# Joint Acquisition of Color and Near-Infrared Images on a Single Sensor

#### THÈSE Nº 6766 (2015)

PRÉSENTÉE LE 30 OCTOBRE 2015

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS LABORATOIRE D'IMAGES ET REPRÉSENTATION VISUELLE PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS

#### ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

#### PAR

#### Zahra SADEGHIPOOR KERMANI

acceptée sur proposition du jury:

Prof. P. Fua, président du jury Prof. S. Süsstrunk, Prof. Y. Lu, directeurs de thèse Prof. D. Alleysson, rapporteur Dr E. Mendez Mendez, rapporteur Prof. P. Vandergheynst, rapporteur



I don't pretend we have all the answers. But the questions are certainly worth thinking about. — Arthur C. Clarke

To my parents...

## **Acknowledgments**

I have had the great fortune of being supported by many people in the past few years. In the following, I would like to express my gratitude for their undeniable contributions to both my professional and personal life.

First of all, I would like to thank my supervisor, Professor Sabine Süsstrunk, for giving me the opportunity to work in the lab and for her endless support during the past five years. Sabine provided me with freedom to work on my ideas, even when she thought certain ideas were likely not to work. Her insight and guidance helped me grow in many aspects as a researcher. I would also like to thank my co-advisor, Professor Yue Lu. Yue, despite being thousands of miles away, always helped me with his insightful comments and ideas.

I had the absolute honor of having my thesis evaluated by a jury of distinguished experts: Professor David Alleysson, Professor Pascal Fua, Dr Erick Mendez Mendez, and Professor Pierre Vandergheynest. I greatly appreciate their comments and feedback on my work.

I would like to acknowledge the generous contributions of the Swiss National Science Foundation, the Hasler Foundation, and Qualcomm Inc., through the Qualcomm Innovation Fellowship program. I thank Dr. Alessandro Mulloni and Dr. Erick Mendez Mendez who mentored my two Qualcomm fellowship projects.

I am very grateful to our secretaries Ms. Jacqueline Aeberhard, Ms. Virginie Rebetez, and Ms. Francoise Behn. My work at EPFL and life in Switzerland would have not been remotely as smooth if it were not for their amazing support. Ms. Holly Cogliati-Bauereis has been incredibly patient when proof-reading my papers and thesis. She taught me a lot, for which I am truly thankful.

I would like to thank the students who worked with me, Lea Vliegen, Sophie Voirol, and Arthur Descoeudres for taking some of the photos used in the experiments of the thesis. Special thanks go to Arthur who also helped with some of the simulations in Chapter 4.

I would like to thank my former and current colleagues, Albrecht, Appu, Bin, Cheryl, Damien, Marjan, Nickolaos, and Radhakrishna for helping with my research in one way or another. Many thanks to Radha and Celine for helping with the French abstract, and to Dr. Javier Vazquez-Corral for commenting on parts of my thesis. I am indebted especially to my office-mate of four years, Gökhan. I always

relied on his help. I cannot possibly return all his favors and can merely thank him.

When I first arrived in Switzerland, Neda supported me in ways that I could not have imagined. I shared with her some of the most joyful moments of the past few years. Neda is more than a friend to me and I forever send her my love and gratitude.

My best friend and roommate, Elham, has never stopped supporting me and giving me confidence when I needed it most. She helped with everything from proof-reading my papers and thesis to listening to me talk about my research over and over again, giving ideas, and asking questions that I, myself, had not thought of. Elham has become a part of my family and I cannot thank her enough for all her help.

Many thanks go to my brothers, Alireza and Mohammad upon whom I could rely anytime I needed help with my affairs in Iran while being in Switzerland. I thank Zeinab for being the sweetest sister a girl could have and for making sure that I was always aware of what the consequences would be if some plans did not work out. Finally and most importantly, I would like to express my deepest gratitude towards my parents. I owe all my accomplishments to their unconditional love, support, and sacrifice. I, thus, dedicate this thesis to my parents.

Lausanne, August 25, 2015

## **Abstract**

In the past few years, fusing NIR and color images has been explored in general computational photography and computer vision tasks, where traditionally only color images are used. The additional information provided by the differences of light and scene reflections in the visible and NIR bands of the electromagnetic spectrum is used in several applications such as image denoising, image dehazing, shadow detection and removal, and high dynamic-range imaging.

In this thesis, we study a system that simultaneously captures color and NIR images on a single silicon sensor. Such a camera could be manufactured with minor changes in the hardware of consumer color cameras and it could be integrated inside small devices such as cell phones. We address two main challenges in color and NIR acquisition. First, we study the spatial and spectral sampling of the scene, which is inevitable in single-sensor acquisition of multiple spectral channels. We then focus on chromatic aberration distortions.

Similarly to color imaging, we use a color filter array (CFA) to sample the scene in visible and NIR bands. We address two main challenges regarding the CFA: (1) designing the CFA, and (2) developing a demosaicing algorithm to reconstruct full-resolution images from the subsampled measurements. We consider a general CFA with filters that transmit different mixtures of color and NIR channels. We develop a framework that, by exploiting the spatial and spectral correlations of color and NIR images, computes the transmittance of each filter and the demosaicing matrix. Our optimized CFA and demosaicing outperform other solutions developed for single-sensor color and NIR acquisition. We also investigate a CFA that is formed by one blue, one green, one red, and one NIR-pass filter. We call it the RGBN CFA and assume that, similarly to color cameras, it uses dye filters that do not have sharp cut-offs. Hence, color and NIR radiations leak into NIR and color filters, respectively. We devise an algorithm that reconstructs full-resolution images from mixed and subsampled sensor measurements. The RGBN CFA and our reconstruction algorithm perform as well as or even better than other single-sensor acquisition techniques that use more complicated hardware components.

The problem of chromatic aberration is caused by deficiencies of optical elements. A simple lens converges light rays with different wavelengths at different distances from the lens. Hence, if the color image is in focus and sharp on the sensor plane,

the NIR image captured with the same focus settings is out of focus and blurred. We propose an algorithm that retrieves the lost details in NIR using the gradients of the sharp color image. As the high-frequency details of color and NIR images are not strongly correlated in all image patches, our method locally adapts the contribution of color gradients in deblurring. To achieve this, we develop a multiscale scheme that iterates between deblurring NIR and estimating the correlation between color and NIR high-frequency components. Our algorithm outperforms both blind and guided deblurring approaches. We also design a method that estimates a dense blur-kernel map when the severity of chromatic aberration changes as the depths of objects vary across the image. Our method performs better than the competing methods both in estimating the blur-kernel map and in deblurring.

**Keywords:** color and NIR imaging, spatial correlation, spectral correlation, color filter array, spatial and spectral sampling, demosaicing, chromatic aberration, deblurring.

## Résumé

Ces dernières années, la fusion de NIR et des images couleurs a été explorée en photographie computationnelle pour les tâches de vision informatique, là où traditionnellement seules les images couleurs étaient utilisées. Les informations additionnelles obtenues à partir des différences de lumière et des réflectances des surfaces entre les bandes visibles et NIR du champ électromagnétique, sont utilisées dans plusieurs applications comme le débruitage des images, détection et suppresion des ombres, et dans l'imagerie à grande gamme dynamique.

Dans cette thèse, nous étudions un système qui prend simultanément des images couleur et NIR sur un seul capteur silicone. Un appareil photo de ce type peut être fabriqué avec des changements mineurs dans le hardware d'appareils photo couleur du commerce et peut être intégré dans de petits appareils tels que les téléphones portables. Nous résolvons deux challenges importants en acquisition des images couleur et NIR. Premièrement, nous étudierons l'échantillonnage spatial et spectral de la scène, ce qui est inévitable dans l'acquisition avec un capteur unique des canaux multi spectre. Ensuite nous nous focaliserons sur les aberrations chromatiques causées par les carences des éléments optiques.

De façon similaire à l'imagerie couleur, nous utilisons une matrice de filtre de couleur (CFA) pour échantillonner les scènes dans les domaines visible et NIR. Nous résolvons deux principaux challenges concernant les CFA: (1) concevoir les CFA et (2) développer un algorithme de démosaiçage pour reconstruire les images en pleine résolution avec des mesures sous-échantillonnées. Nous considérons un CFA général avec des filtres qui transmettent différents mélanges de canaux de couleur et NIR. Nous développons une structure qui, en exploitant les corrélations spatiale et spectrales des images couleur et NIR, calcule la transmittance de chaque filtre et la matrice de démosaiçage. Notre CFA optimisé et notre matrice de démosaiçage surpassent les autres solutions développées pour l'acquisition de couleur et NIR par un capteur unique. Nous étudions aussi le CFA formé de filtres bleu, vert, rouge et NIR. Nous appelons ce CFA "CFA RGBN" et supposons que, comme pour les appareils photo couleur, il utilise des filtres teintés qui n'ont pas de fréquences de coupures parfaites. Par conséquent, les radiations couleur et NIR passent respectivement à travers les filtres NIR et couleur. Nous concevons un algorithme qui reconstruit des images en pleine résolution avec les mesures des capteurs sous-échantillonnées et mélangés. Le CFA RGBN et notre algorithme de reconstruction fonctionnent aussi bien ou même mieux que les autres techniques d'acquisition avec un seul capteur qui utilisent des composants de hardware plus compliqués.

Nous analysons le problème des aberrations chromatiques dans l'acquisition couleur et NIR. Une simple lentille fait converger les rayons de lumière avec des longueurs d'onde á différentes distance de la lentille. Par conséquent, si l'image couleur est focalisée précisément dans le plan du capteur, l'image NIR captée avec les mêmes paramètres de mise au point n'est pas focalisée et floue. Nous proposons un algorithme qui récupère les détails perdus du NIR en utilisant les gradients de l'image couleur nette. Comme les détails haute-fréquences des images couleur et NIR ne sont pas fortement corrélés dans tous les patchs d'images, notre méthode adapte localement la contribution des gradients de couleur dans le déflouage. Pour atteindre cela, nous avons développé une méthode multi-échelle qui alterne entre le déflouage du NIR et l'estimation de la corrélation entre les gradients de couleur et NIR. Notre algorithme surpasse aussi bien les approches de déflouage guidé et aveugle. Nous avons aussi créé une méthode qui estime le noyau dense de flouage quand la sévérité des aberrations chromatiques changent avec la profondeur des divers objets dans l'image. Notre méthode surpasse les méthodes concurrentes aussi bien pour estimer le noyau de flouage que pour le déflouage.

Mot clef: imagerie couleur et NIR, corrélation spatial, corrélation spectrale, matrice de filtre de couleur, échantillonnage spatial et spectral, dématriçage, aberration chromatique, déflouage.

## **Contents**

1	Intr	duction
	1.1	Spatial and Spectral Sampling
	1.2	Chromatic Aberration
	1.3	Image Acquisition
	1.4	Thesis Outline
2	Rela	ted Work 17
	2.1	Color Demosaicing
		2.1.1 Constant-Hue Assumption
		2.1.2 Luminance-Chrominance Decomposition
		2.1.3 Probabilistic Demosaicing
		2.1.4 Demosaicing Based on Sparse Decomposition
		2.1.5 Graph-Based Demosaicing
	2.2	Color Filter Array Design
	2.3	Single-Sensor Multispectral Acquisition
	2.4	Chromatic Aberration
		2.4.1 Chromatic Aberration in Color Imaging
		2.4.2 Deblurring
	2.5	Discussion
	2.6	Summary
3	Opt	mized CFA 43
	3.1	Correlation Characteristics of Color and NIR Images
		3.1.1 Filter-Bank Analysis of Correlation
		3.1.2 Frequency Analysis of Correlation
	3.2	The Optimized CFA and Demosaicing
	3.3	Experiments
		3.3.1 Implementation Details
		3.3.2 Simulation Results
	3.4	Summary

x Contents

4	RGE	BN CF	A and Sparse Reconstruction	67
	4.1	Comp	ressive Imaging	68
		4.1.1	Compressive Sensing in Color Imaging	68
		4.1.2	Compressive Sensing in Multispectral Acquisition	68
	4.2	Reviev	w of Sparse Reconstruction	69
	4.3	The P	roposed Algorithm: Color and NIR Separation	70
		4.3.1	Separating Red and NIR Channels	70
		4.3.2	Green/Blue and NIR Separation	71
		4.3.3	The Measurement Matrix	72
		4.3.4	The Sparsifying Transform	74
	4.4	Simula	ations of Our Proposed Algorithm	
		4.4.1	Implementation Details	79
		4.4.2	Results	
		4.4.3	Separation with Two Dictionaries	
	4.5	Analys	sis of Single-Sensor Color and NIR Acquisition Techniques	86
		4.5.1	Single-Sensor Acquisition Techniques	
		4.5.2	Quality Metrics	
		4.5.3	Simulation Results	
		4.5.4	Discussion	96
	4.6	Summ	nary	96
5	Chr	omatic	Aberration in Color and NIR Joint Acquisition	99
	5.1	Guideo	d Deblurring	100
		5.1.1	Estimating the Blur Kernel	
		5.1.2	Deblurring the NIR Image	101
		5.1.3	Multiscale Deblurring	
	5.2	Experi	iments	
		5.2.1	Data Acquisition	107
		5.2.2	Simulation Results	
	5.3	Summ	nary	
6	Spa	tially V	Varying Chromatic Aberration	121
	<b>6</b> .1	•	illy Varying Defocus Blur	122
	6.2	-	round-Background Segmentation and Deblurring	
		6.2.1	Proposed Algorithm	
		6.2.2	Results	
	6.3		Illy Varying Blur Kernel Estimation and Deblurring	
		6.3.1	Multi-Label Segmentation	
		6.3.2	Spatially Varying Deblurring	
		6.3.3	Results	
	6.4		ssion	
	6.5		nary	
	٠.٠		·····	

7	Conclusion	163
	7.1 Thesis Summary	163
	7.2 Future Research	165
Α	Optimizing CFA and Demosaicing	169
Bi	bliography	171
Cι	urriculum Vitae	187

xii Contents

# List of Figures

1.1	This figure illustrates some of the limitations of digital cameras in capturing the scene and the deficiencies of an algorithm in extracting information from color images. (a) The image captured in a low-light environment without flash contains strong noise [117]. (b) A picture representing a high dynamic-range scene does not preserve the details in the dark regions. The details on the wall (red arrow) are completely lost. (c) Input image [123], and (d) shadow mask computed by the algorithm of Guo et al. [53]. White pixels mark shadow regions. Notice that no shadow pixels are detected on the tree in the upper-left part of the image (red arrow), although it is under the shadow	2
1.2	(a) In high dynamic-range imaging, several pictures with different exposure times are recorded and fused into one image that contains rich details in both dark and bright regions of the scene [36, 101]. Left: several pictures that are captured with different exposure times. Right: the details of both dark and bright regions are visible in the final image produced by the algorithm of Meylan et al. [101]. (b) Petschnigg et al. [117] propose to capture a pair of images—with and without flash—to obtain a clean image in low-light conditions. The clean flash image is used to denoise the no-flash image that preserves the ambient illumination. From left to right: no-flash image, the clean image captured with flash, and the denoised image. These images are made available by authors at http://research.microsoft.com/	
	en-us/um/redmond/projects/flashnoflash/	9
1.3	<ul><li>(a) and (b) Vegetation usually looks brighter in the NIR image, and sky is darker in NIR. Water absorbs NIR and appears dark in this image, while, the stones beneath water are visible in the color image.</li><li>(c) and (d) The scarf in the middle looks more or less the same in the NIR image, despite having different colors.</li></ul>	4
1.4	Some applications of fusing color and NIR images in "everyday" photography. For each row, from left to right: the color image, the NIR	,
	image, and the result	(

xiv List of Figures

1.5	The quantum efficiency of typical CCD and CMOS sensors made of silicon. Both sensors are sensitive to visible (400-700 nm) and NIR (700-1100 nm) bands. Data for the CCD sensor is obtained from [119], and data for the CMOS sensor is provided by [95]	7
1.6	Capturing color and NIR images in two sequential shots leads to registration issues. When there are moving objects in the scene, this acquisition approach is unfeasible.	7
1.7	(a) The Bayer color filter array that is used in most color cameras. (b) The overall sensitivities of a silicon sensor and color filters in Bayer, measured for a consumer color camera	8
1.8	A schematic of CFA sampling and demosaicing in a single-sensor color camera.	S
1.9	(a) The RGBN CFA proposed to sample color and NIR channels. (b) The overall sensitivities of color and NIR filters and a silicon sensor. Each color filter transmits one color channel and part of the NIR band as the NIR-blocking filter is removed. The NIR filter transmits both NIR and red channels. The visible range: 400-700 nm, the NIR band: 700-1100 nm.	10
1.10	(a) A simple lens converges light rays with different wavelengths at different distances. If the sensor is placed at the focus plane of the green light, only the image of scene information with this wavelength is sharp. The disks on the right illustrate the lens blur kernels on the sensor for other wavelengths. The black rays represent NIR radiation.  (b) A pair of color and NIR images that represent the same scene. The color image is in focus and all color channels are sharp, whereas the NIR image, captured with the same lens and focus, suffers from chromatic aberration and is blurred	12
1.11	As the defocus blur is depth dependent, the amount of blur in the NIR image with respect to the color image changes from foreground to the background plane. The foreground object (insets with green borders) is in-focus in the color image and blurred in NIR. Whereas, the background region (insets with red borders) is sharper in the NIR image	13
2.1	(a) Input image. (b) The CFA image shown in color. As the sampling frequency of the green channel is the highest, the image looks greenish. (c) The Fourier spectrum of the CFA image. The central part contains luminance information and chrominance channels (here shown as $C_1$ and $C_2$ ) are modulated to high frequencies.	21
2.2	-, ·	$\frac{21}{25}$

List of Figures xv

(a) A multispectral camera that uses a filter wheel [3]. (b) A schematic of a multispectral acquisition system that uses a beam splitter and several detectors	27
<ul><li>(a) The binary tree used to design the MSFA for acquisition of five spectral bands.</li><li>(b) The corresponding checkerboard decomposition.</li><li>(c) The multi-spectral filter array (MSFA) is obtained by superposing the patterns in leaves of the checkerboard decomposition [102]</li></ul>	28
(a) The MSFA presented in [107] for the acquisition of five spectral channels in the visible band. (b) The hybrid CFA proposed in [75] for capturing color channels and one additional band (colored black). (c) The MSFA proposed in [4] for sampling $k$ spectral channels denoted as $\{C_1, C_2, \dots, C_k\}$	29
The relative response of the multispectral camera developed by Lapray $et~al.~[83]$ . The camera captures one NIR $(C_8)$ and seven visible channels. The plot is produced based on the data provided by the authors of $[83]$	31
The two MSFA filter arrangements used by Lapray $et\ al.$ [83]. Spectral channels $C_1$ to $C_7$ represent the scene in the visible band and $C_8$ is the NIR channel	31
Purple fringes, partly caused by chromatic aberration, are mainly visible around bright edges in the image	34
The arrows show the boundaries of the transition region around the edge. The triplet values are the RGB code values of leftmost and rightmost pixels of the transition region	35
The gradients of natural images have heavy-tailed distributions as the ones shown here	37
A pair of color (left) and NIR (right) images. The color pigments on the pen are transparent in NIR, resulting in differences between the gradients of color and NIR images. Note that the text in the book looks the same in both images, as carbon black used in modern ink	
absorbs light in both visible and NIR wavelength ranges	39
This figure illustrates the differences between the absolute intensities of (a) the Y channel (the average of color channels), and (b) the NIR image. Despite these differences, the object boundaries and high-frequency details of both color and NIR representations are generally	40
An example image and its four subbands. All images have the same resolution in the filter-bank decomposition, and are re-sampled here	42
	of a multispectral acquisition system that uses a beam splitter and several detectors

xvi List of Figures

3.3	The histograms of correlation values for different subbands. The correlation between LL subbands of NIR and Y for majority of images is	
	less than 0.6, whereas the correlation is usually larger than 0.6 for LL	
	subbands of red-green and blue-green pairs. For the high-frequencies	
	subbands, however, in most images the correlation between NIR and	
	Y is higher than 0.5	45
3.4	Top row: color and NIR images. Bottom row shows a patch on the	
	wall in color channels and NIR. The pattern on the wall is visible in all	
	color channels, while the patch is uniform in the NIR image. These	
	inherent differences result in weaker correlation between color and	
	NIR high frequencies compared with the correlation of color channels.	45
3.5	The histograms of absolute correlation values for different subbands.	
0.0	The correlation values between LL subbands are usually small, whereas	
	the high-frequency subbands exhibit stronger correlations	46
3.6	Three sample filters used in our correlation analysis. The filters are	10
5.0	shown in the frequency domain	48
3.7		40
3.1	The $\mathcal{N}$ and $\mathbf{Y}$ channels filtered by the $ij^{\text{th}}$ frequency-selective filter	
	are called $\mathcal{N}_{ij}$ and $\mathbf{Y}_{ij}$ . The NCC block computes the normalized	40
2.0	cross correlation between two input images	49
3.8	The correlation surfaces of $\mathcal N$ and $\mathbf Y$ channels	50
3.8	The correlation surfaces of $\mathcal N$ and $\mathbf Y$ channels (cont.)	51
3.8	The correlation surfaces of $\mathcal N$ and $\mathbf Y$ channels versus horizontal and	
	vertical frequencies. For each example color and NIR images are	
	shown on the left and the correlation surface is on the right (cont.).	52
3.9	The correlation surface averaged for 30 pairs of $\mathcal N$ and $\mathbf Y$ channels.	
	The minimum correlation is 0.5159 and occurs at $f_x = f_y = 0$ , and	
	the maximum is 0.7721 at $(f_x = 0.3, f_y = 0)$ and $(f_y = 0, f_x = 0.3)$ .	52
3.10	A general CFA of size $4 \times 4$ . Each filter in this CFA transmits a mix-	
	ture of primary colors (red, green, blue) and NIR. The $\alpha$ coefficients	
	are the overall transmittance of each filter in the corresponding band.	
	The filters are assumed to equally transmit NIR, hence $\alpha_{\mathcal{N}} = 1$ for	
	all filters. These coefficients are generated randomly for illustration	
	purposes alone	54
3.11	Each patch $\mathbf{x}_0$ of size $M \times M$ is demosaized using a neighborhood of	
	size $(2L+1)M \times (2L+1)M$ . In this example: $M=4$ , and $L=1$ .	
	The size of CFA is $M \times M$ , and every patch of size $M \times M$ is sampled	
	by the same periodic pattern	54
3.12	The Sobel filter does not extract the highest frequencies that might	
	be contaminated with high-frequency noise. On the other hand, the	
	maximum response of the simple derivative filter occurs at the maxi-	
	mum frequency	58

List of Figures xvii

3.13	$4 \times 4$ CFAs designed by (a) Lu's algorithm [91] and (b) our method. The color of each filter indicates the mixture of primary colors it transmits and the numbers on each pixel are the coefficients of each mixture. All filters equally transmit NIR $(\alpha_{\mathcal{N}} = 1)$	59
3.14	Ground-truth color images. In Figure 3.15 we compare the zoomed-in regions of ground-truth images with those of the images reconstructed by Lu's algorithm [91] and our method.	61
3.15	For each example, the top row shows color images and the NIR images are placed in the bottom row	62
3.15	For each example, the top row shows color images and the NIR images are placed in the bottom row (cont.)	63
3.16	The CFA obtained by optimizing four coefficients (transmittance in red, green, blue, and NIR bands) for each filter. Note that six filters completely block the NIR radiation	64
3.17	Both our method and Lu's fail to correctly reconstruct the patches where color and NIR do not share the same high-frequency content.	66
4.1	(a) Color arrangement in one patch of the image sampled by the RGBN CFA. Both red and black pixels contain mixtures of NIR and red intensities. (b) A block of size $4 \times 4$ in the CFA image. Indices in this patch show how we arrange the measurements in $\mathbf{z}$ (see equation (4.4)). (c) Separating NIR and color channels results in subsampled color channels with the shown arrangements	70
4.2	Patches with white borders share the same high-frequency information between red and NIR. The ones in yellow boxes are dissimilar patches where high-frequency components of red and NIR are quite different	75
4.3	Left column: RGB images, and the right column: NIR images. (a) Ground-truth images. (b) By using the similar transform, the differences between color and NIR details are discarded (patches inside white boxes). (c) Although successful in reconstructing dissimilar patches, the dissimilar dictionary does not retrieve the shared edges faithfully (see the "PHILIPS" logo). (d) By fusing the results of similar and dissimilar dictionaries, high-quality estimations of all patches are obtained	77
4.4	Red and NIR edge maps of images reconstructed by the dissimilar dictionary. These maps provide an estimation of the inherent correlation between the high-frequency details of red and NIR images in local patches. For instance, the screen looks different in red and NIR ground-truth images, which is reflected in the differences of edge maps in the corresponding patches. Edge pixels are shown in black.	78
	maps in the corresponding patches. Edge pixels are shown in black.	10

xviii List of Figures

4.5	An illustration of red and NIR filters spectral transmittance for the values shown in Table 4.1 for $T_L = 0.2$	80
4.6	(a) Ground-truth images (b) By using only the transform that is trained for similar patches, the inherent differences between color and NIR are not preserved in the reconstructed images. (c) By combining the results obtained by both similar and dissimilar transforms, our algorithm successfully maintains the differences	84
4.6	(d) Ground-truth images (e) By using only the transform that is trained for similar patches, the inherent differences between color and NIR are not preserved in the reconstructed images. (f) By combining the results obtained by both similar and dissimilar transforms, our algorithm successfully maintains the differences (cont.)	85
4.7	The color filter arrays proposed by five single-sensor acquisition techniques. In all CFAs, the black pixel represents an NIR-pass filter. Our RGBN CFA (e) is the only one that models visible and NIR radiation leaking into NIR and color filters, respectively	87
4.8	From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7	92
4.8	From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7 (cont.).	93
4.8	From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7 (cont.).	94
4.8	From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7 (cont.).	95
5.1	(a) The focus error of simple and compound lenses versus wavelength (re-drawn based on data from [1]). (b) The achromatic doublet lens is a combination of convergent crown glass and divergent flint glass elements	100

List of Figures xix

5.2	(a) Sharp color image. (b) Blurred NIR image. Second row: (c) the result of deblurring guided by the color image without using similarity maps contains false edges that are not present in the NIR image. (d) By using the similarity maps, the algorithm preserves the differences between color and NIR images. Third row: As the similarity maps provide a rough estimation of correlation between gradients, the image in (d) is not as sharp as the image (c). (e) Using the multiscale approach presented in Subsection 5.1.3, the deblurring results in a sharper image while the differences between color and NIR are kept.	103
5.3	A pyramid of $\mathcal{N}$ and $\mathbf{Y}$ images with $p+1$ scales. The kernel in each scale is obtained by downsampling the kernel in the previous finer scale	.104
5.4	A schematic of our multiscale deblurring algorithm for the first two coarsest scales $(p \text{ and } p-1)$ . The procedure shown in (a) and (b) is repeated for all scales until the full-resolution NIR is deblurred	106
5.5	The second row shows a patch where high-frequency details of color and NIR images are inherently different. The similarity maps computed through single-scale and multiscale deblurring have small values (dark pixels) in the corresponding region. The third row shows a patch where color and NIR high-frequency details are similar. Note that compared with the map computed in one scale, the multiscale map has larger values (bright pixels) in this region. The input images	
5.6	of both algorithms are the sharp color (a) and blurred NIR (e) images (a) The blurred NIR image, (b) the result of the blind deblurring	.108
	algorithm of Krishnan <i>et al.</i> [82], and (c) the result of the deblurring step of Krishnan's algorithm using our estimated kernel	110
5.7	(a) Sharp color and (b) blurred NIR images are captured with the same focus settings. (e) Sharp NIR image is captured after re-focusing for the NIR shot. Without using the similarity maps the painting and text on the paper appear in the deblurred image (second and fourth rows), although they are invisible in the ground-truth NIR image. Our algorithm successfully preserves these inherent differences between color and NIR images, and retrieves the lost details in the NIR image	111
5.7	(f) Sharp color and (g) blurred NIR images are captured with the same focus settings. (j) Sharp NIR image is captured after re-focusing for the NIR shot. Second row shows that without using the similarity maps, intrinsic differences between color and NIR details are discarded. Although our algorithm maintains these differences, the results are not as sharp as (h), because the similarity maps are not accurate. In the following, we show how our multiscale approach	
	addresses this challenge.	119

xx List of Figures

5.8	The deblurred images obtained using the TV regularization (5.19) and the deblurring step of Krishnan's algorithm [82] suffer from ringing artifacts. The images deblurred by our algorithm do not exhibit these artifacts. This shows the superiority of using an image-specific prior (the gradients of the sharp color image) over general distributions to regularize deblurring. As the kernel estimated with our algorithm produces better results, to have a fair comparison between deblurring techniques, we used our kernel with all algorithms. The differences are more easily seen when this figure is viewed on a screen	114
5.9	Second row shows that both single-scale and multiscale deblurring preserve the inherent differences between color and NIR images. Multiscale deblurring (with $p=2$ ) results in sharper images as it computes the similarity maps more accurately (third and fourth rows)	116
5.9	Multiscale deblurring (with $p=2$ ) results in sharper images as it computes the similarity maps more accurately (cont.)	117
5.10	(a) Sharp color image. (b) Blurred NIR image. The images deblurred in (c) two scales, (d) three scales, and (e) five scales. (f) Sharp (ground-truth) NIR image. Increasing the number of scales leads to sharper details. However, the differences are not always noticeable, specifically beyond two scales	118
5.11	(a) Sharp color image. (b) Blurred NIR image. (c) The guided image filtering [58] with a small neighborhood (small $r$ ) does not result in sharp details (fifth row). (d) The guided image filtering with $r=50$ eliminates the differences between the gradients of color and NIR images, and does not even preserve the NIR intensities faithfully. Notice how dark the pen is in the second row of (d), although it is bright in the ground-truth NIR image. (e) Our algorithm produces sharper images while maintaining the intrinsic differences between color and NIR. (f) Sharp NIR image	119
6.1	A pair of color and NIR images captured with the same focus settings. The color image is focused in the middle of the scene (zoomed-in on the second row), where the NIR image is out of focus. The NIR image is, however, sharp in the background where the color representation is blurred	122
6.2	2D is the lens diameter. The sensor is placed at distance $x$ from the lens. A source point at distance $d$ from the lens produces a blurred disk of radius $r$ on the sensor. Light rays emitted from distance $d$ are focused at distance $y$ from the lens	199

List of Figures xxi

6.3	Comparison of foreground-background (in-focus/out-of-focus) masks. The foreground objects (the lens cap and the book) are in focus in the color image. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale	128
6.4	Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale	129
6.5	Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale	130
6.6	Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale	131
6.7	Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and the ones in the background are converted to gray scale.	132
6.8	Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale.	133
6.9	(a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138]. (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground where the color image is sharper. Fourth row: Background patches. (c) and (f) The masks obtained by our algorithm and the color-based segmentation that requires user input produce the best deblurring results. (d) and (e) The results of [138] and [23] are blurred and/or suffer from severe halo artifacts	134
	halo artifacts.	104

xxii List of Figures

6.10	(a) Input color image. (b) Input NIR image. (c) Image deblurred	
	using the color-based mask. (d) Image deblurred using the mask	
	of [138]. (e) Image deblurred using the mask of [23]. (f) Our result.	
	The foreground-background masks obtained by different techniques	
	are shown in the first row (c)-(f). Foreground pixels are shown by	
	their original intensities, and background pixels are darkened. Second	
	and third rows: patches on the foreground where the color image is	
	sharper. Fourth row: Background patches. Our algorithm obtains	
	the most accurate mask. Hence, the images deblurred by our mask	
	(f) is sharper than other images (second and third rows)	135

 List of Figures xxiii

6.13	(a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138]. (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground where the color image is sharper. Fourth row: Background patches. All algorithms are successful in obtaining accurate masks for this simple scene. As a result, deblurred images look similar. The mask computed by [138] results in the best deblurring for patch in the second row. Arrows in the color image point to the regions zoomed-in in the second and third rows	138
6.14	(a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138]. (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground where the color image is sharper. Fourth row: a background patch. Second row: our result is slightly sharper compared with other algorithms (differences are most noticeable on the screen). Third row: Our algorithm mislabels this patch as background and does not deblur it. Fourth row: Although this patch is detected as foreground by our algorithm, the deblurred image does not suffer from any artifacts in this region. The arrow in the color image points to the small region that is zoomed-in in the second row	139
6.15	The blur kernel map indicates the $\sigma$ parameter of the blur kernel in each patch. Positive $\sigma$ values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where $\sigma$ is negative. The objects are numbered in the order of their distance to the camera	144
6.16	The blur kernel map indicates the $\sigma$ parameter of the blur kernel in each patch. Positive $\sigma$ values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where $\sigma$ is negative. The objects are numbered in the order of their distance to the camera	145

xxiv List of Figures

6.17	The blur kernel map indicates the $\sigma$ parameter of the blur kernel in each patch. Positive $\sigma$ values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where $\sigma$ is negative. The objects are numbered in the order of their	
6.18	distance to the camera	146
6.19	distance to the camera	147
6.20	distance to the camera	148 149
6.21	The blur kernel map indicates the $\sigma$ parameter of the blur kernel in each patch. Positive $\sigma$ values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where $\sigma$ is negative. The objects are numbered in the order of their distance to the camera	150
6.22	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and to deblur each region separately	151
6.23	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions	
	with the same label and to deblur each region separately	152

List of Figures xxv

6.24	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and to deblur each region separately	153
6.25	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and to deblur each region separately	154
6.26	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and to deblur each region separately	155
6.27	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and to deblur each region separately	156
6.28	(a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and to deblur each region separately	157

xxvi List of Figures

6.29	The blur kernel map obtained by our algorithm can be effectively used	
	to deblur the NIR image in patches where the color image is sharper.	
	Moreover, the color image can be deblurred on the background plane	
	where the NIR image is sharper	159
6.30	For this scene the amount of blur in the NIR image with respect to the	
	color image is spatially invariant. Hence, estimating the blur-kernel	
	map does not improve the deblurring result	160
7.1	The PixelTeq color and NIR camera (PixelCam) monitors coffee beans.	
	Visible and NIR active light sources are used for this demonstration.	
	The left half of the screen shows the color image, and the right part	
	shows the NIR image. Note that the NIR light leaks into the color	
	filters. The NIR radiation is observed as purple stripes in the color	
	image. This photo is taken at the Photonics West 2015 exhibition	165

## **List of Tables**

2.1	Five main categories of color demosaicing algorithms	18
2.2	The summary of the single-sensor multispectral acquisition techniques.	32
3.1	Correlation values between four subbands for 30 pairs of images. The correlation between the low frequencies of $\mathcal{N}$ and $\mathbf{Y}$ channels is usually small. In most cases, the correlation between high-frequency subbands is stronger.	44
3.2	Correlation values between four subbands for 30 pairs of images. The reported values show the correlations between NIR and different color	
3.3	channels	47 60
4.1	The transmittance of red and NIR filters in red and NIR bands (see $(4.7)$ and $(4.8)$ ). For each $T_L$ , the four coefficients define the measurement matrix. We arbitrarily choose these coefficients to study the effect of dictionary on the quality of reconstructed images	80
4.2	The effect of training set on the sparsifying dictionary. The reported results are averaged over 60 pairs of color and NIR images. For different dictionaries, we use the same measurement matrix, with non-zero coefficients shown in Table 4.1. (a) For these experiments, two dictionaries are trained over 12000 and 75000 patches extracted from 30 pairs of color and NIR train images. (b) In this experiment, we use two dictionaries trained over two different sets of images. Each	
4.3	dictionary is trained over 12000 patches	81
4.4	values of $T_L$	82
	cially for NIR images	83

xxviii List of Tables

4.5	Changing the patch size does not significantly affect the performance	
	of our reconstruction algorithm	83
4.6	CPSNR and PSNR results for 60 pairs of color and NIR images sam-	
	pled and reconstructed by five different techniques	90
4.7	SSIM results for 60 pairs of color and NIR images sampled and re-	
	constructed by five different techniques. To compute color SSIM, we	
	compare the Y channels of original and reconstructed color images,	
	as SSIM is developed for gray-scale images	91
4.8	VIF results for 60 pairs of color and NIR images sampled and re-	
	constructed by five different techniques. To compute color VIF, we	
	compare the Y channels of original and reconstructed color images.	91
4.9	$\Delta E$ results for 60 color images sampled and reconstructed by five	
	different techniques. As $\Delta E$ measures the color accuracy, we only	
	compute it for color images.	91
4.10	Percentage of pixels suffering from the zipper artifacts averaged over	
	60 color images sampled and reconstructed by five different techniques.	91
5.1	Blur kernel estimation: for each ground-truth $\sigma$ value (number of	
	pixels) that is used to blur the NIR images, we report the $\sigma$ estimated	
	by using the blurred NIR and sharp color images. The experiment is	
	conducted for 30 pairs of images	109

#### Chapter 1

### Introduction

The quality and functionalities of digital cameras have made immense progress in the 40 years since 1975 when the first prototype was built by Steve Sasson at Kodak [2]. As a result of rapid developments in this market, since 2003 digital cameras have outsold their analog ancestors. According to Tomi Ahonen [7], in 2014 alone, 1.8 billion digital consumer cameras were sold.

Despite extensive developments in digital cameras, in some cases the quality of captured images is still much lower than what is perceived by the human eye (See Figure 1.1). More importantly, the algorithms that aim to understand the contents of the scene by analyzing images are prone to failure in many cases. See one example in Figure 1.1-(c) and (d).

An effective approach to overcoming these limitations is to capture several images representing the scene. The captured images are fused either to create a high-quality image or to extract more information about the scene (see Figure 1.2). In the examples illustrated in Figure 1.2 several color images, representing the scene in the visible band with the wavelength range of 400-700 nm, are fused.

Combining color and near-infrared (NIR) images of the scene has also proven beneficial both in improving the quality of color images and in understanding the scene. NIR is a part of the electromagnetic spectrum next to the red band with the wavelength range of 700 nm to approximately 1100 nm. Figure 1.3 shows two pairs of color and NIR images. Although the spatial information (object shapes and boundaries) contained in these images are generally similar, there are many inherent differences between the color and NIR representations of the scene. For instance, the first example in Figure 1.3 shows that vegetation usually reflects more strongly in NIR, hence it appears brighter in this image. Whereas water absorbs NIR and looks dark in the NIR image, while, depending on the depth, what is beneath water could be visible in the color image. Reflection in the NIR band is mostly material dependent and the objects made of the same material reflect more or less the same in NIR. Figure 1.3-(c) and (d) illustrates an example.

2 Chapter 1.

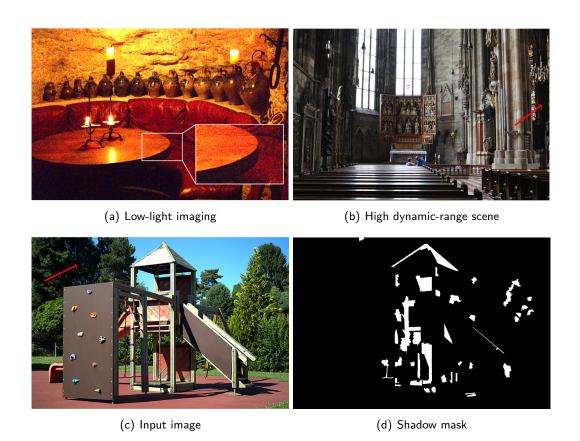
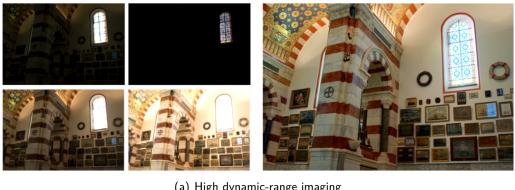


Figure 1.1: This figure illustrates some of the limitations of digital cameras in capturing the scene and the deficiencies of an algorithm in extracting information from color images. (a) The image captured in a low-light environment without flash contains strong noise [117]. (b) A picture representing a high dynamic-range scene does not preserve the details in the dark regions. The details on the wall (red arrow) are completely lost. (c) Input image [123], and (d) shadow mask computed by the algorithm of Guo et al. [53]. White pixels mark shadow regions. Notice that no shadow pixels are detected on the tree in the upper-left part of the image (red arrow), although it is under the shadow.



(a) High dynamic-range imaging



(b) Flash/no-flash photography

Figure 1.2: (a) In high dynamic-range imaging, several pictures with different exposure times are recorded and fused into one image that contains rich details in both dark and bright regions of the scene [36, 101]. Left: several pictures that are captured with different exposure times. Right: the details of both dark and bright regions are visible in the final image produced by the algorithm of Meylan et al. [101]. (b) Petschnigg et al. [117] propose to capture a pair of images—with and without flash—to obtain a clean image in low-light conditions. The clean flash image is used to denoise the no-flash image that preserves the ambient illumination. From left to right: no-flash image, the clean image captured with flash, and the denoised image. These images are made available by authors at http://research.microsoft.com/en-us/um/redmond/projects/flashnoflash/.

Chapter 1.

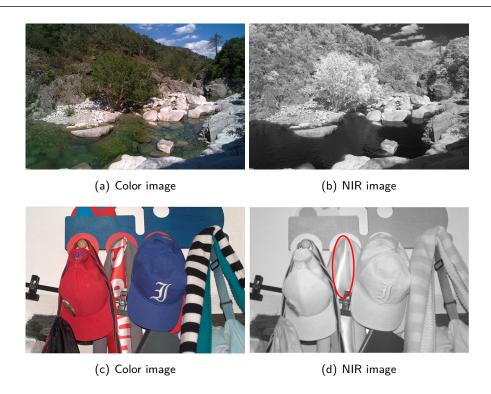


Figure 1.3: (a) and (b) Vegetation usually looks brighter in the NIR image, and sky is darker in NIR. Water absorbs NIR and appears dark in this image, while, the stones beneath water are visible in the color image. (c) and (d) The scarf in the middle looks more or less the same in the NIR image, despite having different colors.

Multispectral data captured in the NIR band of the spectrum has been conventionally used in specialized areas such as security and surveillance, food industry, medical applications, and remote sensing [41, 133, 155], where imaging systems specially designed for industrial and scientific purposes are employed. The additional information provided by the differences between color and NIR images is recently used in several computational photography and computer vision tasks such as high dynamic-range imaging [158], image denoising [152, 103], dehazing [132], shadow detection [123], material-based segmentation [131], and semantic segmentation [129]. Figure 1.4 illustrates some of these applications.

Despite the success of fusing color and NIR images in the above applications, an affordable and convenient approach to capturing these images is not yet available. Capturing color and NIR images is currently possible by two main techniques that use commercially available products. The first approach uses a beam splitter to divide the incoming light to visible and NIR radiations that are, respectively, captured by a visible and an NIR detector. This acquisition system is expensive and bulky. It

is also possible to capture both color and NIR images by a silicon sensor, as silicon is sensitive to visible and NIR bands of the spectrum (see Figure 1.5). Silicon is the light-sensitive material in sensors of most current color cameras. However, in color cameras the NIR light is blocked by an interference filter called "hot mirror". The second technique for capturing color and NIR images is to use a color camera, after removing its hot-mirror, in two sequential shots, when NIR-blocking and visible-light blocking filters are placed in front of the lens. This approach is time-consuming and movements of the camera or scene between two shots might introduce severe registration issues. It is specifically impractical for scenes with moving objects, similar to the one shown in Figure 1.6.

Combining color and NIR images benefits consumer-level photography, where none of the above acquisition systems is acceptable. In this thesis, we study the simultaneous acquisition of color and NIR on a single silicon sensor. This approach would not suffer from the shortcomings of current acquisition systems. As it requires only one sensor, it is affordable and has the potential to be integrated inside small devices such as cell phones. Moreover, perfect alignment is achieved as both images are captured jointly.

We address two main challenges in the single-sensor capture of color and NIR. We first study spatial and spectral sampling of the scene (Chapters 3 and 4). We then focus on the distortions caused by deficiencies of the optical elements, specifically the phenomenon of chromatic aberration (Chapters 5 and 6). In the following we describe these challenges and briefly review our solutions to these problems.

6 Chapter 1.



(a) Skin smoothing [43]



(b) Image dehazing [132]



(c) Shadow detection [123]

Figure 1.4: Some applications of fusing color and NIR images in "everyday" photography. For each row, from left to right: the color image, the NIR image, and the result.

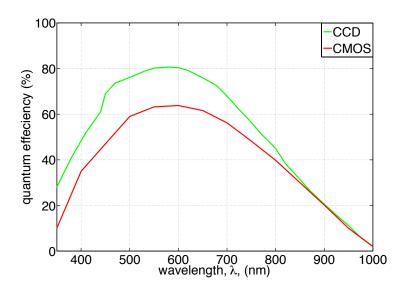


Figure 1.5: The quantum efficiency of typical CCD and CMOS sensors made of silicon. Both sensors are sensitive to visible (400-700 nm) and NIR (700-1100 nm) bands. Data for the CCD sensor is obtained from [119], and data for the CMOS sensor is provided by [95].

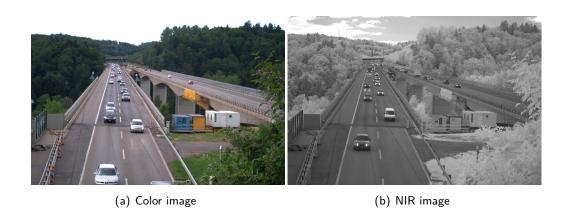


Figure 1.6: Capturing color and NIR images in two sequential shots leads to registration issues. When there are moving objects in the scene, this acquisition approach is unfeasible.

8 Chapter 1.

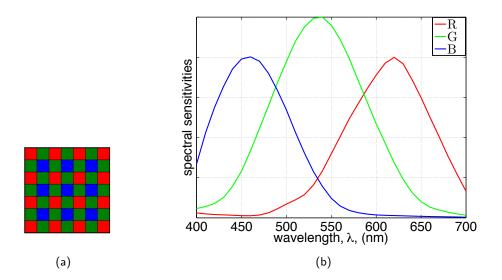


Figure 1.7: (a) The Bayer color filter array that is used in most color cameras. (b) The overall sensitivities of a silicon sensor and color filters in Bayer, measured for a consumer color camera.

### 1.1 Spatial and Spectral Sampling

In single-sensor color imaging a color filter array (CFA) spectrally samples the scene. Each filter in the CFA transmits part of the visible light and blocks the rest. The most popular CFA in color imaging, called the Bayer CFA [13], is shown in Figure 1.7-(a). The overall sensitivities of the silicon sensor and the Bayer filters in a consumer color camera are shown in Figure 1.7-(b). As this figure shows, each filter in the CFA strongly transmits the corresponding band (for instance middle wavelengths for the green filter) and almost blocks the rest of the spectrum.

The image captured using a single sensor and a CFA is a gray-scale image that carries the information about only one color channel at each pixel. This image is usually called the CFA image or the mosaiced image. After acquisition, an algorithm called "demosaicing" estimates the missing color values at each pixel by using the sensor measurements. The output of demosaicing is referred to as the full-resolution image or the demosaiced image. Figure 1.8 is a schematic of CFA sampling and reconstruction in a single-sensor color camera.

Similar to color imaging, we assume that color (red, green, and blue) and NIR channels are sampled by a CFA. We address two main questions regarding this sampling: (1) designing the color filter array, and (2) developing a demosaicing algorithm customized to color and NIR acquisition. To answer these questions, we first study the correlation characteristics of natural color and NIR images. We analyze the correlation in different frequency bands and show that, although not

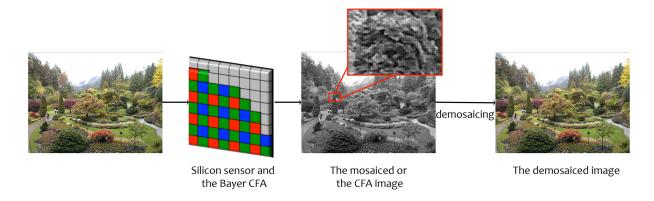


Figure 1.8: A schematic of CFA sampling and demosaicing in a single-sensor color camera.

as strongly as color channels, color and NIR images are mostly correlated in high frequencies. The details of this study and the results of our analysis are explained in Chapter 3.

We exploit the results of the spatio-spectral analysis to custom design a color filter array for color and NIR acquisition. Our algorithm is based on the work of Lu et al. [91], in which each filter in the CFA is assumed to transmit a different mixture of all channels (red, green, blue, and NIR). To reconstruct full-resolution images, using a linear demosaicing is proposed. The optimum transmittance for different filters of the CFA and the optimum demosaicing are estimated by solving an optimization problem that minimizes the error of reconstructing full-resolution images.

We change the optimization problem of Lu et al. [91] by adding a regularization term that enforces the reconstructed color and NIR images to comply with the results of our spatio-spectral analysis. More specifically, our approach designs CFA and demosaicing by minimizing the error of reconstructing color and NIR images, and the difference between the high-frequency components of these images. We call the CFA and demosaicing designed by this method the optimized CFA and the optimized demosaicing. In Chapter 3, we describe this algorithm and compare its performance with the method of Lu et al. [91].

As we will show in Chapter 4, the optimized CFA and the optimized demosaicing outperform all other color and NIR demosaicing methods. However, implementing the optimized CFA is more complicated than the CFAs currently used in color imaging, which causes additional manufacturing costs. To reduce these costs, we study a CFA similar to the Bayer CFA (Fig. 1.7-(a)) that is already placed in most color cameras. The only difference between our proposed CFA and the Bayer CFA is that one of the two green filters is replaced with an NIR-pass filter. Our proposed CFA, called the RGBN CFA, is shown in Figure 1.9-(a).

We assume that, similar to most CFAs in color imaging, inexpensive dye filters

Chapter 1.

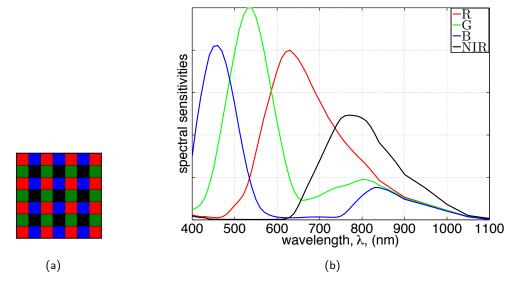


Figure 1.9: (a) The RGBN CFA proposed to sample color and NIR channels. (b) The overall sensitivities of color and NIR filters and a silicon sensor. Each color filter transmits one color channel and part of the NIR band as the NIR-blocking filter is removed. The NIR filter transmits both NIR and red channels. The visible range: 400-700 nm, the NIR band: 700-1100 nm.

are used in the RGBN CFA. The main difficulty in using these filters is that, without significantly increasing the price, it is unfeasible to have a sharp cut-off and narrow transition regions between the filter pass-band and its stop-band. Figure 1.9-(b) shows the overall sensitivities of a silicon sensor and dye filters without the hot mirror. All color filters in the RGBN CFA partly transmit the NIR light, and the NIR filter transmits the visible radiation, especially the red light.

As a result of using such filters, the sensor measures a mixture of one color channel and NIR intensities at each pixel. This further complicates the task of reconstructing full-resolution images, as the channels are not only subsampled but are also mixed together. We develop an algorithm that separates color and NIR intensities given only the mixed subsampled sensor measurements. As the separation problem is highly underconstrained, incorporating prior knowledge about the target signals (color and NIR channels) is required. Once again, we exploit the spatial and spectral correlations of these images. For this problem, however, we adopt the tools developed in compressive sensing and sparse reconstruction, where a high-dimensional signal is recovered from only a few measurements. In Chapter 4, we detail the use of the RGBN CFA and our reconstruction algorithm in the joint acquisition of color and NIR.

### 1.2 Chromatic Aberration

Multispectral imaging systems that use a single sensor for the joint acquisition of different channels suffer from chromatic aberration (CA) distortions introduced by the imperfections of optical elements. Chromatic aberration occurs as the refractive index of a simple lens changes with wavelength, which causes the light rays with different wavelengths to be focused at different distances from the lens. This phenomenon results in the image of only one channel being sharp on the sensor, while the images of other channels are blurred depending on their wavelength difference to the sharp channel. See Figure 1.10-(a) for an illustration.

Chromatic aberration distortions are wavelength-dependent and are amplified as the captured wavelength range increases. In joint acquisition of color and NIR, a range of approximately 700 nm (400-1100 nm) is captured, which is more than twice the range captured in color imaging (400-700 nm). This leads to severe distortions that cannot be completely removed by the solutions used for correcting chromatic aberration in color imaging. Figure 1.10-(b) shows a pair of color and NIR images captured with the same lens and focus settings. The lens, corrected for visible wavelengths, results in sharp color channels, while the NIR image is significantly blurred.

We consider the scenario where the color image is in focus and sharp, and the NIR image captured with the same focus settings is out of focus and blurred. We formulate the task of correcting for chromatic aberration as deblurring the NIR image. Nonetheless, our deblurring methods can be modified for the case where the NIR image is in focus and the color counterpart is blurred. In Chapter 5, we propose a deblurring algorithm that first estimates the amount of blur in NIR. In the next step, the algorithm uses the gradients of the sharp color image to recover the high-frequency details of the NIR image. As the gradients of color and NIR images are not strongly correlated in all patches, our algorithm locally adapts the contribution of color gradients in deblurring. To achieve this performance, we develop a multiscale deblurring scheme that estimates the correlation between color and NIR gradients. In Chapter 5 we show that the proposed algorithm is successful in retrieving the lost details of the NIR image and in preserving the inherent differences between color and NIR.

The amount of out-of-focus blur introduced by chromatic aberration is depth-dependent. In Chapter 5, we study the cases where the blur does not change significantly across the image because the focus plane is located further than the hyperfocal distance of the lens. In this case, the blur can be assumed uniform. However, in some situations, for instance when the depth of field is shallow, the blur introduced by chromatic aberration is spatially variant. Figure 1.11 shows one example where the amount of defocus blur changes across both color and NIR images. As NIR wavelengths are longer than the visible band, the focus plane of NIR is further away from the camera compared with the focus plane of color channels. Hence, the

12 Chapter 1.

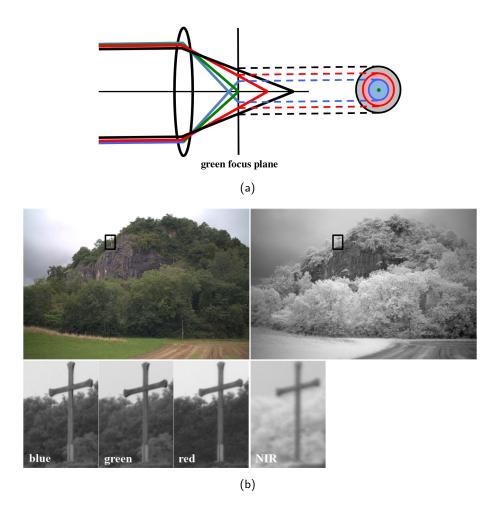


Figure 1.10: (a) A simple lens converges light rays with different wavelengths at different distances. If the sensor is placed at the focus plane of the green light, only the image of scene information with this wavelength is sharp. The disks on the right illustrate the lens blur kernels on the sensor for other wavelengths. The black rays represent NIR radiation. (b) A pair of color and NIR images that represent the same scene. The color image is in focus and all color channels are sharp, whereas the NIR image, captured with the same lens and focus, suffers from chromatic aberration and is blurred.

objects further away from the visible focus plane appear less blurred in the NIR image, and vice versa (see the background in Figure 1.11).

In Chapter 6, we address the case of spatially varying blur kernels, assuming that the blur is uniform in small local patches. In each local patch, we perform an optimization to find the kernel that best describes the blur either in NIR or in the color patch. As uniform patches are indifferent to blur, it is not possible to estimate the blur kernel for these patches. We propagate the kernels estimated in textured patches to uniform patches by solving an optimization that promotes a smooth change of blur kernels across the image. This results in a blur-kernel map that contains the amount of blur in each local patch. Once the blur-kernel map is estimated, we separately deblur each region of the image that is distorted by a uniform blur kernel, using the algorithm presented in Chapter 5.



Figure 1.11: As the defocus blur is depth dependent, the amount of blur in the NIR image with respect to the color image changes from foreground to the background plane. The foreground object (insets with green borders) is in-focus in the color image and blurred in NIR. Whereas, the background region (insets with red borders) is sharper in the NIR image.

Chapter 1.

### 1.3 Image Acquisition

All the images used for the experiments of this thesis are captured by a Canon Rebel T1i camera. We removed the hot-mirror filter, hence the camera is capable of sensing both visible and NIR radiations. The color and NIR images representing each scene are recorded in sequential shots when, respectively, NIR-cut and visible-light-cut filters are placed in front of the lens. For outdoor scenes, the same exposure time is used for both color and NIR images. For indoor scenes, the exposure time is usually increased for the NIR image, as some of the common indoor light sources such as fluorescent do not radiate strongly in the NIR range. To reduce misalignment errors between color and NIR images, we avoid photographing scenes with moving objects, and all the images are captured while the camera is fixed on a tripod. The remaining registration issues are resolved by aligning color and NIR images using feature-point matching.

The image set used in Chapters 3 and 4 contains 60 pairs of color and NIR images that represent diverse indoor and outdoor scenes. To avoid chromatic aberrations, the camera is differently focused for color and NIR images of each scene. We simulate the process of CFA sampling and demosaicing for these images.

For the experiments of Chapter 5 and 6, one color and two NIR images are captured for each scene. The camera and lens are adjusted such that the color image is in focus and sharp. By maintaining the same lens and focus settings, the first NIR image is captured. This image, as a result of chromatic aberration, is out of focus and blurred. The algorithms presented in Chapters 5 and 6 deblur the NIR image by using the sharp color image as the guide. The second NIR image is captured after re-focusing for the NIR shot, resulting in a sharp image that is used as the visual ground-truth in our comparisons.

### 1.4 Thesis Outline

This thesis is structured as follows. In Chapter 2 we review the literature on CFA sampling and demosaicing in color imaging. We explain the techniques proposed for single-sensor acquisition of multispectral images. We discuss chromatic aberration correction in color imaging and deblurring algorithms as a general approach for reducing chromatic aberration distortions.

We analyze the correlation characteristics of color and NIR images in Chapter 3, and explain the algorithm that exploits these characteristics in designing the optimum CFA and the optimum demosaicing.

In Chapter 4 we study the use of the RGBN CFA in the joint acquisition of color and NIR images. We present an algorithm that separates color and NIR channels sampled by the RGBN CFA. As our method is similar to the solutions proposed in compressive sensing, we briefly review the basic principles in this field and discuss the main differences between our algorithm and conventional compressive sensing algorithms. In this chapter, we thoroughly compare different color and NIR joint acquisition techniques with our proposed methods.

We devote chapters 5 and 6 to analyzing chromatic aberration in joint acquisition. In Chapter 5, we present our deblurring algorithm for the cases where the chromatic aberration distortions are assumed to be uniform across the image. We compare the NIR images deblurred by our algorithm with those produced by the general method used in color imaging for correcting chromatic aberration. We then extend our blur kernel estimation and deblurring algorithms for spatially varying chromatic aberrations in Chapter 6.

We conclude the thesis in Chapter 7 by discussing other issues in designing a consumer color-and-NIR camera. Furthermore, we explain how our algorithms and the results of our studies are beneficial in designing a more general multispectral acquisition systems.

16 Chapter 1.

## Chapter 2

## Related Work

As we study the joint acquisition of color and NIR images by using a single silicon sensor, our work is similar to single-sensor color imaging. We review major color demosaicing techniques in Section 2.1. The research community has mainly focused on demosaicing the images sampled by the Bayer CFA, however, in the past few years the design of the CFA has also been investigated. In Section 2.2 we review these studies.

Color and NIR acquisition is also related to multispectral imaging where several spectral bands are captured. Although not as extensive as color imaging, single-sensor multispectral imaging has received some attention in the past few years. In Section 2.3, we describe the main techniques in this field and a few studies that address the joint capture of color and NIR.

In Section 2.4, we explain the principles of chromatic aberration reduction. We start by reviewing the algorithms designed to remove chromatic aberration in color imaging in Subsection 2.4.1. In Chapter 5, we formulate the task of chromatic aberration correction as deblurring the NIR image. Thus, in Subsection 2.4.2, we introduce the problem of deblurring and review the literature in this field.

### 2.1 Color Demosaicing

Color demosaicing has been extensively studied, with numerous solutions available from basic interpolation techniques to more sophisticated methods based on the graph theory, statistical and probabilistic frameworks, sparse recovery, etc. Despite all differences, the fundamental principle used in most color demosaicing algorithms is that color channels are usually strongly correlated. Different algorithms mainly vary in the approach they follow for exploiting the intra-channel correlation. In Subsections 2.1.1-2.1.5, we review five main categories of techniques explored in the literature. Table 2.1 summarizes the main assumptions in each of these categories and presents a brief overview of the demosaicing approaches.

	of the graph.		
	imizing the Laplacian matrix		
	saicing is performed by min-		
	a weighted graph and demo-	smooth.	graph theory $(2.1.5)$
[64]	The image is represented as	Color images are piecewise	Demosaicing based on the
	ples.		
	computed given the CFA sam-	domain.	
	the full-resolution image is	resentations in some transform	sparse decomposition (2.1.4)
[96, 151, 105, 122, 157]	The sparse representation of	Color images have sparse rep-	Demosaicing based on the
	are used to model the image.		
	such as total variation (TV)		
	Different regularization terms	distributions.	
	obtained by a MAP estimator.	represented by a few general	probability theory (2.1.3)
[18, 108, 60, 118, 128, 99, 31]	The full-resolution image is	Natural color images are well	Demosaicing based on the
	the full-resolution image.		
	are interpolated to reconstruct	chrominance channels.	
	and the chrominance channels	components compared with	
	tracted from the CFA image	tains larger high-frequency	decomposition (2.1.2)
[9, 35, 34, 85, 89]	Luminance channel is ex-	The luminance channel con-	Luminance-chrominance
	polated.		
26, 52]	channel differences are inter-	stant.	
148, 140, 28, 114, 115, 27, 88,	ing the color channels, the	color channels is locally con-	
$ \left  \ [45, 84, 54, 49, 112, 150, 87, 62, \ ] \right  $	Instead of directly interpolat-	The ratio or the difference of	Constant-hue (2.1.1)
References	General approach	Main assumption	Algorithm

Table 2.1: Five main categories of color demosaicing algorithms.

### 2.1.1 Constant-Hue Assumption

Many demosaicing algorithms assume that the ratio of color channels, or their difference, is constant [45, 84, 54, 49, 112, 150, 87]. This observation is called the constant-hue assumption. Following this assumption, usually the green channel, with the highest sampling frequency in the Bayer CFA and least amount of aliasing, is first interpolated. Afterwards the color difference signals,  $\mathbf{R} - \mathbf{G}$  and  $\mathbf{B} - \mathbf{G}$ , are interpolated. As the color difference is assumed to be constant inside each object,  $\mathbf{R} - \mathbf{G}$  and  $\mathbf{B} - \mathbf{G}$  contain less high-frequency content compared with the red and blue channels. Hence, interpolating color differences results in higher accuracy estimations compared with interpolating the red and blue channels.

The main difference between these algorithms is the interpolation technique used to reconstruct full-resolution green and color-difference channels. Usually, edge adaptive techniques that interpolate the samples along the edges are used [62, 148, 140, 28, 114, 115, 27, 88, 26]. To preserve the details and avoid blurring when estimating the missing value, these methods do not use the samples across the edge, where the constant-hue assumption is not valid.

To detect the direction of edges in each local neighborhood, usually the difference in the intensities of observed samples in both horizontal and vertical directions are computed. In [62], Hirakawa and Parks take a different approach, where instead of the gradients in a local neighborhood, they measure homogeneity in both horizontal and vertical directions. They define homogeneity as the number of pixels in a local neighborhood that have similar luminance and chrominance intensities. The interpolation is performed in the direction that has the highest homogeneity.

In [52], Gunturk et al. use a filter-bank scheme to decompose color channels into base and detail bands. In the first step of their algorithm, the detail bands of the missing green values are replaced with high-frequency details of red and blue channels in the corresponding pixels. Afterwards, to establish the correlation between the high frequencies of color channels, the detail subbands of red and blue are replaced by those of the reconstructed green channel. This process is iteratively followed by projecting the estimated red and blue channels into observed samples. The convergence analysis and a fast one-step implementation of Gunturk's algorithm is presented by Lu et al. in [92].

#### 2.1.2 Luminance-Chrominance Decomposition

Several demosaicing techniques decompose the CFA image into luminance and chrominance channels and estimate the full-resolution image by separately reconstructing these channels [9, 35, 34, 85, 89]. The luminance channel is usually calculated as a weighted average of color channels and contains spatial information about the scene, whereas, chrominance channels are expressed as the differences of color channels and convey low-frequency color information.

Alleysson et al. analyze the frequency representation of the CFA image in [9].

They express the Fourier spectrum of the CFA image based on the frequency representations of color channels as follows:

$$\hat{\mathbf{I}}_{CFA}(f_x, f_y) = \sum_{i \in \{R, G, B\}} \alpha_i \hat{\mathbf{C}}_i(f_x, f_y) 
+ \frac{1}{8} \sum_{\substack{r,s = -1 \\ r \neq 0, s \neq 0}}^{1} \hat{\mathbf{C}}_R(f_r, f_s) - \hat{\mathbf{C}}_B(f_r, f_s) 
+ \frac{1}{16} \sum_{\substack{r,s = -1 \\ r \neq 0, s \neq 0}}^{1} \hat{\mathbf{C}}_R(f_r, f_s) - 2\hat{\mathbf{C}}_G(f_r, f_s) + \hat{\mathbf{C}}_B(f_r, f_s).$$
(2.1)

In the above equation,  $\hat{\mathbf{I}}_{CFA}$  is the frequency representation of the CFA image,  $\hat{\mathbf{C}}_i$  represents the Fourier spectrum of the *i* color channel.  $f_x$  and  $f_y$  are, respectively, horizontal and vertical frequencies.  $f_r = f_x - r/2$  and  $f_s = f_y - s/2$ . For the Bayer CFA,  $\alpha_R = 1/2$ ,  $\alpha_G = 1$ ,  $\alpha_B = 1/2$ .

Figure 2.1 shows the frequency spectrum of an example image sampled by the Bayer CFA. The central part corresponds to the first term of (2.1) that contains the luminance information of the scene. The second and third terms of (2.1) express the chrominance channels modulated to high frequencies (named  $C_1$  and  $C_2$  in Fig. 2.1).

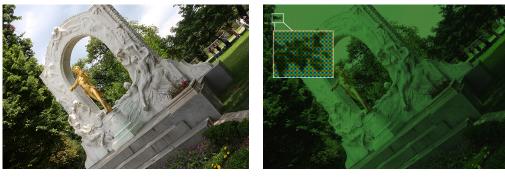
As the luminance information is concentrated at low frequencies of the CFA image, it is estimated by applying a lowpass filter to CFA samples.

The subsampled chrominance channels, expressed as differences of color channels, are calculated by subtracting the luminance channel from the CFA image. As a result of strong correlation between the high-frequency details of color channels, the chrominance channels are mainly lowpass. Hence, even a simple bilinear interpolation results in a high-quality estimation of these channels.

Following the same idea, Chaix de Lavaréne et al. in [35] adopt a Wiener filter-based demosaicing to reconstruct luminance and chrominance channels, and in [34], they generalize the luminance-chrominance demosaicing algorithm to be applicable to any CFA pattern.

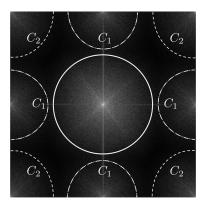
In [85], Leung *et al.* propose a method for designing optimum filters for luminancechrominance demultiplexing. They analyze different filters in terms of their computational complexity and demosaicing performance over a set of training images.

Lian et al. [89] modify the algorithm of Alleysson et al. [9] by estimating the luminance in green pixels by applying a passband filter to the CFA samples. As the Bayer CFA samples the green channel in every row and column of the image, there is no horizontal and vertical aliasing in the samples of this channel. Hence, estimating the luminance channel by a simple linear filter can be performed accurately. In contrast, a linear filter is not suitable for estimating the luminance channel at red and blue pixels. Lian et al. propose an iterative scheme for this task. First, initial estimates of red and blue channels are obtained by bilinear interpolation. These estimates guide the reconstruction of the luminance channel, which in the next



(a) Input image

(b) The CFA image sampled by the Bayer CFA



(c) Frequency spectrum of the CFA image

Figure 2.1: (a) Input image. (b) The CFA image shown in color. As the sampling frequency of the green channel is the highest, the image looks greenish. (c) The Fourier spectrum of the CFA image. The central part contains luminance information and chrominance channels (here shown as  $C_1$  and  $C_2$ ) are modulated to high-frequencies.

iteration is used to refine red and blue intensities. In this process, an edge-adaptive interpolation is used. After estimating the luminance channel, they apply a bilinear interpolation to calculate the full resolution chrominance channels (the difference of the CFA image and luminance).

### 2.1.3 Probabilistic Demosaicing

Similar to other estimation problems, the task of demosaicing can be formulated as the maximum a-posteriori probability (MAP) framework, where the following optimization is solved to reconstruct the full-resolution image [18]:

$$\mathbf{I} = \underset{\mathbf{I}}{\operatorname{argmax}} p(\mathbf{I}|\mathbf{I}_{CFA}), \tag{2.2}$$

where **I** is the full-resolution color image, and p(.) denotes the probability density function (PDF). Following Bayes' rule, the above optimization is re-written as:

$$\mathbf{I} = \underset{\mathbf{I}}{\operatorname{argmax}} p(\mathbf{I}_{CFA}|\mathbf{I})p(\mathbf{I})$$
 (2.3)

As probability functions in the above optimization are usually modeled by normal distributions, the optimum solution is obtained by minimizing the negative logarithm  $(-\log)$  of the above cost function:

$$\mathbf{I} = \underset{\mathbf{I}}{\operatorname{argmin}} - \log(p(\mathbf{I}_{CFA}|\mathbf{I})) - \log(p(\mathbf{I})). \tag{2.4}$$

 $-\log(p(\mathbf{I}_{\text{CFA}}|\mathbf{I}))$  is called the data term that forces the intensities of the estimated image to be similar to the observations ( $\mathbf{I}_{\text{CFA}}$ ).

The task of reconstructing a full-resolution color image from CFA samples is ill-posed, as one measurement is available at each pixel and two unknown color values are estimated. Hence the problem does not have a unique solution, unless there are additional constraints. The second term in (2.4),  $-\log(p(\mathbf{I}))$ , is a regularization term that constrains the ill-posed problem of demosaicing by modeling the statistical characteristics of natural images.

In [18], Brainard models the color image as a linear combination of several sinusoidal basis functions, where the weights are normal random variables. Mukherjee et al. [108] model the smoothness of natural images and the strong correlation between different color channels by a Markov random field (MRF), which is used as a regularization term in (2.4). Hel-Or and Keren use steerable wavelet coefficients to impose the smoothness only along edges at appropriate scales [60].

Portilla *et al.* model the task of demosaicing as an LMMSE (linear minimum mean squared error) estimation [118]. They learn the spatio-chromatic joint correlation characteristics of natural color images. The covariance matrices of training data are used to form local linear filters that are applied to demosaic the image.

In [128], Saito and Komatsu employ the total variation (TV) regularization in demosaicing, where they explore spatial and chromatic correlations, respectively, by minimizing the TV terms of each color channel and color differences. This idea is combined with the luminance-chrominance decomposition by Menon and Calvagno in [99] and Condat in [31]. They minimize the total variations for the difference of color channels and the luminance channel.

Another popular class of regularization terms is the sparsity measure of natural color images. In the next subsection, we review the demosaicing algorithms that are based on sparse recovery.

### 2.1.4 Demosaicing Based on Sparse Decomposition

In compressive sensing [37], ill-posed inverse problems are constrained assuming that the target signal has a sparse representation in some transform domain. For

instance, natural images are shown to be sparse when represented by transforms such as DCT or wavelet [47]. The transform domain that sparsely represents a class of signals is usually called "dictionary". The sparse-recovery algorithms are used in different image processing tasks such as denoising [39], inpainting [40], and super-resolution [153]. We explain the problem of sparse reconstruction in more detail in Chapter 4.

Mairal et al. develop a sparse-recovery technique for the general problem of color image reconstruction [96]. Their method is a generalization of the gray-scale image denoising algorithm [39] that uses the K-SVD algorithm [6]. K-SVD is proposed for training sparsifying transforms for specific classes of signals including gray-scale images. Mairal et al. first modify K-SVD such that it learns the correlation between different color channels. They also change the K-SVD algorithm in a way that it removes nonhomogeneous noise. Modeling the CFA sampling as the color image being corrupted by strong nonhomogeneous impulse noise that results in missing some color information at every pixel, they use this method in color demosaicing.

Wu et al. [151] propose a novel dictionary for the task of demosaicing. They first calculate an initial estimate of the full-resolution image using one of the available demosaicing algorithms. Using the principal component analysis (PCA), a dictionary is trained for each local neighborhood. These local dictionaries are used to refine the estimate of the full-resolution image by solving a sparse recovery problem.

Another demosaicing algorithm based on sparse decomposition is proposed by Moghadam et al. in [105]. In this paper, the spatial correlation between neighboring pixels and the spectral correlation between color channels are explored separately. Moreover, their demosaicing method is not limited to the Bayer CFA only. In [122], Rossi and Calvagno first estimate the luminance channel using all color samples. They modify the sparsity-based demosaicing approach proposed in [105] such that the recovered color image respects the luminance channel estimated in the first step.

Zhang and Tao explore using a super overcomplete dictionary in color demosaicing [157]. The dictionary is formed as a collection of L sparsifying transforms each representing a different class of image patches. Each patch in the training set is classified, based on its sparsity level, into one of the L classes, and is used to train the corresponding sparsifying transform. The L transforms are then used to reconstruct each mosaiced patch, based on its sparsity level.

#### 2.1.5 Graph-Based Demosaicing

Hu et al. in [64] propose a graph-based optimization for color demosaicing. They represent the image as a weighted graph, when each pixel is a node and similarity between two adjacent pixels is the weight of the connecting edge. The similarity metric is a combination of intensity similarity and spatial proximity of two pixels.

The Laplacian matrix of the graph is defined as

$$\mathbf{L}(u,v) = \begin{cases} \sum w(u,v) & u = v \\ -w(u,v), & u \sim v \end{cases}$$
 (2.5)

where u and v are two vertices of the graph, w(u, v) denotes the weight of the edge that connects u and v, and  $\sim$  represents adjacency. Each channel in the demosaiced image is estimated by solving the following optimization problem that enforces the smoothness on the channel:

$$\hat{\mathbf{c}}_i = \underset{\mathbf{c}_i}{\operatorname{argmin}} \mathbf{c}_i^T \mathbf{L} \mathbf{c}, \quad i \in \{R, G, B\},$$
(2.6)

where  $\mathbf{c}_i$  is obtained by stacking the columns of the corresponding color channel into a vector.

As discussed in this section, color demosaicing algorithms usually benefit from the strong correlation between color channels. This points to one of the main challenges in color and NIR joint acquisition, compared with color imaging. We will demonstrate in Chapter 3 that the correlation between color and NIR channels is not as strong as color correlation. Hence, color demosaicing algorithms would not perform well in color and NIR acquisition. Moreover, many of these methods assume that the Bayer CFA samples the scene that cannot be directly used in joint acquisition, thus limiting the applicability of color demosaicing approaches in our problem.

### 2.2 Color Filter Array Design

As mentioned before, the most commonly used CFA in color imaging is the Bayer CFA shown in Figure 2.2-(a). In the following we review other CFAs designed and studied by camera manufacturers and the research community.

The CFA used in the Kodak DSC620x camera is formed by subtractive primary filters, cyan, magenta, and yellow (CMY) [100] (Fig. 2.2-(b)). Some cameras such as Nikon Coolpix 990 use a CMYG CFA<sup>1</sup> to achieve an acceptable compromise between light efficiency and color acuity (Fig. 2.2-(c)). Compton and Hamilton propose to add to the CFA a white filter that transmits the whole visible spectrum [30]. This increases the CFA light efficiency. This idea is implemented in some of Kodak CFAs (Fig. 2.2-(d)).

Hirakawa and Wolfe in [63] study different designs for the CFA by representing the CFA image in the Fourier domain as explained in Subsection 2.1.2. They show that using subtractive filters (such as cyan and magenta) leads to less artifacts compared with sampling the scene with primary color filters (Fig. 2.2-(e)).

Lu and Vetterli propose a CFA that samples a linear combination of different color channels at each pixel [93]. The optimum coefficients for each combination

 $<sup>^1{\</sup>rm The}$  CMYG CFA uses cyan, magenta, yellow, and green filters.

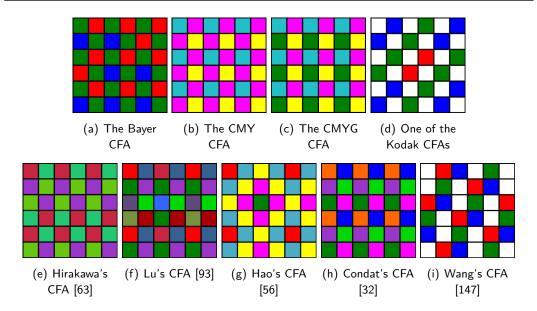


Figure 2.2: Different color filter arrays proposed for color imaging.

are computed by solving an optimization problem that minimizes the demosaicing error. Figure 2.2-(f) shows a CFA optimized by Lu and Vetterli's algorithm.

Hao et al. formulate the frequency structure of a general CFA, which determines the frequency arrangement of the CFA image [56]. Using this structure, they introduce two rules for selecting a good CFA: (1) the aliasing between luminance and subsampled chrominance channels should be minimized (see central and corner lobes in Figure 2.1-c), (2) the correlation between multiplexed chrominance channels (the high-frequency lobes in Figure 2.1-c) should be maximized. The first condition ensures that multiplexed components can be effectively separated, and the second one offers the opportunity of using one chrominance channel to guide the reconstruction of the other one. Based on these criteria, they select a few good CFAs and find the one that results in the highest accuracy when reconstructing benchmark images (Fig. 2.2-(g)).

Following the same idea, Condat analyzes the CFA in the frequency domain [32]. He uses the criteria similar to those suggested in [63] and [56] for designing the CFA. Moreover, to increase robustness to noise, he maximizes the transmittance of each filter in the CFA. Figure 2.2-(h) shows the CFA proposed by Condat.

In [147], Wang et al. study the trade-off between the light efficiency of the CFA and the demosaicing accuracy. They propose to include several white pixels in the CFA and arrange the color filters such that the aliasing between luminance and chrominance channels is minimized. These criteria result in a CFA that consists of 40% white filters, 20% red, 20% green, and 20% blue filters (Fig. 2.2-(i)), which is shown to perform better than the Kodak RGBW CFA.

### 2.3 Single-Sensor Multispectral Acquisition

Primarily designed for fields such as remote sensing, agriculture, product quality assessment, and security and surveillance, most multispectral acquisition systems are only suitable for professional and industrial use. These systems either use a filter wheel or a beam splitter with several detectors to capture different spectral channels of the scene [83]. See Figure 2.3 for an illustration. By using a filter wheel [46] or tunable filters such as liquid crystal filters [48], each spectral channel is recorded in one shot while the corresponding filter is placed in front of the sensor. In a system that uses a beam splitter, the incoming light is divided into different spectral channels, each measured by a different detector. These multispectral acquisition systems are usually too expensive, bulky, and very time consuming.

Some studies, however, address the simultaneous acquisition of multiple spectral channels on a single sensor. The array of filters that samples the scene is, by some researchers, called the multispectral filter array (MSFA). In some single-sensor multispectral acquisition studies this array is, similar to color imaging, called the color filter array (CFA). In the following, when explaining each technique we use the term used by its developers.

Miao et al. in [102] propose an algorithm that designs the arrangement of different spectral filters such that both spatial and spectral consistencies are guaranteed. The algorithm uses the probability of appearance (POA) for each spectral channel. The POA is a sampling rate of each band; for instance, in the Bayer pattern, the POAs for red, green and blue channels are, respectively, 1/4, 1/2 and 1/4. Figure 2.4 shows the process of designing MSFA for the case of five spectral bands denoted by  $\{R, G, B, C, M\}$  with POAs of  $\{1/4, 1/4, 1/4, 1/8, 1/8\}$ , respectively. A binary tree is used to determine the arrangement of filters. The POAs corresponding to nodes at each level are half the POAs of their parents. Constructing the tree continues until the leaf with the smallest POA in the set (1/8 in this example) appears. To assign the location of each spectral band in the MSFA, the scheme shown in Figure 2.4-(b), which corresponds to the binary tree, is generated. Starting from the root, in each node the pattern is decomposed to two complementary patterns that are the downsampled versions of the upper-level arrangement by a factor of 2. For instance, in Fig. 2.4-(b), R and B patterns are generated by downsampling the pattern marked by 2s. The decomposition process continues until the corresponding pattern for each leaf is generated. Combining patterns for all leaves, the MSFA is obtained as shown in Figure 2.4-(c).

The demosaicing algorithm proposed in [102] is customized to the designed MSFA and consists of three main steps: band selection, pixel selection, and interpolation. As the bands with highest POAs (highest sampling rates) are least aliased, the best approach is to first demosaic these bands (similar to green demosaicing in the color imaging). Full resolution images of these bands can be used in reconstructing other channels. Therefore, demosaicing starts with the band corresponding to the upper-

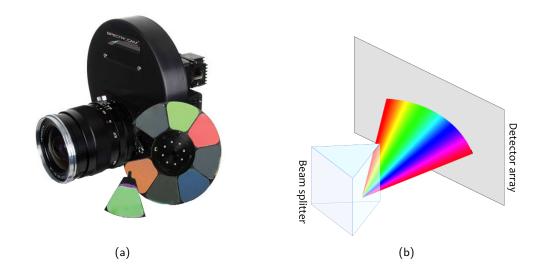


Figure 2.3: (a) A multispectral camera that uses a filter wheel [3]. (b) A schematic of a multispectral acquisition system that uses a beam splitter and several detectors.

most leaf in the binary tree. In the example shown in Figure 2.4, the highest POA corresponds to channels R, G, and B, and the demosaicing process might start with any of these channels.

One of the difficulties in demosaicing multispectral images is that in each spectral band very few known pixels are available, which are insufficient for approximating missing pixels with acceptable accuracy. To address this problem, Miao at al. suggest to sequentially approximate the missing pixels for each band. In each iteration of the proposed algorithm, some missing pixels are estimated using the MSFA samples and the pixels interpolated in previous iterations at the same channel. For the example of Figure 2.4, to reconstruct the full resolution M channel, first the Mmissing values at C pixels are estimated. In the next step, missing pixels at positions of the M channel parent's sibling (G channel in this case) are interpolated. The algorithm then moves one level up and considers the sibling of node 1 (node 2). As node 2 is an internal node, M values at the pixels of this node's children (i.e., R and B bands) are computed. If the algorithm continues one more iteration, it reaches the root of the tree, meaning the M channel is completely demosaiced. At each iteration of demosaicing a channel, missing pixels are estimated as a weighted average of the four closest neighbors. Similar to most color demosaicing techniques, weights are inversely proportional to the magnitude of gradient in the corresponding direction.

Miao et al. mention that the edge locations of different channels in a multispectral image are the same, and the edge information of high-resolution bands with the highest sampling rate can be used to refine the estimations of channels with lower sampling frequencies. However, they do not use this information in the proposed

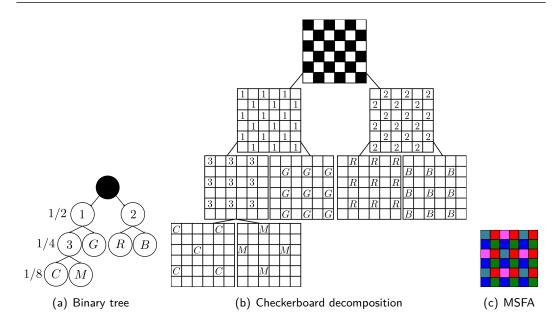


Figure 2.4: (a) The binary tree used to design the MSFA for acquisition of five spectral bands. (b) The corresponding checkerboard decomposition. (c) The multispectral filter array (MSFA) is obtained by superposing the patterns in leaves of the checkerboard decomposition [102].

algorithm.

Brauers and Aach propose using a  $3 \times 2$  CFA to record six spectral channels in the visible range [19]. To reconstruct the full-resolution image, they first apply bilinear interpolation to each spectral channel. Afterwards, assuming a strong correlation between different channels, an improved estimate of the image is obtained by interpolating the channel-difference signals. This algorithm is similar to color demosaicing algorithms that rely on the constant-hue assumption.

Baone and Qi in [12] formulate the task of demosaicing a multispectral image sampled with any MSFA as a MAP estimation, similar to (2.4). They form the regularization term of (2.4) assuming that multispectral images follow a multivariate Gaussian distribution. The parameters of this distribution are computed on an initial estimate of the full-resolution multispectral image obtained by a simple bilinear interpolation.

Hershey and Zhang propose a  $2 \times 2$  CFA, where blue, green, red, and invisible filters sample the scene [61]. The invisible filter could be an ultraviolet or an NIR filter. To estimate the missing channels at each pixel, they propose applying a bilinear interpolation to the samples in the corresponding spectral channel. In [142], Tang et al. study a similar CFA for color and NIR acquisition. They formulate the task of reconstructing full-resolution images as a MAP framework. They propose using quadratic regularizers such as a Gaussian prior for image gradients to constrain

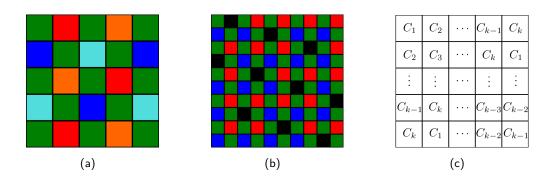


Figure 2.5: (a) The MSFA presented in [107] for the acquisition of five spectral channels in the visible band. (b) The hybrid CFA proposed in [75] for capturing color channels and one additional band (colored black). (c) The MSFA proposed in [4] for sampling k spectral channels denoted as  $\{C_1, C_2, \dots, C_k\}$ .

the reconstruction problem.

Lu et al. design a generic CFA for the joint acquisition of color and NIR images [91]. Each filter in this CFA transmits NIR and a weighted average of color channels. The optimum weights for each filter in the CFA are calculated by solving a spatial optimization problem that minimizes the error of reconstructing full-resolution color and NIR images. By solving this optimization, Lu et al. also obtain an optimum linear demosaicing, formulated as a demosaicing matrix. This algorithm is explained in more detail in Chapter 3, where we propose its extension by taking into consideration the spatial and spectral correlation characteristics of color and NIR images.

Monno et al. present a five-band MSFA that samples the visible part of the electromagnetic spectrum [107]. Their proposed MSFA is shown in Figure 2.5-(a). As the green channel is sampled with the highest rate in the MSFA, this channel is first interpolated. The full-resolution green channel is then used as a guide image in the guided image filtering algorithm [57] to reconstruct the other four channels.

Kiku et al. [75] use a  $10 \times 10$  hybrid CFA, shown in Figure 2.5-(b), to capture three color channels and one additional band. To demosaic the image, they assume no correlation between color channels and the additional band, hence they reconstruct this band separately by applying the super-resolution algorithm of [98]. To demosaic the color channel, the full-resolution green channel is estimated. The interpolated green channel is then used to guide the demosaicing of red and blue channels. This step is performed by the guided image filtering proposed by He et al. in [57].

Aggarwal and Majumdar address the single sensor acquisition of multispectral images in [4]. They propose a uniform MSFA in which spectral channels are sampled on diagonal stripes. Figure 2.5-(c) illustrates this MSFA for sampling k spectral channels denoted as  $\{C_1, C_2, \dots, C_k\}$ . The missing intensities at each pixel are esti-

mated as the weighted average of observed samples in a  $3 \times 3$  neighborhood around the corresponding pixel. The weights for each specific MSFA are trained by minimizing the error of reconstructing full-resolution images over a set of multispectral images.

Wang et al. address the demosaicing of multispectral visible images where k different channels are uniformly sampled by a  $k \times k$  MSFA [146]. They train a demosaicing matrix over a set of multispectral images to perform an initial linear demosaicing. In the next step, they compute the residual error in each channel by subtracting the initial estimate from the MSFA samples in that band. The residual channels are interpolated and added to the initial estimate to obtain the final demosaicing result. This algorithm is an extension of Monno's color demosaicing algorithm presented in [74].

Lapray et al. design a multispectral camera that captures seven spectral channels in the visible band and one NIR band [83]. Their multispectral filters are built by SILIOS Technologies using the COLOR SHADES technology. According to Lapray et al. this technology combines the thin-film deposition and micro-/nano-etching processes. Figure 2.6 shows the spectral transmittance of these filters. They propose two different MSFA arrangements for these eight filters, shown in Figure 2.7. The MSFA of Figure 2.7-(a) has the highest sampling frequency for channel P5 (green wavelengths) and NIR channel, and is recommended for applications that require high spatial resolution in both visible and NIR bands. The second MSFA in this figure is designed by the binary tree algorithm of Miao et al. [102] assuming that the sampling frequency for all channels is 1/8. In [83], the authors focus mostly on transmittance and sensitivity properties of the designed filters and sensor. To reconstruct full-resolution images, they explore bilinear interpolation and also the channel-differences interpolation.

In Chapter 4, we compare the performance of some of the aforementioned algorithms that are applicable to the joint acquisition of color and NIR. Table 2.2 summarizes the single-sensor multispectral acquisition techniques discussed in this section.

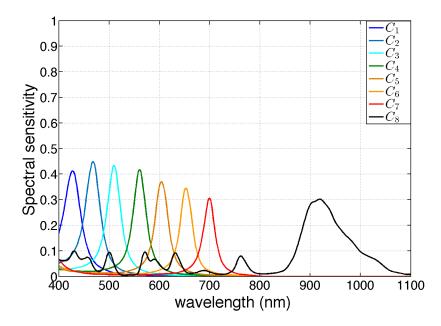


Figure 2.6: The relative response of the multispectral camera developed by Lapray et al. [83]. The camera captures one NIR  $(C_8)$  and seven visible channels. The plot is produced based on the data provided by the authors of [83].

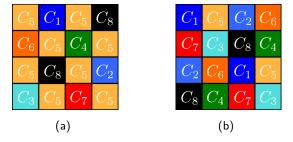


Figure 2.7: The two MSFA filter arrangements used by Lapray *et al.* [83]. Spectral channels  $C_1$  to  $C_7$  represent the scene in the visible band and  $C_8$  is the NIR channel.

Reference	Sampling approach	Demosaicing
Miao <i>et al.</i> [102]	A multispectral filter array (MSFA) that	Edge-adaptive interpolation of spectral
	respects spatial and spectral consistencies	channels.
Brauers and Aach [19]	$\stackrel{\circ}{A}$ 3 × 2 CFA to sample 6 visible bands.	Bilinear interpolation of channel differ-
,		ences.
Baone and Qi [12]	Any general MSFA.	MAP estimation assuming a multivariate
		Gaussian distribution for images.
Hershey and Zhang [61]	A $2 \times 2$ CFA to sample color channels and	Bilinear interpolation.
	one invisible channel.	
Lu et al. [91]	An optimized CFA with filters that trans-	An optimized linear demosaicing.
	mit different mixtures of color and NIR	
	channels.	
Monno <i>et al.</i> [107]	A $5 \times 5$ CFA to sample five channels in	Guided filtering using the green full-
	the visible band (Fig. 2.5-a).	resolution image as the guide.
Kiku <i>et al.</i> [75]	A $10 \times 10$ hybrid CFA to capture three	Guided filtering is applied to demosaic
	color channels and one additional band	the color image and a super-resolution al-
	(ex. NIR) (Fig. 2.5-b).	gorithm is used to reconstruct the addi-
		tional channel.
Aggarwal and Majumdar [4]	A general diagonal MSFA (Fig. 2.5-c).	Optimized linear demosaicing.
Wang $et \ al. \ [146]$	Any general MSFA.	Optimized linear demosaicing.
Lapray et al. [83]	A $4\times4$ CFA to sample one NIR and seven	Bilinear interpolation.
	visible channels (Fig. 2.7).	

Table 2.2: The summary of the single-sensor multispectral acquisition techniques.

#### 2.4 Chromatic Aberration

We review the studies that address the longitudinal chromatic aberration in color imaging in Subsection 2.4.1. Furthermore, as we formulate the task of reducing chromatic aberration in color and NIR acquisition as deblurring the NIR image, our work is closely related to the field of deblurring. We review the main algorithms developed in this field in Subsection 2.4.2.

### 2.4.1 Chromatic Aberration in Color Imaging

Chromatic aberration in color imaging is usually corrected assuming that the edges of color channels are co-located when the image is not distorted by CA. The green channel is usually considered to be in focus on the sensor plane, whereas other channels are out of focus and blurred. Based on these assumptions, some studies [144, 145, 79] propose to simply replace the high-frequency components of the blurred channels with those of the sharp color channel. For instance, the image of the red channel is reconstructed as follows:

$$\hat{\mathbf{R}} = h_{\mathrm{LP}}(\mathbf{R}) + h_{\mathrm{HP}}(\mathbf{G}),\tag{2.7}$$

where  $h_{LP}(.)$  and  $h_{HP}(.)$ , respectively, apply low-pass and high-pass filters to the image. **R** and **G** are observed red and green images, and  $\hat{\mathbf{R}}$  is the reconstructed sharp image of the red channel.

In [72], Kang considers the green channel to be the reference image and computes the blur kernel of red and blue channels, with respect to the green image. He assumes that the edges of blue and red channels are located in a  $5 \times 5$  neighborhood around the location of the corresponding edge in the green channel and uses this information to retrieve the lost details of these channels.

Several algorithms correct for chromatic aberration by removing the purple color fringes caused partly by chromatic aberration [71, 76, 21, 67, 68]. Figure 2.8 shows an example of purple fringes. Most of these approaches search for pixels corrupted by purple fringes around the edges. Those pixels with high intensities in three color channels (bright regions of the image), and higher blue and red intensities compared with their green value are detected as purple fringes. To correct for this effect, the color intensities of these pixels are replaced by an average of the color values for the same pixel. It is also possible to replace the color of the corrupted pixel by an average of intensities in its local neighborhood.

In [24], Chang et al. argue that although the most common, the purple color fringes are not the only false colors caused by chromatic aberration. Thus, their algorithm detects and corrects any type of color fringes in the image. They apply a transition-improvement operator to chrominance channels in order to reduce the blurring effects of longitudinal CA. Such an operator sharpens the image and bounds the values around the edge to maximum and minimum intensities in a close neighborhood to avoid halo artifacts caused by sharpening. In the next step, the color



Figure 2.8: Purple fringes, partly caused by chromatic aberration, are mainly visible around bright edges in the image.

fringes are corrected for by replacing the intensities on the edges with an average of intensities in a small neighborhood around the corresponding pixel.

In [29], Chung et al. assume that if the image does not suffer from chromatic aberration, the transition area for an edge is co-located in all color channels. Based on this, they define the color difference condition for normal edges. According to this condition, the color difference in a transition region either monotonically increases or decreases, hence the color difference is bounded by the minimum and maximum of color difference on both sides of the transition region. Mathematically speaking, if  $\mathbf{D}_R = \mathbf{R} - \mathbf{G}$ , then around a vertical edge

$$\min \{ \mathbf{D}_R(l), \mathbf{D}_R(r) \} \le \mathbf{D}_R(j) \le \max \{ \mathbf{D}_R(l), \mathbf{D}_R(r) \}, \tag{2.8}$$

where l and r are the leftmost and rightmost locations in the transition area, respectively, and j is every pixel in between. Such a transition region is shown in Figure 2.9. The above condition holds for the difference between blue and green channels as well. The algorithm of [29] detects edges that are distorted by chromatic aberration as those that do not respect the above condition. To remove distortions in these regions, Chung  $et\ al.$  propose replacing the color difference by the color difference of the closest neighborhood that does not violate the normal edge color difference condition. The final red and blue channels are computed by adding the green intensities to the corrected color differences.

Kang et al. correct CA distortions in a luminance-chrominance color space such as YCbCr [70]. They first detect the edges and a neighborhood around each edge in the luminance channel. They measure the degree of chromatic aberration distortion as the difference between the gradients of red and green channels (and also between blue and green channels). The color of the distorted pixel is then replaced by a



Figure 2.9: The arrows show the boundaries of the transition region around the edge. The triplet values are the RGB code values of leftmost and rightmost pixels of the transition region.

weighted average of colors in neighboring pixels. The weights are defined as the degree of chromatic aberration distortion in each direction.

Kang et al. in [69] assume that mainly strong edges are affected by chromatic aberration. Hence, they correct for these distortions where a strong edge is detected in the image of the green channel, assuming that this channel is in focus. Distorted regions are reconstructed by minimizing the difference between the gradients of red and green channels. By the same process, the gradients of the blue channel are also matched to those of the green band.

In [139], Singh and Singh formulate the effects of chromatic aberration and CFA sampling as one linear transform. They propose reconstructing the full-resolution aberration-free image by computing the pseudo-inverse of this linear transform. The correlation between color channels is exploited by performing the operations in a luminance-chrominance color space such as YCbCr.

#### 2.4.2 Deblurring

One approach to reducing the effects of longitudinal chromatic aberration is to deblur the spectral channels that are not in focus on the sensor plane. As this technique is closely related to the field of deblurring, in this section, we review the main trends in this field.

The blurring process of an image I is often modeled as follows:

$$\mathbf{I}_b = \mathbf{k} * \mathbf{I} + \mathbf{N},\tag{2.9}$$

where,  $\mathbf{I}_b$  is the observed blurred image.  $\mathbf{k}$  is called the blur kernel or the point spread function (PSF) of the system. '\*' denotes the convolution operator, and  $\mathbf{N}$  is the acquisition noise.

The deblurring algorithms that estimate both  $\mathbf{k}$  and  $\mathbf{I}$ , given only the blurred image, are called blind deblurring. Most of these algorithms formulate the task of estimating the blur kernel and deblurring as the following optimization:

$$\{\hat{\mathbf{I}}, \hat{\mathbf{k}}\} = \underset{\mathbf{I}, \mathbf{k}}{\operatorname{argmin}} \|\mathbf{I}_b - \mathbf{k} * \mathbf{I}\|_F^2 + \lambda_1 \operatorname{Pr}(\mathbf{I}) + \lambda_2 \operatorname{Pr}(\mathbf{k})$$
 (2.10)

In this optimization, the first term is the data fidelity term that enforces the estimated blur kernel and the deblurred image to respect the observed blurry image. As the problem of deblurring is ill-posed, the second and third terms constrain the problem, where Pr stands for prior knowledge, and  $\lambda_1, \lambda_2$  are regularization weights defining the importance of each term. Pr(I) explains the plausible set of sharp images and is usually inspired by natural image characteristics. Pr(k) incorporates the prior information about the blurring process.

The main constraints considered for the blur kernel are that the kernel elements be non-negative, and they must sum up to one to preserve the energy. The norm of the kernel can be minimized through the following Tikhonov regularization [154]:

$$\Pr(\mathbf{k}) = \|\mathbf{k}\|_2^2. \tag{2.11}$$

Moreover, it might be assumed that the kernel changes slowly and hence the norm of the kernel gradients,  $\|\nabla \mathbf{k}\|$ , is minimized.

To regularize the deblurring problem with respect to the sharp underlying image,  $\mathbf{I}$ , usually the distributions for gradients of natural images are used. For instance, it is known that the gradients follow heavy-tailed distributions such as Laplacian or hyper-Laplacian. Figure 2.10 shows the distribution of horizontal and vertical gradients for one example image, where similar to [42], we plot the  $\log_2$  of probability density. A heavy-tailed distribution is modeled by imposing sparsity on gradients of the deblurred image, expressed as  $\Pr(\nabla \mathbf{I}) = \|\mathbf{I}\|_p^p$ , for  $p \in (0,1]$  [81]. The  $\ell_p$  norm for vector  $\mathbf{x}$  is defined as:

$$\|\mathbf{x}\|_p = \sqrt[p]{\sum_i |x_i|^p}.$$
 (2.12)

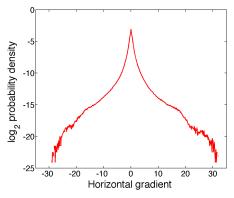
In [82], Krishnan *et al.* argue that assuming sparsity for the gradients favors the blurred image and the no blur solution where  $\mathbf{k} = \delta(m, n)$  (the blur PSF is the delta function). To solve this issue, they propose the following normalized sparsity as the regularization term:

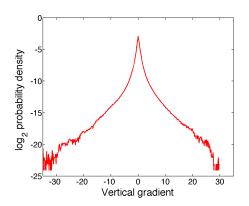
$$\Pr(\mathbf{I}) = \frac{\|\nabla \mathbf{I}\|_1}{\|\nabla \mathbf{I}\|_2}.$$
 (2.13)

Most deblurring algorithms use only one blurred image to estimate the sharp representation of the scene [86, 82, 73, 137]. However, there are few studies that use a guide image in deblurring. He et al. [58] propose the guided filtering for the general problem of image restoration, including image deblurring. The algorithm of Yuan et al. [154] uses a noisy color image to guide the deblurring of a blurred color image of the same scene. Similarly, Zhang et al. [156] propose using multiple



(a) Input image





- (b) Probability density of horizontal gradients
- (c) Probability density of vertical gradients

Figure 2.10: The gradients of natural images have heavy-tailed distributions as the ones shown here.

blurred images of the scene to produce a sharp representation. The main similarity between these algorithms is that both guide and target images represent the scene in the visible band, hence their gradients are strongly correlated and can be used to retrieve the lost details in the blurred image.

Many deblurring algorithms are designed for gray-scale images. To deblur color images they either deblur the luminance channel or treat each color channel separately. As opposed to these methods, Schuler et al. in [134] develop an algorithm that jointly deblurs color channels specifically to correct for different lens aberrations. To reduce chromatic aberration distortions, they assume that each color channel is blurred with a different blur kernel. They explore the correlation between color channels by formulating the regularization terms in the YUV color space, where the total variations of U and V channels (containing chromatic information) are more strongly penalized. They calibrate the lens used in their experiments for a specific set of zoom, aperture, and focus parameters, and use the measured blur kernels to deblur the image. Schuler et al. also incorporate demosaicing into their deblurring

framework and show that this approach slightly outperforms deblurring the already demosaiced color image (0.4 dB improvement over synthetic experiments performed on the Kodak dataset).

In [135], Schuler *et al.* extend their previous algorithm to a blind approach where the blur kernels are also estimated. To estimate spatially varying blur kernels caused by chromatic aberration, they make some assumptions about the symmetry of the blur kernel with respect to the image center, and smoothness of spatial variations in blur kernels. These assumptions constrain the ill-posed problem of estimating blur kernels. Once the kernels are estimated, they use a non-uniform version of Krishan and Fergus's algorithm [81] to deblur the image.

Heide et al. in [59] use a simple lens that introduces different optical aberrations, including chromatic aberration. Similar to [134], they first calibrate the lens by estimating the lens blur kernels at different patches across the image and use these estimations to deblur the image. They assume that the edges of color channels are co-located and regularize the deblurring process by the difference of color gradients normalized by the absolute intensity of each channel in the corresponding pixel.

The common assumption of all the chromatic aberration algorithms explained above is that the gradients of color channels are always strongly correlated. This assumption is not valid for color and NIR images. Figure 2.11 shows an example where the high-frequency details of color and NIR images are not correlated. Because of the differences between the gradients of color and NIR, the algorithms proposed for correcting chromatic aberration cannot be directly applied to our problem. In Chapter 5, we illustrate the effect of assuming a strong correlation between color and NIR gradients in reducing chromatic aberration distortions.

### 2.5 Discussion

By capturing a multi-channel image using a single sensor, both CFA sampling and deficiencies of optical lenses result in loss of information, specifically high-frequency details. In color imaging, the techniques developed for estimating the missing information benefit mainly from the high correlation between the high-frequency components of color channels.

The correlation between color and NIR images is not as strong as are the similarities between color channels. This makes the joint acquisition of color and NIR images on a single sensor more challenging, compared with color imaging, and causes difficulties both in demosaicing and in correcting for chromatic aberration. Throughout the thesis, we propose different algorithms to tackle these difficulties.

2.6. Summary 39

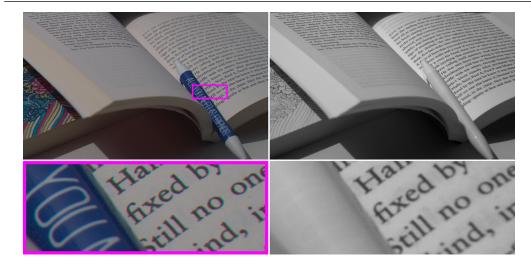


Figure 2.11: A pair of color (left) and NIR (right) images. The color pigments on the pen are transparent in NIR, resulting in differences between the gradients of color and NIR images. Note that the text in the book looks the same in both images, as carbon black used in modern ink absorbs light in both visible and NIR wavelength ranges.

### 2.6 Summary

In this chapter, we have reviewed CFA sampling, chromatic aberration distortions, and the solutions proposed for these problems in single-sensor color and multispectral imaging. Both these tasks are ill-posed, and additional constraints are required to find plausible solutions.

In demosaicing, different regularization terms are used such as constant-hue assumption, sparsity of natural images in a proper transform domain, low-pass nature of chrominance channels, and small total variations in color channels. All these additional constraints exploit two types of correlations in color images: inter-channel and intra-channel correlations. Inter-channel or spatial correlation is the result of strong dependencies between the intensities of neighboring pixels, except at the edges. Intra-channel or spectral correlation models the similarities between different color channels.

To reduce chromatic aberration distortions, the strong correlation between the high-frequency details of color channels is assumed. Hence, the lost high frequencies of blurred channels are retrieved, guided by the sharp details of the in-focus channel.

In the next chapter, we will study the correlation characteristics of color and NIR images, and we present an algorithm that explores these characteristics to design an optimum CFA and demosaicing for the joint acquisition of color and NIR images.

## Chapter 3

# **Optimized CFA**

In this chapter, we first study the correlation characteristics of color and NIR images. The results of our analysis are presented in Section 3.1. We use these results in the design of the optimum CFA and the optimum demosaicing. Our method is based on the work of Lu *et al.* [91] and is presented in Section 3.2. The simulation results comparing the performance of our algorithm with the method of Lu *et al.* [91] are reported in Section 3.3.

### 3.1 Correlation Characteristics of Color and NIR Images

Due to differences in light and scene reflections in visible and NIR bands, there are usually many differences between intensities of color and NIR images. Figure 3.1 illustrates some examples of differences in intensity. Despite these differences, the object boundaries and shapes are usually preserved in both representations. This suggests that the correlation between color and NIR depends on the frequency band. Based on these observations, we hypothesize that the low-frequency components of NIR and color images are not strongly correlated, whereas, the high-frequency details are usually shared between these images. In the following subsections, we take two approaches for analyzing the correlation in natural color and NIR images and testing this hypothesis.

#### 3.1.1 Filter-Bank Analysis of Correlation

In this subsection, inspired by the work of Gunturk *et al.* [52], we analyze the correlation between NIR and color channels in a filter-bank domain. Gunturk *et al.* study the correlation between different color channels in the Kodak dataset.

We first decompose each image to four subbands by using the following horizontal and vertical low-pass  $(h_0^{(h)}, h_0^{(v)})$  and high-pass filters  $(h_1^{(h)}, h_1^{(v)})$ :

$$h_0^{(h)} = [1, 2, 1]/4, \quad h_0^{(v)} = [1, 2, 1]^T/4$$
  
 $h_1^{(h)} = [1, -2, 1]/4, \quad h_1^{(v)} = [1, -2, 1]^T/4,$ 

$$(3.1)$$

42 Chapter 3.



(a) The Y channels of color images



(b) The NIR images

Figure 3.1: This figure illustrates the differences between the absolute intensities of (a) the Y channel (the average of color channels), and (b) the NIR image. Despite these differences, the object boundaries and high-frequency details of both color and NIR representations are generally similar.

By using the above filters, the four subbands of the image  ${\bf I}$  are computed as follows:

$$\mathbf{I}_{LL} = h_0^{(v)} * h_0^{(h)} * \mathbf{I}, \quad \mathbf{I}_{LH} = h_1^{(v)} * h_0^{(h)} * \mathbf{I} 
\mathbf{I}_{HL} = h_0^{(v)} * h_1^{(h)} * \mathbf{I}, \quad \mathbf{I}_{HH} = h_1^{(v)} * h_1^{(h)} * \mathbf{I} 
(3.2)$$

 $\mathbf{I}_{\mathrm{LL}}$  is obtained by applying, in both horizontal and vertical directions, a low-pass filter to the original image.  $\mathbf{I}_{\mathrm{LH}}$  and  $\mathbf{I}_{\mathrm{HL}}$  are the results of low-pass filtering the image in one of the horizontal or vertical directions and high-pass filtering in another. Applying high-pass filters in both directions produces  $\mathbf{I}_{\mathrm{HH}}$ . Figure 3.2 shows one example image and its four subbands.

We compute the correlation between the corresponding subbands of two images  $I_1$  and  $I_2$ , following the normalized cross correlation (NCC) definition:

$$c_{ij} = \frac{\sum_{m,n} (\mathbf{I}_{1ij}(m,n) - \mu_1) (\mathbf{I}_{2ij}(m,n) - \mu_2)}{\sqrt{\sum_{m,n} (\mathbf{I}_{1ij}(m,n) - \mu_1)^2} \sqrt{\sum_{m,n} (\mathbf{I}_{2ij}(m,n) - \mu_2)^2}} \quad \text{for } ij \in \{\text{LL}, \text{LH}, \text{HL}, \text{HH}\}.$$
(3.3)

In the above formulation,  $\mu_1$  and  $\mu_2$  are the average values of  $\mathbf{I}_{1ij}$  and  $\mathbf{I}_{2ij}$  intensities. We first study the correlation between NIR and the luminance channel of the color image. The luminance channel, denoted as  $\mathbf{Y}$ , contains spatial information about the scene in the visible range and we compute it as the average of three color

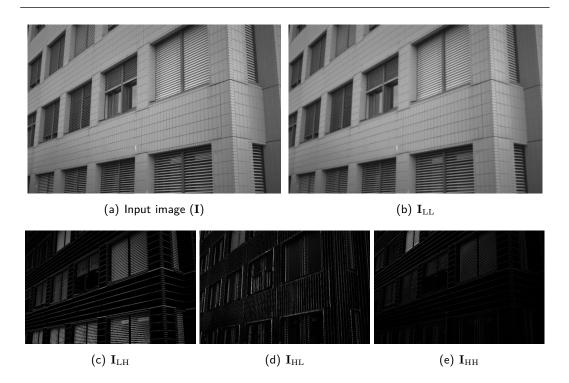


Figure 3.2: An example image and its four subbands. All images have the same resolution in the filter-bank decomposition, and are re-sampled here for illustration purposes.

channels. From now on, we denote the NIR image as  $\mathcal{N}$ . In Table 3.1, we report the correlation values for four subbands of 30 image pairs. This table also shows the correlation values between the red-green and blue-green channel pairs of corresponding images. We show the histograms of correlation values between the subbands of these channels in Figure 3.3. The correlation values between the low-frequency components of NIR and  $\mathbf{Y}$  channels are much smaller than the corresponding values between color channels. The lack of correlation between low-frequencies of NIR and  $\mathbf{Y}$  is explained by the significant differences between the absolute intensities of these channels. Whereas, the high-frequency subbands (LH, HL, HH) exhibit strong correlation. The correlation is not yet as strong as the correlation between color channels for many images. This difference is partly due to the fact that some details in the scene do not appear the same in color and NIR representations, although they usually look very similar in all color channels. For instance, see the example of Figure 3.4 where the pattern on the wall is visible in all color channels, but it disappears in the NIR image.

		$\mathcal{N}$ aı	$\operatorname{ad}\mathbf{Y}$			G a	$\operatorname{nd} \mathbf{R}$			<b>G</b> a	$\operatorname{nd} \mathbf{B}$	
No.	$_{ m LL}$	LH	$_{\mathrm{HL}}$	НН	LL	LH	$_{ m HL}$	НН	LL	LH	$_{ m HL}$	HH
1	0.005	0.53	0.52	0.18	0.95	0.82	0.87	0.75	0.89	0.88	0.9	0.8
2	0.78	0.85	0.8	0.65	0.98	0.92	0.89	0.86	0.99	0.95	0.95	0.91
3	0.41	0.59	0.55	0.44	0.98	0.9	0.87	0.78	0.97	0.93	0.92	0.87
4	0.25	0.69	0.65	0.63	0.94	0.93	0.89	0.86	0.92	0.94	0.91	0.89
5	0.55	0.76	0.84	0.5	0.8	0.84	0.89	0.71	0.92	0.89	0.95	0.78
6	0.4	0.47	0.53	0.28	0.76	0.8	0.77	0.64	0.9	0.86	0.9	0.77
7	0.44	0.72	0.74	0.69	0.64	0.72	0.68	0.62	0.96	0.92	0.94	0.83
8	0.94	0.94	0.95	0.79	0.97	0.85	0.89	0.57	0.99	0.94	0.97	0.75
9	0.57	0.75	0.75	0.6	0.96	0.89	0.84	0.78	0.95	0.94	0.94	0.91
10	0.46	0.75	0.8	0.62	0.97	0.9	0.92	0.77	0.96	0.96	0.94	0.92
11	0.38	0.66	0.58	0.57	0.96	0.83	0.82	0.75	0.94	0.87	0.87	0.82
12	0.36	0.58	0.56	0.57	0.97	0.87	0.87	0.8	0.98	0.92	0.9	0.87
13	0.6	0.67	0.62	0.53	0.78	0.83	0.8	0.68	0.84	0.89	0.84	0.77
14	0.057	0.54	0.5	0.41	0.97	0.91	0.9	0.85	0.95	0.89	0.89	0.84
15	0.079	0.62	0.56	0.5	0.86	0.85	0.84	0.77	0.71	0.82	0.82	0.79
16	0.82	0.86	0.79	0.43	0.96	0.92	0.82	0.34	0.98	0.9	0.76	0.27
17	0.79	0.76	0.75	0.72	0.92	0.79	0.78	0.69	0.87	0.7	0.73	0.56
18	0.88	0.95	0.96	0.9	0.95	0.95	0.92	0.85	0.81	0.96	0.97	0.92
19	0.83	0.94	0.94	0.89	0.92	0.92	0.89	0.85	0.92	0.95	0.94	0.91
20	0.92	0.95	0.92	0.85	0.93	0.89	0.81	0.59	0.9	0.95	0.93	0.84
21	0.033	0.58	0.53	0.55	0.91	0.83	0.82	0.75	0.87	0.83	0.84	0.78
22	0.88	0.96	0.94	0.93	0.96	0.96	0.91	0.88	0.97	0.97	0.95	0.93
23	0.48	0.63	0.57	0.53	0.96	0.89	0.79	0.74	0.94	0.92	0.9	0.88
24	0.096	0.73	0.64	0.63	0.93	0.91	0.86	0.82	0.88	0.82	0.77	0.77
25	0.42	0.77	0.64	0.65	0.9	0.9	0.81	0.76	0.92	0.9	0.87	0.8
26	0.68	0.65	0.88	0.28	0.86	0.78	0.84	0.52	0.96	0.83	0.86	0.59
27	0.69	0.94	0.94	0.89	0.98	0.93	0.91	0.83	0.98	0.95	0.96	0.87
28	0.83	0.84	0.82	0.88	0.98	0.95	0.94	0.9	0.98	0.96	0.95	0.93
29	0.76	0.53	0.51	0.032	0.66	0.41	0.33	0.061	0.9	0.32	0.39	0.11
30	0.58	0.21	0.2	0	0.21	0.2	0.21	0.11	0.97	0.53	0.53	0.087
average	0.53	0.71	0.7	0.57	0.88	0.84	0.81	0.7	0.92	0.87	0.87	0.76

Table 3.1: Correlation values between four subbands for 30 pairs of images. The correlation between the low frequencies of  $\mathcal N$  and  $\mathbf Y$  channels is usually small. In most cases, the correlation between high-frequency subbands is stronger.

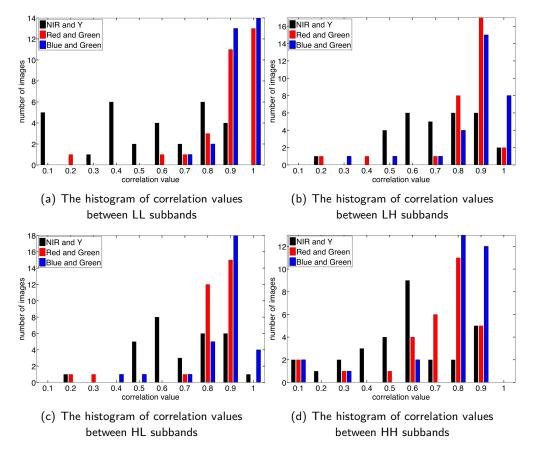


Figure 3.3: The histograms of correlation values for different subbands. The correlation between LL subbands of NIR and Y for majority of images is less than 0.6, whereas the correlation is usually larger than 0.6 for LL subbands of red-green and blue-green pairs. For the high-frequencies subbands, however, in most images the correlation between NIR and Y is higher than 0.5.

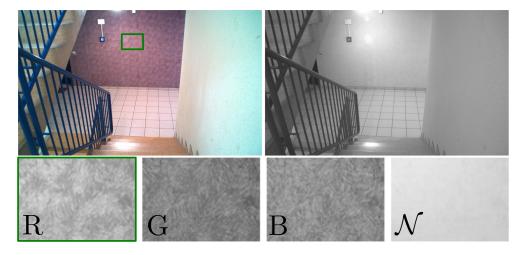


Figure 3.4: Top row: color and NIR images. Bottom row shows a patch on the wall in color channels and NIR. The pattern on the wall is visible in all color channels, while the patch is uniform in the NIR image. These inherent differences result in weaker correlation between color and NIR high frequencies compared with the correlation of color channels.

We compare the correlation between the subbands of NIR and the different color channels in Table 3.2 and Figure 3.5. As before, we observe that the correlation between low-frequency subbands that contain information about the absolute intensities of channels is usually weaker than the correlation between high-frequency details.

Table 3.2 shows that for this dataset, the correlation between the LH and HL subbands of NIR and red (or green) channels is on average higher than the correlation between NIR and blue channels. However, this is not necessarily the case for every image. For images 5, 6, 8, and 29, the correlation values between NIR and blue are higher or comparable with those of green and red channels. Based on these observations, in this thesis, we usually exploit the correlation between the NIR image and the **Y** channel that contains information about all color channels.

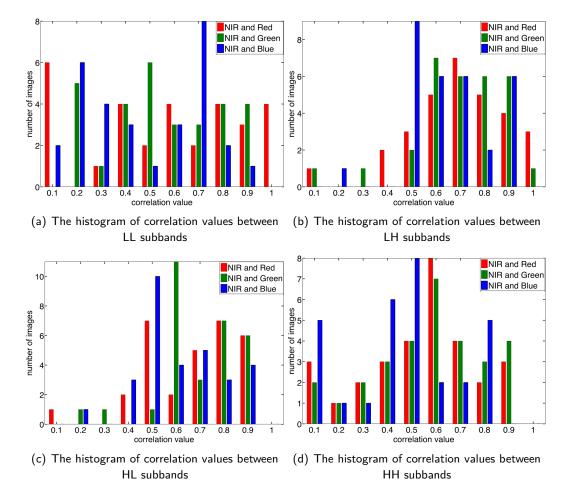


Figure 3.5: The histograms of absolute correlation values for different subbands. The correlation values between LL subbands are usually small, whereas the high-frequency subbands exhibit stronger correlations.

	$\mathcal N$ and $\mathbf R$			$\mathcal N$ and $\mathbf G$			$\mathcal N$ and $\mathbf B$					
No.	LL	LH	HL	НН	LL	LH	HL	НН	LL	LH	HL	НН
1	0.02	0.5	0.49	0.13	0.23	0.55	0.54	0.21	0.19	0.47	0.49	0.13
$\parallel$ 2	0.8	0.84	0.78	0.63	0.8	0.86	0.8	0.66	0.74	0.79	0.74	0.58
3	0.42	0.57	0.53	0.41	0.46	0.61	0.58	0.45	0.33	0.56	0.51	0.41
$\parallel$ 4	0.4	0.75	0.73	0.68	0.17	0.66	0.62	0.59	0.17	0.6	0.53	0.52
5	0.54	0.77	0.83	0.53	0.38	0.69	0.78	0.39	0.66	0.72	0.83	0.45
6	0.13	0.36	0.39	0.23	0.44	0.49	0.56	0.28	0.55	0.48	0.58	0.24
7	0.45	0.67	0.76	0.63	0.31	0.66	0.63	0.63	0.4	0.66	0.65	0.6
8	0.98	0.91	0.92	0.62	0.93	0.9	0.93	0.76	0.88	0.9	0.92	0.66
9	0.58	0.76	0.74	0.61	0.65	0.75	0.75	0.59	0.43	0.69	0.68	0.5
10	0.57	0.73	0.81	0.58	0.49	0.77	0.81	0.63	0.3	0.7	0.72	0.54
11	0.47	0.73	0.61	0.57	0.45	0.68	0.62	0.6	0.19	0.48	0.44	0.41
12	0.37	0.59	0.54	0.53	0.4	0.6	0.6	0.6	0.31	0.5	0.49	0.48
13	0.7	0.72	0.66	0.59	0.52	0.63	0.57	0.47	0.4	0.59	0.56	0.45
14	0.064	0.54	0.5	0.39	0.19	0.59	0.55	0.46	-0.064	0.47	0.45	0.35
15	0.099	0.57	0.5	0.43	0.48	0.67	0.6	0.54	-0.22	0.54	0.51	0.44
16	0.92	0.92	0.82	0.33	0.81	0.85	0.76	0.37	0.68	0.72	0.53	0.13
17	0.77	0.74	0.74	0.68	0.75	0.73	0.71	0.68	0.72	0.61	0.6	0.52
18	0.99	0.95	0.95	0.85	0.9	0.93	0.95	0.88	0.5	0.9	0.9	0.83
19	0.98	0.96	0.94	0.91	0.85	0.91	0.91	0.84	0.59	0.87	0.84	0.77
20	0.99	0.92	0.89	0.66	0.93	0.95	0.9	0.82	0.71	0.89	0.83	0.73
21	-0.11	0.53	0.48	0.49	0.24	0.63	0.58	0.58	-0.19	0.49	0.46	0.43
22	0.91	0.95	0.92	0.9	0.88	0.95	0.94	0.92	0.82	0.91	0.87	0.85
23	0.58	0.61	0.53	0.48	0.48	0.65	0.6	0.54	0.34	0.57	0.49	0.46
24	0.31	0.74	0.61	0.59	0.19	0.78	0.7	0.68	-0.15	0.57	0.49	0.48
25	0.64	0.8	0.67	0.65	0.46	0.78	0.66	0.66	0.14	0.66	0.52	0.51
26	0.78	0.65	0.85	0.26	0.56	0.62	0.85	0.26	0.58	0.51	0.71	0.13
27	0.67	0.94	0.9	0.82	0.69	0.92	0.93	0.86	0.7	0.89	0.91	0.81
28	0.86	0.83	0.82	0.86	0.83	0.84	0.83	0.88	0.76	0.79	0.72	0.8
29	0.81	0.38	0.39	0.004	0.58	0.31	0.31	0.003	0.72	0.47	0.4	0.004
30	0.14	0.098	0.08	0	0.56	0.13	0.17	0.001	0.65	0.17	0.19	0.002
average	0.56	0.7	0.68	0.53	0.55	0.7	0.69	0.56	0.42	0.64	0.62	0.47

Table 3.2: Correlation values between four subbands for 30 pairs of images. The reported values show the correlations between NIR and different color channels.

## 3.1.2 Frequency Analysis of Correlation

In [125], we analyze the correlation characteristics of color and NIR images in different frequency bands by using frequency-selective filters. We use Gaussian filters with different central (peak) frequencies to cover the whole frequency spectrum. Each filter with central linear frequencies  $f_{cx}$  and  $f_{cy}$  in horizontal and vertical directions is formulated in the frequency domain as the following:

$$\hat{h}(f_x, f_y) = \sum_{m=-1, 1} \sum_{n=-1, 1} \exp\left(-\frac{(f_x - mf_{cx})^2 + (f_y - nf_{cy})^2}{\sigma^2}\right), \quad (3.4)$$

where  $f_x$  and  $f_y$  are, respectively, normalized horizontal and vertical frequencies<sup>1</sup> and  $\sigma$  is the standard deviation of the filter that defines the filter bandwidth. Figure 3.6 shows three examples of such filters for central frequencies ( $f_{cx} = f_{cy} = 0$ ), ( $f_{cx} = 0.2, f_{cy} = 0$ ), and ( $f_{cx} = f_{cy} = 0.3$ ).

We change the center frequencies in both directions in the range of [0, 0.3]. The maximum possible frequency is 0.5 in each direction, however, a filter with  $f_{cx} = 0.5$  or  $f_{cy} = 0.5$  mostly extracts high-frequency noise. To avoid the effect of noise, we set the maximum center frequency to 0.3. We call the impulse response of a filter with the  $i^{\text{th}}$  horizontal center frequency and  $j^{\text{th}}$  vertical center frequency  $h_{ij}$ .

To compute the correlation in each frequency band, we first apply the corresponding filter to NIR and  ${\bf Y}$  channels:

$$\mathcal{N}_{ij} = h_{ij} * \mathcal{N}, \quad \mathbf{Y}_{ij} = h_{ij} * \mathbf{Y}. \tag{3.5}$$

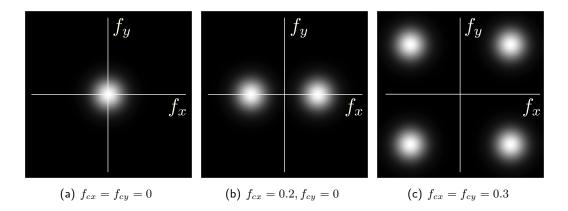


Figure 3.6: Three sample filters used in our correlation analysis. The filters are shown in the frequency domain.

 $<sup>^{1}</sup>$ The frequencies are normalized by the sampling frequency. Here we assume they vary in the range [-0.5, 0.5].

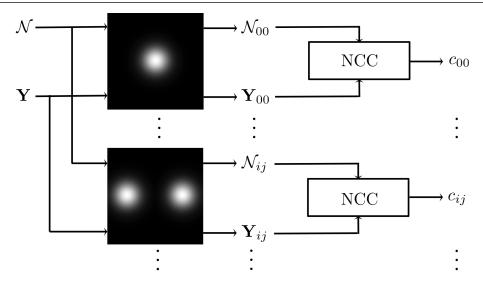


Figure 3.7: The  $\mathcal{N}$  and  $\mathbf{Y}$  channels filtered by the  $ij^{\text{th}}$  frequency-selective filter are called  $\mathcal{N}_{ij}$  and  $\mathbf{Y}_{ij}$ . The NCC block computes the normalized cross correlation between two input images.

The normalized cross correlation in this band is computed as follows:

$$c_{ij} = \frac{\sum_{m,n} (\mathcal{N}_{ij}(m,n) - \mu_{\mathcal{N}}) (\mathbf{Y}_{ij}(m,n) - \mu_{Y})}{\sqrt{\sum_{m,n} (\mathcal{N}_{ij}(m,n) - \mu_{\mathcal{N}})^{2}} \sqrt{\sum_{m,n} (\mathbf{Y}_{ij}(m,n) - \mu_{Y})^{2}}},$$
(3.6)

where m and n are pixel coordinates, and  $\mu_{\mathcal{N}}$ ,  $\mu_{Y}$  are the average intensities of filtered NIR and **Y** channels. Figure 3.7 illustrates the framework of our correlation analysis.

We show a few examples of the correlation surface in the frequency plane in Figure 3.8. The surfaces in the second column of each example are the correlation values plotted versus the horizontal and vertical center frequencies of filters. For instance, the surface value at  $f_x = f_y = 0$  is the correlation between NIR and Y images that are processed by a filter with center frequencies  $f_{cx} = f_{cy} = 0$ , which is a low-pass Gaussian filter. Hence, this value measures the correlation between  $\mathcal{N}$  and Y low-frequency information that represents the absolute intensities.

Similar to the results of filter-bank analysis, for most examples of Figure 3.8, the correlation is minimum at low-frequencies and it increases as the high-frequency details of  $\mathcal{N}$  and  $\mathbf{Y}$  are extracted. There are, however, scenes where the low-frequency components are also highly correlated. For instance, see the correlation surface in Figure 3.8-(d). Nevertheless, in majority of scenes, high-frequency details are strongly correlated.

Figure 3.9 shows the average correlation surface computed for 30 pairs of  $\mathcal{N}$  and  $\mathbf{Y}$  images. We perform this experiment on the same dataset that is used in the filter-bank analysis.

The results of both correlation analyses confirm our hypothesis that color and NIR images are correlated mainly in higher frequency bands. We exploit these results in the next section in order to develop an algorithm for designing the optimum CFA and the optimum demosaicing.

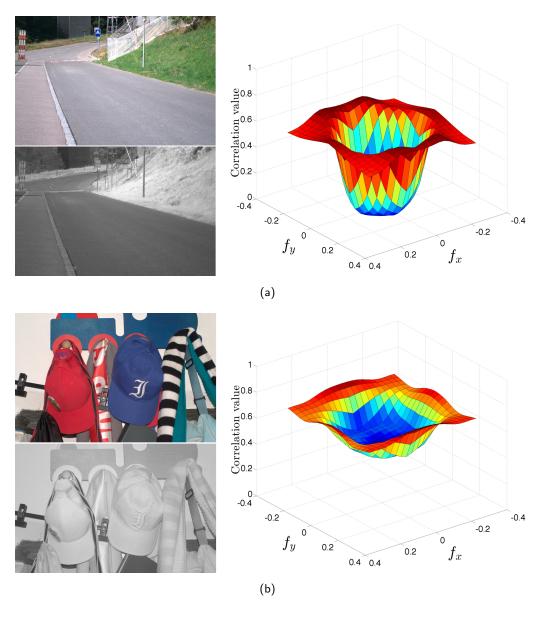


Figure 3.8: The correlation surfaces of  $\mathcal N$  and  $\mathbf Y$  channels.

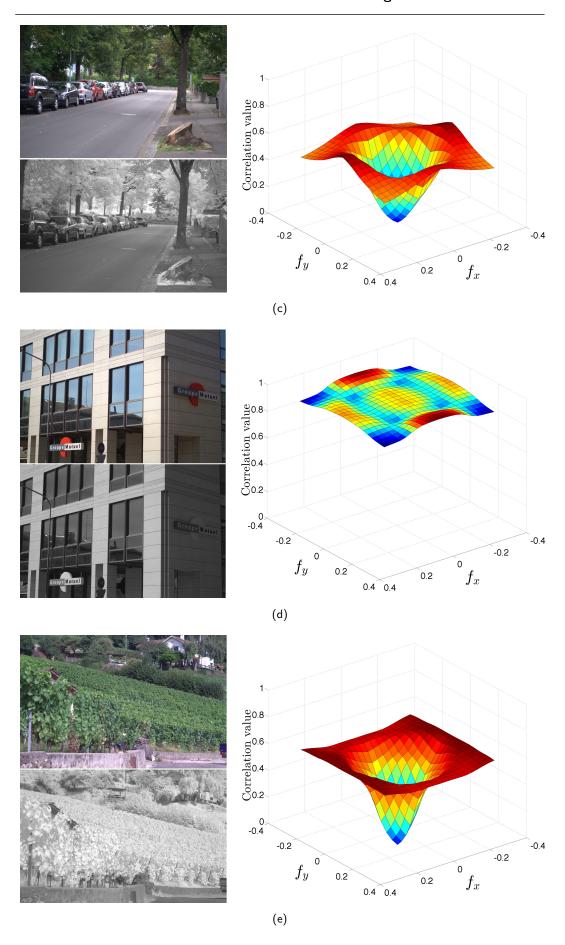


Figure 3.8: The correlation surfaces of  $\mathcal{N}$  and  $\mathbf{Y}$  channels (cont.).

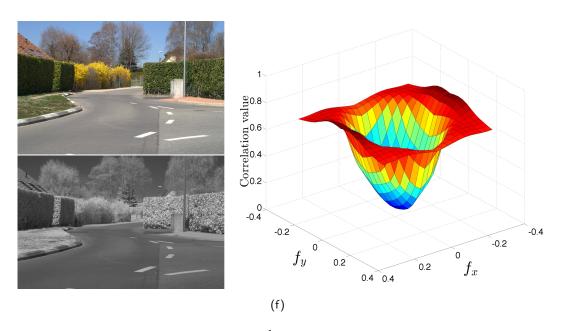


Figure 3.8: The correlation surfaces of  $\mathcal{N}$  and  $\mathbf{Y}$  channels versus horizontal and vertical frequencies. For each example color and NIR images are shown on the left and the correlation surface is on the right (cont.).

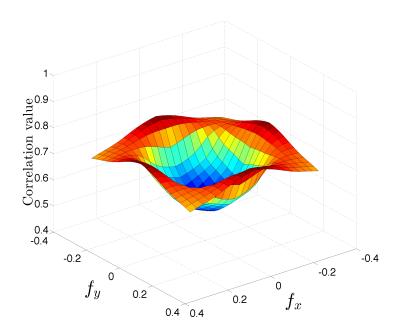


Figure 3.9: The correlation surface averaged for 30 pairs of  $\mathcal{N}$  and  $\mathbf{Y}$  channels. The minimum correlation is 0.5159 and occurs at  $f_x = f_y = 0$ , and the maximum is 0.7721 at  $(f_x = 0.3, f_y = 0)$  and  $(f_y = 0, f_x = 0.3)$ .

# 3.2 The Optimized CFA and Demosaicing

Similar to [91], we assume a general color filter array of size  $M \times M$  is used for sampling the scene. Each filter in the CFA transmits NIR, as well as a mixture of primary colors (red, green, and blue). Figure 3.10 shows an example of such a CFA for M=4. In this figure, the color of each filter represents the mixture of colors the filter transmits, and the coefficients on each pixel are the overall transmittance of the corresponding filter in each band. We denote the overall sensitivity of the sensor and the  $(i,j)^{\text{th}}$  filter in the CFA to red, green, and blue channels,  $\alpha_R^{ij}$ ,  $\alpha_G^{ij}$ ,  $\alpha_B^{ij}$ , respectively. We assume all filters equally transmit NIR, hence  $\alpha_N$  is one for all filters. Thus, in what follows, for simplicity we omit  $\alpha_N$ . With these notations, the value measured by the sensor at  $(i,j)^{\text{th}}$  pixel is

$$z^{ij} = \alpha_R^{ij} \times r + \alpha_G^{ij} \times g + \alpha_B^{ij} \times b + n, \tag{3.7}$$

where r, g, b and n are red, green, blue and NIR intensities at the corresponding pixel.

We consider a patch of size  $(2L+1)M \times (2L+1)M$ . Figure 3.11 shows an example for M=4 and L=1. We form the vector representations of different channels in this patch by stacking their columns:

$$\mathbf{r} = \text{vec}(\mathbf{R}), \mathbf{g} = \text{vec}(\mathbf{G}), \mathbf{b} = \text{vec}(\mathbf{B}), \mathbf{n} = \text{vec}(\mathcal{N}).$$
 (3.8)

We concatenate these vectors into the column vector  $\mathbf{x}$ :

$$\mathbf{x} = [\mathbf{r}^T, \mathbf{g}^T, \mathbf{b}^T, \mathbf{n}^T]^T, \tag{3.9}$$

where T represents the transpose operator.

As shown in (3.7), the intensities of different channels are linearly mapped to the CFA samples. Hence, vector  $\mathbf{x}$  in the full-resolution image is linearly related to the corresponding patch in the CFA image:

$$\mathbf{z} = \mathbf{A}\mathbf{x},\tag{3.10}$$

where z is the vector representation of the patch in the CFA image and A contains the CFA coefficients.

The CFA samples in  $\mathbf{z}$  are used to estimate the intensities of the full-resolution image in the central patch of size  $M \times M$ . This patch is marked as  $\mathbf{x}_0$  in Figure 3.11. By using a linear demosaicing, the full-resolution patch is estimated as

$$\hat{\mathbf{x}}_0 = \mathbf{D}\mathbf{z} = \mathbf{D}\mathbf{A}\mathbf{x}.\tag{3.11}$$

In the above equation, **D** is the demosaicing matrix that maps the CFA samples (**z**) to the intensities of the estimated full-resolution image ( $\hat{\mathbf{x}}_0$ ).

$ \alpha_R : 0.99  \alpha_G : 0.19  \alpha_B : 0.69 $	$\alpha_G:0.57$	$\alpha_G:0.14$	$\alpha_G:0.49$
$ \alpha_R : 0.23  \alpha_G : 0.76  \alpha_B : 0.03 $	$\alpha_G: 0.38$	$\alpha_G:0.49$	$\alpha_G:0.84$
$ \alpha_R : 0.92  \alpha_G : 0.03  \alpha_B : 0.49 $	$\alpha_G: 0.21$	$\alpha_G: 0.01$	$\alpha_G:0.14$
$\alpha_R : 0.64$ $\alpha_G : 0.64$ $\alpha_B : 0.97$	$\alpha_G:0.79$	$\alpha_G: 0.19$	$\alpha_G: 0.73$

Figure 3.10: A general CFA of size  $4\times 4$ . Each filter in this CFA transmits a mixture of primary colors (red, green, blue) and NIR. The  $\alpha$  coefficients are the overall transmittance of each filter in the corresponding band. The filters are assumed to equally transmit NIR, hence  $\alpha_{\mathcal{N}}=1$  for all filters. These coefficients are generated randomly for illustration purposes alone.

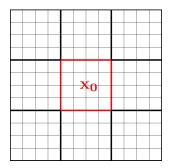


Figure 3.11: Each patch  $\mathbf{x}_0$  of size  $M \times M$  is demosaiced using a neighborhood of size  $(2L+1)M \times (2L+1)M$ . In this example: M=4, and L=1. The size of CFA is  $M \times M$ , and every patch of size  $M \times M$  is sampled by the same periodic pattern.

Lu et al. in [91] propose to find optimum values for the CFA coefficients and the demosaicing matrix by solving the following optimization:

$$\{\mathbf{A}^{\star}, \mathbf{D}^{\star}\} = \underset{\mathbf{A}}{\operatorname{argmin}} E(\|\mathbf{x}_{0} - \hat{\mathbf{x}}_{0}\|_{2}^{2}), \tag{3.12}$$

where E(.) represents the mathematical expectation. This optimization finds the matrices **A** and **D** that minimize the error of reconstructing full-resolution patches. By substituting  $\hat{\mathbf{x}}_0$  from (3.11), we have

$$\{\mathbf{A}^{\star}, \mathbf{D}^{\star}\} = \underset{\mathbf{A}, \mathbf{D}}{\operatorname{argmin}} E(\|\mathbf{S}\mathbf{x} - \mathbf{D}\mathbf{A}\mathbf{x}\|_{2}^{2}). \tag{3.13}$$

**S** is a selection matrix, containing zeros and ones; it extracts the intensities corresponding to the central patch  $(\mathbf{x}_0)$  from  $\mathbf{x}$ :  $\mathbf{x}_0 = \mathbf{S}\mathbf{x}$ .

In the previous section, we show that the high-frequency details of  $\mathcal{N}$  and  $\mathbf{Y}$  are usually strongly correlated. Following this observation, we propose adding the following terms to the cost function of (3.13):

$$\Pr(\mathcal{N}, \mathbf{Y}) = \mathbb{E}(\|h_1 * \hat{\mathcal{N}} - h_1 * \hat{\mathbf{Y}}\|_F^2 + \|h_2 * \hat{\mathcal{N}} - h_2 * \hat{\mathbf{Y}}\|_F^2). \tag{3.14}$$

Here  $\hat{\mathcal{N}}$  and  $\hat{\mathbf{Y}}$  are  $\mathcal{N}$  and  $\mathbf{Y}$  channels of the demosaiced patch.  $h_1$  and  $h_2$  are horizontal and vertical high-pass filters. Minimizing these terms promotes the high-frequency details of reconstructed  $\mathcal{N}$  and  $\mathbf{Y}$  to be strongly correlated.

Denoting the convolution matrices of  $h_1$ ,  $h_2$  filters as  $\mathbf{H}_1$  and  $\mathbf{H}_2$ , we find the optimum CFA and demosaicing by solving:

$$\{\mathbf{A}^{\star}, \mathbf{D}^{\star}\} = \underset{\mathbf{A}, \mathbf{D}}{\operatorname{argmin}} E(\|\mathbf{S}\mathbf{x} - \mathbf{D}\mathbf{A}\mathbf{x}\|_{2}^{2}) + \lambda E(\|\mathbf{H}_{1}\hat{\mathbf{n}} - \mathbf{H}_{1}\hat{\mathbf{y}}\|_{2}^{2} + \|\mathbf{H}_{2}\hat{\mathbf{n}} - \mathbf{H}_{2}\hat{\mathbf{y}}\|_{2}^{2}).$$
(3.15)

**n** and **y** are vector representations of  $\mathcal{N}$  and **Y** channels.  $\hat{\mathbf{y}}$  is computed as the average of red, green, and blue channels. As  $\hat{\mathbf{x}}_0 = [\hat{\mathbf{r}}^T, \hat{\mathbf{g}}^T, \hat{\mathbf{b}}^T, \hat{\mathbf{n}}^T]$ , we write

$$\hat{\mathbf{r}} = \mathbf{S}_r^{M^2 \times 4M^2} \hat{\mathbf{x}}_0 \qquad \hat{\mathbf{g}} = \mathbf{S}_g^{M^2 \times 4M^2} \hat{\mathbf{x}}_0 
\hat{\mathbf{b}} = \mathbf{S}_b^{M^2 \times 4M^2} \hat{\mathbf{x}}_0 \qquad \hat{\mathbf{n}} = \mathbf{S}_N^{M^2 \times 4M^2} \hat{\mathbf{x}}_0$$
(3.16)

Here  $M^2$  is the number of pixels in the estimated patch. We denote an identity matrix of size  $R \times T$  as  $\mathbf{I}^{R \times T}$  and an all-zero matrix of size  $R \times T$  as  $\mathbf{\Theta}^{R \times T}$ . Using these notations, the selection matrices  $\mathbf{S}_r, \mathbf{S}_g, \mathbf{S}_b$  and  $\mathbf{S}_n$  are formed as follows:

$$\mathbf{S}_{r} = [\mathbf{I}^{M^{2} \times M^{2}}, \mathbf{\Theta}^{M^{2} \times 3M^{2}}] \quad \mathbf{S}_{g} = [\mathbf{\Theta}^{M^{2} \times M^{2}}, \mathbf{I}^{M^{2} \times M^{2}}, \mathbf{\Theta}^{M^{2} \times 2M^{2}}]$$

$$\mathbf{S}_{b} = [\mathbf{\Theta}^{M^{2} \times 2M^{2}}, \mathbf{I}^{M^{2} \times M^{2}}, \mathbf{\Theta}^{M^{2} \times M^{2}}] \quad \mathbf{S}_{n} = [\mathbf{\Theta}^{M^{2} \times 3M^{2}}, \mathbf{I}^{M^{2} \times M^{2}}].$$

$$(3.17)$$

Combining (3.11) and (3.16), our optimization problem is written as

$$\{\mathbf{A}^{\star}, \mathbf{D}^{\star}\} = \underset{\mathbf{A}, \mathbf{D}}{\operatorname{argmin}} \operatorname{E}(\|\mathbf{S}\mathbf{x} - \mathbf{D}\mathbf{A}\mathbf{x}\|_{2}^{2})$$

$$+ \lambda \operatorname{E}(\|\mathbf{H}_{1}\mathbf{S}_{n}\mathbf{D}\mathbf{A}\mathbf{x} - \mathbf{H}_{1}\frac{\mathbf{S}_{r} + \mathbf{S}_{g} + \mathbf{S}_{b}}{3}\mathbf{D}\mathbf{A}\mathbf{x}\|_{2}^{2}$$

$$+ \|\mathbf{H}_{2}\mathbf{S}_{n}\mathbf{D}\mathbf{A}\mathbf{x} - \mathbf{H}_{2}\frac{\mathbf{S}_{r} + \mathbf{S}_{g} + \mathbf{S}_{b}}{3}\mathbf{D}\mathbf{A}\mathbf{x}\|_{2}^{2}).$$
(3.18)

Following [91], we decompose matrix  $\mathbf{C} = \mathrm{E}(\mathbf{x}\mathbf{x}^T)$  as  $\mathbf{C} = \mathbf{P}\mathbf{P}$  since  $\mathbf{C}$  is a positive semi-definite matrix. With this decomposition, the above optimization problem is re-written as follows:

$$\{\mathbf{A}^{\star}, \mathbf{D}^{\star}\} = \underset{\mathbf{A}, \mathbf{D}}{\operatorname{argmin}} \|\mathbf{SP} - \mathbf{DAP}\|_{F}^{2}$$

$$+ \lambda(\|\mathbf{H}_{1}\mathbf{S}_{n}\mathbf{DAP} - \mathbf{H}_{1}\frac{\mathbf{S}_{r} + \mathbf{S}_{g} + \mathbf{S}_{b}}{3}\mathbf{DAP}\|_{F}^{2}$$

$$+ \|\mathbf{H}_{2}\mathbf{S}_{n}\mathbf{DAP} - \mathbf{H}_{2}\frac{\mathbf{S}_{r} + \mathbf{S}_{g} + \mathbf{S}_{b}}{3}\mathbf{DAP}\|_{F}^{2}).$$
(3.19)

We simplify the presentation of the above problem by using matrices  $\mathbf{S}_1 \triangleq \mathbf{H}_1 \mathbf{S}_n - \mathbf{H}_1 (\mathbf{S}_r + \mathbf{S}_g + \mathbf{S}_b)/3$  and  $\mathbf{S}_2 \triangleq \mathbf{H}_2 \mathbf{S}_n - \mathbf{H}_2 (\mathbf{S}_r + \mathbf{S}_g + \mathbf{S}_b)/3$ :

$$\{\mathbf{A}^{\star}, \mathbf{D}^{\star}\} = \underset{\mathbf{A}, \mathbf{D}}{\operatorname{argmin}} \|\mathbf{S}\mathbf{P} - \mathbf{D}\mathbf{A}\mathbf{P}\|_F^2 + \lambda(\|\mathbf{S}_1\mathbf{D}\mathbf{A}\mathbf{P}\|_F^2 + \|\mathbf{S}_2\mathbf{D}\mathbf{A}\mathbf{P}\|_F^2).$$
(3.20)

We solve the above optimization problem by iteratively updating  $\mathbf{D}$  and  $\mathbf{A}$ , when the other matrix is fixed. The closed-form solution for  $\mathbf{D}$  is obtained by setting the gradient of the cost function to zero. The solution at iteration k is

$$\mathbf{D}^{(k)} = (\mathbf{I} + \lambda \mathbf{S}_1^T \mathbf{S}_1 + \lambda \mathbf{S}_2^T \mathbf{S}_2))^{\dagger} (\mathbf{SP}) (\mathbf{A}^{(k-1)} \mathbf{P})^{\dagger}$$
(3.21)

Here I is the identity matrix, and  $(.)^{\dagger}$  denotes the pseudo-inverse. Please see Appendix A for the derivation of the above solution.

Formulating matrix  $\mathbf{A}$  as  $\mathbf{A}_0 + \sum_i \alpha_i \mathbf{A}_i$ , we re-write the optimization problem (3.20) as a quadratic problem in terms of  $\alpha$ . Here  $\alpha$  is a vector containing the CFA coefficients that are to be optimized.  $\mathbf{A}_0$  and  $\mathbf{A}_i$ 's are fixed matrices. The quadratic problem has the following form:

$$\alpha^* = \underset{\alpha}{\operatorname{argmin}} \frac{1}{2} \alpha^T \mathbf{Q} \alpha + \mathbf{p}^T \alpha, \quad \text{s.t. } 0 \le \alpha \le 1.$$
 (3.22)

In each iteration  $\mathbf{Q}$  and  $\mathbf{p}$  are fixed with respect to  $\alpha$ . To respect the physical constraints, the transmittance of filters ( $\alpha$ ) is bounded between zero and one. In Appendix A we detail the computation of  $\mathbf{Q}$  and  $\mathbf{p}$ . To solve the above constrained quadratic problem, we use "quadprog" of MATLAB's optimization toolbox. Algorithm 3.1 shows the pseudo-code for solving our optimization problem.

#### Algorithm 3.1 CFA and demosaicing optimization

- 1: Randomly initialize vector  $\alpha$  that contains CFA coefficients. Set k=0 (the iteration number),  $e^0=0$  (the initial error), and  $\delta=0.0001$ .  $\delta$  defines the stop criterion of the iterative optimization.
- 2: **do**
- 3: Update the demosaicing matrix  $(\mathbf{D}^k)$  as (3.21).
- 4: Compute the value of cost function in (3.20) and call it  $e^{k+1}$ .
- 5: Update the CFA coefficients  $(\alpha^{k+1})$  by solving (3.22).
- 6: k = k + 1.
- 7: while  $|e^k e^{k-1}| > \delta$ .
- 8:  $\alpha^* = \alpha^k$  and compute  $\mathbf{D}^*$  by placing  $\alpha^*$  in (3.21).

# 3.3 Experiments

In this section, we compare the performance of the CFA and the demosaicing designed by our algorithm with those computed by the algorithm of Lu *et al.* [91]. Before presenting the results, we review some details about the dataset and implementation of algorithms in the next subsection.

#### 3.3.1 Implementation Details

The mathematical expectation  $E(\mathbf{x}\mathbf{x}^T)$  to compute **C** is replaced by the following ensemble summation:

$$\mathbf{C} = \sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{T}, \tag{3.23}$$

where  $\mathbf{x}_i$ 's are patches extracted from a training set of color and NIR images.

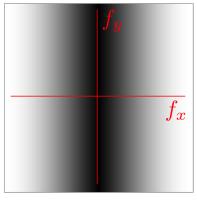
We use the following Sobel filters to extract the high-frequency details of  $\mathcal{N}$  and  $\mathbf{Y}$  channels (see (3.14)):

$$h_1 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \qquad h_2 = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$$
(3.24)

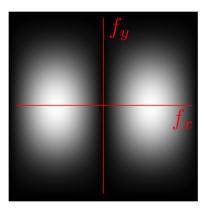
The Sobel filters, as opposed to simple derivative filters [+1, -1], do not amplify noise. Figure 3.12 compares the frequency response of a Sobel filter with that of a simple derivative filter. Note how the Sobel filter suppresses the highest frequencies.

The filtering is applied on small patches with the same size of the CFA that is usually less than 10 pixels on each side. Hence, special care is required to avoid boundary effects. In our optimization, we remove the borders of the filtered patch and only keep the central valid part of the convolution output.

Our optimization problem and the problem of [91] that design CFA and demosaicing are non-convex and the solution depends on the initial point. In the following experiments, we solve each optimization 10 times with 10 different initial points that



(a) The frequency response of the derivative filter (+1,-1).



(b) The frequency response of the Sobel filter,  $h_2$  in (3.24).

Figure 3.12: The Sobel filter does not extract the highest frequencies that might be contaminated with high-frequency noise. On the other hand, the maximum response of the simple derivative filter occurs at the maximum frequency.

are selected randomly. The solution that results in the minimum cost value for the training set is selected to sample and demosaic test images.

Each pair of color and NIR images used in our experiments is captured by a consumer DSLR camera that does not have a hot mirror. The images are captured in two sequential shots when the NIR-blocking and visible-light-blocking filters are placed in front of the lens. Each pair is aligned after acquisition using feature-point matching. All images are subsampled to  $680 \times 1024$  pixels.

After optimizing the CFA and demosaicing matrices, the CFA sampling is simulated for test images following (3.10). The full-resolution images are reconstructed as shown in (3.11).

We compute the following peak-signal to noise ratios (PSNR) for reconstructed color and NIR images:

$$PSNR = 10 \log_{10} \frac{255^{2}}{\frac{1}{P} \sum_{i,j} (\mathcal{N}(i,j) - \hat{\mathcal{N}}(i,j))^{2}}$$

$$CPSNR = 10 \log_{10} \frac{255^{2}}{\frac{1}{3P} \sum_{i,j,k} (\mathbf{I}_{c}(i,j,k) - \hat{\mathbf{I}}_{c}(i,j,k))^{2}}$$
(3.25)

Here CPSNR stands for color PSNR. In the above definitions, matrices with hat ( $\hat{\cdot}$ ) contain intensities of reconstructed images and those without  $\hat{\cdot}$  are ground-truth images.  $\mathbf{I}_c$  is a three-dimensional array containing color channels. P is the number of pixels in each image, and (i,j) indicate the spatial position of each pixel. k indexes different color channels in the image.

In the following experiments, 60 pairs of color and NIR images are used to test the optimized CFA and demosaicing matrices.

#### 3.3.2 Simulation Results

In this subsection, we compare the performance of our algorithm with the method of Lu et al. [91]. Figure 3.13 illustrates the CFAs designed by our algorithm and by the method of [91]. Note that there is a white pixel in our CFA where the coefficients for all the channels is one. There is also one near-white pixel in the CFA designed by Lu's method where coefficients are  $\alpha_R = 1, \alpha_G = 1$  and  $\alpha_B = 0.82$ . As discussed in Chapter 2, the white filters are used in color imaging, for instance in Kodak's RGBW CFA [120]. In [22], Chakrabarti et al. argue that using several white pixels in a CFA results in a high-quality reconstruction of color images. The results of our optimization confirm that having a white filter also benefits the joint acquisition of color and NIR.

We empirically set L=1, which means a neighborhood of size  $(2L+1)M \times (2L+1)M = 12 \times 12$  around each patch of size  $M \times M = 4 \times 4$  is used to demosaic the central patch.

We report the average PSNR values for images sampled and demosaiced by our CFA and demosaicing matrices in Table 4.2. This table shows the results obtained by using the CFA and the demosaicing optimized by the algorithm of Lu *et al.* [91]. On average, the PSNR values obtained by our algorithm, for color and NIR images, are about 1 dB higher than the results of Lu's algorithm [91].

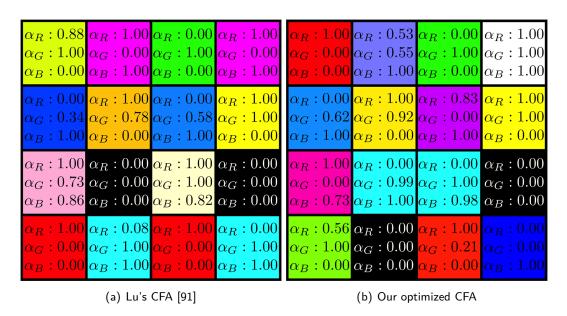


Figure 3.13:  $4 \times 4$  CFAs designed by (a) Lu's algorithm [91] and (b) our method. The color of each filter indicates the mixture of primary colors it transmits and the numbers on each pixel are the coefficients of each mixture. All filters equally transmit NIR ( $\alpha_{\mathcal{N}} = 1$ ).

		Lu's method	Our algorithm	
CPSNR (color)	average	42.39	43.21	
CPSNR (color)	standard deviation	4.46	4.57	
PSNR (NIR)	average	42.29	43.58	
FSIM (MIN)	standard deviation	4.47	4.57	

Table 3.3: The PSNR values of images reconstructed by our algorithm and Lu's [91]. The results are obtained over a dataset of 60 color and NIR pairs. CPSNR and PSNR are computed, respectively, for color and NIR images.

Figure 3.15 shows crops of some ground-truth images<sup>2</sup>, along with the results of both algorithms. The corresponding full-resolution images are shown in Figure 3.14. The visual comparisons confirm the advantage of our algorithm over the method of Lu *et al.* [91].

<sup>&</sup>lt;sup>2</sup>Full-resolution images are illustrated in Figure 3.14.

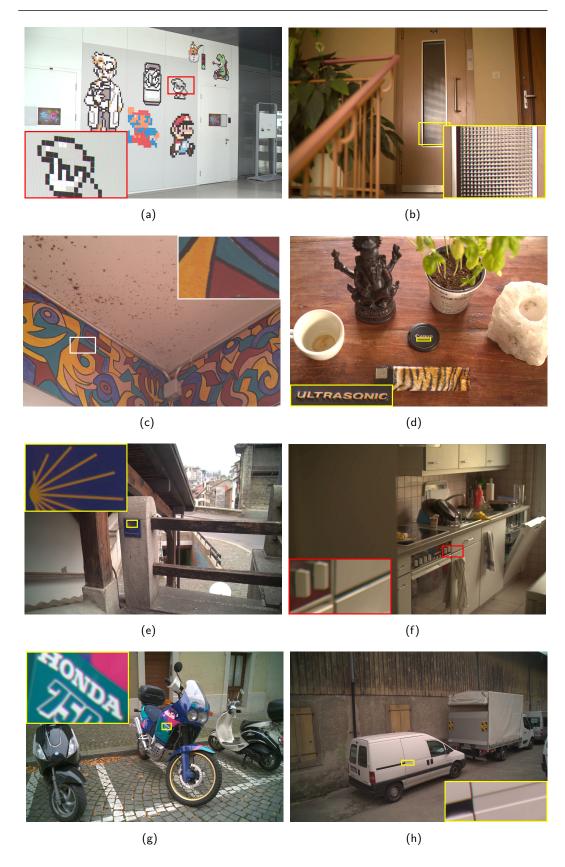
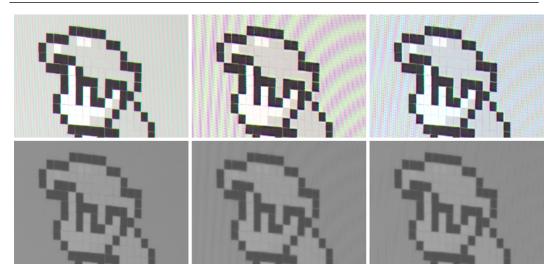
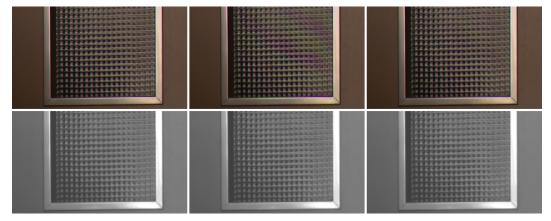


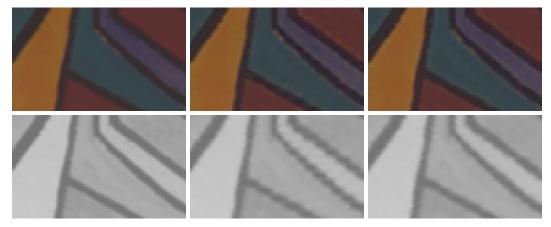
Figure 3.14: Ground-truth color images. In Figure 3.15 we compare the zoomed-in regions of ground-truth images with those of the images reconstructed by Lu's algorithm [91] and our method.



(a) From left to right: Ground-truth, results of [91], and our results. The high-frequency content in the background leads to severe false colors and aliasing in the result of [91]. These details are better preserved in our result.



(b) From left to right: Ground-truth, results of [91], and our results. The false color artifacts are less visible in the image reconstructed by our method.

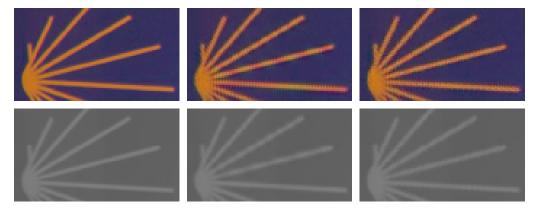


(c) From left to right: Ground-truth, results of [91], and our results. The edges in our result are sharper and exhibit less ringing artifacts.

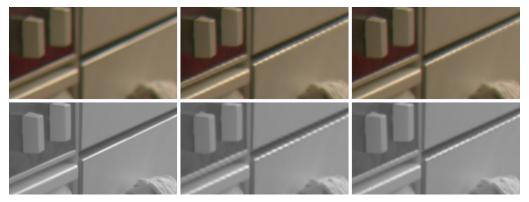


(d) From left to right: Ground-truth, results of [91], and our results. Letters "T", "R", and "S" contain false colors in the result of [91]. The color on these letters looks better in our result.

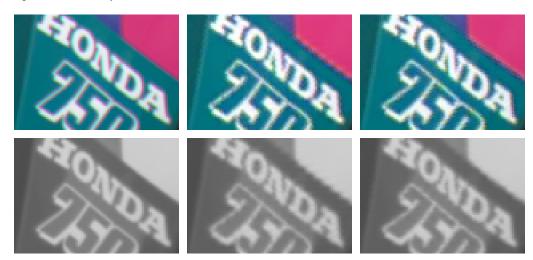
Figure 3.15: For each example, the top row shows color images and the NIR images are placed in the bottom row.



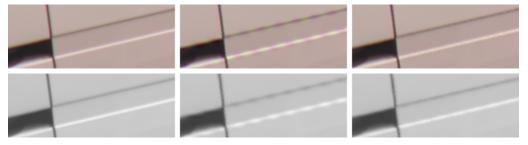
(e) From left to right: Ground-truth, results of [91], and our results. The image reconstructed by our method contains sharper edges and less color artifacts.



(f) From left to right: Ground-truth, results of [91], and our results. Our algorithm reconstructs the edges more faithfully.



(g) From left to right: Ground-truth, results of [91], and our results. Fewer ringing artifacts are visible in our result.



(h) From left to right: Ground-truth, results of [91], and our results. Edges look better with fewer false colors in our result.

Figure 3.15: For each example, the top row shows color images and the NIR images are placed in the bottom row (cont.).

#### Optimizing the NIR Transmittance

Assuming that all the filters in the CFA equally transmit NIR, our algorithm optimizes the transmittance of filters in the visible band only. It is however possible to modify the proposed algorithm such that it optimizes four coefficients (transmittance in red, green, blue, and NIR bands) for each filter. Figure 3.16 shows the coefficients optimized for a  $4 \times 4$  CFA.

We applied this CFA and its corresponding demosaicing matrix to sample and reconstruct the image set described in Subsection 3.3.1. The average PSNR results obtained for color and NIR images are, respectively, 44.75 and 41.66 dB. Comparing these values with the results reported in Table 4.2 shows that the quality of color images is improved by using this CFA, but it results in lower-quality NIR images. This could be explained by a lower NIR sampling frequency in this CFA. Note that, as shown in Figure 3.16, the NIR transmittance is zero for six filters.

$\alpha_R:0.44$	$\alpha_R:0.97$	$\alpha_R:0.01$	$\alpha_R:0.00$
$\alpha_G:0.00$	$\alpha_G:0.00$	$\alpha_G:0.60$	$\alpha_G:0.35$
$\alpha_B:0.00$	$\alpha_B: 1.00$	$\alpha_B:0.00$	$\alpha_B: 1.00$
$lpha_{\mathcal{N}}:0.21$	$\alpha_{\mathcal{N}}:0.00$	$\alpha_{\mathcal{N}}:0.56$	$\alpha_{\mathcal{N}}:0.07$
$\alpha_R:0.00$		$\alpha_R:1.00$	$\alpha_R:1.00$
$\alpha_G: 1.00$		$\alpha_G:0.58$	$\alpha_G: 0.00$
$\alpha_B:0.11$		$\alpha_B:0.00$	$\alpha_B:0.52$
$lpha_{\mathcal{N}}:0.05$		$\alpha_{\mathcal{N}}:0.00$	$\alpha_{\mathcal{N}}:0.73$
$\alpha_R : 0.79$	$\alpha_R : 0.96$		$\alpha_R : 0.19$
$\alpha_G:0.00$	$\alpha_G:0.00$		$\alpha_G: 0.99$
$\alpha_B:0.76$	$\alpha_B:0.00$		$\alpha_B:0.00$
$\alpha_{\mathcal{N}}:0.00$	$\alpha_{\mathcal{N}}:0.60$		$\alpha_{\mathcal{N}}: 1.00$
$\alpha_R:0.00$	$\alpha_R:0.00$	$\alpha_R : 0.66$	$\alpha_R:1.00$
$\alpha_G:0.25$	$\alpha_G: 0.73$	$\alpha_G:0.00$	$\alpha_G: 0.71$
$\alpha_B: 1.00$	$\alpha_B:0.03$	$\alpha_B:0.12$	$\alpha_B:0.00$
$\alpha_{\mathcal{N}}:0.88$	$\alpha_{\mathcal{N}}:0.00$	$\alpha_{\mathcal{N}}:0.38$	$\alpha_{\mathcal{N}}:0.00$

Figure 3.16: The CFA obtained by optimizing four coefficients (transmittance in red, green, blue, and NIR bands) for each filter. Note that six filters completely block the NIR radiation.

#### **Failure Cases**

The most challenging features to preserve when reconstructing a full-resolution image from subsampled measurements are edges and fine details. The task is even more difficult if different channels do not share the same high-frequency content. Figure 3.17 shows one example, where the text on the book cover is almost invisible in the NIR image, because most color pigments are transparent in NIR. In this region, the high-frequency components of NIR and color channels are quite different.

Although the CFA and demosaicing matrices designed by our algorithm and by

3.4. Summary 65

the method of Lu et al. [91] usually result in high-quality reconstructions, they both fail to correctly estimate full-resolution patches where color and NIR high frequencies are not strongly correlated. Figures 3.17-(b) and (c) illustrate that the reconstructed images suffer from two types of artifacts around these regions: (1) the edges are not correctly preserved in the color image, (2) more importantly the text is visible in the reconstructed NIR image.

As most patches in natural images share similar high-frequency details in color and NIR, our training set is dominated by these patches. As a result, the demosaicing matrix optimized over this training set does not perform well in reconstructing patches with different high-frequency components.

# 3.4 Summary

We first computed the correlation between the low-frequency and high-frequency subbands of these images, and showed that the correlation is usually strongest in horizontal and vertical high-frequency subbands (LH and HL). Although the average correlation on our dataset is maximum for NIR/green and NIR/red channels, for some images the correlation is higher for the NIR and blue pair. We further analyzed the correlation in different frequency bands, and concluded that the low-frequency components of color and NIR images are weakly correlated as the absolute intensities of these representations are in many cases different. However, the high-frequencies are usually strongly correlated.

We exploited these results in order to develop a framework for designing an optimum CFA and demosaicing. Our optimization problem minimizes (1) the error of demosaicing over the training set and (2) the differences between the high-frequency components of reconstructed  $\mathcal{N}$  and  $\mathbf{Y}$  channels. The simulation results verify that the quality of images reconstructed by our algorithm is higher than the results of a similar method presented in [91] by Lu et al.

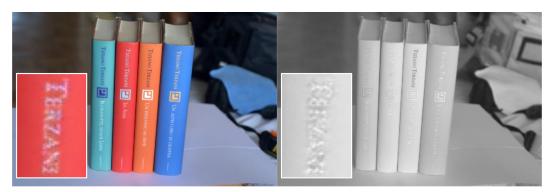
Despite producing high-quality images, both our algorithm and Lu's method suffer from two limitations. First, the optimized CFAs contain two NIR-only filters, which requires the dye filter to have a sharp transition region that is very difficult and costly to implement. We also observe that these algorithms are not as successful when the high-frequency details of color and NIR images are significantly different. This is a challenging case for any inverse underconstrained problem, including demosaicing. As the correlation and redundancy in the signal decrease, it becomes more difficult to reconstruct the signal from its undersampled measurements. In the next chapter, we address these issues by considering the RGBN CFA for the joint acquisition of color and NIR images.



(a) Original images. The text on the book cover is almost invisible in the NIR image as most color pigments (except for carbon black) are transparent in NIR. This leads to a very weak correlation between the high-frequency components of color and NIR images.



(b) The results of Lu's method [91].



(c) The images reconstructed by our optimized CFA and demosaicing.

Figure 3.17: Both our method and Lu's fail to correctly reconstruct the patches where color and NIR do not share the same high-frequency content.

# Chapter 4

# RGBN CFA and Sparse Reconstruction

In this chapter we study the joint acquisition of color and NIR images when the RGBN CFA samples the scene. This CFA, as explained in Chapter 1, is formed by one red, green, blue, and NIR-pass filter. The RGBN CFA is very similar to the popular Bayer CFA.

If the RGBN CFA, similar to current CFAs, is implemented using inexpensive dye filters with smooth transition regions (see Figure 1.9-(b)), NIR radiation leaks into color filters and vice versa. As a result, the sensor records a mixture of NIR and one color channel at each pixel. In this chapter, we propose an algorithm that separates mixed measurements and reconstructs full-resolution color and NIR images sampled by the RGBN CFA.

The problem of separating color and NIR channels is underconstrained, as there is one measurement available in each pixel and two unknowns are to be estimated. Hence, it does not have a unique solution unless additional constraints are added to the problem. To constrain the problem, we explore the spatial and spectral structures of natural color and NIR images. Specifically, we assume these images are sparsely represented in a proper transform domain, meaning that most of the signal coefficients in the transform domain are zero. In this sense, our algorithm is inspired by the solutions proposed in compressive sensing (CS) [37].

To our knowledge, this is the first study in which sparse decomposition is used in the joint acquisition and reconstruction of RGB and NIR images. However, compressive acquisition and sparse reconstruction of color and multispectral images have been studied previously. We review these works and discuss their differences with our problem in Section 4.1. In Section 4.2, the principles of sparse recovery, necessary to comprehend the proposed separation algorithm, are explained. Section 4.3 explains our proposed reconstruction algorithm.

This chapter also includes a thorough comparison of current algorithms proposed for the joint acquisition of color and NIR using a single sensor. We study the

performance of these methods by using different image quality metrics, along with visual comparisons. We conclude by summarizing the advantages and limitations of each technique, both in terms of image quality and hardware implementations.

# 4.1 Compressive Imaging

#### 4.1.1 Compressive Sensing in Color Imaging

Nagesh and Li in [109], and Majumdar and Ward in [97] use compressive sensing in color imaging. They assume that every measurement is a mixture of all pixel intensities in the image, which would require major changes in the sensing process of current cameras. Gan proposes a block sampling of natural images [47], where each block of the image is densely sensed. This means that each measurement is a mixture of several pixel intensities in a local neighborhood. Implementing the hardware of such a system would require a significant modification in the hardware of current consumer cameras.

The compressive acquisition and demosaicing of color images are also studied by Moghadam *et al.* in [104] and Aghagolzadeh *et al* in [5]. In both of these papers, the authors propose using a CFA that, similar to our optimized CFA, transmits a combination of all color channels at every pixel.

There are two main differences between our work and these studies. First, our goal is to capture four channels instead of three channels captured in color imaging. Moreover, all these designs require significant changes in sensing components (sensor and CFA) of conventional color cameras, whereas fabricating our proposed design is possible with minor modifications in current consumer cameras.

#### 4.1.2 Compressive Sensing in Multispectral Acquisition

Sun and Kelly in [141] propose a compressive sensing hyperspectral imager that uses a digital micromirror device. In [94], Duarte and Baraniuk address the compressive acquisition of multi-dimensional signals, including hyperspectral images. The sensing process proposed in both [141] and [94] is quite different from what is used in current color cameras. In these studies, the authors assume that spectral channels are sampled independently, and each measurement is a spatial multiplex of all pixel intensities in the image of one channel. As opposed to these designs, in our proposed model, the color and NIR channels are only spectrally mixed to respect the limitations imposed by the hardware components of current cameras (we do not allow any spatial multiplexing).

Compressive acquisition of hyperspectral images is also addressed in [51, 50], where Golbabaee *et al.* study the problem of reconstructing hyperspectral images from a few noisy measurements. The main assumption in these studies is that only a few materials with specific spectral signatures are present in every image. This assumption holds for target images in remote sensing applications. However, we

envision the color and NIR acquisition system to be used in everyday photography, where the main assumption of [51, 50] is not valid anymore.

# 4.2 Review of Sparse Reconstruction

Let us consider the following system of linear equations:

$$\mathbf{z}^{m \times 1} = \mathbf{M}^{m \times n} \mathbf{s}^{n \times 1} \quad m < n, \tag{4.1}$$

where **s** is the target signal to be recovered, and **M** is the measurement matrix that samples the target signal to produce the measurement vector **z**. Superscripts denote dimensions. The goal is to recover **s**, knowing **z** and **M**. In compressive sensing usually  $m \ll n$ .

As mentioned earlier, CS imposes the sparsity assumption to find the unique solution of the above problem. If this assumption is enforced, the signal recovery is formulated as solving the following optimization problem:

$$\hat{\mathbf{s}} = \underset{\mathbf{s}}{\operatorname{argmin}} \|\mathbf{s}\|_{0} \quad \text{subject to } \mathbf{z}^{m \times 1} = \mathbf{M}^{m \times n} \mathbf{s}^{n \times 1}.$$
 (4.2)

 $\|.\|_0$  is the quasi  $\ell_0$  norm that counts the number of non-zero elements in a vector, which is a measure of sparsity.

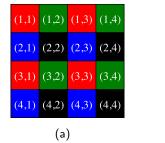
The above problem is usually called "sparse decomposition". Several algorithms are developed to solve this problem or its  $\ell_1$  relaxation. Some of these algorithms are the orthogonal matching pursuit (OMP) [111], the basis pursuit (BP) [25], and the smoothed- $\ell_0$  (SL0) [106].

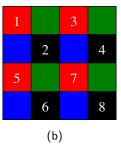
The sparsity assumption does not hold for many classes of natural signals in the original domain. For instance, the majority of pixels in most images have nonzero intensities. However, natural images have sparse representations in a transform domain such as wavelet. As a result, in most cases, problem (4.2) is replaced as follows:

$$\hat{\mathbf{s}} = \underset{\mathbf{s}}{\operatorname{argmin}} \|\mathbf{s}\|_{0} \quad \text{subject to } \mathbf{z} = \mathbf{M}\mathbf{\Phi}\mathbf{s},$$
 (4.3)

where  $\mathbf{s}$  is the sparse representation of the target signal in the transform domain, and  $\mathbf{\Phi}$  is the transform matrix, usually called "dictionary". For instance, if  $\mathbf{\Phi}$  is the discrete cosine transform (DCT) matrix, then  $\mathbf{s}$  contains the DCT coefficients of the signal, which is usually sparse for natural images (for most image patches high-frequency coefficients are approximately zero). The sparsifying transform we design for color and NIR images is explained in Subsection 4.3.4.

As discussed by Candès and Wakin in [20], if the measurement matrix ( $\mathbf{M}$  in (4.2)) is incoherent with the sparsifying transform ( $\mathbf{\Phi}$ ), the perfect recovery of a sparse signal from an underdetermined set of linear measurements is possible with an overwhelming probability. It is established that random matrices are incoherent with most sparsifying transforms [20]. As such, dense random matrices are





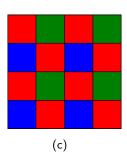


Figure 4.1: (a) Color arrangement in one patch of the image sampled by the RGBN CFA. Both red and black pixels contain mixtures of NIR and red intensities. (b) A block of size  $4 \times 4$  in the CFA image. Indices in this patch show how we arrange the measurements in  $\mathbf{z}$  (see equation (4.4)). (c) Separating NIR and color channels results in subsampled color channels with the shown arrangements.

among the most used measurement operators in compressive sensing. By employing a dense random measurement matrix, every measurement is a linear combination of *all* samples in the signal.

In a single-sensor camera, the structure of the measurement matrix is dictated mainly by the CFA. Hence, the measurement matrix cannot be chosen arbitrarily. In Subsection 4.3.3, we model the measurement matrix in our problem and propose a framework for adaptively designing the measurement operator.

# 4.3 The Proposed Algorithm: Color and NIR Separation

In this section, we first mathematically formulate the task of separating color and NIR channels. As discussed previously, the NIR filter in the RGBN CFA transmits a mixture of the NIR radiation and the red band of the visible spectrum. Hence, by using this CFA, the rate of sampling the mixture of red and NIR is twice the sampling frequency of green/NIR, and blue/NIR mixtures. Therefore, we start by un-mixing red and NIR channels. The output of this step is used in estimating green and blue channels.

## 4.3.1 Separating Red and NIR Channels

Figure 4.1-(a) shows the arrangement of channels in one patch of the image sampled by the RGBN CFA (the mosaiced image), where row and column indices are shown in every pixel. The sensor records a mixture of red and NIR intensities at the pixels colored red and black.

Let us consider a block of size  $\sqrt{p} \times \sqrt{p}$  in the mosaiced image and call the mixture of red and NIR channels at the  $i^{\text{th}}$  pixel  $z_i$ . The measurements in a patch

of size  $\sqrt{p} \times \sqrt{p}$  are stacked in vector  $\mathbf{z}$ :

$$\mathbf{z} = [z_1, z_2, \cdots, z_m]^{\mathrm{T}}, \quad m = \frac{p}{2}$$
 (4.4)

As shown in Figure 4.1-(a), half of the pixels in every patch contain a mixture of red and NIR. The measurements are arranged in vector  $\mathbf{z}$  such that the values measured by one period of the  $2 \times 2$  CFA are placed next to each other in  $\mathbf{z}$ . Figure 4.1-(b) illustrates this ordering in a patch of size  $4 \times 4$ .

The output of the separation algorithm is the red and NIR pixel intensities, in odd rows and columns and in even rows and columns of the patch. We stack the target values in the following vector  $\mathbf{x}$ :

$$\mathbf{x} = [\mathbf{R}_1, \mathbf{R}_2, \cdots, \mathbf{R}_m, \mathcal{N}_1, \mathcal{N}_2, \cdots, \mathcal{N}_m]^{\mathrm{T}}$$
(4.5)

where  $R_i$  and  $N_i$  are, respectively, red and NIR intensities in the  $i^{th}$  pixel.

Every measurement is a weighted average of two samples in the target signal. Hence,  $\mathbf{x}$  and  $\mathbf{z}$  are related through the following linear transformation:

$$\mathbf{z}^{m\times 1} = \mathbf{M}^{m\times 2m} \mathbf{x}^{2m\times 1},\tag{4.6}$$

where M is the measurement matrix.

The goal of the separation step is to find vector  $\mathbf{x}$ , knowing  $\mathbf{z}$  and  $\mathbf{M}$ . We assume that the target signal  $(\mathbf{x})$  has a sparse representation in a transform domain  $\mathbf{\Phi}$ . To separate the channels, we solve the sparse decomposition problem shown in (4.3). Once the sparse representation is computed, the target signal is obtained as  $\hat{\mathbf{x}} = \mathbf{\Phi}\hat{\mathbf{s}}$ . We discuss the dictionary  $(\mathbf{\Phi})$  used to sparsify red and NIR channels of natural images in Subsection 4.3.4.

To separate red and NIR channels, the sparse decomposition problem is solved for all image patches. To increase the estimation accuracy, we partition the image into overlapping patches. The intensity of every pixel is computed several times, and the final value is the average over all the estimations.

#### 4.3.2 Green/Blue and NIR Separation

The NIR values computed in the previous step are interpolated to estimate the missing NIR pixel intensities. This results in the full resolution NIR image. In the next step, we subtract the NIR intensities from the mixtures of green and NIR to estimate green values in the odd rows and even columns of the image<sup>2</sup>. In the same manner, we calculate the blue intensities in the even rows and odd columns (see the indices of rows and columns in Figure 4.1-(a)).

<sup>&</sup>lt;sup>1</sup> For the sake of simplicity, we assume that the image is partitioned into patches that have an even number of pixels  $(p = 2m, m \in \mathbb{N})$ .

<sup>&</sup>lt;sup>2</sup>To perform this operation, the sensitivities of the system (sensor and CFA) in green and NIR bands are taken into account.

Another approach for estimating blue and green intensities is to use the sparse decomposition explained in the previous subsection for separating red and NIR channels. Our experiments show that this approach does not result in significant improvements compared with using the interpolated NIR image as explained above.

After separating the NIR and all the color intensities, the intensities of the subsampled color channels, with the arrangement shown in Figure 4.1-(c), are available. The full-resolution color image can be estimated by applying any color demosaicing algorithm to these samples. Most demosaicing algorithms assume that the Bayer pattern with the highest sampling rate for the green channel samples the scene. We can simply modify any of these methods to demosaic the image for the case when the red channel has the highest sampling frequency (Figure 4.1-(c)).

#### 4.3.3 The Measurement Matrix

In this subsection, we model the sampling of the RGBN CFA at the red and NIR filters by a measurement matrix ( $\mathbf{M}$  in (4.6)). The sensor measures the following value at the location of the red filter:

$$z_1 = \alpha_R^1 R_1 + \alpha_N^1 \mathcal{N}_1, \tag{4.7}$$

where  $R_1$  and  $\mathcal{N}_1$  are the red and NIR intensities at the corresponding pixel.  $\alpha_R^1$  and  $\alpha_N^1$  are the overall sensitivities of the sensor and the filter to red and NIR radiations. The overall sensitivity is the filter transmittance multiplied by the sensor sensitivity. We loosely use the term "filter transmittance" instead of "system overall sensitivity". Similarly, we write the sensor measurement at the location of the NIR filter as follows:

$$z_2 = \alpha_R^2 R_2 + \alpha_N^2 \mathcal{N}_2. \tag{4.8}$$

 $\alpha_R^2$  and  $\alpha_N^2$  are the transmittances of the NIR filter in the red and NIR bands, respectively.

Combining (4.7) and (4.8), it follows that

$$\mathbf{M} = [\mathbf{M}_{\mathrm{R}}, \mathbf{M}_{\mathcal{N}}],\tag{4.9}$$

where:

$$\mathbf{M}_{R} = \begin{bmatrix} \alpha_{R}^{1} & 0 & 0 & \cdots & 0 \\ 0 & \alpha_{R}^{2} & 0 & \cdots & 0 \\ 0 & 0 & \alpha_{R}^{1} & 0 & \cdots \\ \vdots & & & & \end{bmatrix},$$

$$\mathbf{M}_{\mathcal{N}} = \begin{bmatrix} \alpha_{\mathcal{N}}^{1} & 0 & 0 & \cdots & 0 \\ 0 & \alpha_{\mathcal{N}}^{2} & 0 & \cdots & 0 \\ 0 & 0 & \alpha_{\mathcal{N}}^{1} & 0 & \cdots \\ \vdots & & & & \end{bmatrix}.$$

$$(4.10)$$

As the measurement matrix in our problem is highly structured, instead of drawing the coefficients from a random distribution, we find the non-zero coefficients in matrix  $\mathbf{M}$  by solving the following optimization problem:

$$\alpha^* = \underset{\alpha}{\operatorname{argmin}} \sum_{i} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2. \tag{4.11}$$

In this problem,  $\alpha$  contains the non-zero coefficients in matrix M:

$$\alpha = [\alpha_{\mathbf{R}}^1, \alpha_{\mathbf{R}}^2, \alpha_{\mathcal{N}}^1, \alpha_{\mathcal{N}}^2]^T. \tag{4.12}$$

 $\mathbf{x}_i$  in (4.11) is the  $i^{\text{th}}$  patch in the training set. This vector is formed by stacking red and NIR intensities in one patch as shown in (4.5).  $\hat{\mathbf{x}}$  is the reconstructed patch, which is computed as  $\hat{\mathbf{x}} = \mathbf{\Phi}\hat{\mathbf{s}}$ , where  $\hat{\mathbf{s}}$  is obtained by solving (4.3).

We pose the following constraints in order to respect physical limitations in manufacturing the filters:

$$\alpha_{\mathrm{R}}^{1}, \alpha_{\mathrm{R}}^{2}, \alpha_{\mathcal{N}}^{1}, \alpha_{\mathcal{N}}^{2} \leq 1,$$

$$\alpha_{\mathrm{R}}^{1} + \alpha_{\mathcal{N}}^{1} \leq 1, \quad \alpha_{\mathrm{R}}^{2} + \alpha_{\mathcal{N}}^{2} \leq 1$$

$$(4.13)$$

The second and third inequalities model the limited capacities of pixels. We assume that the maximum capacity is normalized to one. Moreover, as discussed before, dye filters used in the RGBN CFA do not have a sharp cut-off, hence all coefficients in  $\alpha$  are required to be non-zero. We use the following additional constraint to guarantee this:

$$\alpha_{\rm R}^1, \alpha_{\rm R}^2, \alpha_{\rm N}^1, \alpha_{\rm N}^2 \ge T_L > 0 \tag{4.14}$$

Here  $T_L$  is a positive constant that models the minimum transmittance of red and NIR filters in red and NIR bands. For instance  $T_L = 0.1$  means that, because of manufacturing limitations, the transmittance of the red filter in the NIR band cannot be less than 10%.<sup>3</sup>

Considering the above constraints, the non-zero coefficients of the measurement matrix are computed by solving the following optimization problem:

$$\alpha^* = \underset{\alpha}{\operatorname{argmin}} \sum_{i} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2. \quad \text{s.t. } T_L \le \alpha < 1, \mathbf{A}\alpha \le \mathbf{b}, \tag{4.15}$$

for:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{4.16}$$

We solve (4.15) for different values of  $T_L$  and present the optimum coefficients in Section 4.4.

<sup>&</sup>lt;sup>3</sup>We assume that the overall transmittance of each filter is normalized to one.

#### 4.3.4 The Sparsifying Transform

The main assumption in our separation algorithm is that the red and NIR channels have a sparse representation in a transform domain denoted as matrix  $\Phi$  in (4.3). In the following, we refer to matrix  $\Phi$ , interchangeably, as the sparsifying transform or the dictionary. In this subsection, we describe the dictionary used in our separation algorithm.

Decorrelating transforms such as Kruhenen-Loève, known as KLT or principal component analysis (PCA), accumulate most of the signal energy in a few non-zero coefficients in the transform domain, when the remaining coefficients are zero or almost zero; this results in a sparse or an approximately sparse representation. Such a representation is obtained by removing the correlation and redundancy in the signal. Similarly, to build a sparsifying transform for the red and NIR channels, we exploit the correlation in our target signal ( $\mathbf{x}$  in (4.5)). In the following, we identify two types of correlations in signal  $\mathbf{x}$ .

#### **Spatial Correlation**

In natural images, the pixel intensities in a small patch are usually correlated because abrupt transitions occur rarely in a small neighborhood. We call this correlation the spatial correlation as it models the spatial relations in a patch. Our target signal  $\mathbf{x}$ , which contains pixel intensities of the red and NIR channels in a local neighborhood, exhibits spatial correlation. Many two-dimensional transforms, such as 2D DCT, 2D wavelet, and curvelet are used by the image processing community to explore the spatial correlation. Any of these transforms are applicable to our problem.

#### **Spectral Correlation**

In addition to a strong spatial correlation, the values in our target signal are correlated in the spectral dimension. This is the result of the correlation between the NIR and red channels at each pixel. To exploit the spectral correlation, we use a two-dimensional PCA that is trained over a set of red and NIR patches. The PCA transform is commonly used in multispectral imaging to remove the spectral redundancy [121, 38].

Figure 4.2 illustrates an example patch where the red and NIR channels are strongly correlated. We refer to these areas of the image as similar patches. As discussed in Chapter 3, differences between scene reflections in visible and NIR bands might result in different high-frequency details in some patches of color and NIR images. Figure 4.2 shows an example of such a patch, called a dissimilar patch here. As both similar and dissimilar categories might be encountered in a pair of color and NIR images, only one dictionary cannot effectively express the diverse spectral structure of our target signal.

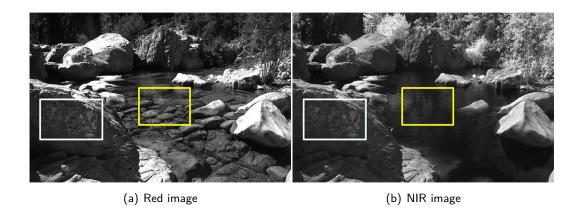


Figure 4.2: Patches with white borders share the same high-frequency information between red and NIR. The ones in yellow boxes are dissimilar patches where high-frequency components of red and NIR are quite different.

To represent the spectral correlation between NIR and red channels, we propose using two dictionaries, each trained over one category of similar or dissimilar patches. To train these transforms, we form two different datasets. The first one contains patches with similar high-frequency information in red and NIR channels, and the other dataset consists of dissimilar patches where high-frequency details of red and NIR images are highly different. To assign each patch in the training set to one of these categories, we compute the DCT coefficients of red and NIR images. We assume that the last H coefficients of the DCT representation carry the high-frequency content of the patch. We call the high-frequency DCT coefficients of the red patch,  $\mathbf{s}_{\mathrm{R}}^{H}$ , and those of the NIR patch,  $\mathbf{s}_{\mathrm{N}}^{H}$ , and we compute the energy of difference between these two vectors normalized by their norm:

$$d = \frac{\||\mathbf{s}_{R}^{H}| - |\mathbf{s}_{N}^{H}|\|_{2}^{2}}{\|\mathbf{s}_{R}^{H}\|_{2}\|\mathbf{s}_{N}^{H}\|_{2} + \epsilon},$$
(4.17)

where  $\epsilon$  is a small constant added to avoid division by zero. If d is smaller than a constant threshold, T, the high-frequency components are strongly correlated and the corresponding pair of patches belongs to the similar dataset (used to train the similar dictionary). If d is above the threshold, the patches are added to the dissimilar dataset. In our experiments we empirically set H to be 0.9 of the number of pixels in the patch and T=4.

We train a different PCA transform over each of these datasets to represent its spectral correlation. We call the PCA matrix of the similar dataset  $\Phi_s^{PCA}$ , and the one trained over the dissimilar transform  $\Phi_{ds}^{PCA}$ . Multiplying these transforms with the spatially decorrelating transform, the following two dictionaries are obtained:

$$\Phi_{s} = \Phi_{s}^{PCA} \times \Phi^{DCT} 
\Phi_{ds} = \Phi_{ds}^{PCA} \times \Phi^{DCT},$$
(4.18)

where  $\Phi^{\rm DCT}$  is the DCT transformation matrix. To better represent our target signal, we further refine above dictionaries applying K-SVD to both datasets. K-SVD is a dictionary learning algorithm proposed by Aharon *et al.* [6]. We use  $\Phi_{\rm s}$  and  $\Phi_{\rm ds}$  in (4.18) to initialize K-SVD for each dataset.

#### Separation with Two Dictionaries

We apply both similar and dissimilar dictionaries in (4.3) to obtain two different estimates of red and NIR channels. We call the NIR channel estimated using  $\Phi_s$ ,  $\mathcal{N}_s$ , and the result of using  $\Phi_{ds}$ ,  $\mathcal{N}_{ds}$ . Similar notations are adopted for the estimated red channels.

Figure 4.3 shows red and NIR images estimated by two dictionaries, along with ground-truth images. The high-frequency details of the red and NIR patches on the screen are very different, as the image on the screen is invisible in the NIR image. Using the similar dictionary ( $\Phi_s$ ) these differences are discarded in the estimated red and NIR images. Whereas,  $\Phi_{ds}$  is successful in estimating these patches. However, the patch with the "PHILIPS" logo is reconstructed better when the similar transform  $\Phi_s$  is used. The reason is that the NIR and red high frequency details are similar in this patch and  $\Phi_s$  is trained to represent the strong spectral correlation; and  $\Phi_{ds}$ , trained over dissimilar patches, is not as successful in recovering this patch.

To obtain an accurate estimation for all patches in the image, we fuse the images reconstructed by both dictionaries. To this end, each patch is categorized as either a similar one where red and NIR high frequencies are correlated or a dissimilar patch where high frequencies are different.

Our experiments show that the edge maps of dissimilar estimations ( $\mathcal{N}_{ds}$  and  $\mathbf{R}_{ds}$ ) provide an acceptable approximation of the level of correlation between the red and NIR edges in each patch. Figure 4.4 shows the edge maps of these images for the example given in Figure 4.3. If the NIR and red edge maps are the same in more than  $90\%^4$  of pixels in the patch, it is categorized as a similar patch. We form a binary map,  $\mathbf{B}$ , by comparing the edge maps in each local neighborhood.  $\mathbf{B}$  is one at a pixel that belongs to a similar patch, otherwise it is zero. Using this map, the final NIR and red channels are computed as follows:

$$\mathcal{N} = \mathbf{B} \odot \mathcal{N}_{s} + (1 - \mathbf{B}) \odot \mathcal{N}_{ds}, 
\mathbf{R} = \mathbf{B} \odot \mathbf{R}_{s} + (1 - \mathbf{B}) \odot \mathbf{R}_{ds}.$$
(4.19)

Here  $\odot$  represents the element-wise multiplication. Figure 4.3-(d) shows the final estimations obtained using both similar and dissimilar dictionaries.

 $<sup>^4\</sup>mathrm{We}$  allow a small deviation from the perfect match between two edge maps because of noise and reconstruction errors.



(a) Ground-truth images



(b) Images reconstructed by the similar transform



(c) Images reconstructed by the dissimilar transform



(d) Final results of our algorithm obtained by fusing (b) and (c)

Figure 4.3: Left column: RGB images, and the right column: NIR images. (a) Ground-truth images. (b) By using the similar transform, the differences between color and NIR details are discarded (patches inside white boxes). (c) Although successful in reconstructing dissimilar patches, the dissimilar dictionary does not retrieve the shared edges faithfully (see the "PHILIPS" logo). (d) By fusing the results of similar and dissimilar dictionaries, high-quality estimations of all patches are obtained.



(a) Red image reconstructed by the dissimilar transform  $(\mathbf{R}_{\mathrm{ds}})$ 

(b) NIR image reconstructed by the dissimilar transform  $(\mathcal{N}_{\mathrm{ds}})$ 



(c) The edge map computed from the reconstructed red image

(d) The edge map computed from the reconstructed NIR image

Figure 4.4: Red and NIR edge maps of images reconstructed by the dissimilar dictionary. These maps provide an estimation of the inherent correlation between the high-frequency details of red and NIR images in local patches. For instance, the screen looks different in red and NIR ground-truth images, which is reflected in the differences of edge maps in the corresponding patches. Edge pixels are shown in black.

#### 4.4 Simulations of Our Proposed Algorithm

#### 4.4.1 Implementation Details

To find the optimum values for the measurement matrix  $\mathbf{M}$ , we solve problem (4.11) by MATLAB's "fmincon" function designed for solving general constrained optimization problems. As the problem is nonlinear and non-convex, we run each optimization with 10 different initial points and choose the solution that results in the lowest value for the cost function. We perform the optimization for different numbers of training patches and report the results in the following.

To separate red and NIR channels, we use patches of size  $p \times p$  with an overlap of p-2 pixels in each of horizontal and vertical directions. The results for p=8 and p=12 are reported in the following subsection. We use the smoothed- $\ell_0$  (SL0) algorithm [106] to solve the sparse decomposition problem (4.3) for each patch. To interpolate the NIR image after the red-NIR separation, we employ the edge-adaptive interpolation technique of Li and Orchard [88]. After the color and NIR intensities are separated, the full-resolution color image is obtained by the color demosaicing algorithm of Hirakawa and Parks [62]. This algorithm is designed for the Bayer CFA where the green sampling frequency is the highest. We modify it for our problem where, after separation, the red sampling rate is the highest.

To evaluate the performance of our algorithm, we use a dataset of 60 pairs of color and NIR images. We run every experiment for three different values of  $T_L$ . From (4.14) recall that  $T_L$  is the minimum achievable transmittance of the red filter in the NIR band, and vice versa for the NIR filter. In the next subsection, we present the results for  $T_L = 0.1$ ,  $T_L = 0.2$ , and  $T_L = 0.3$ .

#### 4.4.2 Results

In the following, we first present the results of training the dictionary and the measurement matrix. We then compare the quality of reconstructed images when different patch sizes are used, and show the effect of using two dictionaries in separation. In Section 4.5, we compare our algorithm with other single-sensor acquisition techniques.

#### **Dictionary Training**

In the first set of experiments, we use different training sets to design the dictionary  $(\Phi)$ . We employ each of these dictionaries in reconstructing 60 pairs of color and NIR test images. In each experiment, we use the same arbitrary measurement matrix with different trained dictionaries. The non-zero coefficients for each measurement matrix respect inequalities (4.13) and (4.14). The coefficients we use for this experiment are shown in Table 4.1. Figure 4.5 shows an example of red and NIR filters' spectral transmittance for  $T_L = 0.2$ .

	$T_L = 0.1$	$T_L = 0.2$	$T_L = 0.3$
$\alpha_{ m R}^1$	0.9	0.8	0.7
$\alpha_N^1$	0.1	0.2	0.3
$\alpha_{\rm R}^2$	0.1	0.2	0.3
$\alpha_N^2$	0.9	0.8	0.7

Table 4.1: The transmittance of red and NIR filters in red and NIR bands (see (4.7) and (4.8)). For each  $T_L$ , the four coefficients define the measurement matrix. We arbitrarily choose these coefficients to study the effect of dictionary on the quality of reconstructed images.

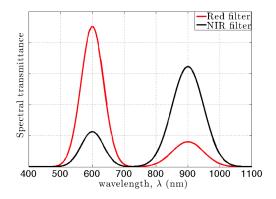


Figure 4.5: An illustration of red and NIR filters spectral transmittance for the values shown in Table 4.1 for  $T_L = 0.2$ .

To design the dictionary, we use a dataset of red and NIR patches. To investigate the effect of the training-set size on the results, we train two dictionaries, one for 12000 red and NIR patches and the other one for 75000 patches, extracted from 30 pairs of train images. The CPSNR and PSNR values for images reconstructed by each dictionary are reported in Table 4.2-(a). These results show that, although not significant, increasing the size of the training set leads to a higher-quality reconstruction.

We also analyze the effect of the training set on the performance of the algorithm. To this end, two different sets each with 30 pairs of images are used to train the dictionary. We extract 12000 patches from each set. The results of using these two dictionaries in reconstructing the test set are shown in Table 4.2-(b). Small differences in the results show that the dictionary learning algorithm performs equally well for both training datasets.

#### Training the Measurement Matrix

We compute the optimum measurement matrix by solving (4.15). The coefficients obtained for a training set of 12000 image patches and different values of  $T_L$  are shown in Table 4.3. For all  $T_L$  values, one filter has a higher transmittance in the red band and the other one transmits the NIR light more than the red radiation, although there is no constraint in our optimization favoring this solution. This outcome perfectly matches the model of the RGBN CFA, where the first filter is a red filter with a higher transmittance in the red part of the spectrum and the second filter is an NIR one that transmits mostly the NIR radiation.

To investigate the effect of training set, we extract 12000 and 75000 patches from

(a)

		training size = $12000$	training size = $75000$
$T_L = 0.1$	CPSNR (color)	40.55	40.76
IL = 0.1	PSNR (NIR)	42.26	43.34
$T_L = 0.2$	CPSNR (color)	40.71	40.78
$I_L - 0.2$	PSNR (NIR)	42.08	42.96
$T_L = 0.3$	CPSNR (color)	40.74	40.63
$I_L = 0.3$	PSNR (NIR)	41.51	42.08

(b)

		First training dataset	Second training dataset
$T_L = 0.1$	CPSNR (color)	40.55	40.64
$I_L = 0.1$	PSNR (NIR)	42.26	42.97
$T_L = 0.2$	CPSNR (color)	40.71	40.74
IL = 0.2	PSNR (NIR)	42.08	42.76
$T_L = 0.3$	CPSNR (color)	40.74	40.74
$I_L = 0.3$	PSNR (NIR)	41.51	42.18

Table 4.2: The effect of training set on the sparsifying dictionary. The reported results are averaged over 60 pairs of color and NIR images. For different dictionaries, we use the same measurement matrix, with non-zero coefficients shown in Table 4.1. (a) For these experiments, two dictionaries are trained over 12000 and 75000 patches extracted from 30 pairs of color and NIR train images. (b) In this experiment, we use two dictionaries trained over two different sets of images. Each dictionary is trained over 12000 patches.

	$\alpha_{ m R}^1$	$\alpha_{\mathcal{N}}^{1}$	$\alpha_{ m R}^2$	$\alpha_N^2$
$T_L = 0.1$	0.58	0.10	0.20	0.80
$T_L = 0.2$	0.80	0.20	0.20	0.80
$T_L = 0.3$	0.70	0.30	0.30	0.70

Table 4.3: The optimum transmittances of red and NIR filters in the RGBN CFA. These values are obtained by solving problem (4.15) for different values of  $T_L$ .

the same set of images to train two sets of measurement matrices. We use each of the measurement matrices to sample the test images and the dictionaries trained for the same datasets are used to reconstruct the CFA images. The nonzero coefficients of measurement matrices and average CPSNR and PSNR results of using each matrix are reported in Tables 4.4-(a) and 4.4-(b).

We observe in Table 4.4 that computing the measurement matrix is not very sensitive to the size of the training set. In fact, except for slight differences in optimum values for  $T_L = 0.1$ , using two training sets results in the same measurement matrices.

The differences between the CPSNR and PSNR results in Table 4.2-(a) and Table 4.4 are in most cases negligible (the maximum difference is less than 0.2 dB). This demonstrates that although using optimized measurement matrices slightly improves the quality, the performance of our reconstruction algorithm is almost not affected by the exact transmittances of CFA filters. Hence our algorithm is not sensitive to manufacturing errors in implementing the filters, that are specifically hard to avoid when inexpensive dye filters are used.

#### Patch Size

Our simulations confirm that changing the size of the processed patch does not lead to noticeable differences in the performance of our reconstruction algorithm. In Table 4.5, we report the results for reconstructing images using  $8 \times 8$  and  $12 \times 12$  patches.

#### 4.4.3 Separation with Two Dictionaries

In Subsection 4.3.4, we propose using two dictionaries (similar and dissimilar transforms) to represent the spectral correlation between the red and NIR channels. To train these dictionaries, we form two datasets each containing 25000 patches.

Figures 4.6-(b) and (e) show the images reconstructed using the similar transform only. The images obtained by fusing the results of similar and dissimilar transforms are illustrated in Figures 4.6-(c) and (f). It is immediately observed that the inherent differences between color and NIR are not preserved in images reconstructed by

using the similar transform only. Whereas, combining the results of both similar and dissimilar dictionaries successfully retrieves these differences.

(a) The optimum transmittance of red and NIR filters in the RGBN CFA. The coefficients are trained over two sets containing 12000 (data 1) and 75000 (data 2) patches.

	$T_L = 0.1$		$T_L$ =	= 0.2	$T_L = 0.3$	
	data 1	data 2	data 1	data 2	data 1	data 2
$\alpha_{ m R}^1$	0.58	0.47	0.80	0.66	0.70	0.70
$\alpha_N^1$	0.10	0.10	0.20	0.20	0.30	0.30
$\alpha_{ m R}^2$	0.21	0.13	0.20	0.20	0.30	0.30
$ \begin{array}{c c} \alpha_{\mathrm{R}}^{1} \\ \alpha_{\mathcal{N}}^{1} \\ \alpha_{\mathrm{R}}^{2} \\ \alpha_{\mathcal{N}}^{2} \end{array} $	0.79	0.87	0.80	0.80	0.70	0.70

(b) CPSNR and PSNR results of images sampled by the above measurement matrices and reconstructed by our algorithm.

	$T_L = 0.1$		$T_L$ =	= 0.2	$T_L = 0.3$	
	data 1	data 2	data 1	data 2	data 1	data 2
CPSNR (color)	40.78	40.82	40.71	40.78	40.74	40.63
PSNR (NIR)	41.92	43.40	42.08	43.04	41.51	42.08

Table 4.4: Analyzing the effect of training set on the optimized measurement matrix. Using a larger training set slightly improves the results especially for NIR images.

		$8 \times 8$ patches	$12 \times 12$ patches
T 0.1	CPSNR (color)	40.76	40.45
$T_L = 0.1$	PSNR (NIR)	43.34	43.37
T 0.0	CPSNR (color)	40.78	40.50
$T_L = 0.2$	PSNR (NIR)	42.96	43.02
$T_L = 0.3$	CPSNR (color)	40.63	40.46
$I_L = 0.3$	PSNR (NIR)	42.08	42.03

Table 4.5: Changing the patch size does not significantly affect the performance of our reconstruction algorithm.



Figure 4.6: (a) Ground-truth images (b) By using only the transform that is trained for similar patches, the inherent differences between color and NIR are not preserved in the reconstructed images. (c) By combining the results obtained by both similar and dissimilar transforms, our algorithm successfully maintains the differences.

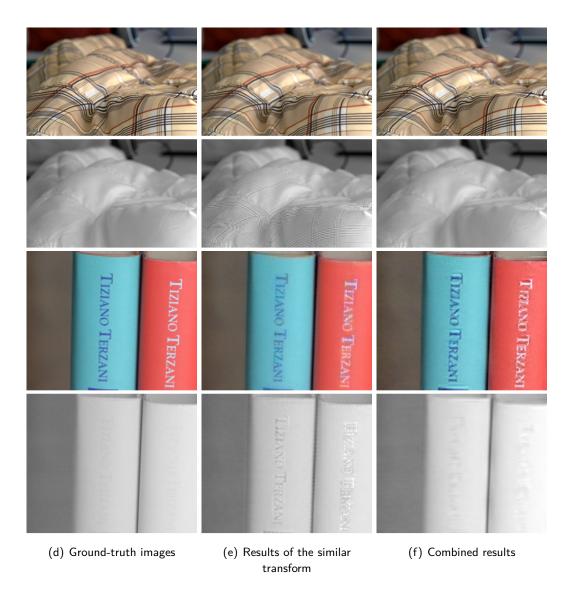


Figure 4.6: (d) Ground-truth images (e) By using only the transform that is trained for similar patches, the inherent differences between color and NIR are not preserved in the reconstructed images. (f) By combining the results obtained by both similar and dissimilar transforms, our algorithm successfully maintains the differences (cont.).

# 4.5 Analysis of Single-Sensor Color and NIR Acquisition Techniques

In this section, we use three single-sensor multispectral imaging techniques [102, 4, 75] for the joint acquisition of color and NIR images and compare their performance with our proposed methods (based on the optimized CFA and the RGBN CFA). In the following, we first briefly review each of these algorithms<sup>5</sup>. We then explain the metrics chosen for evaluating the quality of images reconstructed by each method. We conclude the section by presenting numerical and visual results and discussing the advantages and limitations of each method.

#### 4.5.1 Single-Sensor Acquisition Techniques

#### The Binary-Tree Algorithm

Miao et al. in [102] present an algorithm for designing a generic CFA given the sampling rate of each spectral channel. Their demosaicing algorithm starts by an edge adaptive interpolation of the channel with the highest sampling frequency. Miao et al. do not provide any recommendations regarding optimum sampling rates for different spectral channels. We choose to use their algorithm with an equal sampling rate of 1/4 for the red, green, blue, and NIR channels. The designed CFA is shown in Figure 4.7-(a). The arrangement of filters is similar to our RGBN CFA, however Miao et al. assume that each filter transmits only the corresponding channel. In the following, we refer to this algorithm as BT.

#### **Least-Square Multispectral Demosaicing**

In [4], Aggarwal and Majumdar propose a uniform CFA where different channels are sampled on diagonal stripes. Their four-channel CFA is shown in Figure 4.7-(b). The missing values at each pixel are calculated as weighted averages of measurements in neighboring pixels. The weights are optimized by minimizing the reconstruction error in a training set of images. The algorithm is called the least-square based multispectral demosaicing, referred to as LMSD in what follows.

#### The Hybrid CFA Technique

The hybrid CFA presented by Kiku et al. in [75] samples color channels and an additional band, which is NIR in this section. See Figure 4.7-(c) for an illustration. Kiku et al. propose a disjoint reconstruction of the color image and the image of the additional band. The proposed color demosaicing algorithm is a combination of the guided image filtering [58] and the gradient-based interpolation [113]. To

 $<sup>^5\</sup>mathrm{For}$  more details about these algorithms, please refer to Chapter 2.

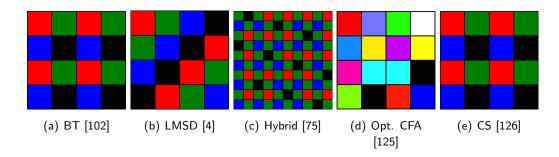


Figure 4.7: The color filter arrays proposed by five single-sensor acquisition techniques. In all CFAs, the black pixel represents an NIR-pass filter. Our RGBN CFA (e) is the only one that models visible and NIR radiation leaking into NIR and color filters, respectively.

reconstruct the additional band, they apply the super-resolution algorithm of [98]. We call this approach the hybrid technique.

#### The Optimized CFA Technique

We explained our algorithm for optimizing CFA and demosaicing matrices in Chapter 3. For the experiments of this section, we use the optimized CFA shown in 4.7-(d). For a detailed explanation on the process of designing this CFA, please refer to Chapter 3. In the following, we use the abbreviation "Opt. CFA" for this method.

#### The RGBN CFA technique

The last sampling technique we analyze in this section uses the RGBN CFA with dye filters. To reconstruct images, we use the separation algorithm proposed in this chapter. The RGBN CFA is shown in Figure 4.7-(e), and we refer to this technique as CS, as the separation algorithm is based on the compressive sensing framework.

#### 4.5.2 Quality Metrics

To compare the performance of different algorithms, we use the following quality metrics that are designed to measure the similarity of the reconstructed image to the reference (ground-truth) image. Ground-truth color and NIR images are, respectively, called  $\mathbf{I}_c$  and  $\mathcal{N}$ , and the reconstructed images are  $\hat{\mathbf{I}}_c$  and  $\hat{\mathcal{N}}$ .

#### **CPSNR** and **PSNR**

Color peak signal to noise ratio (CPSNR) and peak signal to noise ratio (PSNR) measure the quality of reconstructed color and NIR images as

$$PSNR = 10 \log_{10} \frac{255^{2}}{\frac{1}{P} \sum_{i,j} (\mathcal{N}(i,j) - \hat{\mathcal{N}}(i,j))^{2}}$$

$$CPSNR = 10 \log_{10} \frac{255^{2}}{\frac{1}{3P} \sum_{i,j,k} (\mathbf{I}_{c}(i,j,k) - \hat{\mathbf{I}}_{c}(i,j,k))^{2}}$$
(4.20)

P is the number of pixels in each image, (i, j) indicate spatial coordinates, and k indices different color channels.

#### Structural Similarity Index Measure (SSIM)

Developed by Wang *et al.*, the structural similarity index measure (SSIM) compares the average intensity, the contrast, and the structure of the reference and reconstructed images [149]. SSIM is applied to gray-scale images. Here we explain the metric for NIR images; for color images we use the same formulation for the luminance channel (the average of color channels).

Let us call the average intensities and standard deviations of ground-truth and reconstructed NIR images  $\mu_{\mathcal{N}}$ ,  $\mu_{\hat{\mathcal{N}}}$ , and  $\sigma_{\mathcal{N}}$  and  $\sigma_{\hat{\mathcal{N}}}$ . The cross-correlation between original and reconstructed images, representing the similarity between their structure, is computed as

$$\sigma_{\mathcal{N}\hat{\mathcal{N}}} = \frac{1}{P-1} \sum_{i,j} (\mathcal{N}(i,j) - \mu_{\mathcal{N}}) (\hat{\mathcal{N}}(i,j) - \mu_{\hat{\mathcal{N}}})$$
(4.21)

P is the number of pixels. The average intensities, contrast, and structure are compared through the following formulation:

$$SSIM = \frac{(2\mu_{\mathcal{N}}\mu_{\hat{\mathcal{N}}} + \epsilon_1)(2\sigma_{\mathcal{N}\hat{\mathcal{N}}} + \epsilon_2)}{(\mu_{\mathcal{N}}^2 + \mu_{\hat{\mathcal{N}}}^2 + \epsilon_1)(\sigma_{\mathcal{N}}^2 + \sigma_{\hat{\mathcal{N}}}^2 + \epsilon_2)}$$
(4.22)

In the above equation  $\epsilon_1$  and  $\epsilon_2$  are small constants included to avoid zero division. The SSIM metric is in the range of [-1,1], with SSIM = 1 only when two images are exactly the same.

#### Visual Information Fidelity (VIF)

Sheikh and Bovic in [136] present the visual information fidelity (VIF) metric that aims to measure the loss of information in the reconstructed image. The VIF algorithm compares the amount of information carried by the ground-truth image with the information contained in the reconstructed image. The final VIF value is the ratio of these two information measures. The VIF metric is in the range of  $[0, \infty]$ , with VIF = 1 only when two images are exactly the same. If the reconstructed image contain more information compared with the ground-truth image, for instance as a result of contrast enhancement, VIF > 1.

#### Delta E and Zipper Artifacts

Recommended by the International Commission on Illumination (CIE), Delta E  $(\Delta E)$  is a measure of color difference.  $\Delta E$  is computed in the Lab color space that is designed to be perceptually uniform. Representing the color image ( $\mathbf{I}_c$ ) in the Lab space as  $\mathbf{I}_{\text{Lab}}$ ,  $\Delta E$  is computed as follows:

$$\Delta E = \frac{1}{P} \sum_{i,j} (\mathbf{I}_{Lab}(i,j) - \hat{\mathbf{I}}_{Lab}(i,j))^{2}.$$
 (4.23)

In the next subsection, we report the  $\Delta E$  values computed for color images reconstructed by each method.

Another common artifact observed in demosaiced images is the zipper effect, mostly visible around edges. In [90], Lu and Tan argue that when a pixel suffers from the zipper artifact, its color difference with the most similar color in the neighborhood increases, compared with this difference in the original image. Hence, to quantify the zipper effect, for each pixel they find the pixel with the most similar color in a close neighborhood in the ground-truth and compute the color difference, called  $\Delta E_1$ . They compute the color difference between the same pair of pixels in the reconstructed image and call it  $\Delta E_2$ . If  $\Delta E_2 - \Delta E_1 > 2.3$ , this pixel is assumed to suffer from zipper. The percentage of pixels with the zipper effect measures the severity of this artifact. Similar to  $\Delta E$ , the zipper metric is designed for color images only.

#### 4.5.3 Simulation Results

All the results reported in this subsection are computed over 60 pairs of color and NIR images. For each technique, we simulate the CFA sampling based on the patterns shown in Figure 4.7. The full-resolution image is then reconstructed by the corresponding algorithm.

We start by analyzing CPSNR and PSNR results in Table 4.6. Our optimized CFA and demosaicing achieve the best compromise between the color and NIR PSNR results. The average CPSNR values of the LMSD and hybrid techniques is less than 0.1 dB higher than the CPSNR of the optimum CFA algorithm. However our algorithm outperforms the LMSD and hybrid techniques in reconstructing NIR images with, respectively, 2 and 5 dB difference. The results of our CS algorithm are comparable to those of other techniques. This is despite the fact that the CFA used by our algorithm is the simplest among other techniques, and more importantly, our CS algorithm is the only one that models the cross-talk between visible and NIR bands. This practical modeling increases the difficulty of reconstruction considerably, yet our algorithm is successful in producing high-quality images.

Our experiments with the SSIM metric (results in Table 4.7) show that this metric is not powerful enough to measure the quality of demosaiced images. Although other metrics and visual comparisons, as shown later, confirm noticeable differences

in the performance of different algorithms, SSIM is not capable of measuring these differences.

The VIF results are reported in Table 4.8. Our algorithms outperform other techniques in terms of VIF for both color and NIR images, with only one exception. The hybrid technique performs slightly better (0.06) than our CS algorithm in terms of color VIF, however our NIR VIF is on average 0.23 higher than the hybrid's result.

In terms of  $\Delta E$  (Table 4.9), which aims at measuring the quality of color reconstruction, our CS algorithm performs the best. Although still being below the just noticeable difference (JND) of 2.3, the error of color reconstruction is the highest in the images produced by our optimum CFA algorithm. This might be explained by the fact that in the optimized CFA each filter transmits a mixture of color channels, which increases the correlation between the measurements of different filters. Although resulting in a higher-quality spatial reconstruction, this correlation, as shown by Alleysson  $et\ al.\ [10]$ , decreases the color acuity.

As reported in Table 4.10, the percentage of pixels suffering from zipper artifacts is the smallest for the results obtained by the hybrid algorithm. This is mainly due to the high sampling frequency of color channels (90% in total) in this technique. In fact, this algorithm sacrifices the quality of the NIR image to achieve a high-quality color image, which results in severe artifacts in reconstructed NIR images. After the hybrid method, our CS algorithm performs the best in terms of the zipper artifacts. The optimum CFA method does not perform well in terms of the zipper artifacts. The reason is that, to reconstruct the image, this method uses a simple but efficient linear demosaicing and no post-processing is performed to reduce the artifacts. We observed that by applying a simple median filtering to the images obtained by the optimized demosaicing, the average zipper percentage drops from 5.61% to 1.96%, which is the second best after the hybrid technique (1.82% zipper).

Figure 4.8 shows several crops of color and NIR images sampled and reconstructed by different acquisition techniques.

		BT	LMSD	Hybrid	Opt. CFA	CS
CPSNR (color)	ave.	41.98	43.30	43.31	43.22	40.68
	std.	5.15	5.07	4.75	4.57	4.50
PSNR (NIR)	ave.	41.98	41.55	38.52	43.57	43.21
r sinn (inin)	std.	4.51	4.33	4.36	4.57	4.35

Table 4.6: CPSNR and PSNR results for 60 pairs of color and NIR images sampled and reconstructed by five different techniques.

		BT	LMSD	Hybrid	Opt. CFA	CS
Color SSIM	ave.	0.98	0.98	0.99	0.99	0.98
	std.	0.02	0.02	0.01	0.01	0.01
NIR SSIM	ave.	0.98	0.97	0.95	0.98	0.98
NIR SSIM	std.	0.02	0.02	0.03	0.01	0.02

Table 4.7: SSIM results for 60 pairs of color and NIR images sampled and reconstructed by five different techniques. To compute color SSIM, we compare the Y channels of original and reconstructed color images, as SSIM is developed for gray-scale images.

		BT	LMSD	Hybrid	Opt. CFA	CS
Colon VIE	ave.	0.76	0.77	0.89	0.89	0.83
Color VIF	std.	0.08	0.08	0.04	0.08	0.08
NIR VIF	ave.	0.73	0.66	0.55	0.75	0.78
NIR VIF	std.	0.08	0.08	0.10	0.08	0.08

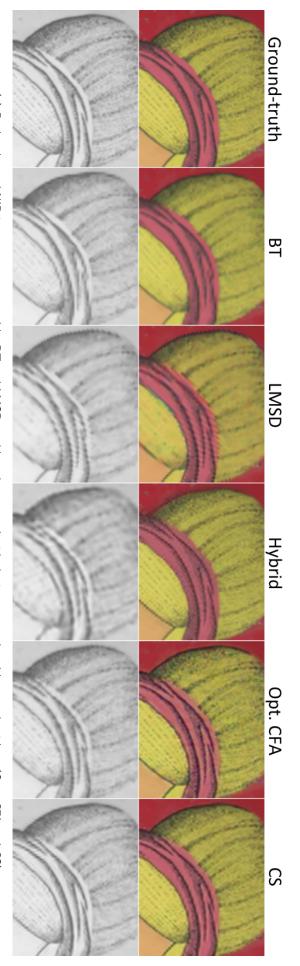
Table 4.8: VIF results for 60 pairs of color and NIR images sampled and reconstructed by five different techniques. To compute color VIF, we compare the Y channels of original and reconstructed color images.

		BT	LMSD	Hybrid	Opt. CFA	CS
$\Lambda E$	ave.	1.34	1.74	1.62	2.15	0.96
$\Delta E$	std.	1.41	1.51	2.39	1.73	0.53

Table 4.9:  $\Delta E$  results for 60 color images sampled and reconstructed by five different techniques. As  $\Delta E$  measures the color accuracy, we only compute it for color images.

		BT	LMSD	Hybrid	Opt. CFA	CS
Zinnon	ave.	11.06	4.41	1.82	5.61	3.02
Zipper	std.	11	4.07	2.04	5.66	2.52

Table 4.10: Percentage of pixels suffering from the zipper artifacts averaged over 60 color images sampled and reconstructed by five different techniques.



(a) Both color and NIR images reconstructed by BT and LMSD are blurred compared with the images produced by our algorithms (Opt. CFA and CS).

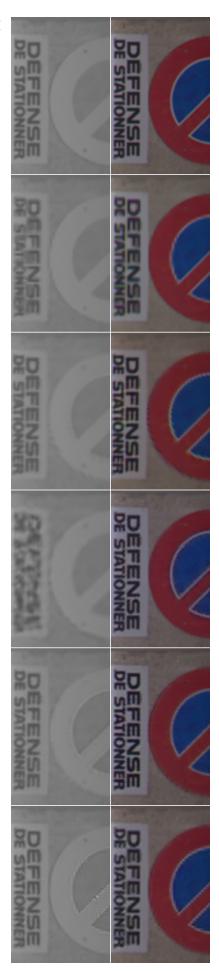
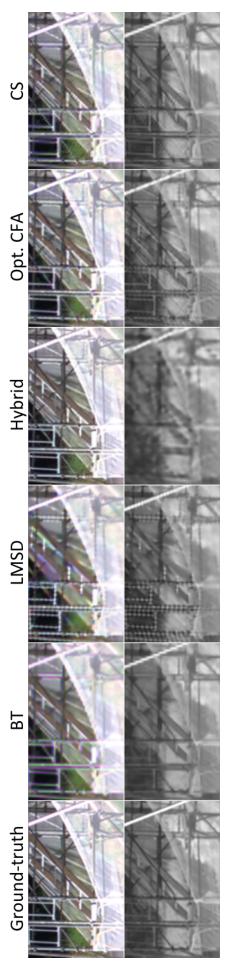
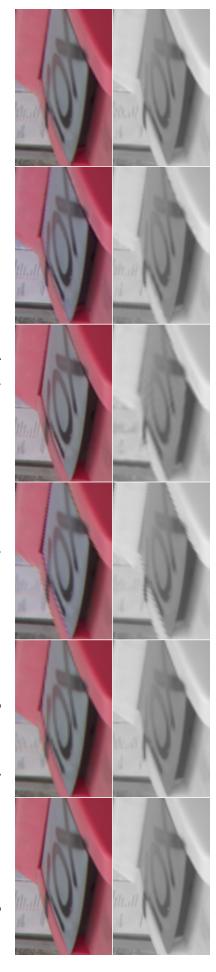


image. (b) The results of BT and LMSD suffer from ringing artifacts and false colors. Despite producing a high-quality color image, the hybrid technique fails to preserve the details in the NIR

Figure 4.8: From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7.

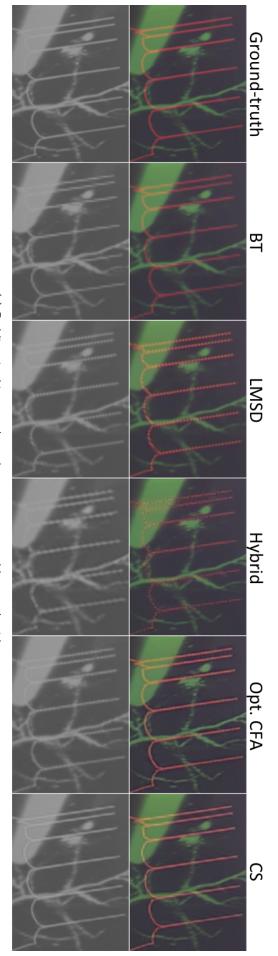


(c) The BT technique results in blurred images suffering from false colors. LMSD is unable to reconstruct horizontal and vertical edges, and the hybrid method fails to reconstruct the NIR image. The results obtained by both our algorithms achieve a better compromise between color and NIR quality.

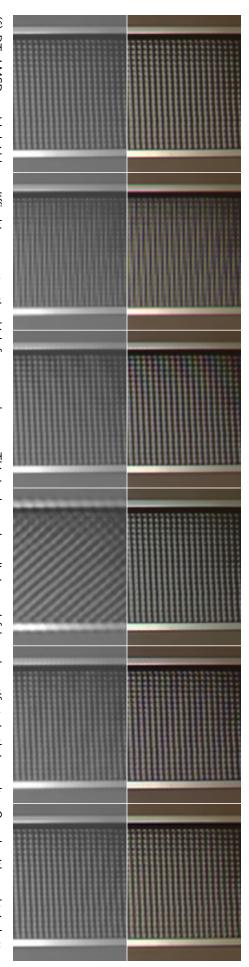


(d) Our results contain less ringing artifacts compared with other methods.

Figure 4.8: From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7 (cont.).



(e) Red lines in this example are better reconstructed by our algorithms.



(f) BT, LMSD, and hybrid have difficulties reconstructing high-frequency changes. This is observed as aliasing and false color artifacts in their results. Our algorithms obtain better

Figure 4.8: From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7 (cont.).

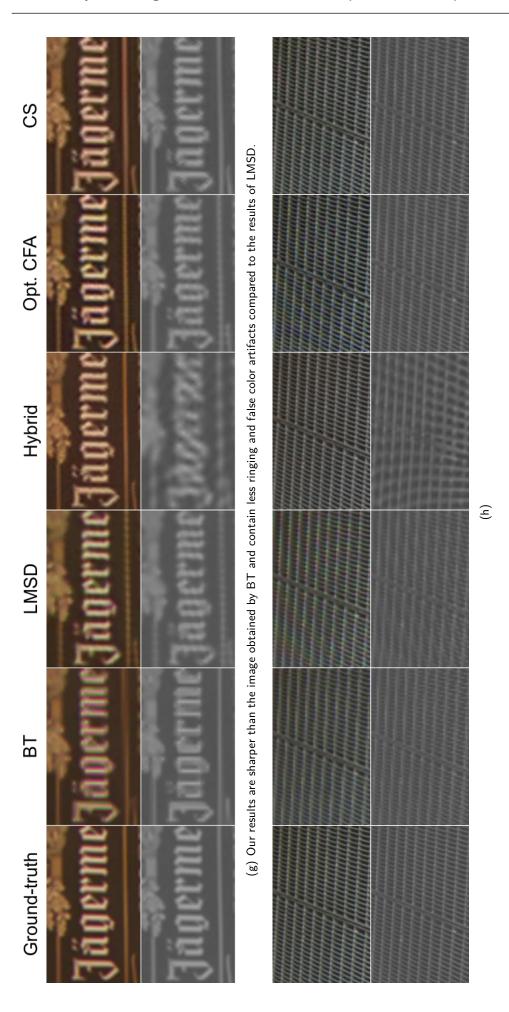


Figure 4.8: From left to right: Ground-truth color and NIR images, the results of five different single-sensor multispectral acquisition techniques: BT, LMSD, hybrid, our optimum CFA method, and our CS algorithm. The CFAs used in different techniques are shown in Figure 4.7 (cont.).

#### 4.5.4 Discussion

The CFA proposed in the LMSD technique [4] samples different channels on diagonal stripes (Figure 4.7-(b)). Although the overall sampling frequency of different channels is still 1/4, the horizontal and vertical sampling frequencies are decreased in this CFA compared with other CFA patterns shown in Figure 4.7. Hence, sampling with this CFA leads to a poor reconstruction of vertical and horizontal edges as shown in Figure 4.8-(c). According to the oblique effect [11], the ability of a human observer to resolve details (hence artifacts) is highest at vertical and horizontal directions. As a result, although the LMSD technique performs quite well in terms of PSNR, its VIF results, which measure the quality of image as perceived by the human eye, are lower compared with other techniques.

The hybrid CFA, Figure 4.7-(c), samples the NIR radiation with the rate of 1/10. As a result, using this CFA leads to high-quality reconstruction of color images only. As shown by all metrics and visual comparisons, the quality of the NIR images demosaiced by this method is lower compared with all other techniques.

Our RGBN CFA (Figure 4.7-(e)) and the CFA designed by the binary tree algorithm [102] are the most similar patterns to the Bayer CFA placed in most current cameras. The binary tree algorithm, however, does not model the leaking of visible and NIR radiations into NIR and color dye filters. This method assumes each filter transmits light only in the corresponding band. Our separation algorithm is the only one, among the techniques analyzed in this section, that models this difficulty in using dye filters. Despite simulating very simple hardware components, the performance of our algorithm is comparable to, or better than, other methods.

### 4.6 Summary

We studied the use of a simple RGBN CFA for the joint acquisition of color and NIR images. The main challenge in sampling the scene with this CFA is that dye filters transmit a mixture of color and NIR channels, as implementing dye filters with sharp cutoffs is extremely challenging if not impossible. To solve this issue, we proposed an algorithm that, relying on the spatial and spectral correlations of color and NIR images, separates the mixed measurements of sensor. We explored the correlation through sparsifying transforms and sparse decomposition. We presented this algorithm in [126].

Although the basic principles of our algorithm are similar to the conventional framework of compressive sensing, there are two main differences. First, limited by the camera hardware, our measurement matrix is sparse and highly structured, whereas in many applications of compressive sensing a dense random sampling is used. Second, we proposed to use two different dictionaries and fuse the reconstructed images. We showed that this is essential to preserve the variations in spectral correlation of color and NIR images. This idea might be extended to us-

4.6. Summary 97

ing multiple dictionaries, each trained to represent a different level of correlation between color and NIR high-frequency details.

In the final section, we analyzed three single-sensor acquisition techniques and compared their performance with our two algorithms based on the optimized CFA and the RGBN CFA. Confirmed by different image quality metrics, as well as visual comparisons, our optimum CFA method performs the best among these techniques.

The RGBN CFA and our reconstruction algorithm produce high-quality images comparable to the results of other methods. This technique offers the most straightforward path towards the mass production of color and NIR cameras, as it requires the least amount of modifications in the hardware of current color cameras.

This chapter concludes the first part of the thesis. In the next two chapters, we study the problem of chromatic aberration in color and NIR joint imaging.

## Chapter 5

# Chromatic Aberration in Color and NIR Joint Acquisition

In this chapter, we study the problem of chromatic aberration (CA) in the joint acquisition of color and NIR images. Chromatic aberration distortions can be reduced by using compound lenses made of multiple lens elements. Figure 5.1-(a) shows the approximate focus errors of a simple lens and some compound lenses. To illustrate, an achromatic doublet lens, shown in Figure 5.1-(b), is a combination of a convergent element of crown glass and a divergent lens of flint glass [8]. There exist professional lenses corrected for both the visible and NIR bands of the spectrum<sup>1</sup>. However, such lenses are very expensive and too bulky to be used with most consumer cameras, especially those integrated inside small devices such as cellphones. Therefore, we are interested mainly in digital correction of CA after acquisition.

We consider the scenario where the color image is in focus and sharp. In this case the NIR image captured with the same lens and same focus settings is blurred as a result of axial chromatic aberration. Hence, we formulate the task of reducing chromatic aberration as deblurring the NIR image. The algorithms presented in this chapter and the following chapter can be simply modified to reduce chromatic aberration if the NIR image is in focus and the color image is blurred.

In the following sections, we present a deblurring algorithm that reduces chromatic aberration distortions, and compare its performance with the general approach for CA correction in color imaging. We also compare our method with deblurring algorithms that use a single image or multiple images.

<sup>&</sup>lt;sup>1</sup>Example of a commercial product: http://www.jenoptik-inc.com/coastalopt-standard-lenses/uv-vis-nir-60mm-slr-lens-mainmenu-155/80-uv-vis-ir-60-mm-apo-macro.html

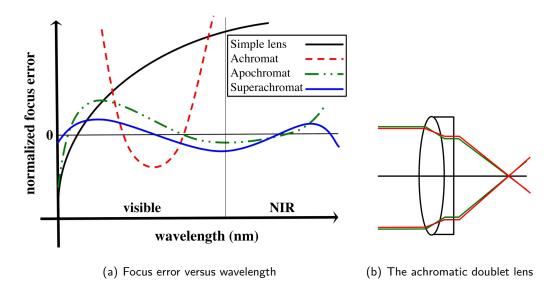


Figure 5.1: (a) The focus error of simple and compound lenses versus wavelength (re-drawn based on data from [1]). (b) The achromatic doublet lens is a combination of convergent crown glass and divergent flint glass elements.

#### 5.1 Guided Deblurring

We mathematically formulate the blurring of the NIR image as follows:

$$\mathcal{N}_b = k * \mathcal{N},\tag{5.1}$$

where  $\mathcal{N}$  is the underlying sharp image to be estimated, k is the kernel point spread function (PSF) that characterizes the out-of-focus lens blur,  $\mathcal{N}_b$  is the NIR image blurred because of chromatic aberration, and \* denotes the convolution operation.

In the following subsections, we explain our deblurring algorithm that first estimates the blur kernel. The algorithm then deblurs  $\mathcal{N}_b$  by exploiting the high-frequency details of the color image.

#### 5.1.1 Estimating the Blur Kernel

Similarly to previous chapters, we denote the luminance channel of the color image as  $\mathbf{Y}$ . By solving the following optimization problem, we estimate the blur kernel k:

$$k^{\star} = \underset{k}{\operatorname{argmin}} \|\nabla_{x} \mathcal{N}_{b} - \nabla_{x}(k * \mathbf{Y})\|_{F}^{2} + \|\nabla_{y} \mathcal{N}_{b} - \nabla_{y}(k * \mathbf{Y})\|_{F}^{2} + \Pr(k), \qquad (5.2)$$

where  $\nabla_x$ ,  $\nabla_y$ , respectively, represent the horizontal and vertical gradient operators. This optimization exploits the fact that, as opposed to absolute intensities, the high-frequency details of NIR and color channels are usually correlated. For a detailed

analysis of the correlation between these channels, please see Chapter 3. As mentioned in previous chapters, color and NIR images do not share similar edges in all patches, nevertheless, our experiments show that if there are some correlated edges in the image pair, the kernel estimated by solving the above problem is reliable.

 $\Pr(k)$  in (5.2) represents the prior information about the kernel. The PSF of a lens is usually modeled as a Gaussian filter [143, 65]. Hence, we assume  $k_{\sigma} = \exp\left(-\frac{(m-m_0)^2+(n-n_0)^2}{2\sigma^2}\right)$ , where  $\sigma$  fully characterizes the kernel and is a measure of the filter's spread.  $m_0, n_0$  indicate the center location of the kernel. In this chapter, we assume that the blur kernel is spatially invariant. In this case, estimating the blur kernel is equivalent to finding one  $\sigma$  value:

$$\sigma^* = \operatorname{argmin} \|\nabla_x \mathcal{N}_b - \nabla_x (k_\sigma * \mathbf{Y})\|_F^2 + \|\nabla_y \mathcal{N}_b - \nabla_y (k_\sigma * \mathbf{Y})\|_F^2$$
s.t.  $k_\sigma(m, n) = \frac{1}{c} \exp\left(-\frac{(m - m_0)^2 + (n - n_0)^2}{2\sigma^2}\right),$ 
(5.3)

where (m, n) are horizontal and vertical coordinates, and c is a normalization factor ensuring that  $\sum_{i} (k_{\sigma})_{i} = 1$ .

#### 5.1.2 Deblurring the NIR Image

We formulate the NIR deblurring as solving the following problem:

$$\mathcal{N}^{\star} = \underset{\mathcal{N}}{\operatorname{argmin}} \|\mathcal{N}_b - k_{\sigma} * \mathcal{N}\|_F^2 + \lambda \Pr(\mathcal{N}), \tag{5.4}$$

where  $k_{\sigma}$  is the blur kernel estimated in the first step (Subsection 5.1.1) and  $\lambda$  is the regularization parameter.

Image deblurring is an ill-posed problem, hence including a regularization term  $(Pr(\mathcal{N}))$  to constrain the solution is necessary. Different prior terms are proposed in the deblurring literature. The most popular regularization is the sparsity of gradients in natural images [14, 80, 66].

In our scenario, the sharp color representation of the scene is available. Thus, instead of using a general distribution of natural images, we propose applying the following regularization term that constrains the gradients of the deblurred NIR image to be similar to those of the sharp color image:

$$\Pr(\mathcal{N}) = \|\nabla_x \mathcal{N} - \mathbf{M}_x \odot \nabla_x \mathbf{Y}\|_F^2 + \|\nabla_y \mathcal{N} - \mathbf{M}_y \odot \nabla_y \mathbf{Y}\|_F^2.$$
 (5.5)

Here,  $\mathbf{M}_x$  and  $\mathbf{M}_y$  are the similarity maps with the same size as the NIR and color images, and  $\odot$  stands for the element-wise multiplication of two matrices. The similarity maps represent the correlation between the gradients of  $\mathcal{N}$  and  $\mathbf{Y}$  channels at each pixel. Therefore,  $\mathbf{M}_x$  has large values at pixels where  $\mathcal{N}$  and  $\mathbf{Y}$  horizontal gradients are highly correlated. Whereas, if the horizontal gradients are not similar, the corresponding component in  $\mathbf{M}_x$  is small.  $\mathbf{M}_y$  has similar properties when the

gradient is computed in the vertical direction. We incorporate the similarity maps to ensure that the edges of the  $\mathbf{Y}$  image contribute to the deblurring result only where gradients of  $\mathcal{N}$  and  $\mathbf{Y}$  are strongly correlated.

The similarity maps should measure the similarity between the gradients of  $\mathcal{N}$  and  $\mathbf{Y}$  channels. If the sharp NIR image is accessible, the similarity levels can be computed as follows:

$$\mathbf{M}_{x}^{0}(m,n) = 1 - \frac{|\nabla_{x}\mathcal{N}(m,n) - \nabla_{x}\mathbf{Y}(m,n)|}{|\nabla_{x}\mathcal{N}(m,n) + \nabla_{x}\mathbf{Y}(m,n)|},$$

$$\mathbf{M}_{y}^{0}(m,n) = 1 - \frac{|\nabla_{y}\mathcal{N}(m,n) - \nabla_{y}\mathbf{Y}(m,n)|}{|\nabla_{y}\mathcal{N}(m,n) + \nabla_{y}\mathbf{Y}(m,n)|},$$
(5.6)

where (m, n) indicate the horizontal and vertical coordinates of the pixel. Note that as the similarity between the gradients of  $\mathcal{N}$  and  $\mathbf{Y}$  channels increases, the values of  $\mathbf{M}_{x}^{0}(m, n)$  and  $\mathbf{M}_{y}^{0}(m, n)$  increase.

In the deblurring problem, we do not have access to the sharp NIR image in advance. Hence, using (5.6) to compute the similarity maps is not possible. In addition, deblurring fails if we directly compare the gradients of the blurred NIR and the sharp Y channel. The reason is that even the edges that are inherently similar between the color and NIR representations, have different profiles as they are blurred in NIR and sharp in the Y channel. To address this difficulty, we first blur the Y channel with the estimated blur kernel:

$$\mathbf{Y}_b = k_{\sigma} * \mathbf{Y},\tag{5.7}$$

and calculate the similarity maps as

$$\mathbf{M}_{x}(m,n) = 1 - \frac{|\nabla_{x}\mathcal{N}_{b}(m,n) - \nabla_{x}\mathbf{Y}_{b}(m,n)|}{|\nabla_{x}\mathcal{N}_{b}(m,n) + \nabla_{x}\mathbf{Y}_{b}(m,n)|},$$

$$\mathbf{M}_{y}(m,n) = 1 - \frac{|\nabla_{y}\mathcal{N}_{b}(m,n) - \nabla_{y}\mathbf{Y}_{b}(m,n)|}{|\nabla_{y}\mathcal{N}_{b}(m,n) + \nabla_{y}\mathbf{Y}_{b}(m,n)|}.$$
(5.8)

Comparing the gradients of blurred Y and NIR channels ensures large components for  $\mathbf{M}_x$  and  $\mathbf{M}_y$  when the edges are similar. Additionally, if in one neighborhood color and NIR edges are fundamentally uncorrelated, they are likely to look different even after blurring (unless the NIR image is severely blurred).

We discussed the significance of using similarity maps ( $\mathbf{M}_x$  and  $\mathbf{M}_y$ ) before. Here, we illustrate this effect with one example. Figure 5.2 shows a pair of blurred NIR and sharp color images. Subfigure (c) is the NIR image deblurred without using similarity maps. It can be immediately observed that false edges (the pattern zoomed-in in the second row) are introduced into the deblurred NIR image. However, when we incorporate the similarity maps into deblurring (Figure 5.2-d), the inherent differences between the color image and its NIR counterpart are preserved (the differences are more noticeable when this figure is viewed on a screen).

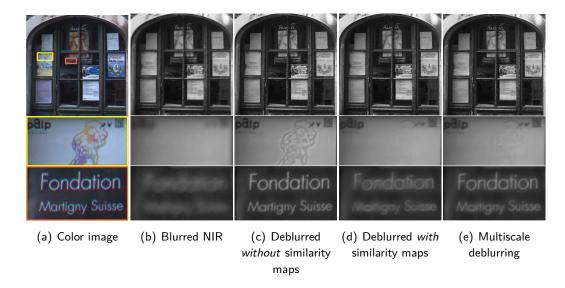


Figure 5.2: (a) Sharp color image. (b) Blurred NIR image. Second row: (c) the result of deblurring guided by the color image without using similarity maps contains false edges that are not present in the NIR image. (d) By using the similarity maps, the algorithm preserves the differences between color and NIR images. Third row: As the similarity maps provide a rough estimation of correlation between gradients, the image in (d) is not as sharp as the image (c). (e) Using the multiscale approach presented in Subsection 5.1.3, the deblurring results in a sharper image while the differences between color and NIR are kept.

The third row of Figure 5.2 shows patches where color and NIR images share similar high-frequency details. The image deblurred using similarity maps in this patch is not as sharp as the one estimated without using any similarity maps. The reason for this is that the maps are computed from blurred  $\mathcal{N}$  and  $\mathbf{Y}$  images, hence they only provide a rough estimation of actual similarities between NIR and color gradients. In the next subsection, we present a multiscale deblurring framework that results in more accurate estimations of similarity maps and sharper images. Figure 5.2-(e) is the result of our multiscale approach.

#### 5.1.3 Multiscale Deblurring

We first form pyramids of  $\mathbf{Y}$  and  $\mathcal{N}_b$  (blurred NIR) images with p+1 scales. The images in each scale are downsampled by a factor of R to form the images of the next coarser level. The full-resolution images are called  $\mathcal{N}_b^{(0)}$  and  $\mathbf{Y}^{(0)}$ , and the coarsest scale is denoted by  $\mathcal{N}_b^{(p)}$  and  $\mathbf{Y}^{(p)}$ . In the following, we use  $(.)_{\downarrow R}$  to represent downsampling by a factor of R, and upsampling is shown by  $(.)_{\uparrow R}$ . Figure 5.3 demonstrates one example image pyramid, and the notations used for the images at

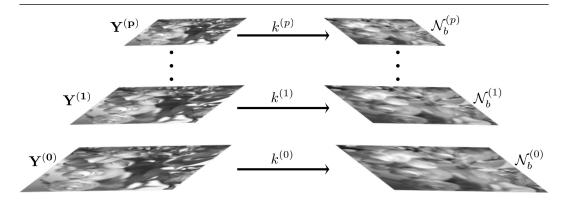


Figure 5.3: A pyramid of  $\mathcal{N}$  and  $\mathbf{Y}$  images with p+1 scales. The kernel in each scale is obtained by downsampling the kernel in the previous finer scale.

each scale.

To estimate the kernel, we use an optimization problem similar to (5.3). The blur kernel in the finest scale is obtained by solving the following optimization problem that computes the variance  $(\sigma)$  of the Gaussian filter:

$$k^{(0)} = f(\mathcal{N}_b^{(0)}, \mathbf{Y}^{(0)}) \triangleq \underset{k}{\operatorname{argmin}} \|\nabla_x \mathcal{N}_b^{(0)} - \nabla_x (k * \mathbf{Y}^{(0)})\|_F^2 + \|\nabla_y \mathcal{N}_b^{(0)} - \nabla_y (k * \mathbf{Y}^{(0)})\|_F^2$$
s.t. 
$$k(m, n) = \frac{1}{c} \exp\left(-\frac{(m - m_0)^2 + (n - n_0)^2}{2\sigma^2}\right),$$
(5.9)

We start the blur kernel estimation from the finest scale of the pyramid because the full-resolution Y and NIR images exhibit larger differences compared with the downsampled image pairs in coarser scales. This results in a more accurate estimation of kernel. After estimating  $k^{(0)}$ , the blur kernels in coarser levels are sequentially computed by downsampling the kernel in the previous finer scale with factor R, i.e.,

$$k^{(1)} = (k^{(0)})_{\downarrow R}, \quad k^{(2)} = (k^{(1)})_{\downarrow R}, \cdots, k^{(p)} = (k^{(p-1)})_{\downarrow R}.$$
 (5.10)

These blur kernels are fixed and are not updated in the following steps of the algorithm.

After estimating the blur kernels, starting from the coarsest scale, we deblur the NIR image in every scale of the pyramid until the full resolution NIR is deblurred. To deblur the coarsest NIR image,  $\mathcal{N}_{b}^{(p)}$ , we solve the following problem:

$$\mathcal{N}_{d}^{(p)} = g(\mathcal{N}_{b}^{(p)}, \mathbf{Y}^{(p)}, \mathbf{M}_{x}^{(p)}, \mathbf{M}_{y}^{(p)})$$

$$\triangleq \underset{\mathcal{N}^{(p)}}{\operatorname{argmin}} \|\mathcal{N}_{b}^{(p)} - k^{(p)} * \mathcal{N}^{(p)}\|_{F}^{2}$$

$$+ \lambda \|\nabla_{x}\mathcal{N}^{(p)} - \mathbf{M}_{x}^{(p)} \odot \nabla_{x}\mathbf{Y}^{(p)}\|_{F}^{2}$$

$$+ \lambda \|\nabla_{y}\mathcal{N}^{(p)} - \mathbf{M}_{y}^{(p)} \odot \nabla_{y}\mathbf{Y}^{(p)}\|_{F}^{2},$$
(5.11)

 $\mathbf{M}_{x}^{(p)}$  and  $\mathbf{M}_{y}^{(p)}$  are, respectively, horizontal and vertical similarity maps in scale p. As before, the second and third terms in the cost function of (5.11) constrain the gradients of the deblurred NIR image to be similar to those of  $\mathbf{Y}^{(p)}$  only where similarity maps have large values.

Only for the coarsest scale of the pyramid, do we compute the similarity maps, as shown in (5.8), by comparing the gradients of blurred NIR and blurred Y images:

$$\mathbf{M}_{x}^{(p)}(m,n) = 1 - \frac{|\nabla_{x}\mathcal{N}_{b}^{(p)}(m,n) - \nabla_{x}\mathbf{Y}_{b}^{(p)}(m,n)|}{|\nabla_{x}\mathcal{N}_{b}^{(p)}(m,n) + \nabla_{x}\mathbf{Y}_{b}^{(p)}(m,n)|},$$

$$\mathbf{M}_{y}^{(p)}(m,n) = 1 - \frac{|\nabla_{y}\mathcal{N}_{b}^{(p)}(m,n) - \nabla_{y}\mathbf{Y}_{b}^{(p)}(m,n)|}{|\nabla_{y}\mathcal{N}_{b}^{(p)}(m,n) + \nabla_{y}\mathbf{Y}_{b}^{(p)}(m,n)|}.$$
(5.12)

Here  $\mathbf{Y}_{b}^{(p)} = k^{(p)} * \mathbf{Y}^{(p)}$ . Figure 5.4-(a) illustrates deblurring the NIR image in the coarsest scale of the pyramid.

Using the gradients of blurred NIR and blurred Y images in (5.12) provides only a rough approximation of similarity levels between the gradients of sharp color and NIR images. As a result, the output image of (5.11),  $\mathcal{N}_d^{(p)}$ , is still blurred compared with the underlying sharp NIR image,  $\mathcal{N}^{(p)}$ . The residual blur in  $\mathcal{N}_d^{(p)}$  can be formulated as

$$\mathcal{N}_d^{(p)} = k_{\text{res}}^{(p)} * \mathcal{N}^{(p)}, \tag{5.13}$$

where  $k_{\text{res}}^{(p)}$  is the residual kernel after deblurring. We estimate the residual kernel by solving

$$k_{\text{res}}^{(p)} = f(\mathcal{N}_d^{(p)}, \mathbf{Y}^{(p)}). \tag{5.14}$$

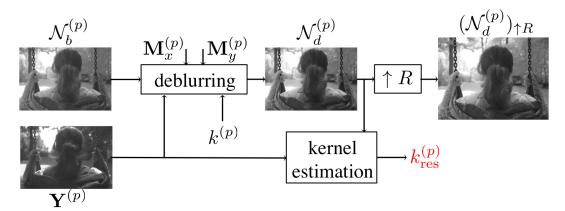
f(.,.) is defined in (5.9). See Figure 5.4-(a) for an illustration. To deblur the NIR image in the next scale,  $\mathcal{N}_b^{(p-1)}$ , we need to compute the corresponding similarity maps,  $\mathbf{M}_x^{(p-1)}$  and  $\mathbf{M}_y^{(p-1)}$ . If, similarly to the previous scale, we use the blurred NIR image, inaccurate estimations of similarity maps are obtained. Instead of using blurred images, we propose to use an upsampled version of the image deblurred in the previous scale,  $(\mathcal{N}_d^{(p)})_{\uparrow R}$ . The gradients of this deblurred image are more similar to those of the sharp NIR image at scale p-1. Hence, comparing its gradients with the Y image provides a more accurate estimation of similarity levels between gradients. Thus, we compute  $\mathbf{M}_x^{(p-1)}$  and  $\mathbf{M}_y^{(p-1)}$  as follows:

$$\mathbf{M}_{i}^{(p-1)} = 1 - \frac{|\nabla_{i}(\mathcal{N}_{d}^{(p)})_{\uparrow R} - \nabla_{i}\mathbf{Y}_{b}^{(p-1)}|}{|\nabla_{i}(\mathcal{N}_{d}^{(p)})_{\uparrow R} + \nabla_{i}\mathbf{Y}_{b}^{(p-1)}|}, \quad i \in \{x, y\}.$$
 (5.15)

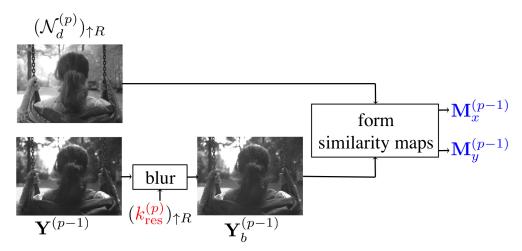
Note that  $\mathcal{N}_d^{(p)}$ , after upsampling by factor R and  $\mathbf{Y}_b^{(p-1)}$ , have the same resolution. Here we omit pixel coordinates for brevity.

Similar to the previous scale,  $\mathbf{Y}_{b}^{(p-1)}$  in (5.15) is the Y image deblurred by the residual blur kernel of  $(\mathcal{N}_d^{(p)})_{\uparrow R}$ . To compute this blur kernel, we upsample  $k_{\text{res}}^{(p)}$ in (5.14) by factor R. So,

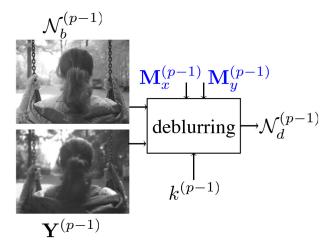
$$\mathbf{Y}_{h}^{(p-1)} = (k_{\text{res}}^{(p)})_{\uparrow R} * \mathbf{Y}^{(p-1)}. \tag{5.16}$$



(a) Deblurring NIR and computing the residual kernel in the coarsest scale.



(b) Forming similarity maps using the NIR image deblurred in the previous coarser scale.



(c) Deblurring the NIR image in scale p-1 using accurate similarity maps.

Figure 5.4: A schematic of our multiscale deblurring algorithm for the first two coarsest scales (p and p-1). The procedure shown in (a) and (b) is repeated for all scales until the full-resolution NIR is deblurred.

Figure 5.4-(b) shows the process of estimating similarity maps in scale p-1.

The similarity maps computed in (5.15) are then used in the following optimization problem to deblur  $\mathcal{N}_{h}^{(p-1)}$  (see Figure 5.4-c):

$$\mathcal{N}_d^{(p-1)} = g(\mathcal{N}_b^{(p-1)}, \mathbf{Y}^{(p-1)}, \mathbf{M}_x^{(p-1)}, \mathbf{M}_y^{(p-1)}). \tag{5.17}$$

Function g(.,.,.) is defined in (5.11).

The deblurring algorithm described above is applied to every scale of the pyramid until  $\mathcal{N}_b^{(0)}$  (the full-resolution NIR) is deblurred to obtain  $\mathcal{N}_d^{(0)}$ , which is the final output of the algorithm. As explained previously, the goal of multiscale processing is to produce accurate similarity maps ( $\mathbf{M}_x$  and  $\mathbf{M}_y$ ). Figure 5.5 illustrates a pair of sharp color and blurred NIR images with two horizontal similarity maps computed by the single-scale and multiscale versions of our algorithm. Darker pixels in similarity maps indicate low similarities between the horizontal gradients of color and NIR images. This figure shows that the similarity maps computed by the multiscale processing better represent the similarities between color and NIR gradients, compared with the maps obtained in a single scale.

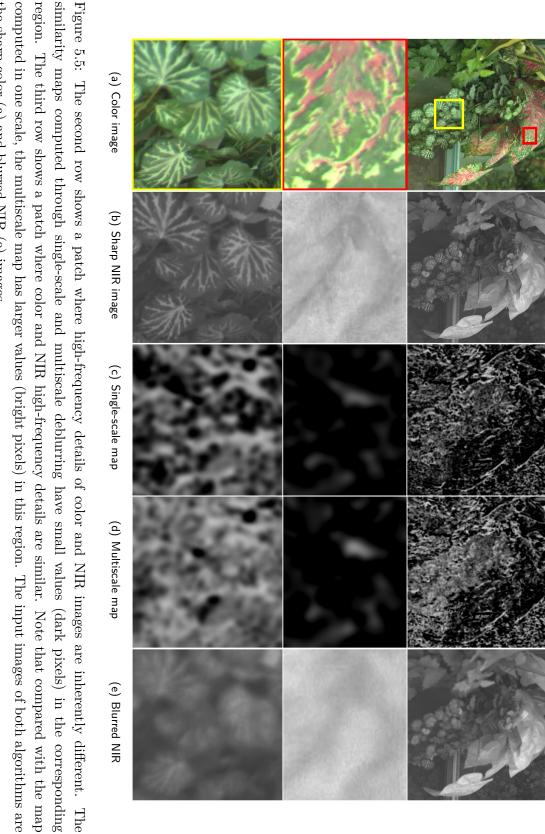
#### 5.2 Experiments

#### 5.2.1 Data Acquisition

All the images used in the following experiments are captured by a Canon Rebel T1i camera, after removing its hot mirror. For each scene, we sequentially captured one color image by placing an NIR-blocking filter in front of the lens and two NIR images using a visible-light blocking filter. The color image is in focus, and the first NIR image, captured with the same lens and same focus settings, is blurred. We then refocused the camera to capture a sharp NIR image, which we use as the visual ground-truth image in the following experiments. For each scene, both NIR images are aligned to the color image. Note that the ideal color-and-NIR camera captures the images in one shot and it is not possible to set the focus differently for color and NIR images. In our experiments, we created this situation by not changing the focus settings between the color shot and the first NIR shot.

#### 5.2.2 Simulation Results

In the following, we first evaluate the performance of our algorithm in estimating the blur kernel. We then present the results of using the single-scale guided deblurring algorithm detailed in Subsection 5.1.2. We compare the performance of our method with the blind deblurring algorithm of Krishnan *et al.* [82] and with the general approach used in color imaging for reducing chromatic aberration. In the final set of experiments, we demonstrate the images deblurred by our multiscale algorithm.



similarity maps computed through single-scale and multiscale deblurring have small values (dark pixels) in the corresponding computed in one scale, the multiscale map has larger values (bright pixels) in this region. The input images of both algorithms are region. The third row shows a patch where color and NIR high-frequency details are similar. Note that compared with the map the sharp color (a) and blurred NIR (e) images.

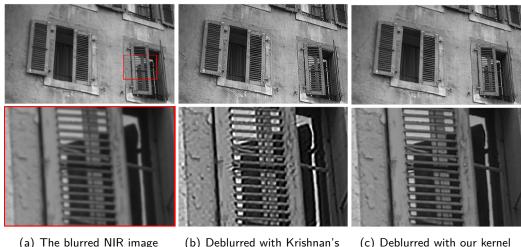
#### **Blur Kernel Estimation**

In a practical scenario, when the color image is in focus and NIR is blurred (out of focus), we do not have access to the ground-truth blur kernel. Therefore, to assess our kernel estimation algorithm, we conducted a synthetic experiment. For this experiment, we used the color and the sharp NIR images of each scene. We blurred the NIR image, as modeled in (5.1), by a Gaussian kernel with a known standard deviation ( $\sigma$ ). The blur-kernel spread ( $\sigma$ ) is estimated by solving the optimization problem (5.3) when the blurred NIR and sharp color images are used. Table 5.1 summarizes ground-truth  $\sigma$  values used to blur the NIR image and the values estimated by our algorithm. The results reported in this table are averaged over 30 pairs of color and NIR images. As can be seen, the estimated values are quite close to the ground-truth. Our experiments show that a small deviation from the true kernel does not greatly affect the deblurring process.

To test the kernel estimation in a more realistic scenario, in the second set of experiments, we use the NIR images blurred by the lens. In this experiment, we compare our blur kernel estimation method with the algorithm of Krishnan et al. [82]. Krishnan et al. propose a blind deblurring method that first iteratively estimates the blur kernel from the blurred image and, at the last step, uses the estimated kernel to deblur the image. Figure 5.6-(b) shows the image deblurred by this method. We also estimate the blur kernel by our algorithm and apply the deblurring step of Krishnan's method to produce the final image using our estimated kernel (Figure 5.6-(c)). As can be seen in Figure 5.6, the image deblurred using our estimated kernel contains fewer artifacts compared with the result of the blind deblurring algorithm of Krishnan et al. [82]. This proves the effectiveness of our algorithm in estimating the blur kernel, as both images are produced by the same deblurring technique and only the estimated blur kernels are different.

Ground-truth $\sigma$		3	4	5	6	7	8	9	10
Estimated $\sigma$	average	2.88	3.87	4.82	5.85	6.81	7.78	8.83	9.55
	std.	0.25	0.32	0.38	0.44	0.61	0.65	0.77	0.77

Table 5.1: Blur kernel estimation: for each ground-truth  $\sigma$  value (number of pixels) that is used to blur the NIR images, we report the  $\sigma$  estimated by using the blurred NIR and sharp color images. The experiment is conducted for 30 pairs of images.



kernel

(c) Deblurred with our kernel

Figure 5.6: (a) The blurred NIR image, (b) the result of the blind deblurring algorithm of Krishnan et al. [82], and (c) the result of the deblurring step of Krishnan's algorithm using our estimated kernel.

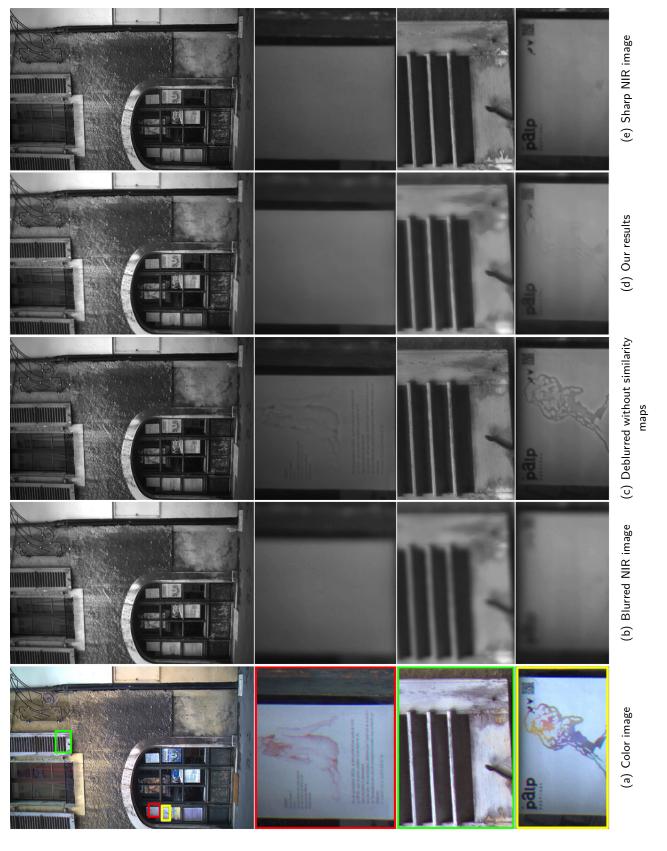
#### Single-Scale Deblurring

We now analyze the performance of our single-scale deblurring algorithm. We first study the importance of using similarity maps in deblurring. In Figure 5.7-(c) and (h), we present the results of simply adding the high-frequency details of the color image into the blurred NIR image. This approach is similar to the chromatic aberration correction algorithms, such as the method of Tisse et al. [144], that assume a strong correlation between the gradients of all color channels. To obtain these results, we solve the following problem that constrains the gradients of the deblurred NIR image to be the same as those of the color image:

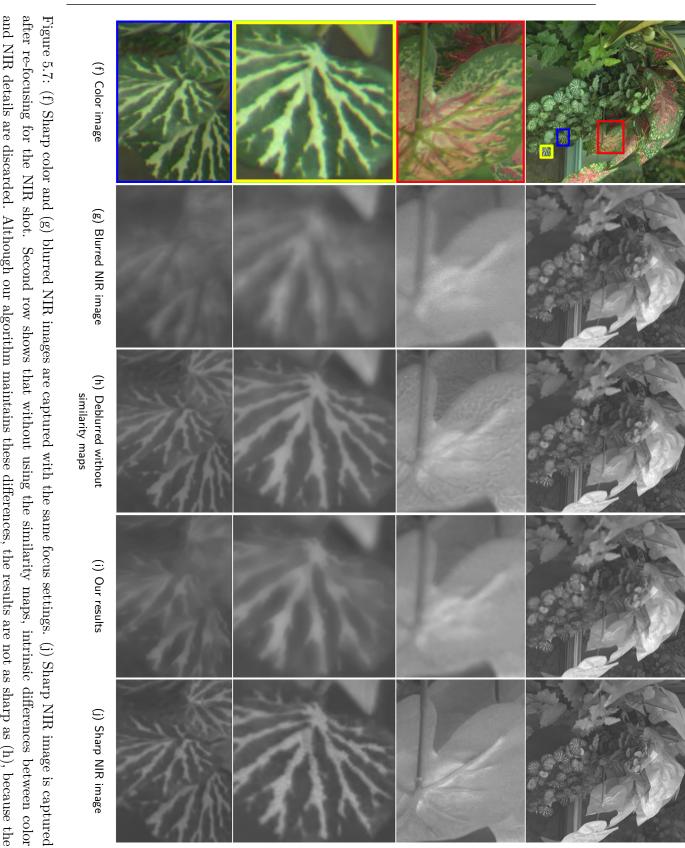
$$\mathcal{N}_d = \underset{\mathcal{N}}{\operatorname{argmin}} \|\mathcal{N}_b - k * \mathcal{N}\|_F^2 + \sum_{i \in \{x,y\}} \|\nabla_i \mathcal{N} - \nabla_i \mathbf{Y}\|_F^2.$$
(5.18)

This method cannot preserve the inherent differences between color and NIR images. For instance, as shown in the second and fourth rows of Figure 5.7, the drawing and the text on the paper are almost invisible in the ground-truth NIR image, however, the image deblurred by (5.18) contains the pattern. Figure 5.7-(d) presents the results obtained by our algorithm that employs similarity maps in deblurring (see (5.4) and (5.5)). We observe that by using the similarity maps the inherent differences between color and NIR images are faithfully preserved.

In the next experiment, we study the performance of the deblurring step when different prior terms for the sharp NIR image,  $Pr(\mathcal{N})$  in (5.4), are explored. Figure 5.8 provides some examples for this comparison. Figure 5.8-(c) shows the deblur-



after re-focusing for the NIR shot. Without using the similarity maps the painting and text on the paper appear in the deblurred Figure 5.7: (a) Sharp color and (b) blurred NIR images are captured with the same focus settings. (e) Sharp NIR image is captured image (second and fourth rows), although they are invisible in the ground-truth NIR image. Our algorithm successfully preserves these inherent differences between color and NIR images, and retrieves the lost details in the NIR image.



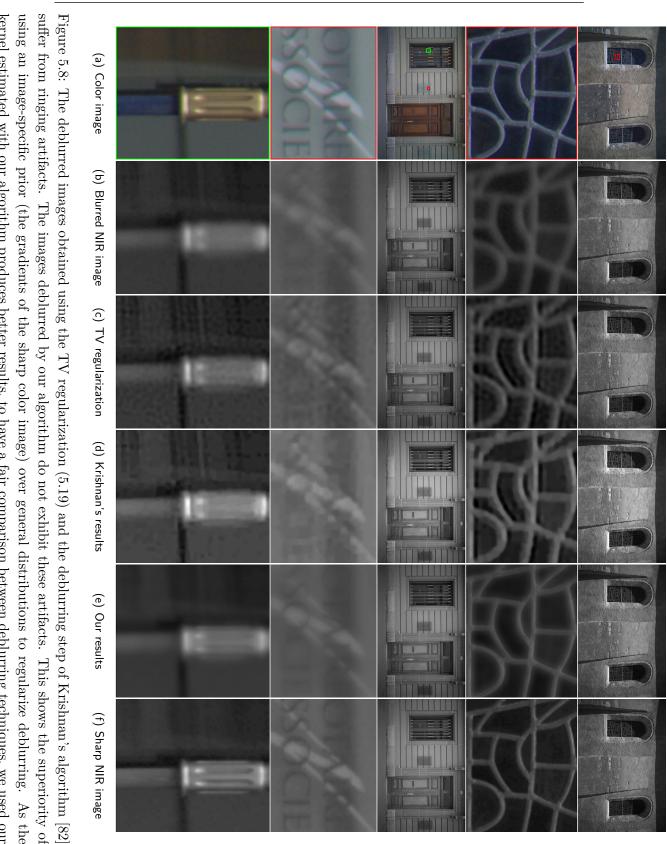
similarity maps are not accurate. In the following, we show how our multiscale approach addresses this challenge. and NIR details are discarded. Although our algorithm maintains these differences, the results are not as sharp as (h), because the after re-focusing for the NIR shot. Second row shows that without using the similarity maps, intrinsic differences between color

ring results when the popular TV (total variation) regularization is used as  $Pr(\mathcal{N})$ . In this case, the following optimization is solved to deblur NIR:

$$\mathcal{N}^{\star} = \underset{\mathcal{N}}{\operatorname{argmin}} \| \mathcal{N}_b - k_{\sigma} * \mathcal{N} \|_F + \lambda \| \nabla \mathcal{N} \|_1.$$
 (5.19)

The deblurring step in the Krishnan *et al.* method [82] is used to produce the images of Figure 5.8-(d). This algorithm exploits a normalized sparsity of image gradients as the prior term. The images of subfigure (e) are the results of our proposed algorithm, when the gradients of the color image are used to constrain the problem. For each image, the blur kernel estimated by our method is used with different deblurring techniques to generate the different results in Figure 5.8.

These results show that using a regularization term specific to each blurred NIR image (the gradients of the sharp color image in our algorithm) is more effective than exploiting general distributions. Our algorithm results in more accurate edges and less noticeable artifacts, compared to other techniques presented in Figure 5.8-(c) and (d).



using an image-specific prior (the gradients of the sharp color image) over general distributions to regularize deblurring. As the suffer from ringing artifacts. The images deblurred by our algorithm do not exhibit these artifacts. This shows the superiority of kernel with all algorithms. The differences are more easily seen when this figure is viewed on a screen. kernel estimated with our algorithm produces better results, to have a fair comparison between deblurring techniques, we used our

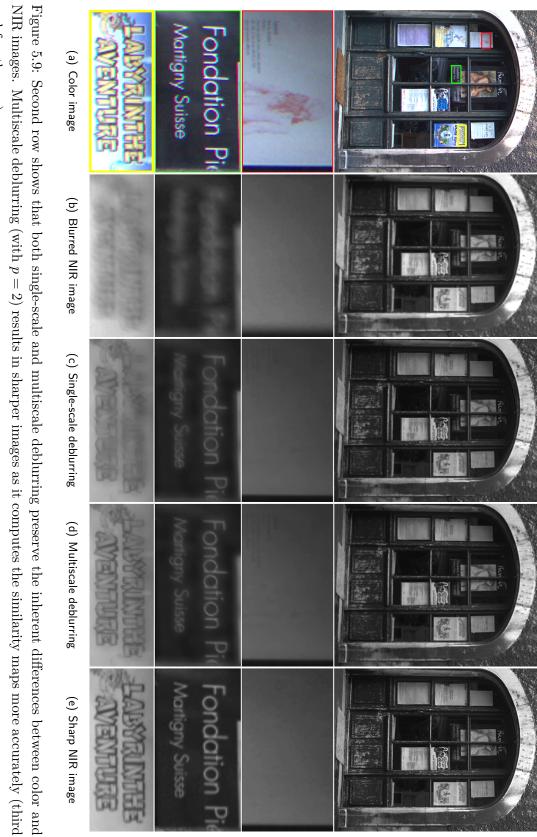
#### Multi-Scale Deblurring

In Figure 5.9 we compare the images deblurred by our single-scale deblurring and its multiscale extension where p=2 (an image pyramid with three scales) is used. It is immediately observed that the accurate similarity maps computed by the multiscale scheme help the algorithm recover sharper details compared with the single-scale deblurring. At the same time, the multiscale algorithm still successfully preserves the inherent differences between the color and NIR representations of the scene.

In Figure 5.10, we study the performance of our algorithm when image pyramids with different number of scales are used in deblurring. As can be seen with the word "Walt" (the second row of the figure), increasing the number of scales results in sharper images. However, the quality of text in the fourth row of Figure 5.10 does not improve beyond two scales. This shows that the optimum number of scales required to obtain accurate similarity maps, depends on the frequency spectrum of the image and also the amount of blur (the size of the blur kernel). If the image is severely blurred and/or contains very fine details, increasing the number of scales improves the deblurring result. We observed that a pyramid with three scales usually results in acceptable deblurred images.

We present the results of the guided image filtering of He  $et\ al.\ [58]$  in Figure 5.11-(c) and (d). For this experiment, we apply the guided filtering algorithm to deblur the NIR image by using the sharp color image as a guide. This algorithm computes each pixel intensity in the output image as a weighted average of neighboring pixels in the guide image. The weights in each neighborhood are computed based on the input image (the blurred NIR in this experiment). We show the results of guided image filtering for two different values of the neighborhood size. As can be seen, for a small value of r (the neighborhood size), the results of guided filtering is blurred compared with the images produced by our multiscale algorithm. On the other hand, increasing r introduces false edges into the NIR image and in the case of r=50, even the pixel intensities of the deblurred NIR image are not similar to the original NIR image. The fifth row of Figure 5.11 shows that although the pen is bright in the NIR image, it is dark in the image produced by the guided image filtering for r=50. Our algorithm produces sharp images while preserving the inherent differences between both the intensities and the gradients of color and NIR images.

116 Chapter 5.



and fourth rows).

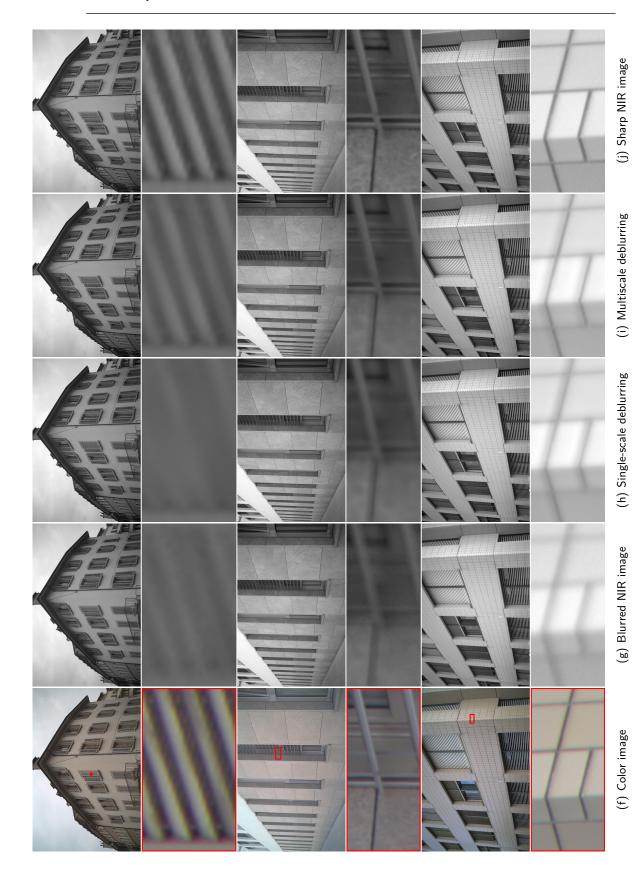
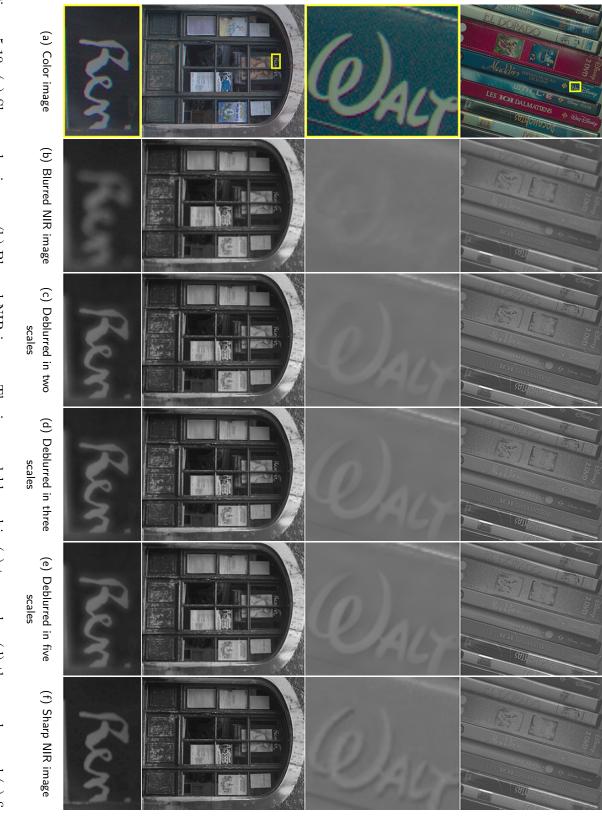
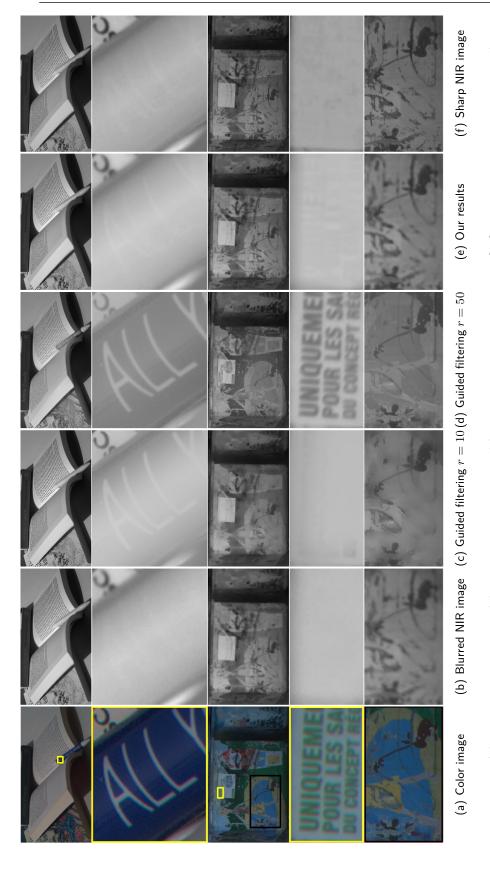


Figure 5.9: Multiscale deblurring (with p = 2) results in sharper images as it computes the similarity maps more accurately (cont.).

118 Chapter 5.



scales. (f) Sharp (ground-truth) NIR image. Increasing the number of scales leads to sharper details. However, the differences are not always noticeable, specifically beyond two scales. Figure 5.10: (a) Sharp color image. (b) Blurred NIR image. The images deblurred in (c) two scales, (d) three scales, and (e) five



r) does not result in sharp details (fifth row). (d) The guided image filtering with r=50 eliminates the differences between the gradients of color and NIR images, and does not even preserve the NIR intensities faithfully. Notice how dark the pen is in the second row of (d), although it is bright in the ground-truth NIR image. (e) Our algorithm produces sharper images while maintaining the Figure 5.11: (a) Sharp color image. (b) Blurred NIR image. (c) The guided image filtering [58] with a small neighborhood (small intrinsic differences between color and NIR. (f) Sharp NIR image.

120 Chapter 5.

# 5.3 Summary

Reducing chromatic aberration distortions in color and NIR acquisition is more challenging compared with color imaging. First, as a wider range of wavelengths is captured in joint acquisition, the distortions are more severe and cannot be avoided by the lenses designed for color imaging. More importantly, the assumption of a strong correlation between details of all channels fails in the case of color and NIR images.

We addressed these challenges by developing a guided algorithm that incorporates similarity maps into deblurring. These maps measure the correlation between color and NIR gradients at every pixel. The contribution of sharp color gradients in reconstructing the NIR image is weighed by the values of similarity maps at each pixel. We presented a multiscale extension of our deblurring algorithm that computes the similarity maps more accurately and results in sharper images. The methods and results explained in this chapter are presented in [127] and [124].

In this chapter, we consider the cases where the amount of blur is uniform across the NIR image, hence we modeled the blur as a spatially invariant kernel. In the following chapter, we address spatially varying chromatic aberration distortions. This phenomenon occurs when differences in objects depths lead to noticeable changes in the amount of blur across the image.

# Chapter 6

# Spatially Varying Chromatic Aberration

In this chapter we study chromatic aberration when the out-of-focus blur kernel varies spatially in both color and NIR images captured with the same lens and focus settings.

In Section 6.1, we show that the amount of out-of-focus blur changes in both color and NIR images as a function of the distance from the camera. More importantly, the amount of blur in the NIR image with respect to the blur in the color image varies depending on the object depth. Our goal is to reconstruct an NIR image that is as sharp as the corresponding color image in different patches.

A commonly encountered situation, for instance in macro or portrait photography, is when the object of interest, usually placed in the foreground, is in focus in the color image and the background is out of focus to create artistic effects and to emphasize the interesting object in the scene. In this case, because of chromatic aberration, the NIR image is severely blurred in the foreground. However, in the background, the blur in the NIR image as compared with the blur in the color image is not as severe. It is even possible that the NIR image is sharper than the color counterpart in the background plane. Figure 6.1 illustrates an example. In Section 6.2, we propose an algorithm that, by comparing the blur clues in color and NIR images, segments the image into foreground and background objects. We then use our guided deblurring algorithm presented in Chapter 5 to deblur the foreground in the NIR image.

In Section 6.3, we extend our foreground-background segmentation algorithm such that it detects several planes in the image where the amount of blur in the NIR image with respect to the blur in the color image is uniform in each plane. We deblur the NIR image in regions where the color image is sharper.

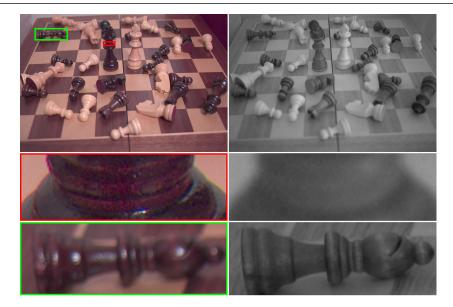


Figure 6.1: A pair of color and NIR images captured with the same focus settings. The color image is focused in the middle of the scene (zoomed-in on the second row), where the NIR image is out of focus. The NIR image is, however, sharp in the background where the color representation is blurred.

# 6.1 Spatially Varying Defocus Blur

In this section, by using the thin lens model, we show how the amount of defocus blur changes in color and NIR images as a function of object distance from the camera.

Let us consider the schematic of Figure 6.2 where 2D is the lens diameter, and the distance between the lens and the sensor is x. Light rays emitted from distance  $d^0$  are converged on the sensor, if

$$\frac{1}{d^0} + \frac{1}{x} = \frac{1}{f},\tag{6.1}$$

where f is the focal length of the lens. If the object is placed closer or further away than the focus plane, the image formed on the sensor is blurred. The amount of blur is characterized by the radius of a disk that is the blurred image of a point light source on the sensor. In Figure 6.2 we denote the radius as r, which is computed as follows:

$$r = D|1 - \frac{x}{u}|\tag{6.2}$$

In the above equation, u is the distance from the lens where the light rays coming from d are focused. Using the thin lens model, we have

$$\frac{1}{u} + \frac{1}{d} = \frac{1}{f},\tag{6.3}$$

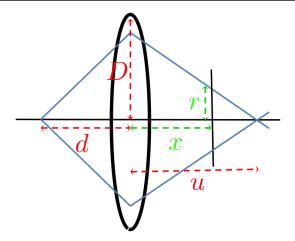


Figure 6.2: 2D is the lens diameter. The sensor is placed at distance x from the lens. A source point at distance d from the lens produces a blurred disk of radius r on the sensor. Light rays emitted from distance d are focused at distance u from the lens.

and,

$$r = D\left|1 - \frac{x}{f} + \frac{x}{d}\right| \tag{6.4}$$

We assume the focus plane for this setting (distance x between the lens and the sensor) is at distance  $d^0$  from the lens. So,

$$r = \begin{cases} D(1 - \frac{x}{f} + \frac{x}{d}) & \text{if } d < d^0 \\ D(-1 + \frac{x}{f} - \frac{x}{d}) & \text{if } d > d^0 \end{cases}$$
 (6.5)

We now study the amount of lens blur in the green<sup>1</sup> and NIR channels for an object placed in distance d. As the refractive index of the lens changes with wavelength, the focal length is wavelength-dependent. We call the focal length for the green channel  $f_{\rm G}$ , and the one for the NIR radiation  $f_{\mathcal{N}}$ . The blur radii in green and NIR images, are

$$r_{\mathcal{N}} = D|1 - \frac{x}{f_{\mathcal{N}}} + \frac{x}{d}|,$$

$$r_{\mathcal{G}} = D|1 - \frac{x}{f_{\mathcal{G}}} + \frac{x}{d}|.$$
(6.6)

Hence, the amount of blur in the NIR image with respect to blur in the green image depends on the object distance:

$$\frac{r_{\mathcal{N}}}{r_{\rm G}} = \frac{|1 - \frac{x}{f_{\mathcal{N}}} + \frac{x}{d}|}{|1 - \frac{x}{f_{\rm G}} + \frac{x}{d}|}.$$
 (6.7)

<sup>&</sup>lt;sup>1</sup>We use green as one example channel in the visible range.

### 6.2 Foreground-Background Segmentation and Deblurring

In this section, we consider a scenario where the foreground object is in focus and sharp in the color image, and blurred in the NIR representation. The background plane is either blurred in both images, or blurred in color and sharp in NIR. The goal is to deblur the NIR image in the regions where the color image is sharper than NIR. To this aim, we present an algorithm that segments the image into foreground (in focus) and background (out of focus) regions by using the blur clues.

#### 6.2.1 Proposed Algorithm

To detect the objects that need to be deblurred in NIR, for each patch we compute a sharpness measure for color and NIR images. Similarly to the previous chapter, we use the **Y** channel of the color image that carries the spatial information.

To compare the sharpness of  $\mathbf{Y}$  and  $\mathcal{N}$  in each patch, we use the measure proposed by Crete *et al.* [33]. Their algorithm is based on the observation that blurring an already blurred image does not lead to significant changes in the gradients of the image. However, blurring a sharp image changes the edges considerably. Therefore, to measure the sharpness level of an image, called  $\mathbf{I}$ , the image is first blurred in both horizontal and vertical directions:

$$\mathbf{I}_b^x = h_1 * \mathbf{I}$$

$$\mathbf{I}_b^y = h_2 * \mathbf{I}$$
(6.8)

 $h_1$  and  $h_2$  are horizontal and vertical lowpass filters. Crete *et al.* suggest using a uniform box filter to blur the image. However, our experiments show that filtering image rows and columns with a Gaussian filter results in a more reliable sharpness measure.

To compare the gradients of the original image and its blurred versions, the following horizontal and vertical gradient images are formed:

$$\nabla_{x}\mathbf{I} = |g_{1} * \mathbf{I}| \quad \nabla_{y}\mathbf{I} = |g_{2} * \mathbf{I}|$$

$$\nabla_{x}\mathbf{I}_{b}^{x} = |g_{1} * \mathbf{I}_{b}^{x}| \quad \nabla_{x}\mathbf{I}_{b}^{y} = |g_{2} * \mathbf{I}_{b}^{y}|$$

$$(6.9)$$

Here  $g_1$  and  $g_2$  are horizontal and vertical gradient filters. In [33], simple derivative filters  $g_1 = [+1, -1]$  and  $g_2 = [+1, -1]^T$  are used. We use the following Sobel operators that are more robust to noise:

$$g_1 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \qquad g_2 = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$$
 (6.10)

In (6.9) the absolute values of gradients are computed, as the gradient direction does not carry information about the sharpness level.

To measure the amount of change in the gradients of the image after blurring, the following variation matrices are formed:

$$\mathbf{V}_{x} = \max(0, \nabla_{x}\mathbf{I} - \nabla_{x}\mathbf{I}_{b}^{x})$$

$$\mathbf{V}_{y} = \max(0, \nabla_{y}\mathbf{I} - \nabla_{y}\mathbf{I}_{b}^{y})$$
(6.11)

Note that the difference is important only where the gradients of the blurred image are weaker than those of the original image  $(\nabla_x \mathbf{I} - \nabla_x \mathbf{I}_b^x > 0)$ . Otherwise the variation value is set to zero. The average magnitudes of entries in variation matrices indicate the level of sharpness:

$$e_{\mathbf{V}}^{x} = \frac{1}{P} \sum_{m,n} \mathbf{V}_{x}(m,n) \qquad e_{\mathbf{V}}^{y} = \frac{1}{P} \sum_{m,n} \mathbf{V}_{y}(m,n), \tag{6.12}$$

P is the number of pixels and (m,n) indicate horizontal and vertical coordinates. To ensure that the blur measure is not affected by the contrast of the original image, average intensities of the image gradients are computed as

$$e_{\mathbf{I}}^{x} = \frac{1}{P} \sum_{m,n} \nabla_{x} \mathbf{I}(m,n) \qquad e_{\mathbf{I}}^{y} = \frac{1}{P} \sum_{m,n} \nabla_{y} \mathbf{I}(m,n),$$
 (6.13)

and used to normalize the blur measure:

$$b^{x} = 1 - \frac{e_{\mathbf{V}}^{x}}{e_{\mathbf{I}}^{x}} \qquad b^{y} = 1 - \frac{e_{\mathbf{V}}^{y}}{e_{\mathbf{I}}^{y}}.$$
 (6.14)

In the above equation,  $b^x$  and  $b^y$  measure the amount of blur in horizontal and vertical directions. If the average magnitudes of variations  $(e_{\mathbf{V}}^x \text{ and } e_{\mathbf{V}}^y)$  with respect to the gradients of the original image are large, which means the image is sharp, the blur measures,  $b^x$  and  $b^y$ , are small. The sharpness of the image is quantified as

$$s = 1 - \max(b^x, b^y). \tag{6.15}$$

We compute the sharpness for all patches in both  $\mathcal{N}$  and  $\mathbf{Y}$  images to obtain two sharpness maps, called  $\mathbf{S}_{\mathcal{N}}$  and  $\mathbf{S}_{\mathbf{Y}}$ , respectively. We, then, form the following difference map:

$$\mathbf{S}_d = \mathbf{S}_{\mathbf{Y}} - \mathbf{S}_{\mathcal{N}},\tag{6.16}$$

which has a positive value for a pixel where the **Y** image is sharper and is negative where  $\mathcal{N}$  is sharper. Uniform patches in the image are not affected by the defocus blur and their sharpness measure might be altered by even an insignificant amount of noise. To handle this, we assign a void value (zero) to the difference map where the patch is uniform in both **Y** and  $\mathcal{N}$ . A patch is assumed to be uniform if the maximum of averages for horizontal and vertical gradients ( $e_{\mathbf{I}}^x$  and  $e_{\mathbf{I}}^y$  in (6.13)) is smaller than a fixed threshold T. We empirically set T = 0.02 when the image intensities are normalized to the range of [0, 1].

To find precise segmentation boundaries, to remove the noise in the difference map, and to assign the uniform patches to either foreground or background, we combine the blur clues presented in the difference map with color clues, similarly to the method of Chakrabarti *et al.* in [23]. Let us call the foreground-background segmentation mask, **U**. This mask is one where the color image is sharper than NIR, and is otherwise zero. We minimize the following energy function to find **U**:

$$E(\mathbf{U}_{m,n}) = \sum_{m,n} \mathbf{B}_{m,n}(\mathbf{U}(m,n)) + \sum_{m,n} \mathbf{C}_{m,n}(\mathbf{U}(m,n)) + \sum_{\substack{(m,n)\\(m',n')}} \delta(\mathbf{U}(m,n) - \mathbf{U}(m',n')),$$
(6.17)

where

$$\delta(i,j) = \begin{cases} 0 & \text{if } i = j \\ 1 & \text{if } i \neq j \end{cases}$$
 (6.18)

The blur clues contribute to the first term of the above energy, where,

$$\mathbf{B}_{m,n}(\mathbf{U}(m,n)) = \begin{cases} -\mathbf{S}_d(m,n) & \text{if } \mathbf{U}(m,n) = 1\\ \mathbf{S}_d(m,n) & \text{if } \mathbf{U}(m,n) = 0 \end{cases}$$
(6.19)

The second term,  $\mathbf{C}_{m,n}(\mathbf{U}(m,n))$ , models the color of foreground and background regions as mixtures of Gaussians [23]. The last term in (6.17) favors smooth segmentation maps and adds a penalty if the labels of neighboring pixels ((m,n) and (m',n')) are different. We minimize  $E(\mathbf{U}_{m,n})$  by the toolbox provided by Chakrabarti *et al.* [23]. This toolbox uses the graph-cut implementation of Vladimir Kolmogorov [16].

After forming the foreground-background (in-focus/out-of-focus) map, we estimate the blur kernel and deblur the NIR image in the foreground region where  $\mathbf{U}$  is one. This is performed by our guided deblurring algorithm presented in Chapter 5.

#### 6.2.2 Results

#### **Foreground-Background Segmentation**

We compare the foreground-background masks obtained by our method with the masks computed by the algorithms of [23] and [138]. In [23], Chakrabarti et al. study the spatially varying blur kernel by analyzing the responses of local patches to a set of windowed Gaussian filters. The output of their algorithm is a mask that segments the image into blurred and sharp regions. They specifically focus on situations where one object in the scene is motion blurred. We modify their algorithm such that it detects the out-of-focus blur<sup>2</sup>. Chakrabarti's algorithm uses only a color image. The method of Shi et al. [138] segments the image into blurred and sharp regions. For this task, they use a combination of gradient features, the properties of the local patch frequency spectrum, and the response to a set of filters trained to discriminate

 $<sup>^2</sup>$ We replace the set of motion blur filters by a set of Gaussian filters.

between blurred and sharp patches. We also present the masks computed by the graph-cut algorithm [16] that refines the segmentation based on the user input. For each image, we provide the algorithm with one line in the foreground and one line in the background. This method does not exploit blur clues, and we include its results to demonstrate how much the segmentation gains from color information in each scene.

Figures 6.3-6.8 show the masks computed by different algorithms. In these figures, we overlay the masks with the color image, so that segmentation boundaries can be simply compared. The pixels that belong to the foreground are represented by their original intensities, and those in the background are shown by gray values. The user input to the grab-cut algorithm is indicated by a red line for the foreground and a green line for the background in the result image.

As shown in Figures 6.3-6.8, our algorithm consistently performs better than other methods. The segmentation results of our method are not, however, perfect. As expected, the segmentation errors usually occur in uniform patches where no information about blur is available. As we show in the following results, our application is not usually concerned with these errors because deblurring does not affect uniform regions.

#### **Deblurring**

We present the results of deblurring the NIR image in the following experiments. To deblur the NIR image, we apply our kernel estimation and deblurring algorithm (explained in Chapter 5) only to the foreground region that is marked one in the foreground-background mask.

For each image, we present four deblurring results obtained using (1) our mask, (2) Chakrabarti's mask [23], (3) Shi's mask [138], and (4) the mask obtained using color clues and user input. The same deblurring algorithm is applied to obtain the results. Figures 6.9-6.14 show the deblurred images.

Although the same deblurring algorithm is used to obtain deblurred images for each scene, as the foreground masks are different, the results are different. In most cases, using our masks leads to sharper images. This is mainly because other masks usually segment a large portion of the background as foreground. This results in an underestimation of the blur-kernel size, which in turn leads to blurry images.



(a) The foreground (inside the yellow box) is in focus in the color image.

(b) The background (marked blue) is sharper in NIR.



(c) Color segmentation with user input. Red line: foreground (d) The result of [138]. A large portion of the background is seeds and green line: background seeds. For this scene, segmen- detected as part of the background. Some regions of the foretation based on color produces the most accurate mask. Note ground (red text on the book) are labeled as background. that this approach requires the user input.

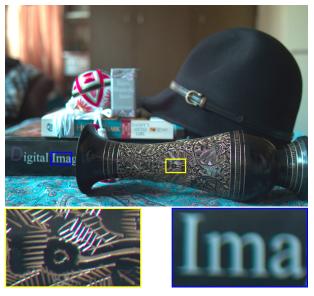


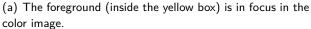
(e) The result of [23]. As the background is highly textured, (f) Our algorithm produces the best result similar to the color-Chakrabarti's algorithm detects it as part of the foreground.

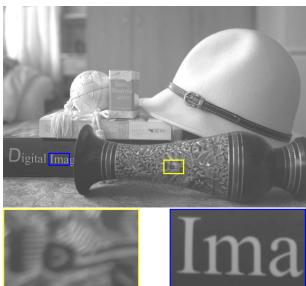


based segmentation in (c). Our method does not use any user input to refine the segmentation.

Figure 6.3: Comparison of foreground-background (in-focus/out-of-focus) masks. The foreground objects (the lens cap and the book) are in focus in the color image. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale.







(b) The background (marked blue) is sharper in NIR.



(c) Color segmentation with user input. Red line: foreground seeds and green line: background seeds. Large regions in the background are detected as foreground because their color (the black book and the black hat) is similar to the foreground object (the black vase).



(d) The result of [138].

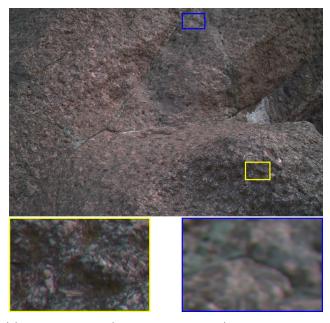


(e) The result of [23].

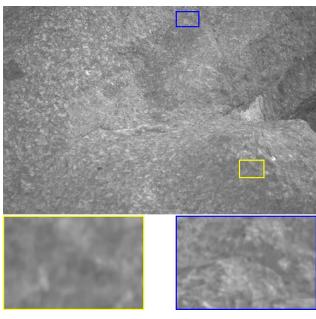


(f) Our result. Part of the hat is detected as foreground by our algorithm, because the corresponding patches are uniform and do not provide any blur information. Moreover, the color of the hat is similar to the foreground object. This does not greatly affect the deblurring result as uniform patches are insensitive to deblurring.

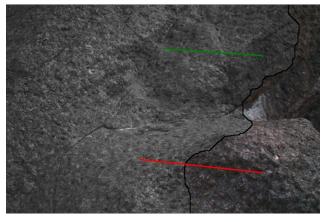
Figure 6.4: Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale.



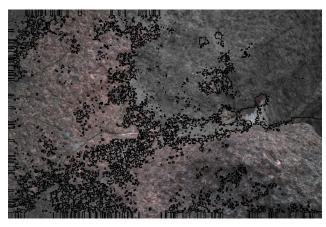
(a) The foreground (inside the yellow box) is in focus in the color image.



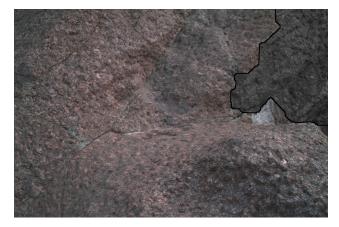
(b) The background (marked blue) is sharper in NIR.



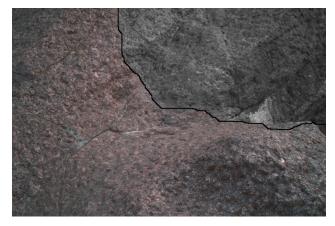
(c) Color segmentation with user input. Red line: foreground seeds and green line: background seeds. The lack of color contrast between foreground and background planes fails the segmentation based on color clues only.



(d) The result of [138]. The background plane is detected correctly, however many small regions of the foreground are labeled as background.

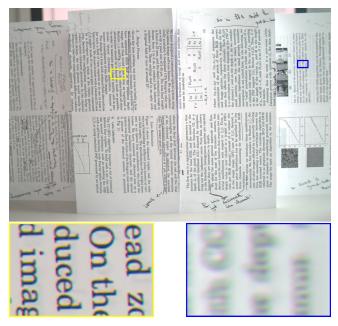


(e) The result of [23]. As shown in (a) and (b), a region of the upper part in this image belongs to background, which is missed in this mask.

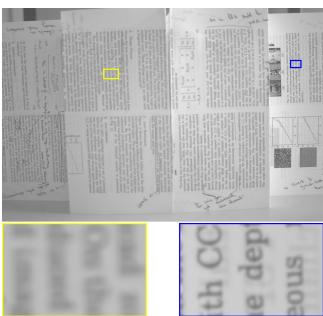


(f) Our method produces the most reliable mask compared with other algorithms.

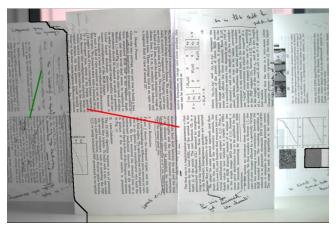
Figure 6.5: Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale.



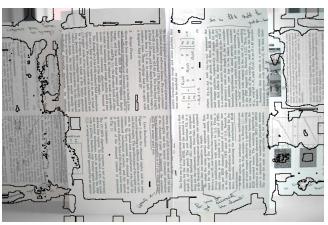
(a) The foreground (inside the yellow box) is in focus in the color image. The central paper-sheets are in the foreground plane. Left and right papers belong to the background plane.



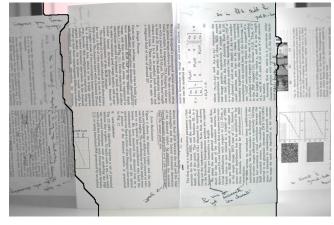
(b) The background (marked blue) is sharper in NIR.



(c) Color segmentation with user input. Red line: foreground seeds and green line: background seeds. As the color of objects in the background and foreground planes is the same, the color-based segmentation fails to label the piece of paper on the right as part of the background.



(d) The result of [138].



(e) The result of [23].

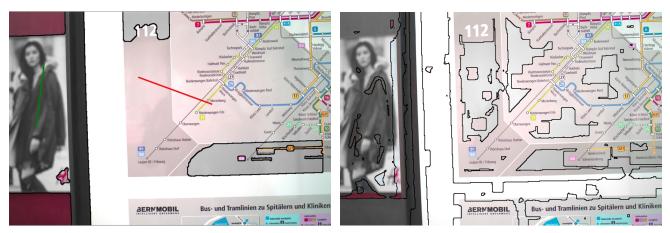


(f) Our result. Our algorithm and Chakrabarti's (e) obtain the most accurate masks for this scene.

Figure 6.6: Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale.



- (a) The foreground (inside the yellow box) is in focus in the color image.
- (b) The background (marked blue) is less blurred in NIR.



(c) Color segmentation with user input. Red line: foreground (d) The result of [138]. Most uniform regions on the foreground seeds and green line: background seeds.

plane are labeled as background.



Figure 6.7: Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and the ones in the background are converted to gray scale.



color image.

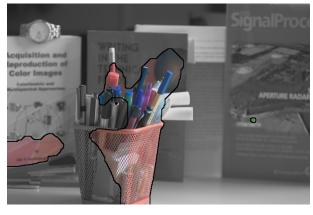
(a) The foreground (inside the yellow box) is in focus in the (b) The background (marked blue) is less blurred in NIR.



(c) Color segmentation with user input. Red line: foreground (d) In the result of [138] many regions are mislabeled as seeds and green line: background seeds. Similarities of ob- foreground or background. jects colors results in the failure of color-based segmentation despite the fact that the user input is provided to this algorithm.

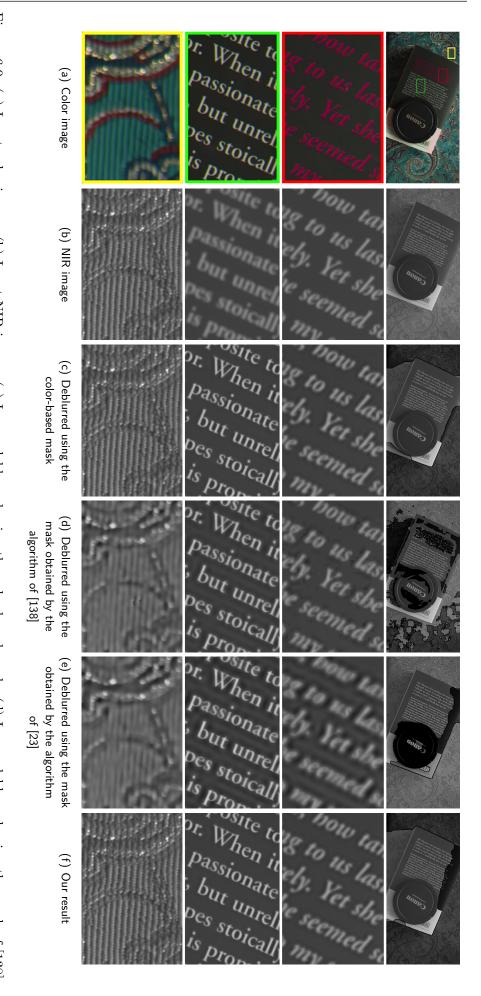


(e) The result of [23]. Although the foreground object is (f) Our result. Our mask is the most accurate one compared detected correctly, several regions of the background are la- with the results of other segmentations. Part of the book on beled as foreground.

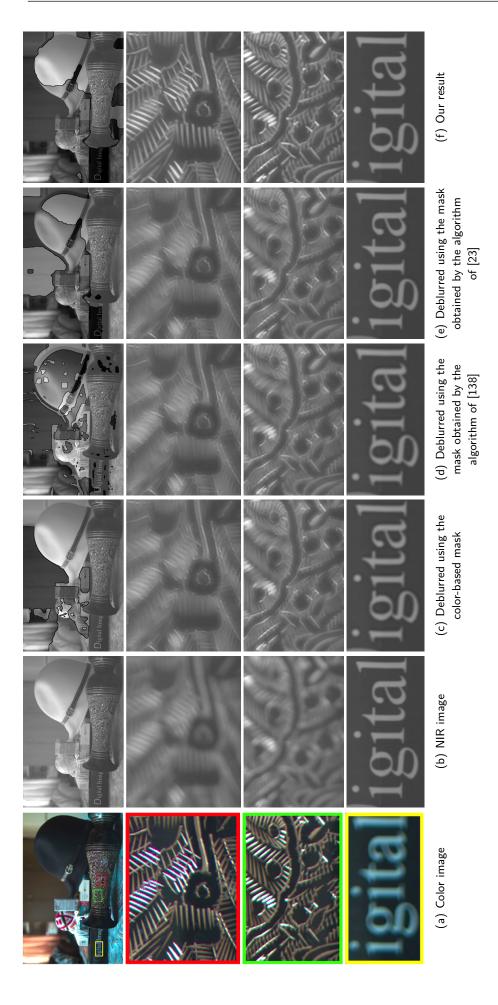


the left is detected as foreground because color and NIR images are inherently different in this region. This mistake does not lead to noticeable errors in the deblurred image as our deblurring algorithm preserves these inherent differences.

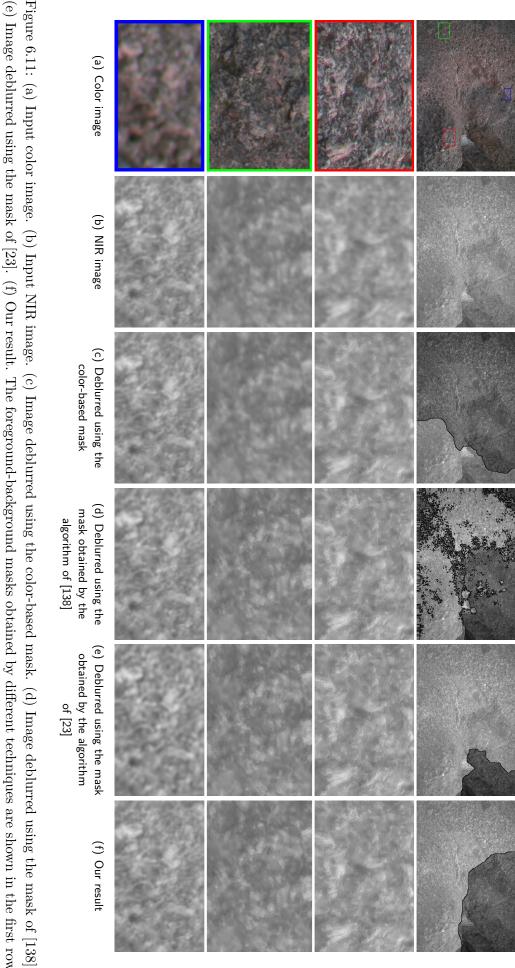
Figure 6.8: Comparison of foreground-background (in-focus/out-of-focus) masks. Pixels detected as foreground (where color is sharper than NIR) are shown by their original color, and those in the background are converted to gray scale.



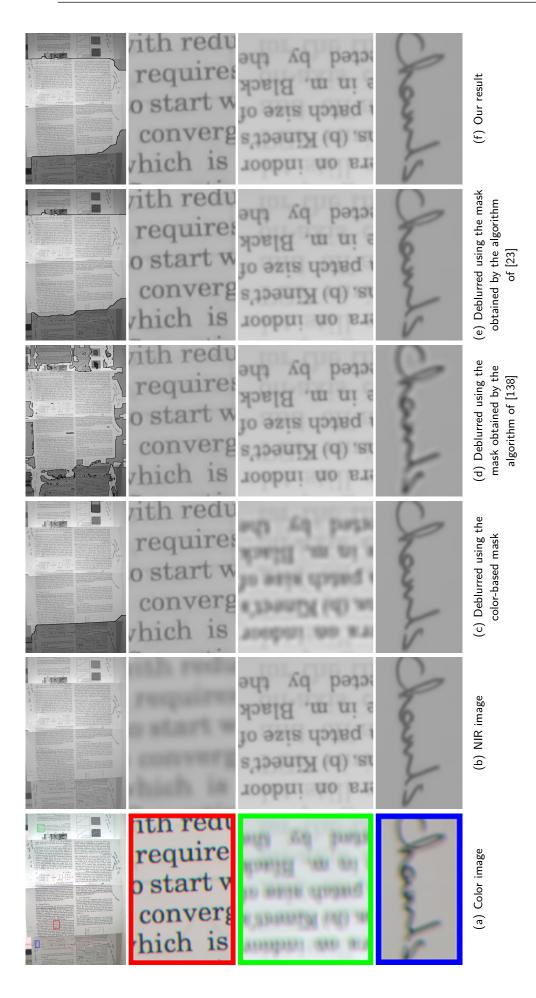
that requires user input produce the best deblurring results. (d) and (e) The results of [138] and [23] are blurred and/or suffer from severe halo artifacts where the color image is sharper. Fourth row: Background patches. (c) and (f) The masks obtained by our algorithm and the color-based segmentation Figure 6.9: (a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138]. (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row



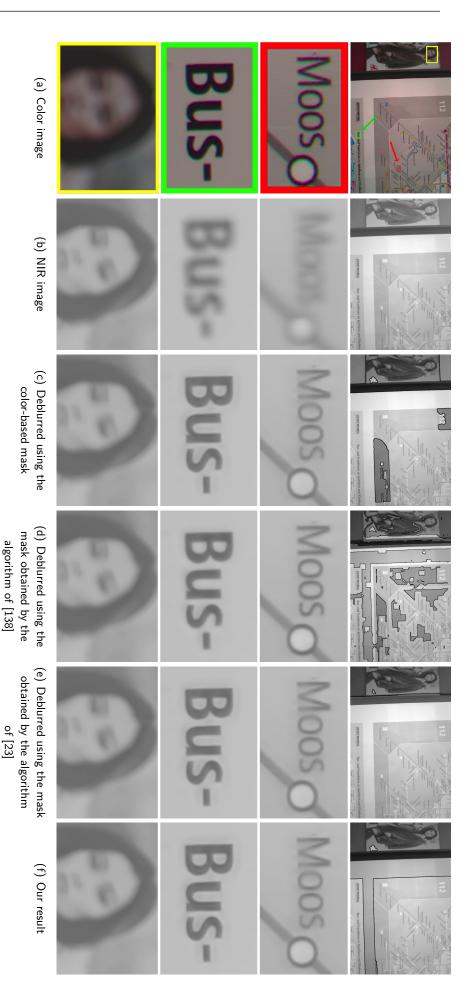
(e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground where the color image is sharper. Fourth row: Background patches. Our algorithm obtains the most accurate mask. Hence, the images deblurred by our Figure 6.10: (a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138]. mask (f) is sharper than other images (second and third rows).



color-based segmentation. Hence, it is not deblurred. The differences are more easily seen when this figure is viewed on a screen. where the color image is sharper. Fourth row: Background patches. Second row: (c) This patch is incorrectly detected as part of the background by the (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground



row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second row: patches on the foreground where the color image is sharper. Third and fourth rows: Background patches. The masks obtained by our algorithm and Chakrabarti's [23] result in the best deblurred images. Third row: Color-based segmentation detects this patch to be part of the foreground, which results in severe artifacts in the deblurred (f) Our result. The foreground-background masks obtained by different techniques are shown in the first Figure 6.12: (a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138] (e) Image deblurred using the mask of [23].



point to the regions zoomed-in in the second and third rows. result, deblurred images look similar. The mask computed by [138] results in the best deblurring for patch in the second row. Arrows in the color image where the color image is sharper. Fourth row: Background patches. All algorithms are successful in obtaining accurate masks for this simple scene. As a (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row Figure 6.13: (a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138].



are most noticeable on the screen). Third row: Our algorithm mislabels this patch as background and does not deblur it. Fourth row: Although this where the color image is sharper. Fourth row: a background patch. Second row: our result is slightly sharper compared with other algorithms (differences (e) Image deblurred using the mask of [23]. (f) Our result. The foreground-background masks obtained by different techniques are shown in the first row (c)-(f). Foreground pixels are shown by their original intensities, and background pixels are darkened. Second and third rows: patches on the foreground patch is detected as foreground by our algorithm, the deblurred image does not suffer from any artifacts in this region. The arrow in the color image Figure 6.14: (a) Input color image. (b) Input NIR image. (c) Image deblurred using the color-based mask. (d) Image deblurred using the mask of [138] points to the small region that is zoomed-in in the second row

# 6.3 Spatially Varying Blur Kernel Estimation and Deblurring

In this section, we study chromatic aberration in cases where multiple planes with different levels of blur in color and NIR images could be detected. We first present an algorithm that segments the image into several regions where the blur is uniform inside each region. We then use this segmentation with an extension of our guided deblurring algorithm to deblur the NIR image in patches where the color representation is sharper than the NIR counterpart.

#### 6.3.1 Multi-Label Segmentation

Let us consider an image patch that is sharper in the color representation. The blur kernel in the corresponding patch of the NIR image might be estimated by minimizing the following cost function:

$$E(k_{\sigma}) = |s_{\mathbf{Y}_b} - s_{\mathcal{N}}|, \quad \text{for } \mathbf{Y}_b = k_{\sigma} * \mathbf{Y}. \tag{6.20}$$

In the above cost function s is the sharpness value of the patch computed by the algorithm of Crete *et al.* [33] explained in Section 6.2. Similarly to Chapter 5, we model the defocus blur as a Gaussian filter characterized by  $\sigma$ . We assume that the blur kernel is uniform in the patch.

Direct minimization of (6.20) does not lead to a reliable estimation of the blur kernel, because the amount of information carried by a local patch is critically low, and the sharpness estimation might be affected by noise. Moreover, one patch might contain sub-patches with different levels of blur, which fails the assumption of kernel uniformity across the patch. Instead of minimizing (6.20), we compute the cost function in all patches of the image for a set of  $\sigma$  values in the range of  $[-\sigma_{\text{max}}, \sigma_{\text{max}}]$ , where  $\sigma_{\text{max}}$  is a positive value empirically set to 6 in our experiments. For negative values of  $\sigma$ , the following cost is computed:

$$E(k_{\sigma}) = |s_{\mathbf{Y}} - s_{\mathcal{N}_b}|, \quad \text{for } \mathcal{N}_b = k_{|\sigma|} * \mathcal{N}.$$
 (6.21)

This value would be small for patches where NIR is sharper than the color image.

We define matrix  $\mathbf{U}$  to be the segmentation map, which contains the  $\sigma$  parameter of the blur kernel that affects the corresponding pixel. Positive values of  $\mathbf{U}$  indicate that the color image is sharper than the NIR representation in those pixels and vice versa for negative values of  $\mathbf{U}$ . We form the following multi-level cost function based on blur clues:

$$\mathbf{B}_{m,n}(\mathbf{U}(m,n)) = \begin{cases} |s_{\mathbf{Y}_b} - s_{\mathcal{N}}| & \text{for } \mathbf{Y}_b = k_{\sigma} * \mathbf{Y}, \text{ if } \mathbf{U}(m,n) > 0\\ |s_{\mathbf{Y}} - s_{\mathcal{N}_b}| & \text{for } \mathcal{N}_b = k_{|\sigma|} * \mathcal{N}, \text{ if } \mathbf{U}(m,n) < 0 \end{cases}$$
(6.22)

In the above formulation  $\sigma = \mathbf{U}(m, n)$ , and sharpness values are computed in a small neighborhood around coordinates (m, n). We empirically found a neighborhood of

size  $50 \times 50$  results in reliable estimations for images of size  $2300 \times 1500$  pixels used in our experiments.

As before, if a patch is uniform, for all values of  $\sigma$  we assign a very small cost to the corresponding elements of **B**. A uniform patch does not carry any information about the amount of blur, hence it should not add any cost to the blur term (**B**).

To obtain a smooth segmentation, we define the following cost function:

$$\mathbf{W}_{m,n,m',n'}(\mathbf{U}) = \lambda_{m,n,m',n'} \delta(\mathbf{U}(m,n) - \mathbf{U}(m',n')) \tag{6.23}$$

The above cost function penalizes assigning different labels to neighboring pixels if the weight parameter  $\lambda$  is large.  $\delta$  is defined in (6.18).  $\lambda$  is computed as the following:

$$\lambda_{m,n,m',n'} = \exp(-\frac{|\mathbf{I}_c(m,n) - \mathbf{I}_c(m',n')|^2}{t}),$$
(6.24)

where  $\mathbf{I}_c(m,n)$  and  $\mathbf{I}_c(m',n')$  are intensities of neighboring pixels (m,n) and (m',n') in the color image. On object boundaries, the difference between the color values of two neighboring pixels is usually large, resulting in a small  $\lambda$ . In this case,  $\mathbf{W}(\mathbf{U})$  does not penalize different labels for those pixels. We empirically choose t = 0.1.

By combining blur clues in (6.22) and the smoothness term in (6.23), we obtain the segmentation map by solving the following optimization:

$$\mathbf{U}^{\star} = \underset{\mathbf{U}}{\operatorname{argmin}} \sum_{m,n} \mathbf{B}_{m,n}(\mathbf{U}(m,n)) + \sum_{\substack{(m,n)\\(m',n')}} \mathbf{W}_{m,n,m',n'}(\mathbf{U}). \tag{6.25}$$

We solve the above optimization using the graph-cut algorithm [17, 78, 15].

#### 6.3.2 Spatially Varying Deblurring

Let us assume that **U** assigns L positive labels ( $\sigma_i$  for  $i = 1, \dots, L$ ) to the image. Recall that the color image is sharper than the NIR image in the patches with positive labels. Hence, we need to deblur the NIR image in these regions. The NIR image is deblurred L times with different values of  $\sigma_i$  in **U**. We call each deblurred image,  $\mathcal{N}_i^d$ .

To combine deblurred images and obtain the final NIR image, we first form L binary masks,  $\mathbf{U}_i$  for  $i = 1, \dots, L$ , as follows:

$$\mathbf{U}_{i}(m,n) = \begin{cases} 1 & \text{if } \mathbf{U}(m,n) = \sigma_{i} \\ 0 & \text{otherwise} \end{cases}$$
 (6.26)

To have smooth transitions between different regions and avoid artifacts on the boundaries of regions, we slightly blur each binary mask:

$$\mathbf{U}_i^b = \mathbf{U}_i * h, \tag{6.27}$$

where h is a Gaussian filter of size 15 × 15. Using smooth masks, we compute the final deblurred image as

$$\mathcal{N}_d = \sum_i (\mathbf{U}_i^b \mathcal{N}_i^d) + \tilde{\mathbf{U}} \mathcal{N}, \tag{6.28}$$

where  $\tilde{\mathbf{U}} = 1 - \sum_{i} \mathbf{U}_{i}^{b}$ .

#### 6.3.3 Results

#### **Blur-Kernel Map Estimation**

In this subsection, we compare the performance of our algorithm in computing the blur-kernel map with the method of Zhang and Hirakawa [159]. Their algorithm estimates spatially varying blur kernels by analyzing the discrete wavelet coefficients of local patches in a single image, where the lens defocus blur is modeled as a disk filter. For each scene, we apply their algorithm to estimate the size of the disk kernel in the local neighborhoods in the color image. We call the matrix, which contains this information at every pixel,  $\mathbf{U}_{RGB}$ . We then use their algorithm to estimate the spatially varying blur kernel in the NIR image, resulting in  $\mathbf{U}_{\mathcal{N}}$ . We compute the amount of blur in NIR with respect to the color counterpart as follows:

$$\mathbf{U} = \mathbf{U}_{\mathcal{N}} - \mathbf{U}_{\text{RGB}}.\tag{6.29}$$

Figures 6.15-6.21 show **U** kernel maps computed by our algorithm (see (6.25)) and the results of Zhang and Hirakawa's approach (**U** in (6.29)). For each scene, we number the objects on the color image in the order of their distance from the camera. For each example, in the caption we indicate the object that is in focus in the color image. For all examples, our algorithm clearly outperforms Zhang and Hirakawa's method [159] in estimating the blur-kernel map.

In almost all cases, our algorithm finds a different blur kernel for every object (or sometimes even a part of an object) placed at a different depth. Objects further away from the color focus plane, where the NIR image is sharper than the color image, are correctly labeled by negative  $\sigma$  values. Although color clues usually ease the segmentation, they cannot always be beneficial. For instance, the color contrast in the example of Figure 6.17 is not high. Notice that the color of the tower in the Stephansdom model (the top part of object number 2) is quite similar to the background color (object number 3). Yet, the blur clues used by our algorithm are strong enough to assign different labels to these regions.

There are still some errors in blur kernel maps obtained by our algorithm. For instance, consider the example in Figure 6.15. In this scene, part of the book (object number 3) has the same label as the bird (object number 2) in our blur-kernel map. This patch of the book is almost uniform and the color edge between the bird and the book is not strong enough to cut the segmentation at that region. In Figure 6.21, object number 3 does not contain rich texture information, and its color is similar to the background (object number 4). Hence, it is incorrectly segmented as part of

the background. These mistakes do not usually affect the final output as uniform regions do not change with deblurring.

#### **Spatially Varying Deblurring**

In Figures 6.22-6.28, we compare the performance of our spatially-varying deblurring algorithm with two other techniques. The first technique is the deblurring algorithm proposed in [159], where the blur-kernel maps obtained as (6.29) are used. We also compare with the results of using our binary segmentation and deblurring algorithms presented in Section 6.2.

The images deblurred by Zhang and Hirakawa's algorithm are in most cases blurred compared with the results of our binary and spatially varying deblurring methods. For all scenes, using our multi-label map results in sharper images without producing artifacts. The images deblurred by our method are almost as sharp as the color image. This holds for patches where the color representation is in focus, and also for regions where, although not in focus, the color image is still sharper than NIR (for instance, see the third row of Figure 6.24). This is because by using a blur segmentation map, a more reliable kernel is estimated for every region. Our binary deblurring could fail if two planes with different amounts of blur in the NIR image are both segmented as foreground.

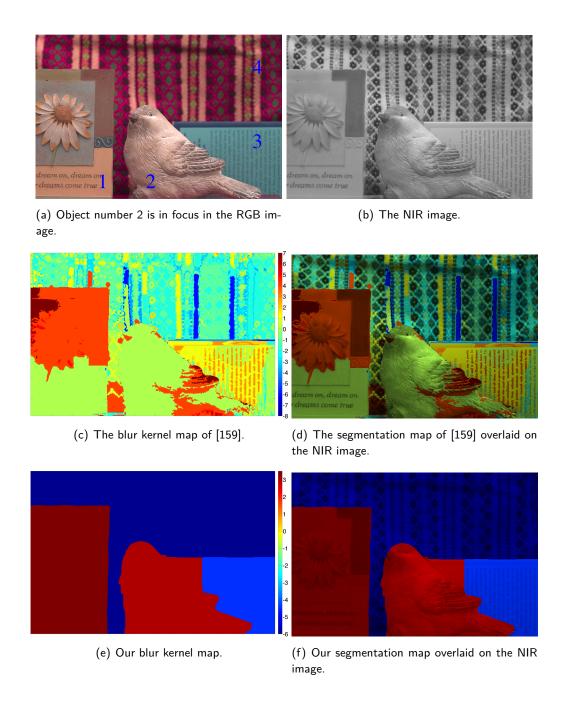


Figure 6.15: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.

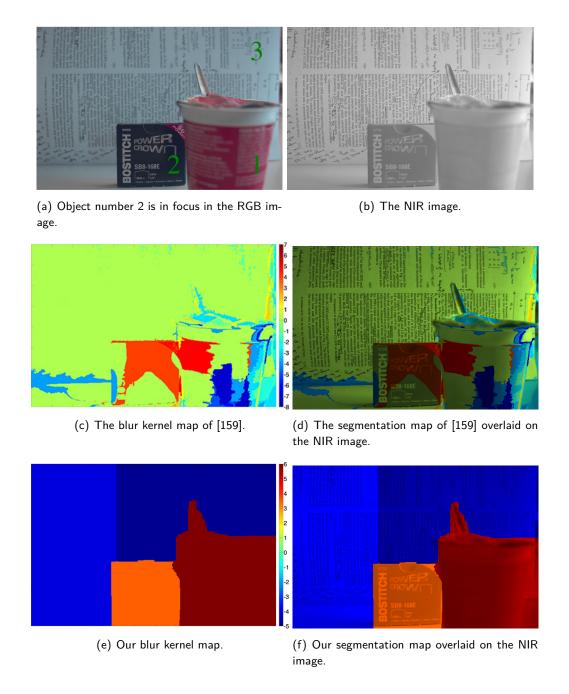


Figure 6.16: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.

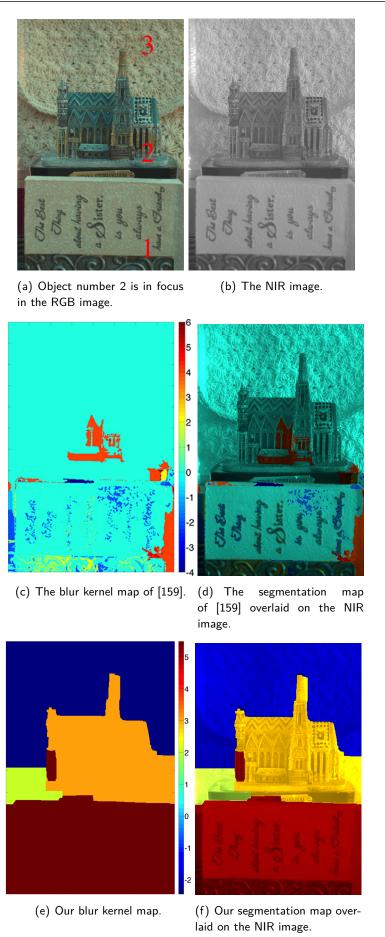


Figure 6.17: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.

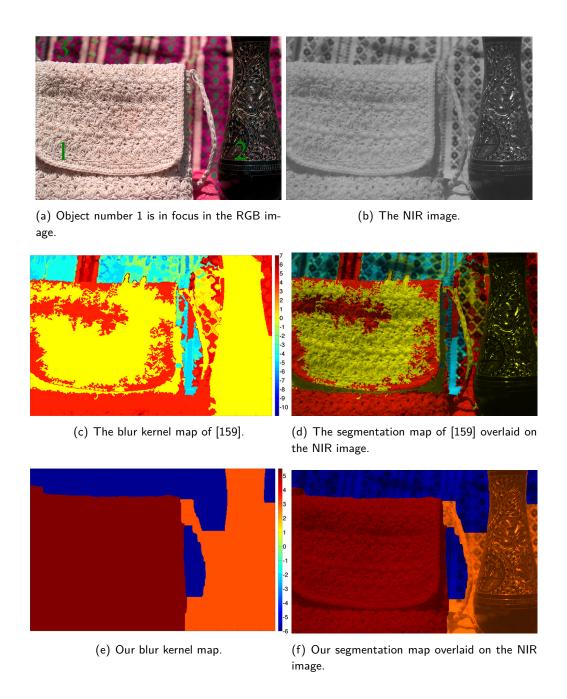


Figure 6.18: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.

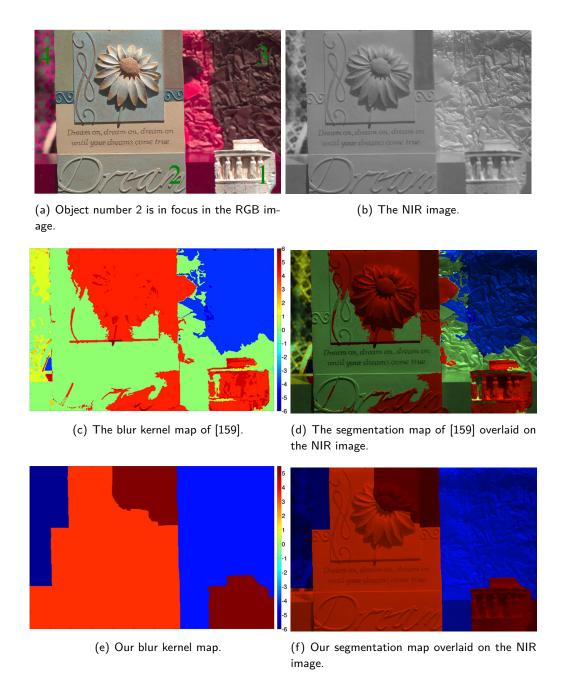


Figure 6.19: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.

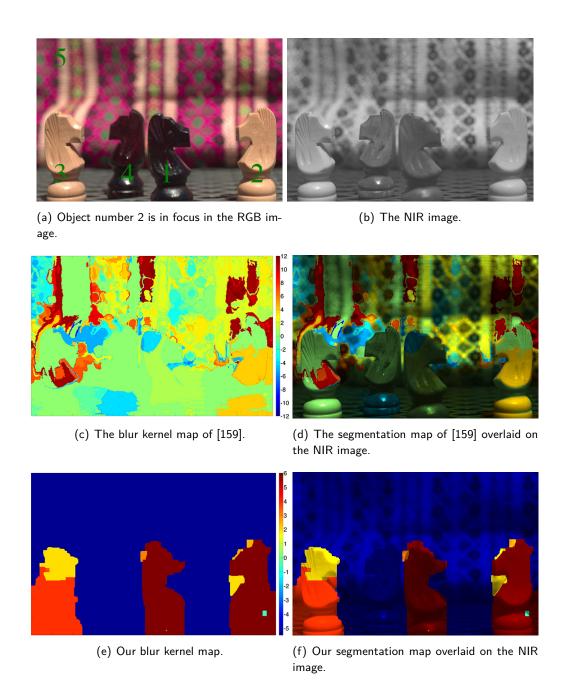


Figure 6.20: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.

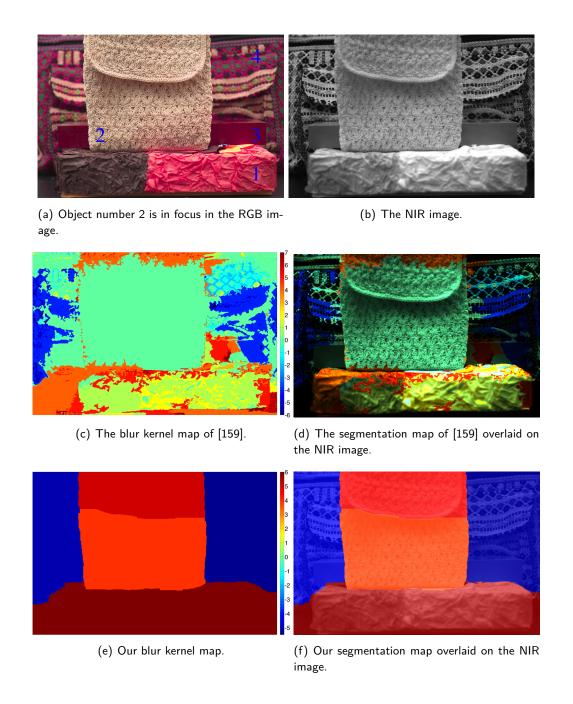
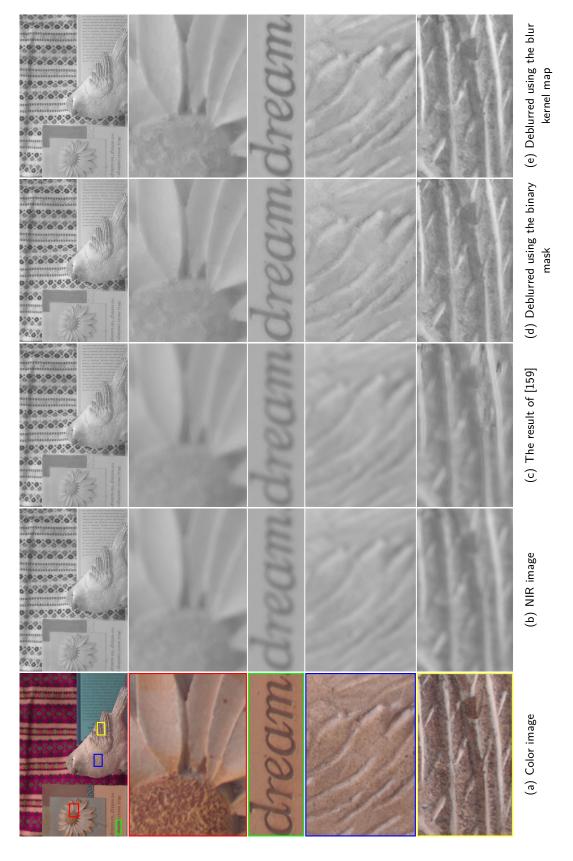
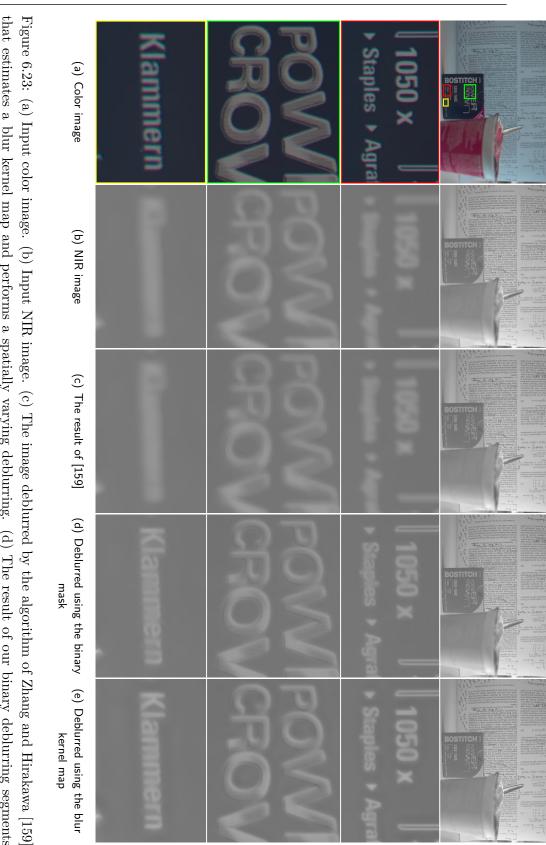


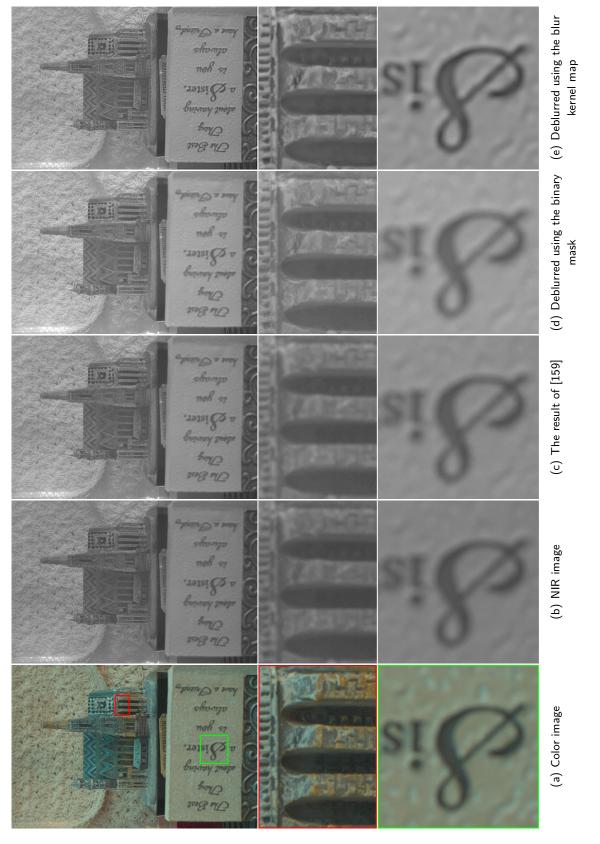
Figure 6.21: The blur kernel map indicates the  $\sigma$  parameter of the blur kernel in each patch. Positive  $\sigma$  values in the map mean that the color image is sharper than NIR in the corresponding patches. NIR is sharper where  $\sigma$  is negative. The objects are numbered in the order of their distance to the camera.



that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and Figure 6.22: (a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] to deblur each region separately.



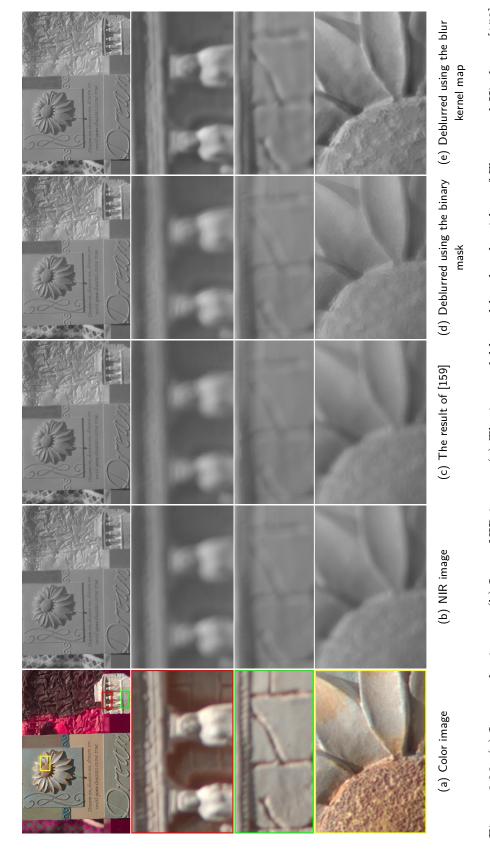
to deblur each region separately. the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments



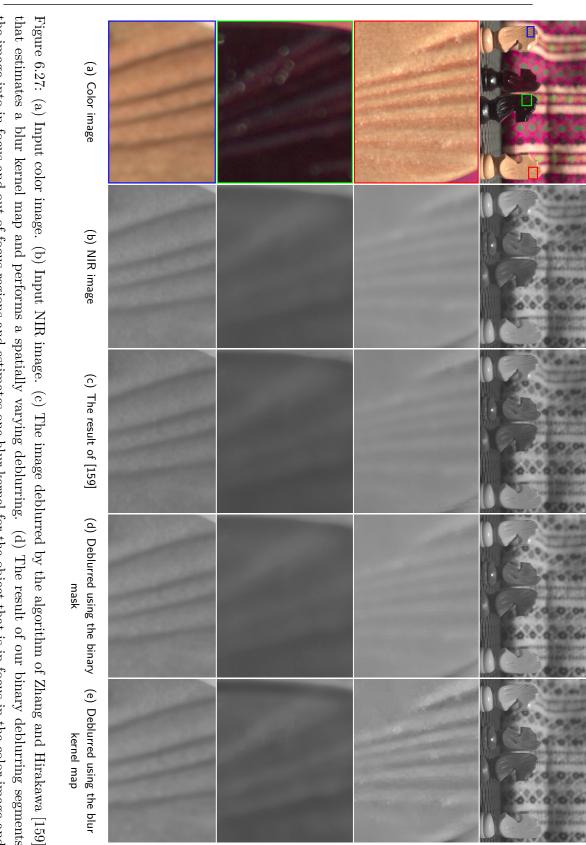
that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and Figure 6.24: (a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] to deblur each region separately.



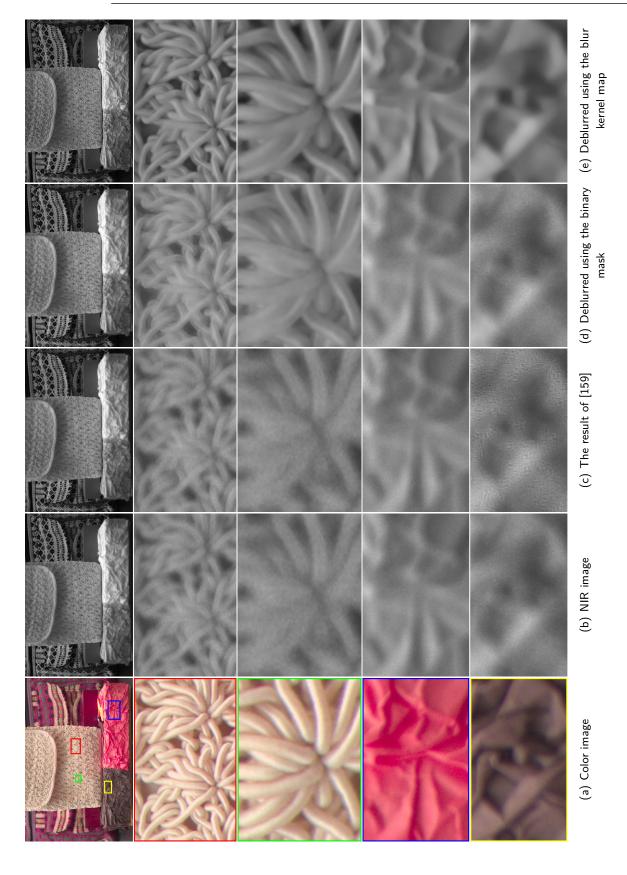
to deblur each region separately. blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and



blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and Figure 6.26: (a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] to deblur each region separately.



to deblur each region separately. blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments



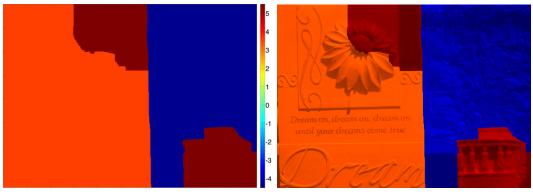
that estimates a blur kernel map and performs a spatially varying deblurring. (d) The result of our binary deblurring segments the image into in-focus and out-of-focus regions and estimates one blur kernel for the object that is in-focus in the color image and blurred in NIR. (e) The blur kernel map obtained by our algorithm is used to split the image into regions with the same label and Figure 6.28: (a) Input color image. (b) Input NIR image. (c) The image deblurred by the algorithm of Zhang and Hirakawa [159] to deblur each region separately.

### 6.4 Discussion

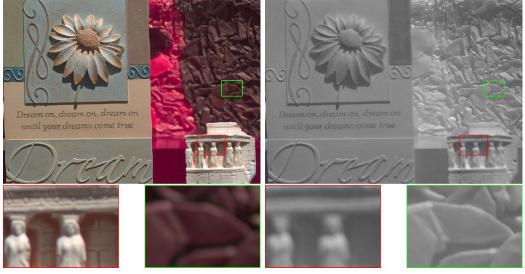
In this chapter, we mainly addressed deblurring the NIR image in regions where the color image is sharper than NIR. By using a dense blur-kernel map obtained by our algorithm, performing two other processing steps is possible. We can blur the NIR image in areas where the color image is more blurred than NIR. Furthermore, we can deblur the color image in patches where the NIR representation is sharper. Figure 6.29 illustrates an example where (1) the color image is deblurred in the background plane and (2) the NIR image is deblurred in the foreground and blurred in the background planes. By deblurring the background object in the color image, the depth-of-field slightly increases. Deblurring and blurring the NIR image in foreground and background planes, respectively, result in an NIR image that has the same focus plane of the original color image.

As mentioned earlier, the amount of blur changes in the image when the objects are placed close<sup>3</sup> to the camera. Otherwise, the blur kernel can be assumed uniform. Figure 6.30 shows one example of such a scene. We apply our blur-kernel map estimation to this example and show the result of using only the blur clues (without exploiting color information) in Figure 6.30-(c). As can be seen, our algorithm estimates a uniform kernel for the majority of patches. Figure 6.30-(d) shows the blur-kernel map computed by both blur and color clues. This map contains one plane for the whole image. Hence, for this case, applying a spatially variant deblurring algorithm does not change the result compared with using the deblurring algorithm proposed in Chapter 5.

<sup>&</sup>lt;sup>3</sup>In the order of a few centimeters to one or two meters depending on the camera settings.



- (a) The blur kernel map
- (b) The blur kernel map overlaid on the NIR image



- jects are in focus and the background object is out jects are blurred. The background object is sharper of focus.
- (c) The original color image. The foreground ob- (d) The original NIR image. The foreground obthan the color representation.



- (e) The background object is deblurred. This in- (f) The foreground objects are deblurred and the creases the depth-of-field compared with the orig- background object is blurred in the NIR image. inal color image in (c).
- The focus plane of this processed image is similar to the original color image shown in (c).

Figure 6.29: The blur kernel map obtained by our algorithm can be effectively used to deblur the NIR image in patches where the color image is sharper. Moreover, the color image can be deblurred on the background plane where the NIR image is sharper.

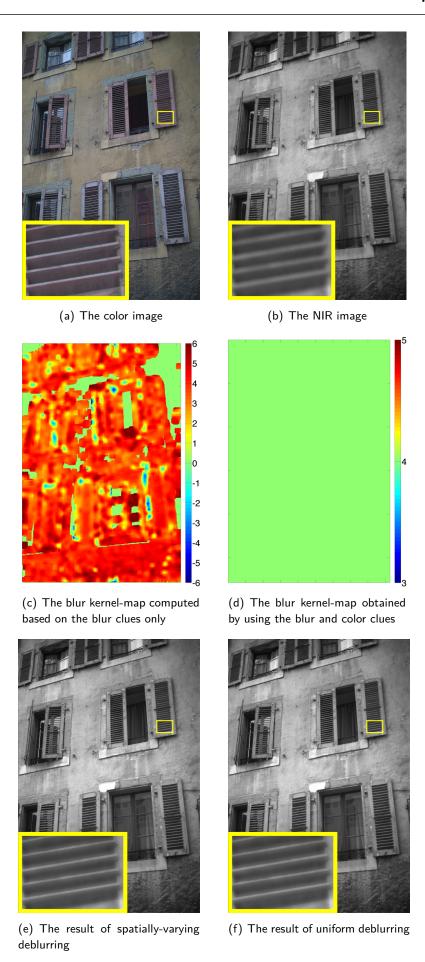


Figure 6.30: For this scene the amount of blur in the NIR image with respect to the color image is spatially invariant. Hence, estimating the blur-kernel map does not improve the deblurring result.

6.5. Summary 161

### 6.5 Summary

The severity of chromatic aberration distortions in the NIR image changes with the object depth if the scene is close to the camera or with a shallow depth-of-field. In this case the assumption of spatially varying blur kernels does not hold anymore.

We developed an algorithm that locally analyzes and compares the sharpness levels in a pair of color and NIR images. The differences in local sharpness values provide information about the amount of blur kernel in one of color or NIR representations with respect to the other. Combined with color information, the local blur clues result in dense blur kernel maps that assign different  $\sigma$  values to objects placed at different distances from the camera. Our deblurring algorithm explained in the previous chapter is adopted to separately deblur each region that has a uniform blur kernel. A smooth fusion of deblurred regions results in visually pleasing images.

We also presented a simplified version of this algorithm that binary segments the image into regions where the color image is sharper than NIR (foreground), and patches where NIR is sharper. The binary mask is used to deblur the foreground plane in the NIR image.

## Chapter 7

# **Conclusion**

### 7.1 Thesis Summary

Current color-and-NIR acquisition techniques, customized mainly for industrial and scientific use, are too expensive and inconvenient for "everyday" photography. We proposed an acquisition solution that can be implemented using the hardware components similar to those of consumer color cameras. Our proposed system reduces the manufacturing costs and paves the way for the mass production of consumer color-and-NIR cameras. Throughout the thesis, we showed that the joint acquisition of color and NIR is more challenging than conventional color imaging. First, because data with a higher spectral resolution (four channels instead of three channels captured in color imaging) is recorded. More importantly, as quantified in Chapter 3, the dependency between NIR and color channels is usually weaker than the correlation between color channels.

In the first part of the thesis, we studied the spectral and spatial sampling of the scene, and illustrated that color and NIR acquisition on a single sensor is feasible as silicon-based sensors are sensitive to both visible and NIR bands of the spectrum. We developed an algorithm that, based on the correlation characteristics of these images, designs an optimum color filter array. The optimized CFA is accompanied by a linear demosaicing that reconstructs full-resolution images. Our algorithm optimizes the CFA and the demosaicing matrix by (1) minimizing the error of reconstructing full-resolution images and (2) maximizing the correlation between the high-frequency information of color and NIR images. Both objective and subjective comparisons confirm the superiority of our method over a similar algorithm presented in [91].

We also analyzed the use of the RGBN CFA made of inexpensive dye filters, similar to those used in current color cameras. As discussed in Chapter 4, the main limitation of using the RGBN CFA is the cross-talk between color and NIR filters. As a result of this deficiency, the sensor records a mixture of visible and NIR radiations at every pixel. Our solution for this problem un-mixes the measurements by relying on the spatial and spectral correlations of color and NIR images. We exploited the

164 Chapter 7.

correlation through training sparsifying transforms that compactly represent the signal.

We compared the performance of our optimized CFA and the RGBN CFA with the state-of-the-art techniques developed for single-sensor multi-band acquisition. Our studies showed that the optimized CFA performs the best in terms of most quality metrics. The RGBN CFA and our reconstruction algorithm achieve the image quality comparable with the results of other techniques. Compared with the complicated sampling strategies proposed by other techniques, the RGBN CFA is the most similar to the CFAs used in current color cameras. Moreover, our algorithm is the only one that models and addresses the cross-talk between color and NIR radiations caused by the imprecise cut-offs of dye filters.

In Chapters 5 and 6, we analyzed chromatic aberration distortions when using simple lenses that reduce chromatic aberration in color channels but remain uncorrected for the NIR band. We tackled this issue by deblurring the NIR image when the color image on the sensor plane is in focus and sharp. Our main goals were to retrieve the lost details of NIR by using the gradients of the sharp color image and to maintain the inherent differences between these images. We achieved these by iteratively deblurring NIR and measuring the local correlation between color and NIR gradients in a multiscale fashion. Our algorithm performs better than blind deblurring algorithms in retrieving the sharp edges without producing visual artifacts. We also showed that the general approach employed in color imaging to reduce chromatic aberration distortions discards the intrinsic differences between color and NIR images. This is critically problematic as the algorithms that fuse color and NIR rely mainly on the differences between these images. Our guided deblurring algorithm successfully preserves these differences.

In Chapter 6, we demonstrated that variations in object distances could cause noticeable changes in the amount of blur across the image. This is specifically pronounced in macro photography. We developed an algorithm that estimates a dense blur-kernel map by combining local sharpness measures computed for the pair of color and NIR images, and color information. By using the blur-kernel map, our guided deblurring algorithm is applied separately to each region of the NIR image that is blurred by a uniform kernel. The final image is obtained by smoothly combining different deblurred regions. Our algorithm outperforms the state-of-theart methods [23], [159], and [138] in estimating the blur-kernel map. As a result, the images deblurred by our algorithm, compared with the results of these methods, contain sharper edges without noticeable artifacts. Our blur-kernel estimation and deblurring methods do not require any information about the lens and the focus settings.

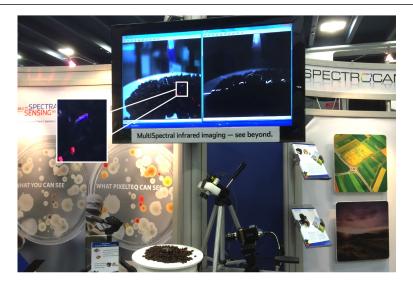


Figure 7.1: The PixelTeq color and NIR camera (PixelCam) monitors coffee beans. Visible and NIR active light sources are used for this demonstration. The left half of the screen shows the color image, and the right part shows the NIR image. Note that the NIR light leaks into the color filters. The NIR radiation is observed as purple stripes in the color image. This photo is taken at the Photonics West 2015 exhibition.

#### 7.2 Future Research

Motivated by diverse applications of color and NIR images, a few camera manufacturers, such as Omnivision and PixelTeq, have recently focused on a single-sensor acquisition solution. The algorithms developed in this thesis for spatial and spectral sampling and chromatic aberration reduction benefit the challenges these manufacturing efforts come across. At the Photonics West 2015 conference, PixelTeq exhibited a camera that uses a color and NIR mosaic. Figure 7.1 shows the camera while it monitors coffee beans. Visible and NIR active light sources are used for this demonstration. The zoomed-in region shows that the NIR radiation leaks into color filters as purple stripes (the red dots are the projections of the visible-light source). Our separation algorithm, presented in Chapter 4, is designed to solve this problem. Analyzing the performance of our algorithm in this camera could be a part of the future research in this field.

As shown in Chapter 6, the amount of chromatic-aberration blur in the NIR image with respect to the color image depends on the object depth. It would be interesting to investigate the use of our blur kernel map in estimating the depths of objects in the scene. In order to calculate absolute depth values, the camera and the lens should be calibrated in different acquisition settings, similar to what is done for a color camera by Trouvé et al. in [145]. This approach will probably not

166 Chapter 7.

be as accurate as active depth-estimation systems such as Kinect or time-of-flight cameras [77, 55]. It will, however, be useful for outdoor scenes where Kinect fails to calculate reliable depth maps.

We addressed two of the most important issues in the single-sensor acquisition of color and NIR images. There still remain a few questions that require further research. For instance, without an NIR-blocking filter, every pixel receives both color and NIR radiations. This, considering the limited charge capacity of pixels, is problematic when recording scenes with extremely high dynamic-range especially if the light source radiates strongly in both visible and NIR bands. This problem could be analyzed thoroughly once a color and NIR camera prototype is accessible.

Some of the commonly used light sources, such as fluorescent, radiate weakly in the NIR band. This causes problems when taking pictures in environments that are illuminated solely by fluorescent light sources. An NIR flash could be used in such a situation. As NIR is invisible and harmless, using an NIR flash does not disturb the user or the environment. To resolve this issue, it would also be interesting to investigate exposing the NIR pixels more than the color pixels. This technique is explored in high dynamic-range imaging where an optical mask with spatially varying transmittance is placed next to the sensor [110].

All the ground-truth images used in this thesis were captured in two sequential shots when NIR-blocking and visible-light-blocking filters are placed in front of the lens. This method limits the images to representing scenes that do not contain moving objects. Although not fundamentally different, color and NIR acquisition should be tested for scenes with moving objects.

Some of the in-camera processing steps, implemented in current color cameras, should be adopted when the additional NIR band is captured. For instance, compression standards employed in current cameras are optimized for color images. The framework needs to be customized for joint compression of color and NIR images. In [130], we conducted an initial study on the joint compression by removing the spatial and spectral redundancies of these images. Moreover, the NIR information could be used to improve the white-balancing in the camera, as common light sources, despite having similar power spectra in the visible range, behave very differently in the NIR band. These differences are explored by Fredembach and Süsstrunk in [44] to detect the dominant illumination in the scene.

In this thesis, the NIR was defined as the spectral band that covers the wavelength range of 700 nm (right next to the visible spectrum) up to the silicon sensitivity limit (wavelength of 1100 nm). Hence, the wavelength range captured in joint acquisition is more than twice the range recorded in color imaging. This, as discussed throughout the thesis, introduces challenges in different aspects of acquisition. Limiting the recorded NIR information to a smaller wavelength range eases these difficulties as the correlation between color and NIR channels increases. This, however, might restrict the applications that fuse color and NIR images, as they rely on the differences between these images. Another possible direction for future

research is to find an NIR sub-band that achieves the best compromise between the acquisition quality and the gain obtained by combining color and NIR information.

Our work addressed a special case of multispectral imaging where four spectral channels are captured. The design issues studied in this thesis are also encountered in a more general single-sensor multispectral acquisition system, especially if both visible and infrared radiations are captured. Our algorithms and the results of our study could be extended to solve the challenges in such an acquisition system.

168 Chapter 7.

## Appendix A

# **Optimizing CFA and Demosaicing**

In Chapter 3, we presented an optimization problem for designing CFA and demosaicing matrices used in the joint acquisition of color and NIR images. The optimization problem is solved by iteratively updating the demosaicing matrix and the CFA coefficients. In this chapter, we derive the details of this optimization.

Let us call the cost function of our optimization problem (see (3.20)),  $J(\mathbf{A}, \mathbf{D})$ . Recall that  $\mathbf{A}$  contains the CFA coefficients and  $\mathbf{D}$  is the demosaicing matrix. The cost function is:

$$J(\mathbf{A}, \mathbf{D}) = \|\mathbf{SP} - \mathbf{DAP}\|_F^2 + \lambda(\|\mathbf{S}_1 \mathbf{DAP}\|_F^2 + \|\mathbf{S}_2 \mathbf{DAP}\|_F^2). \tag{A.1}$$

To expand the above cost function, we use the following equation that holds for any matrix,  $\mathbf{Z}$ :

$$\|\mathbf{Z}\|_F^2 = \text{Tr}(\mathbf{Z}^T \mathbf{Z}),\tag{A.2}$$

where Tr is the matrix trace (sum of its diagonal elements), and T denotes the transpose operator. We expand (A.1) as follows:

$$J(\mathbf{A}, \mathbf{D}) = \text{Tr}((\mathbf{SP} - \mathbf{DAP})^T (\mathbf{SP} - \mathbf{DAP}) + \lambda (\mathbf{S}_1 \mathbf{DAP})^T (\mathbf{S}_1 \mathbf{DAP}) + \lambda (\mathbf{S}_2 \mathbf{DAP})^T (\mathbf{S}_2 \mathbf{DAP}))$$
(A.3)

We start by finding the optimum point of (A.1) with respect to  $\mathbf{D}$  when  $\mathbf{A}$  is fixed. After expanding (A.3) and removing the terms that do not depend on  $\mathbf{D}$ , the cost function with respect to  $\mathbf{D}$  is:

$$J_1(\mathbf{D}) = \text{Tr}(-2\mathbf{D}\mathbf{A}\mathbf{P}\mathbf{P}\mathbf{S}^T) + \text{Tr}(\mathbf{P}^T\mathbf{A}^T\mathbf{D}^T(\mathbf{I} + \lambda \mathbf{S}_1^T\mathbf{S}_1 + \lambda \mathbf{S}_2^T\mathbf{S}_2)\mathbf{D}\mathbf{A}\mathbf{P}), \quad (A.4)$$

where I is the identity matrix.

The following equations hold for all matrices **Z**, **B**, and **C** [116]:

$$\frac{\nabla}{\nabla \mathbf{Z}} \text{Tr}(\mathbf{B} \mathbf{Z} \mathbf{C} \mathbf{Z}^T \mathbf{B}^T) = \mathbf{C}^T \mathbf{Z} \mathbf{B} \mathbf{B}^T + \mathbf{C} \mathbf{Z} \mathbf{B} \mathbf{B}^T, 
\frac{\nabla}{\nabla \mathbf{Z}} \text{Tr}(\mathbf{Z} \mathbf{B}) = \mathbf{B}^T.$$
(A.5)

170 Appendix A.

Using (A.5), we have:

$$\frac{\nabla J_1(\mathbf{D})}{\nabla \mathbf{D}} = -2\mathbf{SPPA}^T + 2(\mathbf{I} + \lambda \mathbf{S}_1^T \mathbf{S}_1 + \lambda \mathbf{S}_2^T \mathbf{S}_2)\mathbf{DAPPA}^T.$$
(A.6)

Setting the above gradient to zero, we obtain the optimum demosaicing matrix:

$$\mathbf{D} = (\mathbf{I} + \lambda \mathbf{S}_1^T \mathbf{S}_1 + \lambda \mathbf{S}_2^T \mathbf{S}_2))^{\dagger} (\mathbf{SP}) (\mathbf{AP})^{\dagger}. \tag{A.7}$$

To update CFA coefficients, as explained in Chapter 3, we write:

$$\mathbf{A} = \mathbf{A}_0 + \sum_i \alpha_i \mathbf{A}_i. \tag{A.8}$$

where  $\mathbf{A}_0$  and  $\mathbf{A}_i$ s are fixed matrices, and  $\alpha_i$  are CFA coefficients. For the sake of brevity, we define the following matrices:

$$\mathbf{X}_{0} \triangleq \mathbf{SP} - \mathbf{D}\mathbf{A}_{0}\mathbf{P} \quad \mathbf{X}_{i} \triangleq \mathbf{D}\mathbf{A}_{i}\mathbf{P}$$

$$\mathbf{Y}_{0} \triangleq \mathbf{S}_{1}\mathbf{D}\mathbf{A}_{0}\mathbf{P} \quad \mathbf{Y}_{i} \triangleq \mathbf{S}_{1}\mathbf{D}\mathbf{A}_{i}\mathbf{P}$$

$$\mathbf{Z}_{0} \triangleq \mathbf{S}_{2}\mathbf{D}\mathbf{A}_{0}\mathbf{P} \quad \mathbf{Z}_{i} \triangleq \mathbf{S}_{2}\mathbf{D}\mathbf{A}_{i}\mathbf{P}.$$
(A.9)

Using (A.3), (A.8), and (A.9), the cost function with respect to the CFA coefficients is:

$$J_{2}(\alpha) = \sum_{i} \alpha_{i} \operatorname{Tr}(\mathbf{Y}_{i}^{T} \mathbf{Y}_{0} + \mathbf{Z}_{i}^{T} \mathbf{Z}_{0} - \mathbf{X}_{i}^{T} \mathbf{X}_{i})$$

$$+ \sum_{i} \sum_{j} \alpha_{i} \alpha_{j} \frac{1}{2} \operatorname{Tr}(\mathbf{X}_{i}^{T} \mathbf{X}_{j} + \mathbf{Y}_{i}^{T} \mathbf{Y}_{j} + \mathbf{Z}_{i}^{T} \mathbf{Z}_{j}).$$
(A.10)

In the above formulation,  $\alpha$  is a vector that contains CFA coefficients. We define the vector  $\mathbf{p}$  and the matrix  $\mathbf{Q}$  as follows:

$$\mathbf{p}_{i} = \operatorname{Tr}(\mathbf{Y}_{i}^{T}\mathbf{Y}_{0} + \mathbf{Z}_{i}^{T}\mathbf{Z}_{0} - \mathbf{X}_{i}^{T}\mathbf{X}_{i})$$

$$\mathbf{Q}_{ij} = \operatorname{Tr}(\mathbf{X}_{i}^{T}\mathbf{X}_{j} + \mathbf{Y}_{i}^{T}\mathbf{Y}_{j} + \mathbf{Z}_{i}^{T}\mathbf{Z}_{j}),$$
(A.11)

where  $\mathbf{p}_i$  is the  $i^{\text{th}}$  element of  $\mathbf{p}$ , and  $\mathbf{Q}_{ij}$  is the element at row i and column j. Using these definitions, the CFA coefficients ( $\alpha$ ) are updated by solving the following quadratic optimization problem:

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \frac{1}{2} \alpha^T \mathbf{Q} \alpha + \mathbf{p}^T \alpha, \quad \text{s.t. } 0 \le \alpha \le 1.$$
 (A.12)

# **Glossary**

- **R** The red channel in a color image. 17, 31, 32, 42, 45, 51, 74, 76
- **G** The green channel in a color image. 17, 31, 32, 42, 45, 51
- **B** The blue channel in a color image. 17, 42, 45, 51
- $\mathcal{N}$  The NIR image. xii, xiii, xv, xxiii, 41, 42, 45–53, 55–57, 63, 74, 76, 85, 86, 98–103, 105, 108, 111, 121–123, 138–140
- I A gray-scale, color, or multi-spectral image. 18–20, 33, 34, 40, 41, 56, 85–87, 122, 123, 139
- k The blur kernel that models the shape and amount of blur in an image. 33, 34
- N Noise that usually contaminates the image during acquisition or further processing. 33
- $\nabla$  The gradient operator. 34, 108, 111, 122, 123
- Y The luminance channel of the color image that contains spatial (achromatic) information about the scene. It is either computed as the average of color channels or is the luminance channel in the YCbCr color space. xii, xv, xxiii, 40–42, 44, 46–50, 53, 55, 63, 98–103, 105, 108, 122, 123, 138

172 Glossary

- [1] "Chromatic aberration," http://en.wikipedia.org/wiki/Chromatic\_aberration.
- [2] "Digital cameras," http://en.wikipedia.org/wiki/Digital\_camera.
- [3] "Filter-wheel in multispectral acquisition," http://www.indiamart.com/comtekscientific-instruments/spectral-cameras.html.
- [4] H. Aggarwal and A. Majumdar, "Single-sensor multi-spectral image demosaicing algorithm using learned interpolation weights," *IEEE International Geoscience and Remote Sensing Symposium*, 2014.
- [5] M. Aghagolzadeh, A. A. Moghadam, M. Kumar, and H. Radha, "Compressive demosaicing for periodic color filter arrays," in Proc. *IEEE International Conference on Image Processing*, 2011.
- [6] M. Aharon, M. Elad, and A. Bruckstein, "The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representations," *IEEE Transactions on Image Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [7] T. Ahonen, "Communities dominate brands," http://communities-dominate.blogs.com/brands/2014/08/camera-stats-world-has-48b-cameras-by-4b-unique-camera-owners-88-of-them-use-cameraphon
- [8] E. Allen and S. Triantaphillidou, *The Manual of Photography*. Focal Press, 2011.
- [9] D. Alleysson, S.Süsstrunk, and J. Hérault, "Linear demosaicing inspired by the human visual system," *IEEE Transactions on Image Processing*, vol. 14, no. 4, pp. 439–449, 2005.
- [10] D. Alleysson, S. Süsstrunk, and J. Marguier, "Influence of spectral sensitivity functions on color demosaicing," *Color Imaging Conference*, 2003.
- [11] S. Appelle, "Perception and discrimination as a function of stimulus orientation: the "oblique effect" in man and animals," *Psychol Bull.*, vol. 78, no. 4, pp. 266–278, 1972.

[12] G. Baone and H. Qi, "Demosaicking methods for multispectral cameras using mosaic focal plane array technology," *Proc. of IS&T/SPIE, Spectral Imaging: Eighth International Symposium on Multispectral Color Science*, 2006.

- [13] B. E. Bayer, "Color imaging array," Patent US 3 971 065, 1976.
- [14] J. Bioucas-Dias, M. Figueiredo, and J. Oliveira, "Total variation-based image deconvolution: A majorization-minimization approach," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2006.
- [15] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/maxflow algorithms for energy minimization in vision," *IEEE Transactions on* Pattern Analysis and Machine Intelligence, vol. 26, no. 9, pp. 1124–1137, 2004.
- [16] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient n-d image segmentation," *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006.
- [17] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.
- [18] D. Brainard, "Bayesian method for reconstructing color images from thrichromatic samples," *Proc. IS&T 47th Annual Meeting*, 1994.
- [19] J. Brauers and T. Aach, "A color filter array based multispectral camera," 12. Workshop Farbbildverarbeitung, 2006.
- [20] E. Candes and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.
- [21] F. Cao, F. Guichard, H. Hornung, and C. Sibade, "Characterization and measurement of color fringing," *Proc. of IS&T/SPIE*, *Electronic Imaging: Digital Photography IV*, 2008.
- [22] A. Chakrabarti, W. Freeman, and T. Zickler, "Rethinking color cameras," *IEEE International Conference on Computational Photography*, 2014.
- [23] A. Chakrabarti, T. Zickler, and W. Freeman, "Analyzing spatially-varying blur," *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [24] J. Chang, H. Kang, and M. G. Kang, "Correction of axial and lateral chromatic aberration with false color filtering," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 1186–1198, 2013.
- [25] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[26] X. Chen, G. Jeon, J. Jeong, and L. He, "Multi-directional weighted interpolation and refinement method for Bayer pattern CFA demosaicking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2014.

- [27] F. Cheng, C. Tsai, S. Hsu, and Y. Chang, "A novel demosaicing method based on the YCbCr space," *IEEE 9th Conference on Industrial Electronics and Applications*, 2014.
- [28] K. Chung, W. Yang, W. Yan, and C. Wang, "Demosaicing of color filter array captured images using gradient edge detection masks and adaptive heterogeneity-projection," *IEEE Transactions on Image Processing*, vol. 17, no. 12, pp. 2356–2367, 2008.
- [29] S. Chung, B. Kim, and W. Song, "Detecting and eliminating chromatic aberration in digital images," *IEEE International Conference on Image Processing*, 2009.
- [30] J. Compton and J. Hamilton, "Image sensor with improved light sensitivity," Patent US 8,139,130, 2005.
- [31] L. Condat, "A generic variational approach for demosaicking from an arbitrary color filter array," *IEEE International Conference on Image Processing*, 2009.
- [32] —, "A new color filter array with optimal properties for noiseless and noisy color image acquisition," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2200–2210, 2011.
- [33] F. Crete, T. Dolmiere, P. Ladret, and M. Nicolas, "The blur effect: perception and estimation with a new no-reference perceptual blur metric," *Proc. of IS&T/SPIE: Human Vision and Electronic Imaging*, 2007.
- [34] B. C. de Lavarène, D. Alleysson, and J. Hérault, "Efficient demosaicing using recursive filtering," IEEE International Conference on Image Processing, 2007.
- [35] —, "Practical implementation of LMMSE demosaicing using luminance and chrominance spaces," *Computer Vision and Image Understanding*, vol. 107, no. 1, pp. 3–13, 2007.
- [36] P. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," SIGGRAPH, 2008.
- [37] D. Donoho, "Compressed sensing," IEEE Transactions on Information Theory, vol. 52, no. 4, pp. 1289–1306, 2006.
- [38] Q. Du and J. Fowler, "Hyperspectral image compression using JPEG2000 and principal component analysis," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 2, pp. 201–205, 2007.

[39] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

- [40] M. Elad, J.-L. Starck, P. Querre, and D. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 340 358, 2005.
- [41] G. ElMasry, D.-W. Sun, and P. Allen, "Near-infrared hyperspectral imaging for predicting colour, pH and tenderness of fresh beef," *Journal of Food Engi*neering, vol. 110, no. 1, pp. 127–140, 2012.
- [42] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," SIGGRAPH, 2006.
- [43] C. Fredembach, N. Barbuscia, and S. Süsstrunk, "Combining visible and near-infrared images for realistic skin smoothing," *Proc. IS&T/SID 17th Color Imaging Conference (CIC)*, 2009.
- [44] C. Fredembach and S. Süsstrunk, "Illuminant estimation and detection using near-infrared," *Proc. of IS&T/SPIE Electronic Imaging: Digital Photography* V, 2009.
- [45] W. Freeman, "Median filter for reconstructing missing color samples," Patent US 4 724 395, 1988.
- [46] Z. Frentress, L. Young, and H. Edwards, "Field photometer with nine-element filter wheel," *Applied Optics*, vol. 3, no. 2, pp. 303–308, 1964.
- [47] L. Gan, "Block compressed sensing of natural images," *International Conference on Digital Signal Processing*, 2007.
- [48] N. Gat, "Imaging spectroscopy using tunable filters: a review," Proc. of IS&T/SPIE: Wavelet Applications VII, 2000.
- [49] J. Glotzbach and R. Schafer, "A method of color filter array interpolation with alias cancellation properties," *IEEE International Conference on Image Processing*, 2001.
- [50] M. Golbabaee, S. Arberet, and P. Vandergheynst, "Compressive source separation: Theory and methods for hyperspectral imaging," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5096–5110, 2013.
- [51] M. Golbabaee and P. Vandergheynst, "Hyperspectral image compressed sensing via low-rank and joint-sparse matrix recovery," in Proc. *International Conference on Acoustics, Speech, and Signal Processing*, 2012.

[52] B. Gunturk, Y. Altunbasak, and R. Mersereau, "Color plane interpolation using alternating projections," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 997–1013, 2002.

- [53] R. Guo, Q. Dai, and D. Hoiem, "Single-image shadow detection and removal using paired regions," *IEEE International Conference on Computer Vision* and Pattern Recognition, 2011.
- [54] J. Hamilton and J. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," Patent US 5 629 734, 1997.
- [55] M. Hansard, S. Lee, O. Choi, and R. Horaud, "Time of flight cameras: Principles, methods, and applications," *Springer Briefs in Computer Science*, 2012.
- [56] P. Hao, Y. Li, Z. Lin, and E. Dubois, "A geometric method for optimal design of color filter arrays," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 709–722, 2011.
- [57] K. He, J. Sun, and X. Tang, "Guided image filtering," *The European Conference on Computer Vision*, 2010.
- [58] —, "Guided image filtering," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, pp. 1397–1409, 2013.
- [59] F. Heide, M. Rouf, M. Hullin, B. Labitzke, W. Heidrich, and A. Kolb, "High-quality computational imaging through simple lenses," ACM Transactions on Graphics (TOG), 2013.
- [60] Y. Hel-Or and D. Keren, "Demosaicing of color images using steerable wavelets," HP Labs Israel, Tech. Rep., 2002.
- [61] J. Hershey and Z. Zhang, "Multispectral digital camera employing both visible light and non-visible light sensing on a single image sensor," Patent US 7 460 160, 2008.
- [62] K. Hirakawa and T. Parks, "Adaptive homogeneity-directed demosaicing algorithm," *IEEE Transactions on Image Processing*, vol. 14, no. 3, pp. 360–369, 2002.
- [63] K. Hirakawa and P. Wolfe, "Spatio-spectral color filter array design for optimal image recovery," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1876–1890, 2008.
- [64] C. Hu, L. Cheng, and Y. Lu, "Graph-based regularization for color image demosaicking," *IEEE International Conference on Image Processing*, 2012.

[65] S. Hwang, K. Kim, J. Shin, J. Paik, B. Abidi, and M. Abidi, "Real-time digital autofocusing using a priori estimated set of PSFs," SPIE Proceedings: Real-Time Imaging VIII, vol. 5297, 2004.

- [66] N. Joshi, C. Zitnick, R. Szeliski, and D. J. Kriegman, "Image deblurring and denoising using color priors," 2009.
- [67] H. J. Ju, D.-K. Lee, and R.-H. Park, "Chromatic aberration removal in narrow color regions of digital images," *IEEE International Conference on Consumer Electronics*, 2012.
- [68] H. J. Ju and R.-H. Park, "Colour fringe detection and correction in YCbCr colour space," *Image Processing*, *IET*, vol. 7, no. 4, pp. 300–309, 2013.
- [69] H. Kang, S.-H. Lee, J. Chang, and M. G. Kang, "Partial differential equation-based approach for removal of chromatic aberration with local characteristics," Journal of Electronic Imaging, vol. 19, no. 3, pp. 033016–033016–8, 2010.
- [70] J. Kang, H. Ok, J. Lim, and S.-D. Lee, "Chromatic aberration reduction through optical feature modeling," *Proc. of IS&T/SPIE*, *Electronic Imaging: Digital Photography V*, 2009.
- [71] S. Kang, "Automatic removal of purple fringing from images," Patent US 20 070 153 341, 2007.
- [72] S. B. Kang, "Automatic removal of chromatic aberration from a single image," *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
- [73] E. Kee, S. Paris, S. Chen, and J. Wang, "Modeling and removing spatially-varying optical blur," *IEEE International Conference on Computational Photography*, 2011.
- [74] D. Kiku, Y. Monno, M. Tanaka, and M. Okutomi, "Residual interpolation for color image demosaicking," *IEEE International Conference on Image Process*ing, 2013.
- [75] —, "Simultaneous capturing of RGB and additional band images using hybrid color filter array," *Proc. of IS&T/SPIE Electronic Imaging: Digital Photography X*, 2014.
- [76] B.-K. Kim and R.-H. Park, "Automatic detection and correction of purple fringing using the gradient information and desaturation," *European Signal Processing Conference*, 2008.
- [77] A. Kolb, E. Barth, R. Koch, and R. Larsen, "Time-of-flight cameras in computer graphics," *Computuer Graphics Forum*, vol. 29, no. 1, pp. 141–159, 2010.

[78] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, 2004.

- [79] J. Korneliussen and K. Hirakawa, "Camera processing with chromatic aberration," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4539–4552, 2014.
- [80] D. Krishnan and R. Fergus, "Dark flash photography," ACM SIGGRAPH, 2009.
- [81] —, "Fast image deconvolution using hyper-laplacian priors," NIPS, 2009.
- [82] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [83] P. Lapray, X. Wang, J. Thomas, and P. Gouton, "Multispectral filter arrays: Recent advances and practical implementation," Sensors, vol. 14, no. 11, pp. 21626–21659, 2014.
- [84] C. Laroche and M. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," Patent US 5 373 322, 1994.
- [85] B. Leung, G. Jeon, and E. Dubois, "Least-squares luma-chroma demultiplexing algorithm for Bayer demosaicking," *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp. 1885–1894, 2011.
- [86] A. Levin, Y. Weiss, F. Durand, and W. Freeman, "Understanding and evaluating blind deconvolution algorithms," IEEE Conference on Computer Vision and Pattern Recognition, 2009.
- [87] X. Li, "Demosaicing by successive approximation," IEEE Transactions on Image Processing, vol. 14, no. 3, pp. 370–379, 2005.
- [88] X. Li and M. T. Orchard, "New edge-directed interpolation," IEEE Transactions on Image Processing, vol. 10, no. 10, pp. 1521–1527, 2001.
- [89] N. Lian, L. Chang, Y. Tan, and V. Zagorodnov, "Adaptive filtering for color filter array demosaicking," *IEEE Transactions on Image Processing*, vol. 16, no. 10, pp. 2515–2525, 2007.
- [90] W. Lu and Y.-P. Tan, "Color filter array demosaicking: new method and performance measures," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1194–1210, 2003.

[91] Y. Lu, C. Fredembach, M. Vetterli, and S. Süsstrunk, "Designing color filter arrays for the joint capture of visible and near-infrared images," *IEEE International Conference on Image Processing*, 2009.

- [92] Y. Lu, M. Karzand, and M. Vetterli, "Demosaicking by alternating projections: Theory and fast one-step implementation," *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 2085–2098, 2010.
- [93] Y. Lu and M.Vetterli, "Optimal color filter array design: quantitative conditions and an efficient search procedure," SPIE Electronic Imaging: Digital Photography V, 2009.
- [94] M. M. F. Duarte and R. G. Baraniuk, "Kronecker compressive sensing," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 494–504, 2012.
- [95] P. Magnan, "Detection of visible photons in CCD and CMOS: A comparative view," Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 504, no. 1-3, pp. 199–212, 2003.
- [96] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.
- [97] A. Majumdar and R. K. Ward, "Compressed sensing of color images," *Elsevier Signal Processing*, vol. 90, no. 12, pp. 3122–3127, 2010.
- [98] S. Mallat and Y. Guoshen, "Super-resolution with sparse mixing estimators," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2889–2900, 2010.
- [99] D. Menon and G. Calvagno, "Regularization approaches to demosaicking," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2209–2220, 2009.
- [100] —, "Color image demosaicking: An overview," Signal Processing Image Communication, vol. 26, no. 8-9, pp. 518–533, 2011.
- [101] L. Meylan, D. Alleysson, and S. Süsstrunk, "Model of retinal local adaptation for the tone mapping of color filter array images," *Journal of the Optical* Society of America A, vol. 24, no. 9, pp. 2807–2816, 2007.
- [102] L. Miao, H. Qi, R. Ramanath, and W. Snyder, "Binary tree-based generic demosaicking algorithm for multispectral filter arrays," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3550–3558, 2006.
- [103] T. Mikami, D. Sugimura, and T. Hamamoto, "Capturing color and near-infrared images with different exposure times for image enhancement under extremely low-light scene," in Proc. IEEE International Conference on Image Processing, 2014.

[104] A. A. Moghadam, M. Aghagolzadeh, M. Kumar, and H. Radha, "Compressive demosaicing," in Proc. IEEE International Workshop on Multimedia Signal Processing, 2010.

- [105] A. Moghadam, M. Aghagolzadeh, M. Kumar, and H. Radha, "Compressive framework for demosaicing of natural images," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2356–2371, 2013.
- [106] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed  $\ell^0$  norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, 2009.
- [107] Y. Monno, M. Tanaka, and M. Okutomi, "Multispectral demosaicking using guided filter," *Proc. of IS&T/SPIE*, *Electronic Imaging: Digital Photography VIII*, 2012.
- [108] J. Mukherjee, R. Parthasarathi, and S. Goyal, "Markov random field processing for color demosaicing," *Pattern Recognition Letters*, vol. 22, no. 3-4, pp. 339–351, 2001.
- [109] P. Nagesh and B. Li, "Compressive imaging of color images," in Proc. International Conference on Acoustics, Speech, and Signal Processing, 2009.
- [110] S. Nayar and T. Mitsunaga, "High dynamic range imaging: spatially varying pixel exposures," *IEEE Conference on Computer Vision and Pattern Recog*nition, 2000.
- [111] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in Proc. 27th Annual Asilomar Conference on Signals, Systems, and Computers, 1993.
- [112] S. Pei and I. Tam, "Effective color interpolation in CCD color filter array using signal correlation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 6, pp. 503–513, 2003.
- [113] I. Pekkucuksen and Y. Altunbasak, "Gradient based threshold free color filter array interpolation," in Proc. *IEEE International Conference on Image Processing*, 2010.
- [114] —, "Edge strength filter based color filter array interpolation," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 393–397, 2012.
- [115] —, "Multiscale gradients-based color filter array interpolation," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 157–165, 2012.

[116] K. Petersen and M. Pedersen, "Matrix cookbook," http://www2.imm.dtu.dk/pubdb/views/publication\_details.php?id=3274, 2012.

- [117] G. Petschnigg, M. Agrawala, H. Hoppe, R. Szeliski, M. Cohen, and K. Toyama, "Digital photography with flash and no-flash image pairs," *SIGGRAPH*, 2004.
- [118] J. Portilla, D. Otaduy, and C. Dorronsoro, "Low-complexity linear demosaicing using joint spatial-chromatic image statistics," *IEEE International Con*ference on Image Processing, 2005.
- [119] M. Raffel, C. Willert, S. Wereley, and J. Kompenhans, Particle Image Velocimetry. Springer, 2007.
- [120] M. Rafinazari and E. Dubois, "Demosaicking algorithm for the Kodak-RGBW color filter array," *Proc. of IS&T/SPIE*, Color Imaging XX: Displaying, Processing, Hardcopy, and Applications, 2015.
- [121] P. Ready and P. A. Wintz, "Information extraction, SNR improvement, and data compression in multispectral imagery," *IEEE Transactions on Communications*, vol. 21, no. 10, pp. 1123–1131, 1973.
- [122] M. Rossi and G. Calvagno, "Luminance driven sparse representation based demosaicking," *IEEE International Conference on Image Processing*, 2014.
- [123] D. Rüfenacht, C. Fredembach, and S. Süsstrunk, "Automatic and accurate shadow detection using near-infrared information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, pp. 1672–1678, 2014.
- [124] Z. Sadeghipoor, Y. Lu, E. Mendez, and S. Süsstrunk, "Multiscale guided deblurring: Chromatic aberration correction in color and near-infrared imaging," *European Signal Processing Conference*, 2015.
- [125] Z. Sadeghipoor, Y. Lu, and S. Süsstrunk, "Correlation-based joint acquisition and demosaicing of visible and near-infrared images," *IEEE International Conference on Image Processing*, 2011.
- [126] —, "A novel compressive sensing approach to simultaneously acquire color and near-infrared images on a single sensor," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2013.
- [127] ——, "Gradient-based correction of chromatic aberration in the joint acquisition of color and near-infrared images," *Proc. of IS&T/SPIE Electronic Imaging: Digital Photography and Mobile Imaging XI*, vol. 9404, 2015.
- [128] T. Saito and T. Komatsu, "Demosaicing approach based on extended color total-variation regularization," *IEEE International Conference on Image Pro*cessing, 2008.

[129] N. Salamati, D. Larlus, G. Csurka, and S. Süsstrunk, "Semantic image segmentation using visible and near-infrared channels," 4th Workshop on Color and Photometry in Computer Vision at ECCV, 2012.

- [130] N. Salamati, Z. Sadeghipoor, and S. Süsstrunk, "Compression of multispectral images: Color (rgb) plus near-infrared (nir)," *IEEE International Workshop* on Multimedia Signal Processing (MMSP), 2012.
- [131] N. Salamati and S. Süsstrunk, "Material-based object segmentation using near-infrared information," IS&T/SID 18th Color Imaging Conference, 2010.
- [132] L. Schaul, C. Fredembach, and S. Süsstrunk, "Color image dehazing using the near-infrared," *IEEE International Conference on Image Processing*, 2009.
- [133] K. Schlager, "Non-invasive near infrared measurement of blood analyte concentrations," 1989, US Patent 4,882,492.
- [134] C. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Non-stationary correction of optical aberrations," *IEEE International Conference on Computer Vision*, 2011.
- [135] —, "Blind correction of optical aberrations," European Conference on Computer Vision, 2012.
- [136] H. Sheikh and A. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, 2006.
- [137] C. Shen, W. Hwang, and S. Pei, "Spatially-varying out-of-focus image deblurring with  $\ell 1-2$  optimization and a guided blur map," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2012.
- [138] J. Shi, L. Xu, and J. Jia, "Discriminative blur detection features," *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.
- [139] M. Singh and T. Singh, "Joint chromatic aberration correction and demosaicking," *Proc. of IS&T/SPIE Electronic Imaging: Digital Photography VIII*, 2012.
- [140] C. Su, "Highly effective iterative demosaicing using weighted-edge and color-difference interpolations," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 2, pp. 639–645, 2006.
- [141] T. Sun and K. F. Kelly, "Compressive sensing hyperspectral imager," in Proc. Computational Optical Sensing and Imaging, 2009.
- [142] H. Tang, X. Zhang, S. Zhuo, F. Chen, K. N. Kutulakos, and L. Shen, "High resolution photography with an rgb-infrared camera," *IEEE International Conference on Computational Photography*, 2015.

[143] B. Tatian, "Method for obtaining the transfer function from the edge response function," *Journal of the Optical Society of America*, vol. 55, pp. 1014–1019, 1965.

- [144] C. Tisse, H. P. Nguyen, R. Tessiéres, M. Pyanet, and F. Guichard, "Extended depth-of-field (EDoF) using sharpness transport across colour channels," in Proc. SPIE Proceedings: Novel Optical Systems Design and Optimization XI, 2008.
- [145] P. Trouvé, F. Champagnat, G. Besnerais, J. Sabater, T. Avignon, and J. Idier, "Passive depth estimation using chromatic aberration and a depth from defocus approach," *Applied Optics*, vol. 52, no. 29, pp. 7152–7164, 2013.
- [146] C. Wang, X. Wang, and J. Hardeberg, "A linear interpolation algorithm for spectral filter array demosaicking," Lecture Notes in Computer Science, 2014, vol. 8509, pp. 151–160.
- [147] J. Wang, C. Zhang, and P. Hao, "New color filter arrays of high light sensitivity and high demosaicking performance," in Proc. *IEEE International Conference on Image Processing (ICIP)*, 2011.
- [148] X. Wang, W. Lin, and P. Xue, "Demosaicing with improved edge direction detection," *IEEE International Symposium on Circuits and Systems*, 2005.
- [149] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [150] X. Wu and N. Zhang, "Primary-consistent soft-decision color demosaicking for digital cameras," *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1521–1527, 2004.
- [151] X. Wu, D. Gao, G. Shi, and D. Liu, "Color demosaicking with sparse representations," *IEEE International Conference on Image Processing*, 2010.
- [152] Q. Yan, X. Shen, L. Xu, S. Zhuo, X. Zhang, L. Shen, and J. Jia, "Cross-field joint image restoration via scale map," *IEEE International Conference on Computer Vision*, 2013.
- [153] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [154] L. Yuan, J. Sun, L. Quan, and H. Shum, "Image deblurring with blurred/noisy image pairs," *ACM Transactions on Graphics (TOG)*, 2007.

[155] P. W. Yuen and M. Richardson, "An introduction to hyperspectral imaging and its application for security, surveillance and target acquisition," *The Imaging Science Journal*, vol. 58, no. 5, pp. 241–253, 2010.

- [156] H. Zhang, D. Wipf, and Y. Zhang, "Multi-image blind deblurring using a coupled adaptive sparse prior," *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.
- [157] M. Zhang and L. Tao, "A patch aware multiple dictionary framework for demosaicing," Asian Conference on Computer Vision, 2014.
- [158] X. Zhang, T. Sim, and X. Miao, "Enhancing photographs with near infrared images," *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008.
- [159] Y. Zhang and K. Hirakawa, "Blur processing using double discrete wavelet transform," *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

## **Curriculum Vitae**

#### Zahra Sadeghipoor Kermani

Images and Visual Representation Lab (IVRL) School of Computer and Communication Sciences (IC) Ecole Polytechnique Fédérale de Lausanne (EPFL) 1015 Lausanne, Switzerland

Email: zahra.sadeghipoor@epfl.ch

Web: http://ivrl.epfl.ch/people/kermani

### **Education**

- 2010-2015 PhD in Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne, Switzerland
- 2007-2009 M.Sc. in Electrical Engineering, Sharif University of Technology, Iran.
- 2003-2007 B.Sc. in Electrical Engineering, Tehran Polytechnic, Iran.

### Work Experience

• Sep. 2013-Jan. 2014 Research intern at Apple Inc., Cupertino, California.

#### **Publications**

- Z. Sadeghipoor, Y. Lu, E. Mendez, and S. Süsstrunk, "Multiscale Guided Deblurring: Chromatic Aberration Correction in Color and Near-Infrared Imaging," EUSIPCO, 2015.
- Z. Sadeghipoor, Y. Lu, and S. Süsstrunk, "Gradient-based Correction of Chromatic Aberration in the Joint Acquisition of Color and Near-infrared Images," Proceedings of IS&T/SPIE Electronic Imaging: Digital Photography and Mobile Imaging XI, 2015.

188 Curriculum Vitae

• C. Lau, Y. Schwartzburg, A. Shaji, **Z. Sadeghipoor**, and S. Süsstrunk, "Creating Personalized Jigsaw Puzzles," 12th International Symposium on Non-Photorealistic Animation and Rendering, 2014.

- Z. Sadeghipoor, Y. Lu, and S. Süsstrunk, "A Novel Compressive Sensing Approach to Simultaneously Acquire Color and Near-Infrared Images on a Single Sensor," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2013.
- Z. Sadeghipoor, M. Babaie-Zadeh, and C. Jutten, "Dictionary Learning for Sparse Decomposition: A New Criterion and Algorithm," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2013.
- N. Salamati, **Z. Sadeghipoor**, and S. Süsstrunk, "Compression of Multispectral Images: Color (RGB) plus Near-Infrared (NIR)," *IEEE International Workshop on Multimedia Signal Processing*, 2012.
- Z. Sadeghipoor, Y. Lu, and S. Süsstrunk, "Optimum Spectral Sensitivity Functions for Single Sensor Color Imaging," *Proceedings of IS&T/SPIE Electronic Imaging: Digital Photography VIII*, 2012.
- Z. Sadeghipoor, Y. Lu and S. Süsstrunk, "Correlation-based Joint Acquisition and Demosaicing of Visible and Near-Infrared Images," *IEEE International Conference on Image Processing*, 2011.
- N. Salamati, **Z. Sadeghipoor**, and S. Süsstrunk, "Analyzing Near-infrared Images for Utility Assessment," *IS&T/SPIE Electronic Imaging: Human Vision and Electronic Imaging*, 2011.
- E. Taghizadeh, **Z. Sadeghipoor**, and M. Manzuri, "Semi-polynomial Kernel Optimization Based on the Fisher Method," *IEEE Workshop on Machine Learning and Signal Processing*, 2011.
- Z. Sadeghipoor, M. Babaie-Zadeh, and C. Jutten, "An Adaptive Thresholding Approach for Image Denoising Using Redundant Representations," *IEEE Workshop on Machine Learning and Signal Processing*, 2009.
- **Z. Sadeghipoor**, "Signals and Systems", *Arshad press*, 2009, (written for students who prepare themselves for Electrical Engineering National University Entrance Exam in Iran).

### **Honors and Awards**

• 2013 & 2014 Winner of two Qualcomm Innovation Fellowships (QInF), Austria.

Curriculum Vitae 189

- 2013 Google Anita Borg Scholarship Finalist.
- 2009 Rank 6 among M.Sc. students in communication systems, Sharif University of Technology.
- 2007 Rank 3 among around 20,000 participants at Iran National University Entrance Exam.
- 2007 Rank 1 among all B.Sc. students in electrical engineering, Tehran Polytechnic.
- 2006 & 2007 Finalist in the universities' Olympiad in the Electrical Engineering field (each year only 23 finalists are selected from all Electrical Engineering students in Iran).

### Supervised Projects

- Fall 2014 M.Sc. semester project, "Comprehensive Analysis of Single-sensor Multispectral Imaging Techniques".
- **Spring 2014** B.Sc. semester project, "Analyzing Chromatic Aberration in Color and NIR Joint Acquisition".
- **Spring 2013** M.Sc. thesis, "Single Video Super-Resolution" (in collaboration with TechniColor, Hannover, Germany).
- Fall 2012 M.Sc. semester project, "RGB and NIR Joint Demosaicing Using Compressive Sensing".
- **Spring 2012** B.Sc. semester project, "Joint and Disjoint Demosaicing of RGB and NIR Images".
- Fall 2011 M.Sc. semester project, "Face Detection Using NIR Images"

### **Teaching Experiences**

- 2012-2014 Teaching assistant, "Digital Photography", EPFL.
- 2011-2012 Teaching assistant, "Foundations of Imaging Sciences", EPFL.
- 2007, 2009 Teaching assistant, "Communication Systems", Sharif University.
- 2009 Laboratory instructor, "Electrical Engineering Principles", Sharif University.

### Computer Skills

• C++, MATLAB, MathCad.