirement about the distance of the background or reflection layer. In many real-life scenes, such requirement cannot be fully fulfilled. Besides, it requires the normal axis of the LF camera to be aligned perpendicular to the scene. As shown in the images in Fig. 3.8 and Fig. 3.9, we often take pictures with an angle to the scene. It is about the style of photography that is hard to put restriction to. Since the scenes in Fig. 3.8 and Fig. 3.9 do not fully fulfill the requirements, the performance of LS-DS is only marginally satisfactory in most cases. Without the abovementioned limitations, the proposed algorithm can well reconstruct each layer

Original scene

The scene in another moment

Our background

Our reflection



LS-DS background

LS-DS reflection

Fig. 3.10. A dynamic scene case: a television behind a glass window. Since the content of the television display is changing in time, other methods that require multiple shots of the scene cannot work in this case. Therefore, only the results of LS-DS and the proposed method are shown. It can be seen that the proposed method gives much better performance than LS-DS.

and show the best performance in all cases. We also show a case with a dynamic background in Fig. 3.10, where a television is showing a video behind a glass window. Since the TV display content is changing, the methods that require multiple shots of the scene from different angles cannot capture the same background thus cannot be used in this case. So, we only test LS-DS and our method for this scene.

Method	First layer	Second layer
Synthetic input	13.0249	12.6774
LS-SIFTF	18.4999	18.9543
SID	15.2370	19.3243
LS-MF	16.5286	16.2398
LS-DS	18.6339	18.6433
Proposed	**21.9918**	**21.8188**

Table 3.1. The average PSNR values of the synthetic input images and the results of different methods.

Since the normal axis of the camera is not perpendicular with the scene, we can see that LS-DS leaves a large number of reflection residues in the estimated background image, while the proposed algorithm can give much better performance than LS-DS.

3.4.2. Quantitative Evaluation

For the quantitative evaluation of LS-DS and the proposed algorithm, we first use an LF camera to capture 20 LF images. Ten of them are selected as background while the other ten are selected as reflection. They are manually added together to simulate the images we needed for the evaluation. Since the background is known, we can always measure the PSNR of the separated background with the true one. To generate the images required for the evaluation of the multiple-image methods, we need to have background and reflection images of different viewing angles for each scene. To do so, we do not only take one LF image for each scene as mentioned

above. We put the light field camera on a tripod and shift the camera at five fixed vertical heights to capture five LF images for every scene. Then we use only the central view of each LF image such that for every scene, there are 5 images taken from 5 fixed vertical positions. And since all views of different scenes are taken at 5 fixed vertical heights, we can superimpose any two scenes together to simulate a background image with reflection taken from 5 different viewing angles. These images are then used in the evaluation of the multiple-image methods. Since all separated images may contain biases, we adjust the bias of each separated image to achieve the maximum PSNR as compared with the ground truths. Then we compare the average maximum PSNR of the separated background and reflection images for all 10 scenes generated by all methods. The final results are shown in Table 3.1. We can see that the proposed method outperforms all compared methods. The results are in line with the qualitative evaluation results.

3.5. Summary

In this chapter, we proposed a novel algorithm for solving the reflection removal problem based on light field imaging and the background gradient regeneration strategy. One major improvement of the new algorithm is in its robustness, since it does not have the various restrictions on the scene or the camera orientation as in the existing approaches. In this chapter, we first explored the behavior of the strong

gradient points in the EPI of LF images when they are superimposed with reflection images. It provides the theoretical support for using the light field imaging to estimate the disparities of different layers of such kind of images. We also proposed a general sandwich model to describe the disparity ranges of the components of the background and reflection layers. It is the major part of how the proposed algorithm can be more versatile than the existing methods. Based on this model, we proposed a two-step strategy (initial aggressive separation and background gradient regeneration) to well reconstruct the background layer in an iterative enhancement process. In the evaluation part, we showed the proposed algorithm has better and more robust performance compared to the state-of-the-art reflection removal methods.

Chapter 4.

Improved Multiple-Image Reflection Removal Algorithm Using Deep Neural Networks

(This chapter is extracted from my paper [86]: Tingtian Li, Yuk-Hee Chan, and Daniel P.K. Lun, “Improved multiple-image based reflection removal algorithm using deep neural networks,” IEEE Transactions on Image processing, 2019. (under review))

In Chapter 3, we introduced a multiple-image reflection removal algorithm using different optimization methods. While the algorithm is effective, the time-consuming optimization processes introduce much difficulty when applying it to some real-time applications. It has been a trend in recent years to use deep learning approaches in solving image processing problems. In these approaches, huge datasets are used to train different deep neural network (DNN) models for solving the problems with good performance and efficiency. It is because due to the massive parallel structures of these network models, they can be easily implemented using GPUs to dramatically reduce the computation time. Following the trend, we present

in this chapter a novel DNN-based approach for solving the reflection removal problem following the background gradient regeneration strategy. In fact, it is not totally new for using DNN in reflection removal. However, existing DNN-based methods [7, 25] require that the reflection must be blurry. It affects the generality of their application. In this chapter, we propose a novel DNN-based framework for solving the reflection removal problem using multiple images. The algorithm exploits the depth information of the scene provided by the multiple input images to help separate the background and reflection. It does not require that the reflection must be blurry hence it is more general and robust. Experimental results show that the proposed algorithm achieves superior performance similar to the method we proposed in Chapter 3, but it has a much faster speed when implementing with GPUs.

4.1. Introduction

Owing to the direct feedforward process and efficient use of GPU, DNN-based methods have shown superior performance and much faster speed compared to the traditional optimization-based methods in many image processing applications. As a branch of DNN, GAN [87] has also drawn dramatic attention from researchers. A GAN contains a generator that produces new samples. It also has a jointly trained discriminator that tries to distinguish the sample produced by the generator if they are the same as the real samples in the target dataset. When the discriminator cannot

distinguish the generated samples from the real samples, it means the generator has been successfully trained to synthesize new samples following the distribution of the target set. However, because of the min-max training process, the training is difficult to stably converge. For conquering this problem, [57] proposes the WGAN that applies the Wasserstein distance to the loss function for training the GAN. It shows much faster and more stable convergence than the original GAN. Besides data synthetization, GAN or WGAN also shows its potential in solving various inverse problems, like super-resolution [88], inpainting [61] and denoising [89]. However, the reason why GAN or WGAN, which is designed originally for data synthesizing, can be used for inverse problems is still not clearly explained. In this chapter, we propose a novel DNN-based reflection removal method. It is different from the existing methods [7, 25] that we assume multiple input images are available for obtaining the depth information of the scene. It also follows the background gradient regeneration strategy as we have mentioned in Chapter 3. They allow a much robust performance as compared with the existing DNN-based reflection removal approaches. For the proposed algorithm, we firstly use a convolutional neural network (CNN) to estimate the disparity values along the image edges. Following the background gradient regeneration strategy, only the image edges with distinct disparity values will be used to obtain two partial edge maps based on the disparity model shown in Fig. 4.1. After that, a WGAN is used to regenerate the missing background edges from these two partial edge maps. The WGAN contains an auto-encoder and two discriminators. The auto-encoder

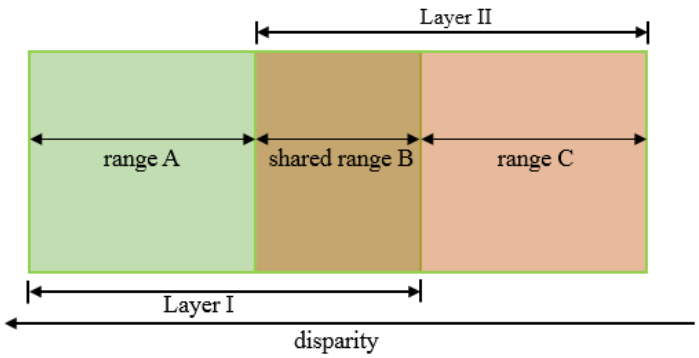


Fig. 4.1. The disparity sandwich model. In this model, the first layer is closer than the second layer to the camera and some of their components may share the same disparity range in the middle. The components in the disparity ranges A and C only belong to the first and second layers respectively. Some of the components of these two layers are mixed in the disparity range B.

regenerates the missing background edges and tries to fool the two jointly trained discriminators to believe they are the real background edges. Finally, all background edges are fed to another CNN for reconstructing the background image. Besides proposing the algorithm, we also try to explain why WGAN combined with a distance function can be used for solving the inverse problems. The flowchart of the entire framework is shown in Fig. 4.2.

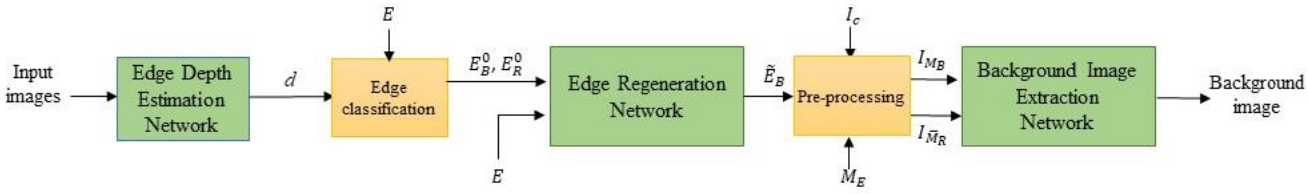
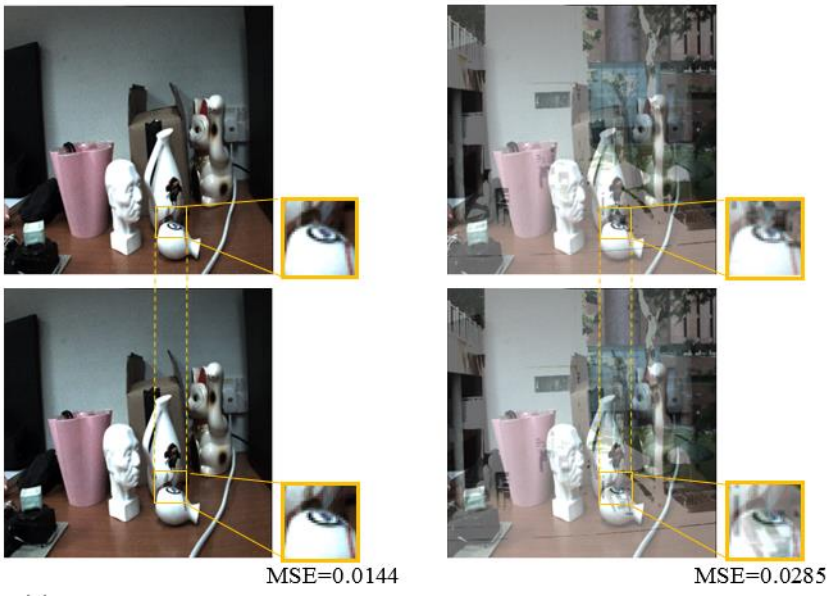


Fig. 4.2. The flowchart of the entire framework.

The rest of the chapter is organized as follows: After the introduction in Section 4.1, we briefly introduce the edge disparity network in Section 4.2 for generating edge disparity map of the input images. In Section 4.3, we present why a WGAN constrained with a distance function can be used for the inverse problems. We then explain how to use it to regenerate the background edges in the shared disparity range. In Section 4.4, we introduce the CNN we have used for extracting the background image from the original image guided by the edge maps. In Section 4.5, we show the experimental and comparison results. Finally, we summarize this chapter in Section 4.6.

4.2. Edge Disparity Network

Disparity estimation has been extensively studied for many decades. The main strategy is to match the corresponding patches in stereo pair or multiple rectified images taken at slightly different viewpoints [32, 90, 91]. However, for images with reflection, the pixels of the background and reflection images are overlapped. It is far more difficult to find patch pairs for estimating the disparities. Fig. 4.3 shows an example when the background image pair is superimposed by another image pair. It can be seen that the matching error of the background patches becomes much larger. It is because the second image pair has different disparity; the pixel shifts of the second image pair are different from the background image pair. However,



(a) Original image pair (b) Image pair with reflection

Fig. 4.3. The image pair before and after superimposed by another image pair. (a) is an image pair. (b) is an image pair after (a) is superimposed by another image pair. Since the disparity of the second pair of images is different from the original image pair, the pixel shifts of this second pair of images are different. Therefore, the matching error increases as shown in the figure.

according to the edge independence property, the strong edges are seldom overlapped. Therefore, instead of matching image patches for every pixel, we propose to estimate the disparities only along the edges. In fact, we have demonstrated in Chapter 3 that we can estimate the edge disparities of light field images by using a sparse regularization process. To improve the computational efficiency, here we train a CNN to achieve the task. In addition, we limit the number of input images to only 5 so that the algorithm can also be used in some array camera systems, which are popular in nowadays mobile devices. The network architecture is shown in Fig. 4.4. The network contains eight layers with 256 channels at the

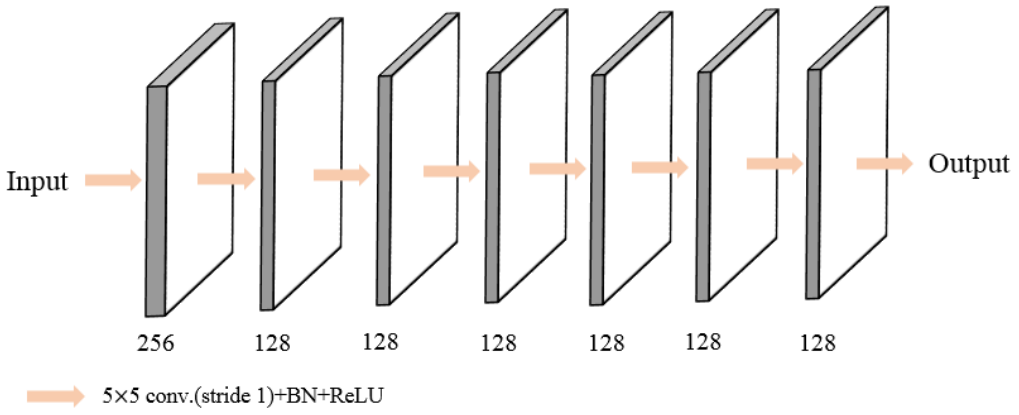


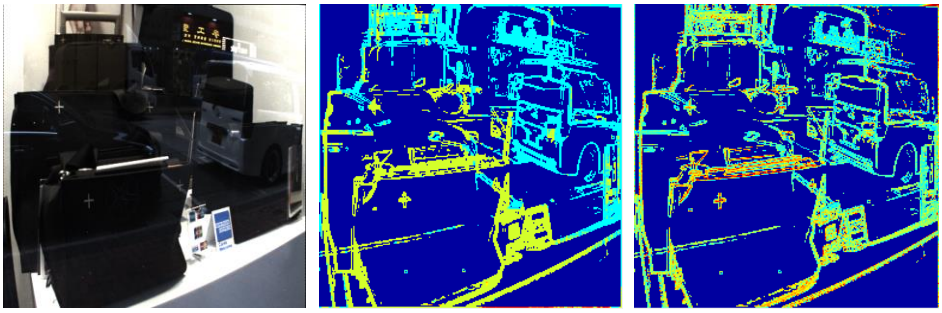
Fig. 4.4. The edge disparity network architecture.

beginning, 128 channels in the middle six layers and one channel at the last layer for outputting the edge disparity map. The kernel size is five. There is also a batch normalization layer and ReLU following every convolutional layer except the last one. We train the network by minimizing the following loss function,

$$L_d = \sum_{n,x} \left\| A_n(x) \cdot I_n(x) - A_n(x) \cdot I_c\left(x + B_{n,c} \cdot d(x)\right) \right\|^2, \qquad (4.3)$$

where d is the disparity; x is the pixel coordinate; n is the index of the input images which are supposed to have been aligned following the orientation of the reference image; c is the index of the reference image which is just one of the input images; A represents the gradient magnitude which lets the loss function focus on the edges; $B_{n,c}$ is the baseline between the reference image and the nth image. Note that in this loss function, we do not need any ground truth disparity map. This unsupervised

training strategy can avoid using the ground truth disparity map, which is always difficult to obtain [92-94]. In Fig. 4.5, we compare our approach with another CNN based method for disparity estimation [95]. In the figure, the disparity values of which the pixel gradient values are below a threshold are discarded. We can see the result of [95] has many errors. For instance, it suggests the top right-hand corner and bottom left-hand corner have similar disparities which are obviously not the case. The errors are caused by the aforementioned problem that [95] estimates the disparities based on the traditional pixel patch matching method, which will have large errors for images with reflection. In contrast, the proposed approach emphasizing on the image edges shows higher accuracy and resolution.



(a) (b) (c)

Fig. 4.5. The edge disparity results. (a) The input image with reflection. (b) The edge disparity map estimated using method [94]. (c) Edge disparity map estimated using the proposed network. In (b) and (c), the red and blue colors represent the large and small disparity values.

4.3. Edges Regeneration Using WGAN

The proposed approach in this chapter also follows the background gradient regeneration strategy. Rather than using a sparse regularization method as in Chapter 3 for edge regeneration, we train a WGAN to achieve the task. We show in this section that a WGAN combined with a distance function is suitable for the inverse problems such as background edge regeneration.

4.3.1. Wasserstein Generative Adversarial Networks

The objective of GAN is to train a generator that synthesizes novel samples which cannot be distinguished from real samples by its discriminator. The training process of GAN can be described as the following min-max game,

$$\min_{G}\max_{D}\mathbb{E}_{x\in\chi}\left[\log\left(D(x)\right)\right]+\mathbb{E}_{z\in\mathcal{Z}}\left[\log\left(1-D\left(G(z)\right)\right)\right], \tag{4.4}$$

where *G* and *D* represent the generator and discriminator respectively. *G* is trained to minimize the loss function for mapping the input z, which follows a distribution $\mathcal{Z}$, to the target *x*, which follows another distribution χ. A discriminator *D* is also jointly trained to distinguish the generated $G(z)$ from the real sample *x* by maximizing the loss function. The goal is to train a generator *G* which can generate fake samples that the discriminator *D* cannot distinguish. Therefore, $G(z)$ will have a distribution very close to that of the real sample *x*. However, such minimax

training is unstable and difficult to converge. As mentioned above, we adopt WGAN in the background edge regeneration. WGAN inherits the ability of GAN but exhibits more stable and fast convergence by using the Wasserstein distance in the loss function. The training process of WGAN can be described as follows:

$$\min_{G} \max_{D} \mathbb{E}_{x \in \chi}[D(x)] - \mathbb{E}_{z \in Z}\big[D\big(G(z)\big)\big]. \tag{4.5}$$

To fully implement WGAN, it also requires to remove the sigmoid activation in the last discriminator layer and clip the weight range of the discriminator to force it to be 1-Lipschitz [57]. With such modifications, we can efficiently train a WGAN to regenerate the background edges.

4.3.2. Bridge from Inverse Problems to WGAN

Originally, GAN is used for synthesizing novel samples but recently, we can also find many applications of GAN or WGAN in solving the inverse problems like super-resolution [88], inpainting [61] and denoising [89]. The reason why GAN or WGAN can work well for recovering images is still not clearly explained. Here, we investigate the reason and build a bridge between the inverse problems and GAN by linking it to the traditional regularization theory. Traditionally, for solving the inverse problems in image processing, we can train an estimator f with the parameters θ by minimizing a 2-norm distance between the estimation output image and the ground truth image as follows:

$$L = \|f(\mathrm{z};\theta) - x\|_2^2, \tag{4.6}$$

where z is the input and x is the ground truth. However, such simple pixel-wise distance minimization often renders the output image blurry and gives low perceptual quality output. In traditional prior regularization theory, it is known that we can produce a better result by adding the prior knowledge of x to the objective function as follows:

$$\min_{\theta} \|f(z;\theta) - x\|_2^2 + p\big(f(z;\theta)\big). \tag{4.7}$$

The prior function p should give low response if $f(z;\theta)$ follows the prior knowledge of x and vice versa. For instance, if the distribution of x is known, we can use it as prior knowledge. Then p should give low response if $f(z;\theta)$ follows the distribution of x. If we consider the generator G of a WGAN is also an estimator, we can rewrite (4.7) as follows:

$$\min_{\theta} \|G(\mathrm{z}) - x\|_2^2 - p\big(G(\mathrm{z})\big). \tag{4.8}$$

Note that the discriminator D of a WGAN is trained to distinguish the generated sample $G(z)$ from the real sample x. It gives high response if it finds $G(z)$ is the same as the real sample and low response if it is fake. Therefore, $D\big(G(z)\big)$ can be used as a prior function as follows:

$$\min_{G} \|G(z) - x\|_2^2 - D\big(G(z)\big). \tag{4.9}$$

To allow the discriminator to give high response to real samples and low response to fake samples, it needs to be jointly trained using the following cost function:

$$\min_{D} D\big(G(z)\big) - D(x). \tag{4.10}$$

It can be seen from the above discussion that a WGAN combined with a distance function can be a special form of the traditional prior regularization method for solving the inverse problems. It, however, can give much better performance than the general prior regularization methods since usually a huge image dataset will be used for training the generator and discriminator.

4.3.3. Partial Edge Maps as Hints

In the last sub-section, we have explained why a WGAN combined with a distance function can be used for solving the inverse problems. In this section, we show how we can use WGAN to regenerate the background edges, which is a typical inverse problem. As it is discussed in Chapter 3, we can easily separate the background and reflection edges if they have distinct depth ranges. However, if they have close depth values, or even share the same depth range, it will be very difficult or even impossible to separate them just from their depths. Any errors in the separation will either remove the background components or include the reflection

residues in the resulting background image.

Following the background gradient regeneration strategy, we first classify the edges that have distinct depth values. They are extracted to form two partial edge sets. The edges of which the depth values cannot be used for their classification will be ignored. Then we make use of a WGAN and the two partial edge sets to regenerate the missing background edges to form the complete background edge set. More specifically, as described by the sandwich model in Fig. 4.1, we only extract the edge components in disparity ranges A and C with very large and small disparity values (i.e. small and large depth values respectively). Here, we use $\dot{E}_1$ and $\dot{E}_2$ to denote the extracted edge components supposed to belong to the first and second image layers respectively (one of them is the background, the other is the reflection). To determine the thresholds for defining the boundaries of ranges A and C of the sandwich model shown in Fig. 4.1, we apply the K-means method [79] to the edge disparity values similar to that in Section 3.3.2. Two clusters with two centers are then obtained. The values around these two centers are selected as the two thresholds. The edge components with disparity values above and below the large and small thresholds respectively will be extracted as $\dot{E}_1$ and $\dot{E}_2$. With the hints of $\dot{E}_1$ and $\dot{E}_2$, we can use a WGAN to regenerate the missing background edges and form the complete background edge set.

4.3.4. Edge Map Reconstruction Using WGAN

Indeed, the reason why a WGAN can regenerate the missing background edges is due to the different distributions between the edges of normal images and those with reflection. Let us use an experiment to illustrate this argument. In the experiment, we use 100 real-life images (each with reflection) from the benchmark dataset SIR2 [66]. Since the ground truth background images are also provided in SIR2, we can compute the histograms of the images with and without reflection. The results are shown in Fig. 4.6. We can see that the edges of the images with reflection have a different bias and skewness compared with the images without

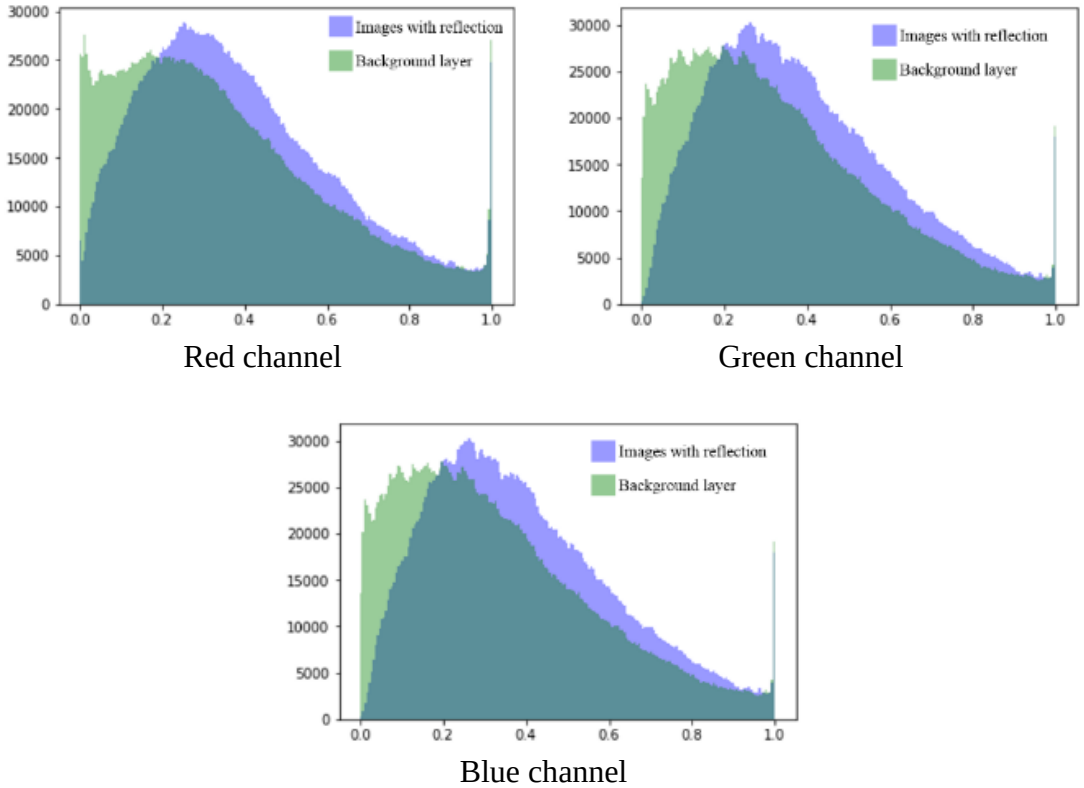


Fig. 4.6. The distributions of the edges for images without reflection (green), and with reflection (blue) in the red, green and blue channels, respectively.

reflection. It is because of the additive light arisen from the superimposition of the uncorrelated background and reflection images. A WGAN can be trained to generate edges following the distribution of the background edges with reflection. The architectures of the generator and discriminator of the proposed WGAN are shown in Fig. 4.7. The generator is an auto-encoder and concatenations are added to connect the down-sampling and up-sampling sides for increasing the resolutions of the up-sampling side features [48]. Two discriminators are built to distinguish the generated background and reflection edges. These two discriminators have the same structure with six down-sampling blocks. We stack the original image edges E, and the two partial edge sets $\dot{E}_1$and $\dot{E}_2$ as the input signal z and feed to the proposed WGAN. We expect the partial edge sets $\dot{E}_1$and $\dot{E}_2$ can be the hints for the network

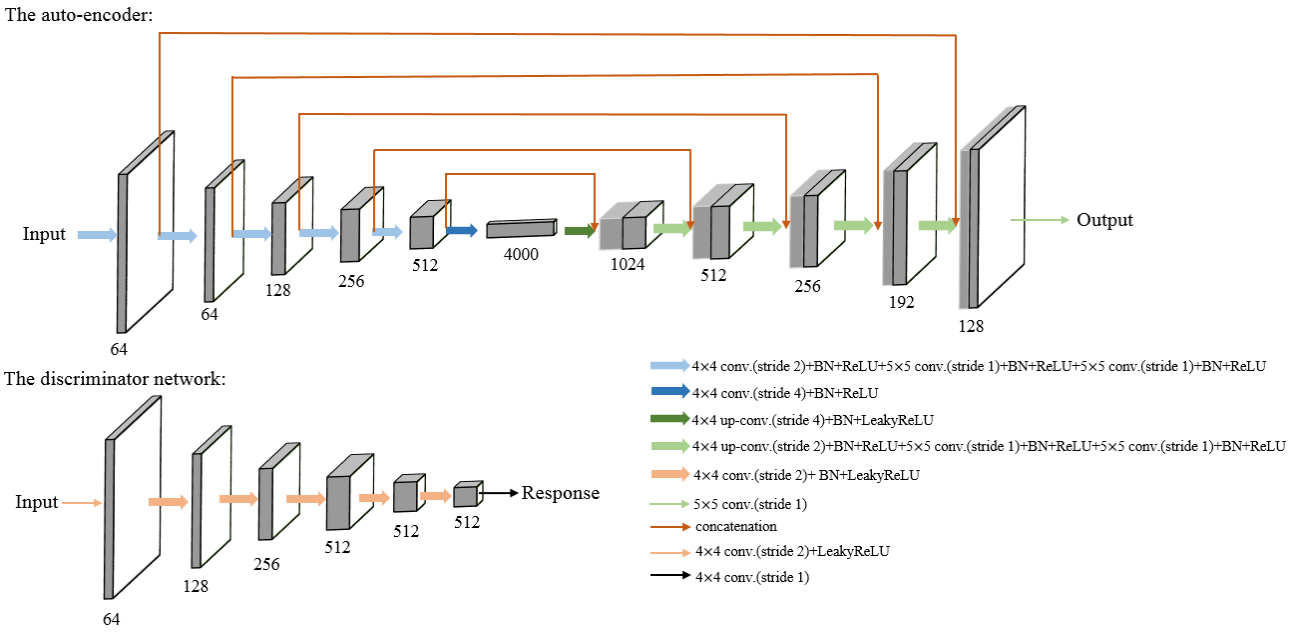


Fig. 4.7. The network architectures of the generator (auto-encoder) and the discriminator.

to regenerate the missing background edges and form the complete background edge set.

For training the networks, we first define an L2 norm loss function as the first term in (4.8) for forcing the regenerated edges to be similar to the ground truth as follows:

$$L_{rec}^{E} = \|G^{E}(z) - E_1\|_2^2, \tag{4.11}$$

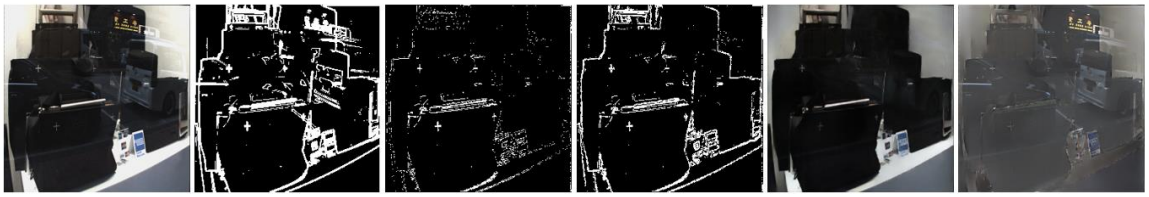
where G^E is the generator that regenerates the missing background edges and gives the complete background edge set, E_1 is the ground truth background edges. Then we use two adversarial loss functions for forcing the output of G^E to follow the distribution of background edges as follows:

$$L_{adv1}^{E} = -D_1^E\big(G^E(z)\big); \tag{4.12}$$

$$L_{adv_2}^{E} = -D_2^E\big(E - G^E(z)\big). \tag{4.13}$$

D_1^E and D_2^E are the two image edge discriminators. They act as the prior functions in the regularization process as discussed before. If $G^E(z)$ and $\big(E - G^E(z)\big)$ are close to the background and reflection edges, L_{adv1}^{E} and $L_{adv_2}^{E}$ will give a low response. The overall loss function for the generator G^E is as follows:

$$L^E = L_{rec}^{E} + \lambda_1\big(L_{adv_1}^{E} + L_{adv_2}^{E}\big). \tag{4.14}$$



(a) (b) (c) (d) (e) (f)

Fig. 4.8. The intermediate results of the proposed algorithm. (a) The input image with reflection. (b) The entire edge map M_E. (c) The initial partial background edge map $M_{\dot{E}_1}$. (d) The estimated complete background edge map using the proposed WGAN. (e) The reconstructed background using the edge map shown in (d). (f) The reflection obtained by deducting the background image from the input image. The mean value of (f) is adjusted to the input images for clear visualization.

The discriminators D_1^E and D_2^E can be trained with the following loss functions,

$$L_{D_1}^E = D_1^E\big(G^E(z)\big) - D_1^E(E_1); \tag{4.15}$$

$$L_{D_2}^E = D_2^E\big(E - G^E(z)\big) - D_2^E(E_2), \tag{4.16}$$

where E_2 is the ground truth edges of the second layer (reflection). We jointly train G^E, D_1^E and D_2^E until they converge. We define the regenerated edges given by the generator as $\tilde{E}_1 = G^E(z)$. The final background binary edge map $\widetilde{M}_1$ is obtained by thresholding $\tilde{E}_1$ with the value 0.05. Fig. 4.8 shows an example of $\widetilde{M}_1$ obtained from the proposed algorithm. We denote the binary edge maps for E and $\dot{E}_1$ as M_E and $M_{\dot{E}_1}$ respectively. It can be seen that M_E contains both the background and reflection edges and the initial partial edge map $M_{\dot{E}_1}$ only contains a portion of the

background edges. From Fig. 4.8(d), we can see that the proposed WGAN successfully estimates the background edges and ignores the reflection edges.

4.4. Background Image Extraction Based on Edges

As we have demonstrated in Chapter 3, we can use different optimization techniques [10, 31] to extract the background image from the original one guided by its edge map. However, the involved iterative optimization processes with huge matrices are very time-consuming. Considering the fast speed of DNN over the traditional optimization processes, here we also use a DNN to generate the background image guided by its edges.

To the best of our knowledge, there are very few DNN approaches for extracting the background images based on their edges. The only one we are aware of is the I-CNN in the method CEILNet [7]. However, the performance of I-CNN is rather unstable that the resulting image can lose many background details while keeping the reflection residual. It is because I-CNN works based on the assumption that the reflection is blurry. When the image contains reflection with strong edges, it is difficult for I-CNN to totally remove them. To solve the problem, we develop a new Background Image Extraction Network, which has an auto-encoder structure the same as that used for edge map estimation in Fig. 4.7 (upper). To remove the strong edges of the reflection remained in the resulting image, we pre-process the input image by removing the reflection edges. To do so, we first compute from the

estimated background edges $\widetilde{M}_2 = M_E - \widetilde{M}_1$, which mainly indicates the positions of the reflection edges. Then we obtain an image $I_{\widetilde{M}_2} = \left(I_c - I_c \cdot \widetilde{M}_2\right)$, which is the original reference image without the reflection edges. We stack $I_{M_1} = \left(I \cdot \widetilde{M}_1\right)$ and $I_{\bar{M}_2}$ as the input signal z and fed to the proposed Background Image Extraction Network. For training the network, we first use the following L2-norm loss function to confine the resulting image to follow the ground truth background at the pixel level,

$$L_{rec}^{I} = \|G^{I}(z) - I_1\|_2^2, \tag{4.17}$$

where I_1 is the ground truth background image and $G^I(z)$ is the network output give the input z. In addition, we add the following perceptual loss function which can ensure the resulting image to follow the human perception,

$$L_P^I = \left\|V\left(G^I(z)\right) - V(I_1)\right\|_2^2, \tag{4.18}$$

where V represents the feature maps of the 14th layer of the pre-trained VGG-16 network. Using the intermediate responses of high-level features is an effective way to measure the perceptual similarity [65]. Thus, the following overall loss function is used to train the proposed Background Image Extraction Network:

$$L^I = L_{rec}^I + \lambda_2 L_P^I \tag{4.19}$$

Fig. 4.8(e) and (f) show an example of the background image and its residual (reflection layer) generated using the estimated background edge map shown in Fig. 4.8(d). We can see that the network successfully extracts the background image from the original image guided by its edge map. To show the effectiveness of the proposed network, Fig. 4.9 (i) shows a simulation case using the I-CNN and the proposed Background Image Extraction Network respectively. To isolate the performance in background image extraction, both the proposed Background Image Extraction Network and I-CNN use the estimated background edge map generated by the Edge Regeneration Network. Since the reflection is not particularly blurred in the synthesized image, we can see that the strong edges of the reflection remain in the result of I-CNN. We also notice that many background details are missing. The proposed Background Image Extraction Network can well recover the background components while removing the reflection since there is no assumption about the blurriness of the reflection and we also incorporate the human perception in the training process. More detailed comparisons can be found in the next section.

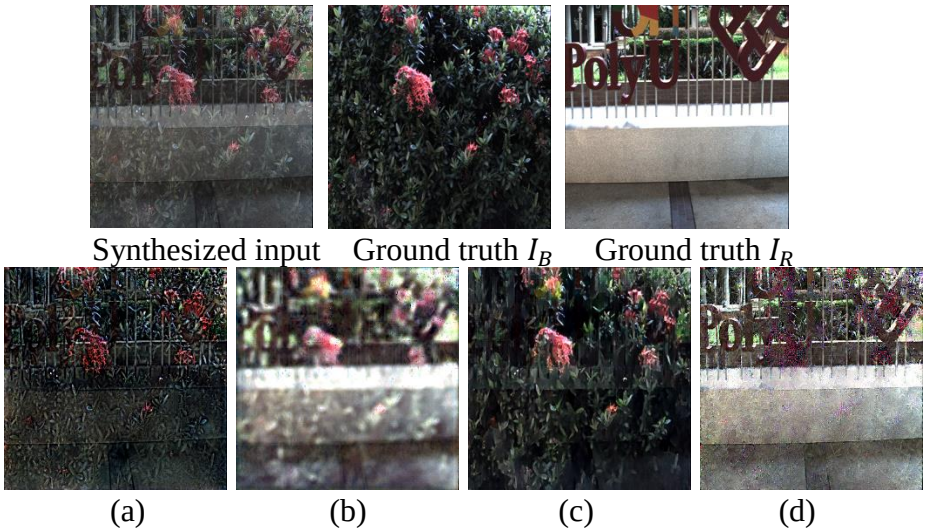


Fig. 4.9 (i). The background images generated by using I-CNN and the proposed background image extraction network. (a) and (b) are the generated background image and its residual respectively using the Edge Disparity Network + Edge Regeneration Network + I-CNN. (c) and (d) are the generated background image and its residual respectively using the Edge Disparity Network + Edge Regeneration Network + the background image extraction network.

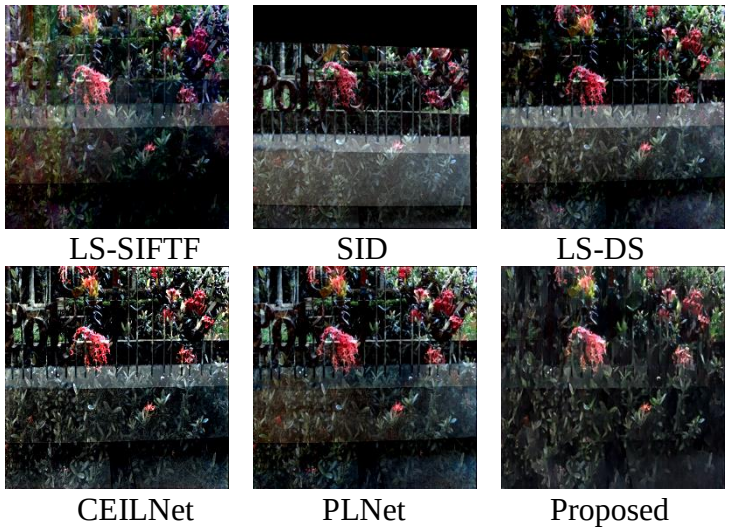


Fig. 4.9 (ii). The background images obtained from using different approaches.

4.5. Experiments and Evaluation

For evaluating the performance of the proposed algorithm, we compare it with the state-of-the-art methods both quantitatively and qualitatively. Before showing the comparison results, let us clearly explain the details of training the networks and how the comparisons are carried out.

4.5.1. Training Details

We assume that five images of slightly different viewing angles are available as the input of the proposed Edge Depth Estimation Network. For convenience, we obtain the required images for the training of the network by using a light field (LF) camera, which can directly capture array images of the target scene in a single shot. We extract five of the captured images and input them to the network after alignment to the same viewing angle. For quantitative evaluation, we synthesize the required training images with reflections by randomly adding two sets of LF images together with different weights. More specifically, we capture 318 sets of LF images and resize them to 256×256 pixels. They are randomly added together and finally, 112,225 images with reflection are synthesized as the training samples. To further increase the training samples, we augment the data by cropping the images into many 128×128 patches at every interval of 16 pixels, then randomly flipping and rotating them at every 90 degrees. The Edge Disparity Estimation Network is trained

using the ADAM solver [96] with learning rate 2×10^{-5}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. For training the Edge Regeneration Network and the Background image Extraction Network, we only use the flipped and rotated images to augment the dataset. It is because a cropped patch may not have sua fficient amount of edges for training, as edges are sparse in nature. Similar to [57], we use RMSprop solver [97] to train the generator and the discriminators of the Edge Regeneration Network with learning rates 2×10^{-4} and 2×10^{-5} respectively. For the Background Image Extraction Network, we also use the RMSprop solver [97] with learning rate 2×10^{-4} for its training. The parameters λ_1 and λ_2 are set as 2.5×10^{-3} and 1.25 respectively. The training and testing are both performed on a desktop computer with Core i7 7820X CPU using a GTX 1080 Ti.

4.5.2. Quantitative Evaluation

A quantitative comparison is made between the proposed algorithm and a few recent methods, including the traditional optimization based approaches such as SIFT flow (LS-SIFTF) [10], superimposed image decomposition using low rank (SID) [9], image layer separation based on the disparity signs (LS-DS) [12]; as well as two other CNN-based methods CEILNet [7] and PLNet [25]. Except for LS-DS which is implemented by us according to their paper, other methods are implemented by the source codes published in their websites. Because LS-SIFTF and SID require relatively large disparities between images, the images we captured

using the LF camera with small baselines cannot be directly used to test these two approaches. To solve the problem, we put the LF camera on a tripod and shift the camera up and down to five preset heights. For each height, we capture one set of LF image for each scene. Using only the central view of each LF image, we can obtain, for each scene, five images of relatively large disparities. We capture 20 groups of such images of different scenes and create ten groups of images with reflections by adding ten of them to the other ten with the weights 0.6 and 0.4. These images are used to test the LS-SIFTF and SID methods. On the other hand, the method LS-DS requires LF images as input. For each group of LF image captured, this time we just use one of them for each scene. We extract the central 5×5 images of each LF image so that we have twenty sets of 5×5 images. They are mixed with a similar method as mentioned above to form ten testing images (with reflection) for LS-DS. CEILNet and PLNet are single-image reflection removal methods, thus we directly input the central view of each LF image to test these networks. Because LS-SIFTF, SID, LS-DS can only perform well with relatively higher resolution images, we feed images with resolution 625×434 to those methods and resize their results to 256×256 pixels for comparison. CEILNet and PLNet are directly fed with images with size 256×256 pixels. Fig. 4.9 (ii) shows one of the comparison results based on the testing images mentioned above. It can be seen that the proposed algorithm gives the best result compared to other methods. The average PSNRs of all the testing algorithms are shown in Table 4.1. Because the results of LS-SIFTF and SID can have large biases in the mean value which can

give very low PSNRs, we normalize the mean values of all the results to be the same as the ground truths. As shown in Table 4.1, the proposed method significantly outperforms the other competing methods. It is because all other methods have different assumptions about the input images. For instance, LS-SIFTF requires the gradients of the background to be much larger than the reflection; SID requires the background to be planar; LS-DS requires the background and reflection to be at different sides of the focal plane and the normal line of the camera must be perpendicular to the scene; CEILNet and PLNet have a stringent assumption that the reflection must be blurry. They all introduce the errors to the reflection removal process in case the input images do not follow exactly the respective assumptions. We also evaluate the influence of the input terms I_{M_1}, $I_{\bar{M}_2}$ for the Background Image Extraction Network. The PSNR values of the generated background images are shown in Table 4.1. Since I_{M_1} emphasizes useful edges and $I_{\bar{M}_2}$ hides useless reflection edges, the Background Image Extraction Network can achieve the best performance when both of them are input to the network.

Method	**PSNR of the recovered background (dB)**
Synthetic input	13.094
LS-SIFTF [10]	18.912
SID [9]	15.488
LS-DS [12]	18.855
CEILNet [7]	17.714
PLNet [25]	19.092
Proposed w/o Edge Regeneration	22.774
Proposed w/o discriminators	23.224
Proposed w/o $I_{\bar{M}_2}$	23.340
Proposed w/o I_{M_1}	23.220
Proposed	**24.031**

Table 4.1 The average PSNR in dB of the resulting background images generated by different methods with respect to their ground truths.

4.5.3. Effectiveness of The WGAN for Background Edge Estimation

We also evaluate the effectiveness of the WGAN contributing to the estimation of the background edge map $\widetilde{M}_1$. We compare the proposed WGAN (the auto-encoder jointly trained with the discriminators using the loss function (4.14)) and the same auto-encoder without the discriminators trained only using the loss function (4.11) to estimate $\widetilde{M}_1$. We investigate the regenerated edge distributions of the proposed WGAN and only the auto-encoder trained without the adversarial terms. Fig. 4.10 shows the comparison results. In the figure, the histograms and the fitted distributions of the regenerated background edge components $\tilde{E}_1$ are shown. We can see that the $\tilde{E}_1$ estimated by the proposed WGAN has the distributions very

close to the ground truths. It is because the proposed WGAN tends to constrain the generated samples to follow the distributions of the ground truths, such that the discriminators cannot distinguish them from the real ones. Without the discriminators, the network can only minimize the mean square difference between the generated result and ground truth. The distributions may deviate from the ground truth. Table 4.1 also shows the PSNR of the final background images generated with the Edge Regeneration Network but without using the discriminators. It can be seen that without the discriminators, the PSNR decreases by about 1.3 dB.

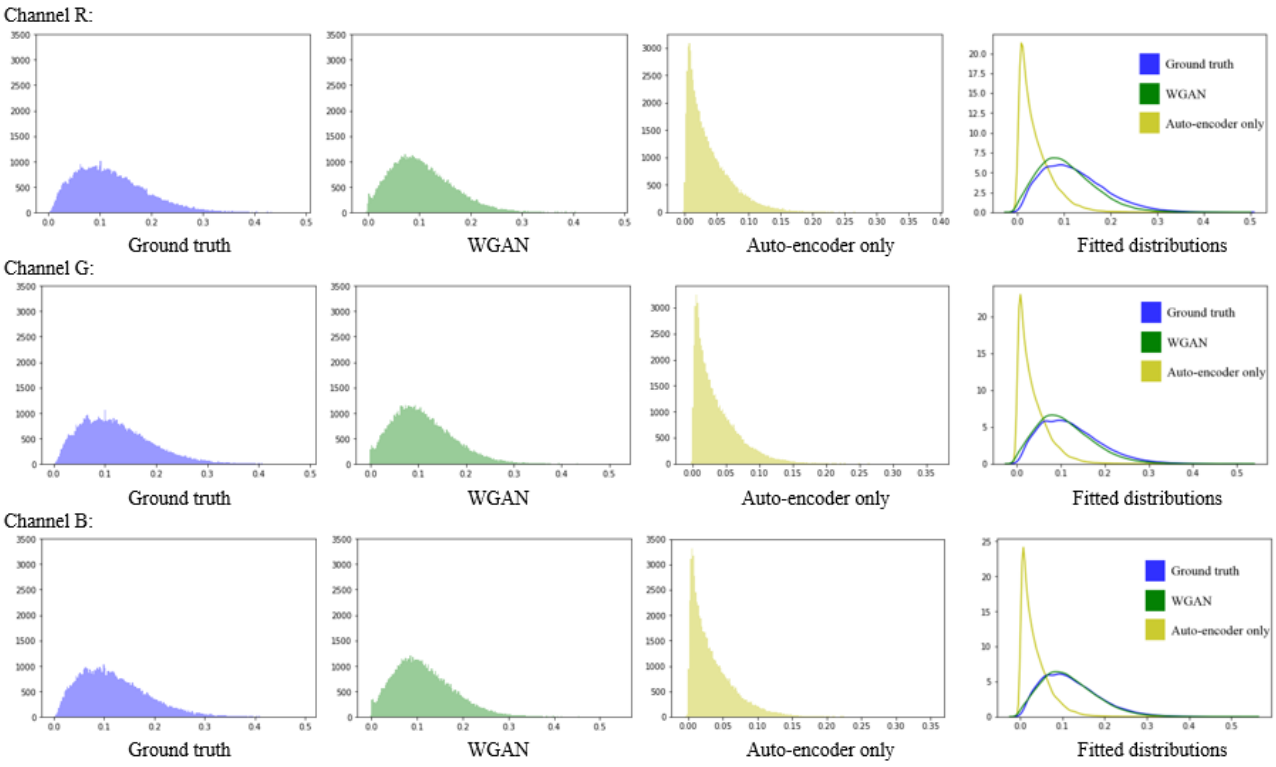


Fig. 4.10. The histograms and fitted distributions of the estimated background edges $\tilde{E}_B$ given by the proposed WGAN and only the auto-encoder at different color channels. The first column shows the histograms of the ground truth edges; the second column shows the histograms of the edges generated from the proposed WGAN; the third column shows the histograms of the edges generated by the auto-encoder trained without the discriminators; the last column is their fitted distributions. Different rows represent different color channels.

4.5.4. Qualitative Evaluation

For qualitative evaluation, we compare the visual quality of the extracted background and reflection images using different methods. In this evaluation, the testing images are directly captured in a real-life environment, such as in front of a glass, etc., so that a reflection of an unwanted scene is added to the image. Since we do not have the ground truth background of these images, we can only evaluate the performance by visual inspection. The comparison results are shown in Fig. 4.11. For LS-SIFTF, it cannot correctly separate the reflections from the backgrounds when both of them have strong gradients. Its performance is acceptable only for the fourth scene where the reflection is relatively weak. However, there are still many residuals remained in the regions with strong reflections. For SID, it assumes the background layer is planar and uses the homography to register the background while blurring the reflection. Thus, it can only deal with planar background scenes. In fact, even the background is planar, the features of the reflection can affect the homography estimation. Therefore, we can see that the resulting images are blurry due to inaccurate registration. For LS-DS, it requires the background and reflection to have absolutely different depth ranges and it also requires the camera to be perpendicular to the target scene. Such stringent requirements to the pose and photography environment introduce much difficulty to remove the reflection in practice. For CEILNet and PLNet, they assume the reflection is much smoother than the background. They fail to remove the strong and sharp reflection components in

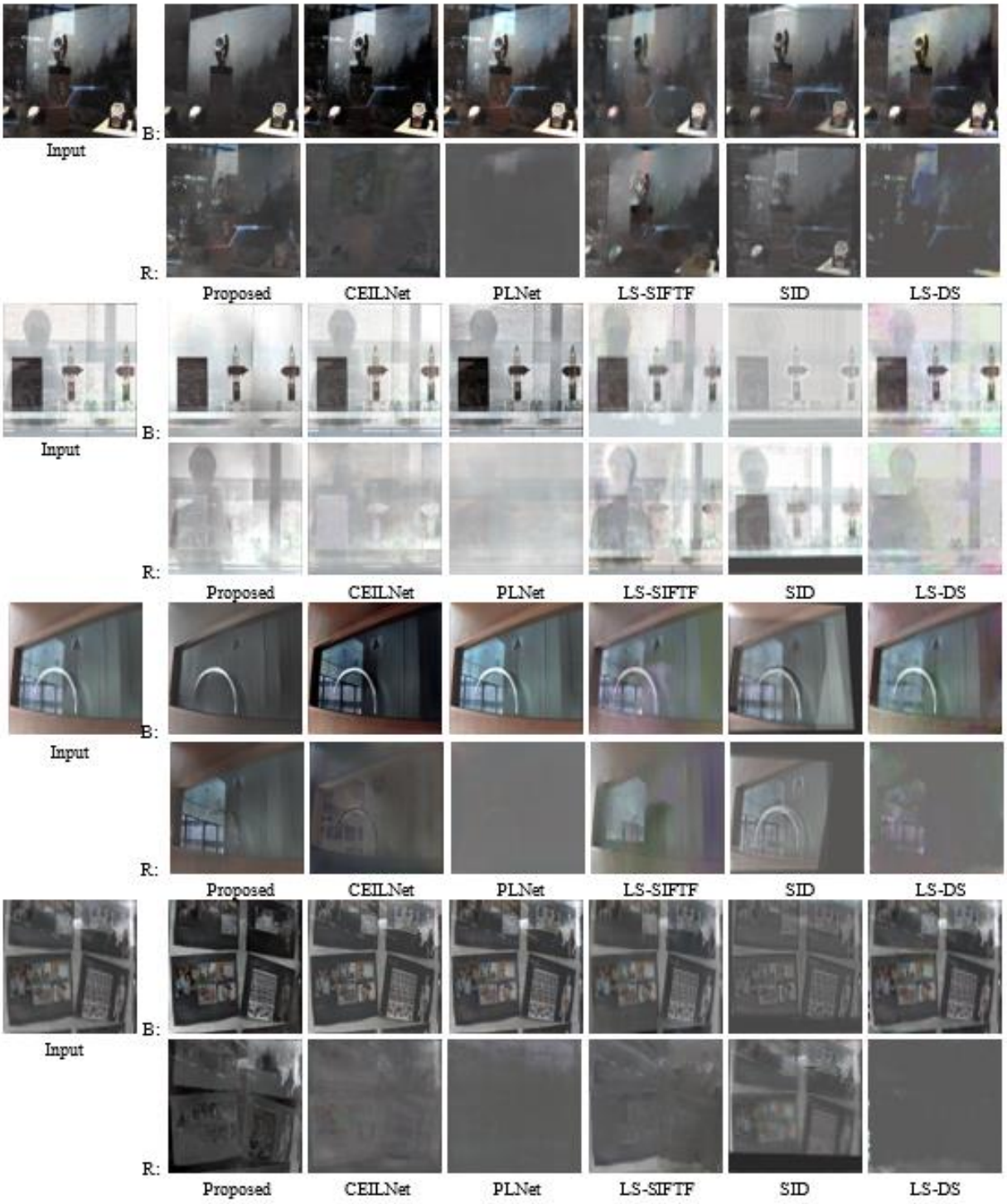


Fig. 4.11. The qualitative comparison results of different methods.

the images. Without the abovementioned limitations, the proposed method

Method	Average Time
LS-SIFTF	130.59 s
SID	58.95 s
LS-DS	17.01 s
LS-LFGS	69.51s
CEILNet	0.82 s
PLNet	1.15 s
Proposed	**0.88 s**

Table 4.2. The average execution times of different methods

successfully extracts the backgrounds and separates the reflections for all scenes.

4.5.5. Running Time

We also compare the running times of different testing methods by taking the average processing times of these methods on five real-life images. This time we only evaluate the computational cost regardless of performance. Therefore, we feed images with size 256×256 to all methods. The results are shown in Table 4.2. We can see that the traditional optimization-based methods LS-SIFTF [10], SID [9], LS-DS [12] and LS-LS-LFGS [67] are much slower compared to the DNN-based methods, such as CEILNet [7], PLNet [25] and the proposed one. It is because those optimization-based methods require iterative operations on huge matrices, which can take a very long time. In contrast, CNN based approaches with parallel-conducting kernel architectures are very suitable to GPU. They can efficiently utilize the massively parallel structure of GPU and complete the whole process within only one second.

4.6. Summary

In this chapter, we proposed a novel DNN-based reflection removal algorithm following the background gradient regeneration strategy. For a target scene, the proposed algorithm only requires 5 images at different viewing angles as the input, which is in contrast to the approach we proposed in Chapter 3 where the light field image is needed. The relaxed hardware requirement allows the proposed algorithm to be more readily used in some array camera systems, which is increasingly popular in nowadays mobile devices. For the proposed algorithm, the input images are first aligned to the referenced viewing angle and fed to the Edge Disparity Network for estimating the edge disparities. Only the edges with distinct disparities are used to obtain two partial edge maps. The edges of which the disparity values are not distinct enough for their classification are ignored. They are regenerated by the Edge Regeneration Network which is implemented by a WGAN with one generator and two discriminators. The complete background edge set is then fed to the Background Image Extraction Network for extracting the resulting background image. Comparing to other single-image DNN-based methods, the proposed algorithm does not require the reflection to be blurry. Comparing to the traditional multiple-image optimization-based methods, the proposed algorithm is more robust since it does not have the various assumptions on the images and imaging environment. It has shown superior performance compared to the state-of-the-art methods. Even comparing with the method we proposed in Chapter 3, this DNN-based method also shows

significant improvement on the computation speed, particularly when implementing with GPUs.

Chapter 5. Single-Image Reflection Removal via a Two-Stage Background Recovery Process

(This chapter is extracted from my paper [98]: Tingtian Li, and Daniel P.K. Lun, "Single-Image Reflection Removal via a Two-Stage Background Recovery Process," IEEE Signal Processing Letters, 2019).

In Chapter 3 and 4, we proposed two multiple-image reflection removal methods using the traditional optimization and deep learning methods respectively. However, in a practical situation, we are often required to deal with the reflection removal problem with only a single image on hand. In this chapter, we propose a novel deep learning-based reflection removal method using only a single image as the input. Due to the use of the background gradient regeneration strategy, the proposed algorithm gives superior performance compared with other state-of-the-art single-

image DNN-based reflection removal methods. It is particularly suitable for the images with blurry reflection, which is not uncommon in daily photography.

5.1. Introduction

As mentioned in Chapter 4, many existing single-image reflection removal methods are inspired by [8] that assumes the reflection is defocused and has a distinct distribution from the focused background. It is true that, in daily photography, the background scene, which is the interest of the photographer, is often focused while the reflection scene is not. The reflection scene is thus often blurry as shown in the image and has a short-tailed distribution [8]. Based on this distinct distribution, researchers develop different methods including the DNN methods [7, 25] to recognize and remove the blurry reflection components. For example, [7] firstly uses a convolutional CNN to distinguish the sharp background edges and then uses another CNN to reconstruct the background. [25] also trains a CNN to recognize the blurry reflection components by minimizing a VGG perceptual feature distance [65]. Although the assumption that the reflection layer is blurry is valid in many situations, we notice that the strong edges of the blurry reflection can still have high gradient values. One example is shown in Fig 5.1(a) (the circled region). In this case, these DNN-based methods [7, 25] will mistakenly treat these high gradients reflection components as the background components and

leave them in the resulting image as shown in Fig 5.1(b) and (c).

(a) Input (b) CEILNet [7] (c) PLNet [25]

(d) B_{ini} (e) B_{ini} without L_{FR} (f) $I - B_{ini}$

(g) Confidence map C_{rf} (h) B_{final} (i) Ground truth

Fig. 5.1. A reflection removal example. (a) The original image with reflection. Note that the reflection is blurry due to defocus. (b) and (c) Results of the traditional single-image DNN-based approaches. (d) to (g) The intermediate results of the proposed algorithm. (h) The final result of the proposed algorithm. (i) The ground truth background image. For visualizing the estimated initial reflection image clearly, the intensity of (f) is scaled up by two times.

In this chapter, we present a novel two-stage approach to remove the reflection using deep neural networks. The approach follows the background gradient regeneration strategy such that it first aggressively removes the reflection components in the image to ensure that only the background components remain in the image. Since such aggressive reflection removal process can accidentally remove the background components also, a refinement stage is carried out to regenerate the missing background components. To achieve the above strategy, we propose at the first stage of the algorithm to include a feature reduction term in the loss function when training a CNN to achieve the abovementioned aggressive reflection removal process. Then at the second stage of the proposed algorithm, we use the initial background estimation result to generate a confidence map for identifying the strong reflection and background gradients. A generative adversarial network (GAN) is used to reconstruct the background image from the classified gradients. The GAN can also help in regenerating the background gradients that are accidentally removed in the first stage. Experimental results show that the proposed algorithm can give superior performance compared to other single-image deep learning-based reflection removal methods. It shows a strong reflection removal capability even when the reflection scene contains strong gradients, which are often problematic to the traditional single-image reflection removal approaches.

5.2. Initial Background Estimation with Feature Reduction Term

Perceptual features are widely used in deep-learning approaches for solving inverse problems [61, 65, 99]. Compared to the pixel-wise intensity, minimizing the perceptual feature distances can generate an image closer to the human perceptual expectation. The perceptual features can be obtained by extracting the intermediate layer features of a pre-trained network such as VGG-16, VGG-19 [5] trained on a large dataset. In fact, [25] also tries to remove the blurry reflection by minimizing the perceptual feature distance. Because this method highly depends on the assumption that the reflection components are blurry, it will fail when some parts of the reflection still show high gradient values. Just the perceptual feature distance is not enough to totally remove the reflection. For better solving this problem, in this section, we investigate the perceptual feature properties of the images with reflection and propose to include a feature reduction term in the loss function when training the network for further suppressing the reflection.

When an image I_2 is superimposed on another image I_1, the resulting image I will contain the textures from both I_1 and I_2. Intuitively, such increase in texture will lead the superimposed image I to have more perceptual features than the

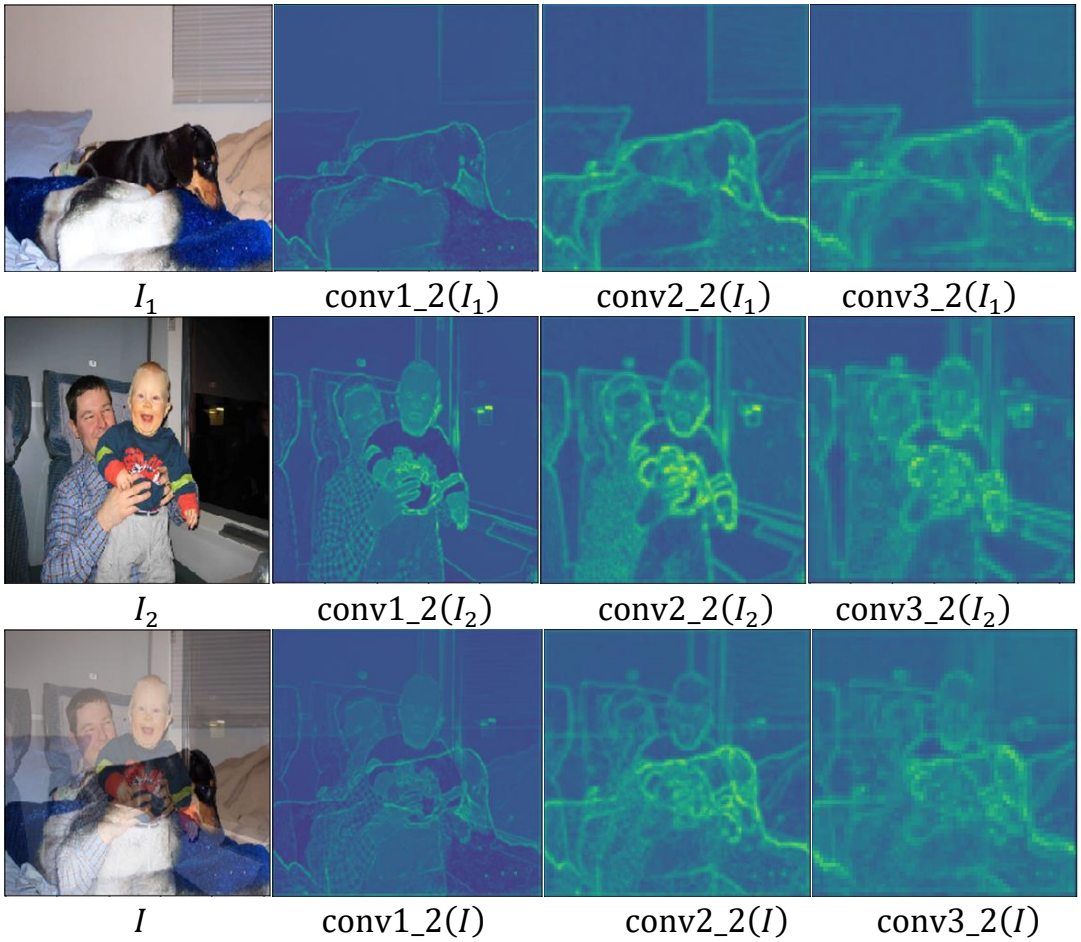


Fig. 5.2. The VGG-19 perceptual feature magnitudes of the superimposed image I and two single layer images I_1 and I_2 at '$conv1\_2$', '$conv2\_2$', '$conv3\_2$' layers, where $I = \alpha * I_1 + (1 - \alpha) * I_2$, $\alpha = 0.6$. The perceptual feature magnitudes of each image as shown in the figure are obtained by adding the perceptual feature magnitudes generated by a VGG-19 network across all channels at the denoted layers.

original image I_1. For validating this, we show in Fig. 5.2 the summed perceptual feature magnitudes of an image with reflection across all channels at different layers. Here, we use the VGG-19 network which is pre-trained on ImageNet [100] to produce the perceptual features. We can see the superimposed image I will contain both the perceptual features of I_1 and I_2. Based on this observation, it is understood that a good reflection removal process should also minimize the perceptual features

in the resulting image. Thus, we propose to include a feature reduction term in the loss function when training the network for reducing the low-level perceptual features in the resulting image. For the 1st stage of the proposed method, a CNN is trained with a loss function L_{ini} as follows:

$$L_{ini} = L_{rec} + L_{FR}. \tag{5.2}$$

$$L_{rec} = \sum_{i=1}^{5} \lambda_1 \left\| \phi_i\left(F_1(I)\right) - \phi_i(I_B) \right\|_2^2 + \lambda_2 \| F_1(I) - I_B \|_1 \tag{5.3}$$

$$L_{FR} = \sum_{i=1}^{3} \lambda_2 \left\| \phi_i\left(F_1(I)\right) \right\|_1 \tag{5.4}$$

where Φ_i denotes the features at 'conv(i_2)' layer of a VGG-19 network pre-trained on the ImageNet dataset [100]. I_B is the ground truth background image. λ_1, λ_2, and λ_3 are the hyper-parameters, which are chosen as 3, 0.4 and 3, respectively, in our experiments. F_1 represents the proposed CNN. So $B_{ini} = F_1(I)$ represents the initial estimation of the background image. L_{ini} in (5.2) consists of two loss functions L_{rec} and L_{FR}. L_{rec} serves to preserve the background. It is a weighted sum of the feature distance and pixel-wise distance from the background ground truth as shown in (5.3). Since the background images we used to train the network are all sharp and clear, L_{rec} in effect guides the network to remove the pixels or perceptual features come from the blurred parts of the image. But if there exist some high gradient components in the blurred regions, the network will be confused. It will keep the

components and perhaps also the neighboring pixels. To solve the problem, we propose to incorporate a feature reduction term L_{FR} as shown in (5.4) when training the CNN. It gives the total feature magnitudes of the first few layers of a VGG-19 network with B_{ini} as the input. It serves to minimize the low level perceptual features of B_{ini}. Since L_{FR} will lead to the suppression of all features and L_{rec} will try to preserve the background features, it ends up that the reflection features will be suppressed more comparing to the background features. More importantly, for the high gradient components in the blurred regions, L_{FR} together with L_{rec} will let the network have a stronger power to remove them although it is at the expense of the sharpness of the background layer since the gradients of the background will also be slightly reduced. Fig. 5.1(d) and (e) show a comparison between L_{ini} with and without L_{FR}. We can see that if L_{FR} is not included, the result in Fig. 5.1(e) is similar to Fig. 5.1(b) and (c). There are obvious reflection edges remaining in the result. On the contrary, the one including L_{FR} in Fig. 5.1(d) has much weaker reflection edge residuals. However, as expected, including L_{FR} may also result in the removal of some background feature components, which leads to a blurrier background image than the ground truth. In the next section, we will discuss the second stage of the proposed algorithm for refining the background.

5.3. Background Refinement at The Second Stage

The reduction of the low-level features in B_{ini} renders the attenuation of its gradient values. Interestingly, it provides us useful information to identify the strong gradients of the background and reflection layers. In fact, as discussed in [10, 67], the background layer can be reconstructed from its strong gradients, while those flat regions with weak or none gradients can be easily inferred by the networks or optimization processes. Now, let us consider the residue of the initial background estimate, i.e. $(I - B_{ini})$. It contains mainly the reflection layers plus the attenuated background gradients as shown in Fig. 5.1(f). Comparing $(I - B_{ini})$ with B_{ini}, the attenuated background gradients in $(I - B_{ini})$ overlap with the background gradients in B_{ini}. And according to the gradient independence property [9, 10, 67], the strong gradients of the background and reflection layers seldom overlap since they are often uncorrelated. It means that at the positions where the strong reflection gradients in $(I - B_{ini})$ are found, we will not find any strong background gradients in B_{ini}. Based on the above, we define a confidence map for identifying the strong reflection gradients as follows:

$$C_{rf} = log\left(\frac{G_{I-B_{ini}}}{G_{B_{ini}} + \varepsilon} + 1\right) \cdot M \tag{5.5}$$

where G represents the gradient magnitude, ε is a very small constant. M is a mask which has the value of 1 for those pixels in I with the Sobel gradient magnitude

larger than 1, and 0 otherwise. It masks out only the positions in I where strong gradients are found for the subsequent operations. As mentioned above, at the positions where $G_{I-B_{ini}}$ contains strong reflection gradients, $G_{B_{ini}}$ will have small or even 0 values. And at the positions where $G_{I-B_{ini}}$ contains the attenuated background gradients, $G_{B_{ini}}$ will have the original background gradients which have larger values. Thus, only the reflection strong gradients will have high confidence values in C_{rf} as shown in Fig. 5.1(g). Then we can run a K-means clustering process ($K = 2$) on this confidence map to generate an adaptive threshold ξ for classifying the values in C_{rf} into two groups. The reflection strong gradients E_R and background strong gradients E_B can then be identified as follows:

$$E_R = E_I \cdot (C_{rf} > \xi);\ \ E_B = E_I \cdot (C_{rf} < \xi), \tag{5.6}$$

where E_I denotes those pixels in I whose gradient magnitudes above 1. We concatenate the image I with E_B and E_R to form the input z and sent to a new network F_2 for background reconstruction. A loss function is defined as follows

$$\begin{aligned} L_2 = \sum_{i=1}^{5} \lambda_1 \left\| \phi_i\left(F_2(z)\right) - \phi_i(I_B) \right\|_2^2 + \lambda_2 \| F_2(z) - I_B \|_1 \\ - \lambda_4 D\left(F_2(z)\right) \end{aligned} \tag{5.7}$$

Similar to L_{rec} in the first stage, the first two terms are used to reconstruct the background. Because E_B and E_R may contain outliers, we also use an adversarial term $-\lambda_4 D\left(F_2(z)\right)$ to guide the results to follow the distribution of natural images.

CEILNet+2nd stage PLNet+2nd stage B_{final}(B_{ini}+2nd stage)

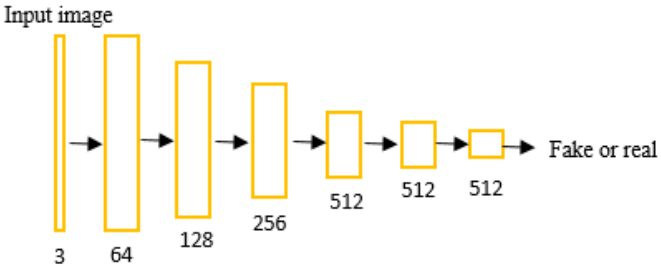
Fig. 5.3. A comparison of final results by using different methods as the first stage.

(5.7) can be implemented using a GAN, where D is the discriminator for measuring the similarity between the inferred background $F_2(I)$ and the ground truth background I_B. λ_4 is a hyper-parameter which is chosen as 0.05 in our experiments. The discriminator D will show high values when $F_2(I)$ follows the distribution of natural images. This discriminator can be jointly trained by minimizing the following loss function:

$$L_{adv} = D\big(F_2(z)\big) - D(I_B) \tag{5.8}$$

Fig. 5.1(h) shows an example of the final result B_{final}. It shows that the final background is shaper compared to the initial result in Fig. 5.1(d) and without reflection residues. For validating the significance of using the initial background estimate B_{ini} as the input for the second stage, we also use the results of CEILNet [7] and PLNet [25] instead of using B_{ini} to generate the confidence map and then

(a) The network structure of F_1 (CNN in stage 1) and F_2 (generator in stage 2). They have the same U-net like structure.



(b) The architecture of the discriminator D

Fig. 5.4. The structures of the networks used in the proposed algorithm.

reconstruct the final background. A comparison of using different approaches is shown in Fig. 5.3. We can see that only B_{ini} can support the background regeneration at the second stage without reflection residues. It is because the reflection residues in the results of CEILNet and PLNet will offset the reflection gradients in the reflection confidence map.

5.4. Experiments and Results

5.4.1. Network Architecture

For the first stage of the proposed algorithm, a U-net like auto-encoder as shown in Fig. 5.4(a) is used to generate the initial background estimation. U-net has been widely used in solving the inverse problems [20, 48, 61]. For the proposed network, the encoder contains 6 levels of stride-two convolutional layers, each followed by a batch normalization layer and ReLU. The decoder part consists of also 6 levels of deconvolutional layer, followed by batch normalization layer and leaky ReLU. We also concatenate the features at the encoder side to the decoder side at each level for increasing the resolution of the results [48]. For the second stage, the structure of the generator network is the same as the auto-encoder used in stage 1. The discriminator network is relatively simple as shown in Fig. 5.4(b). It is composed of six blocks of stride-two convolutional layers, batch normalization layers, and leaky ReLU. The output of the discriminator is a scalar value indicating its judgment that the perceptual features are real or fake.

5.4.2. Training Data Preparation

For training the networks, we synthesize images with reflection using the images in the VOC2012 dataset [101]. The synthetization strategy is similar to [7, 25] and metioned in Section 2.5.4. A training sample is synthesized by superimposing one image serving as a reflection on another image serving as the background. Images in the dataset are mixed together randomly so that many training samples can be obtained. We simulate the blurring effect of the reflection layer by smoothing the reflection images before adding them to the background images. We also simulate the possible ghost effect [26] by convolving the reflection images with a kernel with two very close impulses. The synthesized images are then resized to 256×256. Rotation and flipping are also used for data augmentation. The networks of the first and second stages are trained sequentially for avoiding overfitting. The batch size we used is 3. We use the RMSprop solver [97] to train F_1, F_2 and D. Their learning rates are set to be 2×10^{-4}, 2×10^{-4} and 2×10^{-5} respectively. For F_2 and D, the gradient clipping is also used. The training and testing are both conducted on a computer using the GPU GTX 1080 Ti.

5.4.3. Evaluation and Comparison

For evaluating the performance of the proposed approach, we conduct a series of quantitative and qualitative comparisons with three recent DNN-based single-image reflection removal methods: CEILNet [7], PLNet [25] and BDN [64]. They are both implemented by the source codes published at their websites and their pre-trained models are used in all comparisons. We test these methods using the benchmark dataset SIR2 [66] which contains 452 real scene images with reflections and ground truth backgrounds. Table 5.1 shows the performance. While having a similar SSIM score (if not better), the proposed method significantly outperforms the other methods in terms of average PSNR with a small standard deviation. We also show the performances of the proposed method with and without the adversarial term in (5.6). It shows that the adversarial term can further improve the performance by guiding the resulting image to follow the distribution of natural images.

Method	**PSNR**	**SSIM**	**PSNR STD**
CEILNet [7]	21.75	0.835	4.52
PLNet [25]	20.28	0.823	2.85
BDN [64]	21.43	0.848	2.27
Proposed w/o the adversarial term in (5.7)	22.63	0.844	2.62
Proposed	**23.41**	**0.852**	**2.70**

Table 5.1 The performance of different methods testing with the benchmark real scene dataset SIR2 (452 images).

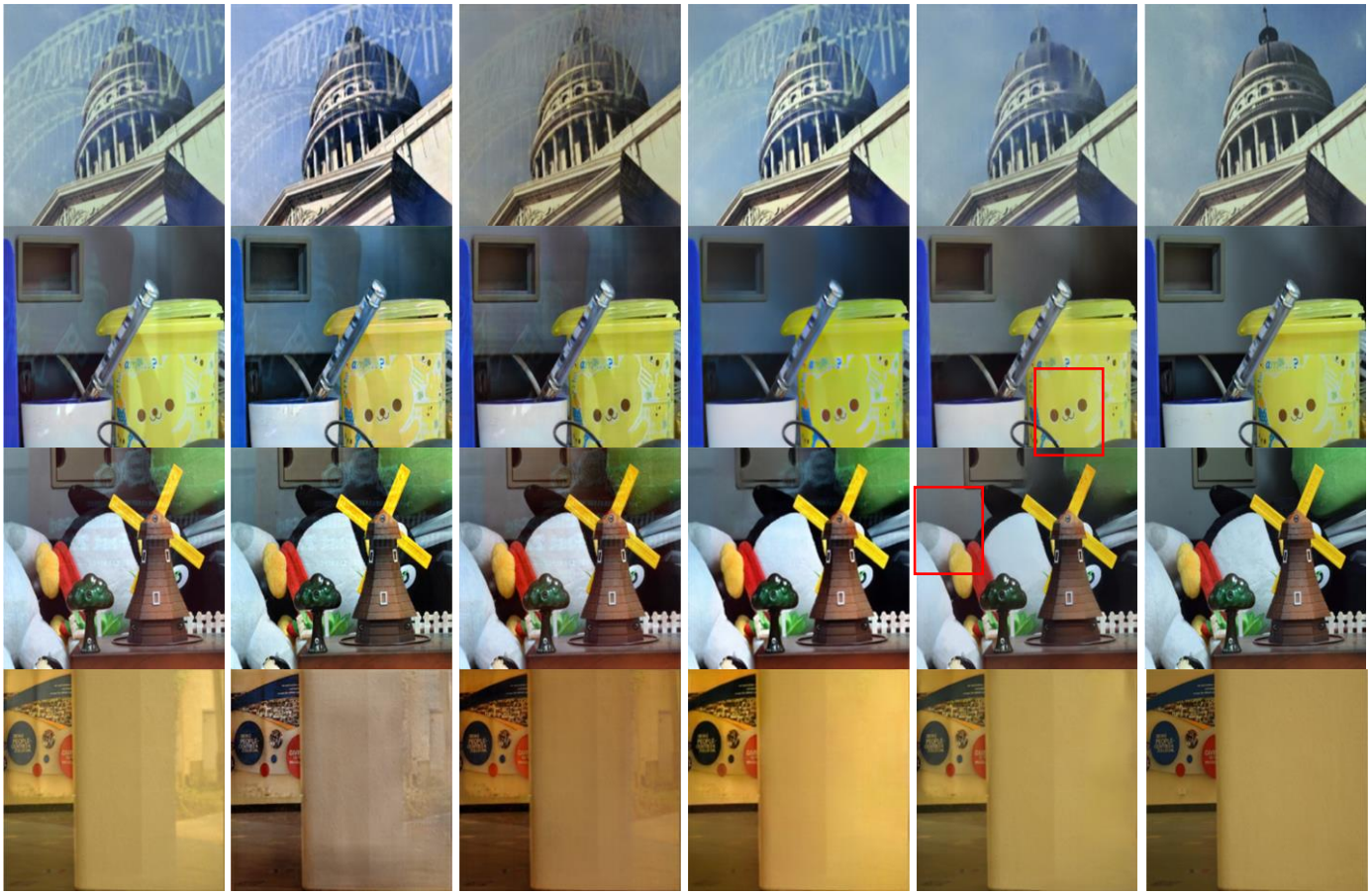


Fig. 5.5. The reflection removal results using different approaches on the images from a benchmark dataset SIR2.

The qualitative comparison results are shown in Fig. 5.5. It can be seen that all competing methods have obvious reflection residues in their results. They mainly come from the strong edges of the reflection layers which have high gradient values. On the contrary, the proposed algorithm can better suppress the reflection components as shown in Fig. 5.5. As mentioned above, the initial stage of the proposed algorithm can blur the background as shown in Fig. 5.6. We make use of

(a) (b)

Fig. 5.6. Blow-ups of the red boxes in Fig. 5.5 to compare the initial and final results. (a) Left: initial, right: final; (b) Left: initial, right: final.

the information provided in the initial result to refine the background estimates at the second stage. The resulting images have the best quality as can be seen in Fig. 5.5 and are much clearer than the initial estimates as shown in Fig. 5.6. We also count the averaging execution times of different methods running on the images used in Section 4.5.5. The results are shown in Table 5.2. We can see they show similar fast speeds as they all use deep neural networks with feedforward structures and GPU acceleration.

5.5. Summary

In this chapter, we proposed a novel two-stage reflection removal algorithm using the deep neural networks based on the background gradient regeneration strategy. The new algorithm can fully remove the reflection residues which often appear in

Method	Average Time
CEILNet	0.82 s
PLNet	1.15 s
BDN	0.79
Proposed	0.73 s

Table 5.2 The averaging execution times of different methods running on the images used in Section 4.5.5.

the results of the traditional methods when the reflection also contains strong gradient components. In this chapter, we first investigated the perceptual feature property of the images with reflection and proposed to include a feature reduction term in the loss function when training the network for further suppressing the reflection strong gradients. As the background gradients may also be suppressed at the first stage, we proposed the second background regeneration network to refine the result. We used the initial result to obtain a reflection edge confidence map and used another auto-encoder trained with an adversarial term to regenerate the background image. Our experimental results have demonstrated the superior performance compared to other state-of-the-art DNN-based reflection removal methods. The proposed algorithm is particularly suitable to images with blurry reflection, which is not uncommon in daily photography.

Chapter 6. Conclusion and Future Works

6.1. Conclusion

In this thesis, we propose three algorithms for removing the reflection in images. These algorithms all follow the same strategy: background gradient regeneration. Since the reflection removal problem is severely ill-posed, existing reflection removal methods need to make different assumptions on the properties of the background and reflection for their separation. Unfortunately, due to the similar morphological properties of the background and reflection images, these weak assumptions often cannot be fulfilled in many practical situations. Many reflection residues thus remain in their inferred background results. Rather than following the existing approaches in searching for a perfect assumption that can accurately distinguish the background and reflection in all situations, we believe it is more realistic and effective to look for a remedial strategy in case the separation is unsuccessful. The proposed background gradient regeneration strategy suggests to firstly remove the reflection components in an aggressive manner even in the expense of losing some of the background components. The missing background

components are then regenerated based on the remaining ones using different estimation methods. The advance in the data regeneration techniques renders the successful implementation of the proposed strategy in this research study.

The proposed reflection removal methods have been elaborated in detail in Chapter 3 to 5. These three methods have fully demonstrated the effectiveness of the proposed strategy in reflection removal. For the first method using the traditional optimization methods based on the background gradient regeneration strategy, it outperforms the traditional multiple-image reflection removal methods by more than 3dB in PSNR as shown in our experiments. For improving the speed, the second proposed method integrates the strategy with different deep neural network (DNN) techniques. It achieves superior performance similar to the first proposed algorithm while providing a more than 1,000 times speedup when implementing with GPUs. The first two proposed algorithms require multiple input images. The third proposed algorithm requires only a single image as the input. Following the background gradient regeneration strategy, a 1.7dB improvement in PSNR is achieved compared to other single-image DNN-based reflection removal methods. They are indeed some significant contributions to the field of study.

In recent years, computational imaging technology has been widely adopted in digital cameras, mobile devices, and image processing software. Although the advance of such technology has made possible many new functions in imaging, there are still problems that cannot be totally resolved. One of them is reflection removal. In fact, the reflection problem is often encountered in daily photography and greatly affecting the image quality. However, a truly robust solution is yet to be

Method	Advantages	Disadvantages
Light-field optimization-based method	• Robust performance • Does not require any training	• Not fit for the situation when the background and reflection have their depth ranges largely overlapped • Low speed and require large memory storage • Need special imaging device
Multiple-image DNN-based method	• Robust performance • High speed	• Not fit for the situation when the background and reflection have their depth ranges largely overlapped • Need special imaging device • Performance can be affected by the size and variety of the training dataset
Single-image DNN-based method	• Does not require any special imaging device • High speed	• Require reflection to be defocused • Performance can be affected by the size and variety of the training dataset

Table 6.1 A comparison of different proposed approaches.

developed due to the ill-posedness of the problem. The proposed reflection removal algorithms can be readily implemented by existing imaging hardware and can provide real-time performance when implementing with GPUs. Most importantly, they provide the much-needed robustness which the existing approaches cannot achieve. They have provided some practical solutions to the problem and we believe they will arouse great interest from the competitive digital imaging industry.

We also summarize the advantages and disadvantages of these three methods in Table 6.1. The first and second proposed methods can give robust performances in different environments. Although these two algorithms can give considerable performance improvement over the existing methods, they still have their own weaknesses. They cannot deal with the situations that the depth ranges of the background and reflection overlap largely. In this case, the depth range of the pure background components becomes very narrow such that there will not be enough background components for regenerating the missing ones. And because the first method is optimization-based, it takes much longer time than the second approach for its computation. Also as light field images are used, this method requires large memory storage. For the third proposed method, it requires only a single input image thus it does not need special hardware device for image capture. It is DNN-based so it is also very fast. It has less reflection residuals as compared with the existing single-image reflection removal methods. However, the method still requires the reflection to be de-focused. Although it is not uncommon to have blurry reflection

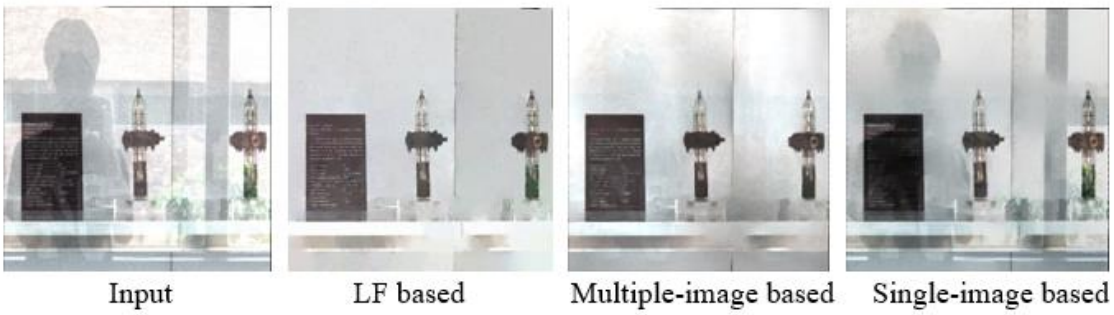


Fig. 6.1. Comparison results of different proposed approaches

images, there are still quite a lot of images having a sharp reflection. Finally, for both proposed DNN-based approaches, the performance can be affected by the choice of the training dataset, while it is not the case for the proposed optimization-based method which does not require any training. These weaknesses of the proposed algorithms can be the directions for further research. Fig. 6.1 shows the comparison results of different proposed approaches. Here we can see that the LF based and multiple-image based methods can remove the reflection very well, since they use a similar strategy and framework. On the other hand, the single-image based method cannot totally remove the reflection, because the reflection is not blurry. It is a limitation of that method.

6.2. Future Works

As an extension to the existing work, removing the reflection in videos can be a valuable direction. In videos, the differences in the motions of background and reflection between adjacent frames naturally provide a useful clue to distinguish the background and reflection components. Especially nowadays some digital cameras can capture videos at very high frame rates (e.g. 960 fps) for slow-motion video recording. Considering the movement of a camera held by a photographer can be approximated as linear and homogeneous during a very short time interval (e.g. 0.01 seconds), we can capture many images (e.g. 9 images) in that short time interval by using slow-motion video recording. It is equivalent to capturing the images by a linear row of cameras. If we can develop a method to rectify the slight jitters in the captured images, we can use our first or second proposed algorithm to remove the reflection with only a single camera.

Furthermore, the recurrent neural network (RNN) and long short term memory (LSTM) neural network, which have been adopted in video object recognition [102, 103] and tracking [104], can also be used for further exploring the motion relationship between video frames. The power of RNN and LSTM in exploring the potential background and reflection motion information in the video frames will be helpful for obtaining more accurate motion models for reflection removal.

References

[1] M. Born and E. Wolf, *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier, 2013.

[2] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks,* vol. 3361, no. 10, p. 1995, 1995.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770-778.

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

[6] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1-9.

[7] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "A generic deep architecture for single image reflection removal and image smoothing," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, vol. 2, no. 3, p. 4.

[8] Y. Li and M. S. Brown, "Single image layer separation using relative smoothness," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2752-2759.

[9] X. Guo, X. Cao, and Y. Ma, "Robust separation of reflection from multiple images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2187-2194.

[10] Y. Li and M. S. Brown, "Exploiting reflection change for automatic reflection removal," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2432-2439.

[11] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman, "A computational approach for obstruction-free photography," *ACM Trans. Graph.,* vol. 34, no. 4, p. 79, 2015.

[12] Y. Ni, J. Chen, and L.-P. Chau, "Reflection Removal Based on Single Light Field Capture," in *Proc. IEEE Int. Sympos. Circuits Syst.*, 2017, pp. 1-4.

[13] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented Lagrangian method for total variation video restoration," *IEEE Transactions on Image Processing*, vol. 20, no. 11, pp. 3097-3111, 2011.

[14] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock, "An introduction to total variation for image analysis," *Theoretical foundations and numerical methods for sparse recovery*, vol. 9, no. 263-340, p. 227, 2010.

[15] R. H. Chan, M. Tao, and X. Yuan, "Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers," *SIAM Journal on imaging Sciences*, vol. 6, no. 1, pp. 680-697, 2013.

[16] H. Liu, R. Xiong, X. Zhang, Y. Zhang, S. Ma, and W. Gao, "Nonlocal gradient sparsity regularization for image restoration," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 9, pp. 1909-1921, 2016.

[17] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996, pp. 31-42: ACM.

[18] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," *Computer Science Technical Report CSTR*, vol. 2, no. 11, pp. 1-11, 2005.

[19] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, "Compressive light field photography using overcomplete dictionaries and optimized projections," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, p. 46, 2013.

[20] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4509-4522, 2017.

[21] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142-3155, 2017.

[22] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608-4622, 2018.

[23] L. Li, J. Pan, W.-S. Lai, C. Gao, N. Sang, and M.-H. Yang, "Learning a discriminative prior for blind image deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6616-6625.

[24] Z. Shen, W.-S. Lai, T. Xu, J. Kautz, and M.-H. Yang, "Deep semantic face deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8260-8269.

[25] X. Zhang, R. Ng, and Q. Chen, "Single Image Reflection Separation with Perceptual Losses," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.

[26] Y. Shih, D. Krishnan, F. Durand, and W. T. Freeman, "Reflection removal using ghosting cues," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3193-3201.

[27] Y. Y. Schechner, J. Shamir, and N. Kiryati, "Polarization-based decorrelation of transparent layers: The inclination angle of an invisible surface," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, 1999, vol. 2, pp. 814-819: IEEE.

[28] N. Kong, Y.-W. Tai, and S. Y. Shin, "High-quality reflection separation using polarized images," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3393-3405, 2011.
[29] N. Kong, Y.-W. Tai, and J. S. Shin, "A physically-based approach to reflection separation: from physical modeling to constrained optimization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 2, pp. 209-221, 2014.
[30] A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 828-835, 2005.
[31] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, 2007.
[32] K. Gai, Z. Shi, and C. Zhang, "Blind separation of superimposed moving images using image statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 19-32, Jan 2012.
[33] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978-994, Aug 2011.
[34] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," vol. 1, no. 4, pp. 541-551, 1989.
[35] Y. LeCun, L. Bottou, Y. Bengio, and P. J. P. o. t. I. Haffner, "Gradient-based learning applied to document recognition," vol. 86, no. 11, pp. 2278-2324, 1998.
[36] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, 1992, pp. 144-152: ACM.
[37] R. Arora, A. Basu, P. Mianjy, and A. J. a. p. a. Mukherjee, "Understanding deep neural networks with rectified linear units," 2016.
[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. J. T. J. o. M. L. R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," vol. 15, no. 1, pp. 1929-1958, 2014.
[39] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6154-6162.
[40] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440-1448.
[41] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961-2969.
[42] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE*

Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117-2125.

[43] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980-2988.

[44] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779-788.

[45] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91-99.

[46] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431-3440.

[47] V. Badrinarayanan, A. Kendall, R. J. I. t. o. p. a. Cipolla, and m. intelligence, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," vol. 39, no. 12, pp. 2481-2495, 2017.

[48] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234-241: Springer.

[49] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150-3158.

[50] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2359-2367.

[51] A. Dosovitskiy *et al.*, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758-2766.

[52] T.-W. Hui, X. Tang, and C. Change Loy, "Liteflownet: A lightweight convolutional neural network for optical flow estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8981-8989.

[53] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2462-2470.

[54] C. Dong, C. C. Loy, K. He, X. J. I. t. o. p. a. Tang, and m. intelligence, "Image super-resolution using deep convolutional networks," vol. 38, no. 2, pp. 295-307, 2015.

[55] J. Yamanaka, S. Kuwashima, and T. Kurita, "Fast and accurate image super resolution by deep CNN with skip connection and network in network," in

International Conference on Neural Information Processing, 2017, pp. 217-225: Springer.

[56] A. Radford, L. Metz, and S. J. a. p. a. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015.

[57] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning*, 2017, pp. 214-223.

[58] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems*, 2017, pp. 5767-5777.

[59] D. Berthelot, T. Schumm, and L. J. a. p. a. Metz, "BEGAN: Boundary Equilibrium Generative Adversarial Networks," 2017.

[60] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2794-2802.

[61] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2536-2544.

[62] J. Chen, J. Chen, H. Chao, and M. Yang, "Image Blind Denoising With Generative Adversarial Network Based Noise Modeling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3155-3164.

[63] S.-J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee, "SRFeat: Single Image Super-Resolution with Feature Discrimination," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 439-455.

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

[65] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*, 2016, pp. 694-711: Springer.

[66] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, "Benchmarking single-image reflection removal algorithms," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017.

[67] T. Li, D. P. K. Lun, Y. H. Chan, and Budianto, "Robust Reflection Removal Based on Light Field Imaging," *IEEE Trans. Image Process.*, vol. 28, no. 4, 2019.

[68] A. Isaksen, L. McMillan, and S. J. Gortler, "Dynamically reparameterized light fields," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 297-306: ACM Press/Addison-Wesley Publishing Co.

[69] N. Joshi, S. Avidan, W. Matusik, and D. J. Kriegman, "Synthetic aperture tracking: tracking through occlusions," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1-8: IEEE.

[70] L. Jianqiao and L. Ze-Nian, "Continuous depth map reconstruction from light fields," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, 2013, pp. 1-6.

[71] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 673-680.

[72] T.-C. Wang, A. A. Efros, and R. Ramamoorthi, "Occlusion-aware depth estimation using light-field cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3487-3495.

[73] S. Wanner and B. Goldluecke, "Variational light field analysis for disparity estimation and super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 606-619, Mar 2014.

[74] T. Li and D. P. K. Lun, "Super-resolution imaging with occlusion removal using a camera array," in *ISCAS*, 2016, pp. 2487-2490: IEEE.

[75] R. C. Bolles, H. H. Baker, and D. H. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *Int. J. Comput. Vis.*, vol. 1, no. 1, pp. 7-55, Mar 1987.

[76] S. Wanner and B. Goldluecke, "Globally consistent depth labeling of 4D light fields," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 41-48: IEEE.

[77] J. Bigun, "Optimal orientation detection of linear symmetry," ed: Linköping University Electronic Press, 1987.

[78] K. G. Derpanis, "The harris corner detector," *York University*, 2004.

[79] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881-892, Jul 2002.

[80] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, vol. 1, pp. 886-893: IEEE.

[81] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," in *Proceedings of the 18th ACM international conference on Multimedia*, 2010, pp. 1469-1472: ACM.

[82] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627-1645, 2010.

[83] S. Z. Li, "Markov random field models in computer vision," in *Proc. Eur. Conf. Comput. Vis.*, 1994, pp. 361-370: Springer.

[84] Q. Chen, D. Li, and C.-K. Tang, "KNN matting," *PAMI,* vol. 35, no. 9, pp. 2175-2188, 2013.
[85] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE transactions on pattern analysis and machine intelligence,* vol. 26, no. 9, pp. 1124-1137, 2004.
[86] T. Li, Y.-H. Chan, and D. P. K. Lun, "Improved multiple-image based reflection removal algorithm using deep neural networks," *IEEE Transactions on Image processing,* 2019 (under review).
[87] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672-2680.
[88] C. Ledig *et al.*, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4681-4690.
[89] Q. Yang *et al.*, "Low dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss," *IEEE transactions on medical imaging*, 2018.
[90] R. Toldo, F. Fantini, L. Giona, S. Fantoni, A. J. I.-I. A. o. t. P. Fusiello, Remote Sensing, and S. I. Sciences, "Accurate multiview stereo reconstruction with fast visibility integration and tight disparity bounding," no. 1, pp. 243-249, 2013.
[91] E. Trucco and A. Verri, *Introductory techniques for 3-D computer vision*. Prentice Hall Englewood Cliffs, 1998.
[92] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision*, 2012, pp. 746-760: Springer.
[93] A. Saxena, S. H. Chung, and A. Y. Ng, "3-d depth reconstruction from a single still image," *International journal of computer vision,* vol. 76, no. 1, pp. 53-69, 2008.
[94] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. A. Dodgson, "Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid," in *European conference on Computer vision*, 2010, pp. 510-523: Springer.
[95] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research,* vol. 17, no. 1-32, p. 2, 2016.
[96] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Rep.*, 2014.
[97] T. Tieleman and G. Hinton, "Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning,* vol. 4, no. 2, pp. 26-31, 2012.
[98] T. Li and D. P. K. Lun, "Single-Image Reflection Removal via a Two-Stage Background Recovery Process," *IEEE Signal Processing Letters,* 2019 (under review).

[99] D. Engin, A. Genc, and H. Kemal Ekenel, "Cycle-Dehaze: Enhanced CycleGAN for Single Image Dehazing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 825-833.

[100] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248-255: Ieee.

[101] M. Everingham and J. Winn, "The pascal visual object classes challenge 2012 (voc2012) development kit," *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*, 2011.

[102] Y. Lu, C. Lu, and C.-K. Tang, "Online video object detection using association LSTM," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2344-2352.

[103] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. J. I. A. Baik, "Action recognition in video sequences using deep bi-directional LSTM with CNN features," vol. 6, pp. 1155-1166, 2017.

[104] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.