MOTION DEBLURRING METHODOLOGIES: GOING BEYOND CONVENTIONAL CAMERAS

A THESIS

submitted by

MAHESH MOHAN M. R.

for the award of the degree

of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

January 2021

THESIS CERTIFICATE

This is to certify that the thesis titled **Motion Deblurring Methodologies: Going Beyond Conventional Cameras** submitted by **Mahesh Mohan M. R.** to the Indian Institute of Technology, Madras, for the award of the degree of **Doctor of Philosophy**, is a bona fide record of the research work carried out by him under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Place: Chennai
Date: January, 2021

Dr. A. N. Rajagopalan,
(Research Guide),
Professor
Dept. of Electrical Engg.
IIT Madras
Chennai - 600 036.

ACKNOWLEDGEMENTS

It has been a snake-and-ladder game – ladders, snakes, a big snake, ladder, ... Oh lost count of the snakes! First and foremost, I thank God, or nature or whatever that force is called, to give me the strength, even at my lowest ebb, to reach again to that dice, dream, and take chance. For sure, I as one cannot help myself in those times; many helped me in that act, even sometimes when I couldn't see any chance forward, and I believe no-one could. I am grateful to my guide Prof. A. N. Rajagopalan to help me tide over some difficult times and for not giving up on me, without which this thesis would not have been possible. I started this journey knowing almost nothing about doing and communicating research; now if I know something about research and look forward to learn and do more, I owe this mindset to his tiresome effort and guidance. Also, I am particularly grateful to Prof. Aravind R., who was always a well-wisher for me since the start of this journey, and whom I always believe will defend for me – a belief that used to give me immense strength throughout this journey.

Though I do not know much Signal Processing, I am glad to tell that I love that subject and I try to see all research problems in a Signal Processing perspective. I take this opportunity to thank few hidden figures who worked to kindle a flare of this subject within me; the credit of this thesis belongs to them as well. The list starts with my father who made me comfortable with numbers in my childhood, and my mother who believed in all my endeavours. Next in my list are my great teachers; to name a few, Karuppunni Sir and Neelakantan Sir instilled in me a taste of mathematics in my school-days; Prof. Suresh helped me to retain this spirit in my undergrad; and Prof. Pradeep Sarvepalli and Prof. Krishna Jagannathan in IIT Madras enlightened me with a notion that each mathematical claim has to have a solid proof. Also, I am indebted to Prof. A. N. Rajagopalan, Prof. Aravind R., Prof. Kaushik Mitra, and Prof. Sheeba V.S. in teaching me advanced Signal Processing subjects.

My list continues with my well-wishers, who motivated me when I could (or can) not see any chance forward. Teacher Sherin, Tr. Lekshmi, Tr. Nirmala, Tr. Beena, and Tr. Jayasree from my school, Chinmaya Vidyalaya Kunnumpuram, always provided me

with much needed hope. The unwavering support of Anusree teacher in my undergrad was my strength during several times; I still recollect she telling me “Mahesh can do”, even when I found myself very ill-equipped. My friends Dr. Dinesh Krishnamoorthy and Gopi Raju, and Prof. Kaushik Mitra play her role now, towards wishing me a Post-doc position. The next in my list are those from whom I noted many life-lessons: Prof. David Koilpillai for his affection, Mani Sir for his sincerity, Prof. Rajagopalan and Prof. Aravind for their discipline, and Prof. Pradeep Sarvepalli and Prof. Sheeba V. S. for their teaching preparations. My list is indeed long, and sadly, some figures are still hidden; but I am always thankful for them for making the good in me.

The works of Dr. Oliver Whyte immensely helped me in my literature study. I thank my doctoral committee members and the anonymous reviewers of my works whose valuable comments and suggestions helped me in shaping my thesis. Also, I thank my Thesis reviewers for providing constructive comments to improve this Thesis. I also thank all members of our IPCV lab: Sahana, Purna, Karthik, Abhijith, Vijay, Subeesh, Nimisha, Kuldeep, Arun, Sheetal, Praveen, Maithreya, Akansha, and Saurabh, and many others for their cheerful company. It was also exciting to work with Sharath, Sunil, and Nithin. I also thank my friends Nithin S., Dinesh K., Emmanuel, Dibakar, Gopi, Br. Vinod, Soumen, Anil, and Rana, who came for me whenever I needed any help. I would also like to express my love to my beautiful campuses GEC Thrissur and IIT Madras for all the blessings showered on me. I also acknowledge the financial support from Ministry of Human Resource and Develoment, India, and travel grants from Google, Microsoft, and ACM to participate in international conferences.

Finally, I would like to thank my family: Dr. S. Mohanachandran, Radhamany S, Maneesha Mohan M. R., Vishnu M., and Dhyuthi Mohan (late) for their love and support. My parents have made countless sacrifices for me, and have provided me with unwavering support and encouragement. Then there were often solo times in my life which could easily slip towards loneliness and lack of purpose, but in many of those times, there has been my Guardian Angel who doesn’t let me lonely, and takes me to a happy world, gives dreams, and waits eagerly till I start fighting for those dreams. This dissertation is dedicated to my parents, my teachers, and my Guardian Angel.

ABSTRACT

KEYWORDS: Blind motion deblurring, motion blur models, rolling shutter cameras, light field cameras, unconstrained dual-lens cameras, dynamic scene deblurring, deep learning.

Motion blur is a common artifact in hand-held photography. Presently, consumer cameras have gone beyond the conventional cameras in order to have additional benefits and functionalities. Three important such imaging devices are rolling shutter camera (with extended battery life, lower cost and higher frame rate), and light field camera and unconstrained dual-lens camera (which enable post-capture refocusing, varying the aperture (f-stopping), and depth sensing). Their increasing popularity has necessitated the need for tackling motion blur in these devices. In this thesis, we develop models and methods for these cameras aimed at "restoring" motion blurred photographs, where we have no particular information about the camera motion or the structure of the scene being photographed – a problem referred to as blind motion deblurring.

First, we tackle motion deblurring in rolling shutter cameras. Most present-day imaging devices are equipped with CMOS (complementary metal oxide semiconductor) sensors. Because CMOS sensors mostly employ a rolling shutter (RS) mechanism, the deblurring problem takes on a new dimension. Although few works have recently addressed this problem, they suffer from many constraints including heavy computational cost, need for precise sensor information, does *not* cater for wide-angle lenses (which most cell-phone and drone cameras have), and inability to deal with irregular camera trajectory. In Chapter 3, we propose a model for RS blind motion deblurring that mitigates these issues significantly. Comprehensive comparisons with state-of-the-art methods reveal that our approach not only exhibits significant computational gains and unconstrained functionality but also leads to improved deblurring performance.

Next, we consider the case of light field (LF) cameras. For LFs, the state-of-the-art blind deblurring method for general 3D scenes is limited to handling only downsampled

LF, both in spatial and angular resolution. This is due to the computational overhead involved in optimizing for a very high dimensional full-resolution LF *altogether* (e.g., a typical LF camera, `Lytro Illum`, contains 197 RGB images of size 433x625). Moreover, this optimization warrants high-end GPUs, which is seldom practical from a consumer-end. In Chapter 4, we introduce a new blind motion deblurring strategy for LFs which alleviates these limitations significantly. Our model achieves this by isolating 4D LF motion blur across the 2D subaperture images, thus paving the way for independent deblurring of these subaperture images. Furthermore, our model accommodates common camera motion parameterization across the subaperture images. Consequently, blind deblurring of any single subaperture image elegantly paves the way for cost-effective non-blind deblurring of the other subaperture images. Our approach is CPU-efficient computationally and can effectively deblur full-resolution LFs.

Subsequently, we move to the case of unconstrained dual-lens cameras. Recently, there has been a renewed interest in leveraging multiple cameras, but under unconstrained settings. They have been quite successfully deployed in smartphones, which have become the de facto choice for many photographic applications. However, akin to normal cameras, the functionality of multi-camera systems can be marred by motion blur. Despite the far-reaching potential of unconstrained camera arrays, there is not a single deblurring method for such systems. In Chapter 5, we propose a generalized blur model that elegantly explains the intrinsically coupled image formation model for dual-lens set-up, which are by far most predominant in smartphones. While image aesthetics is the main objective in normal camera deblurring, any method conceived for our problem is additionally tasked with ascertaining consistent scene-depth in the deblurred images. We reveal an intriguing challenge that stems from an inherent ambiguity unique to this problem which naturally disrupts this coherence. We address this issue by devising a judicious prior, and based on our model and prior propose a practical blind motion deblurring method for dual-lens cameras, that achieves state-of-the-art performance.

Finally, we focus on motion blur caused by dynamic scenes in unconstrained dual-lens cameras. In practice, apart from camera-shake, motion blur happens due to object motion as well. While most present-day dual-lens (DL) cameras are aimed at supporting extended vision applications, a natural hindrance to their working is the motion blur encountered in dynamic scenes. In Chapter 6, as a first, we address the problem of dynamic scene deblurring for unconstrained dual-lens cameras using Deep Learn-

ing and make three important contributions. We first address the root cause of view-inconsistency in the generic DL deblurring network using a coherent fusion module. We then tackle the inherent problem in unconstrained DL deblurring that violates the epipolar constraint by introducing an adaptive scale-space approach. Our signal processing formulation allows accommodation of lower image-scales in the same network without increasing the number of parameters. Finally, we propose a filtering scheme to address the space-variant and image-dependent nature of blur. We experimentally show that our proposed techniques have substantial practical merit.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

BMD	Blind motion deblurring
CC	Conventional camera
CCD	Charge-coupled device
CMOS	Complimentary metal oxide semiconductor
GS	Global shutter
RS	Rolling shutter
RGB	Red green blue
n**D**	n dimensional (e.g., 2D, 3D, 4D, and 6D.)
FOV	Field of view
PSF	Point spread function
MDF	Motion density function
LASSO	Least absolute shrinkage and selection operator
LARS	Least angle regression
ADMM	Alternating direction method of multipliers
EFF	Efficient filter flow
FFT	Fast Fourier transform
MAP	Maximum a posteriori
TV	Total variation
PSNR	Peak signal-to-noise ratio
dB	Decibel
SSIM	Structural similarity measure
RMSE	Root mean square error
IFC	Information fidelity criterion
VIF	Visual information fidelity
LFC	Light field camera
LF	Light field
SA	Sub-aperture
SAI	Sub-aperture image

CPU	Central processing unit
GPU	Graphical processing unit
DL	Dual-lens
HDR	High dynamic range
COR	Center of rotation
Enc/Dec	Encoder/Decoder
CNN	Convolutional neural network
ReLU	Rectified linear unit
SvId	Space-variant Image-dependent
ASPP	Atrous spatial pyramid pooling
BS	Boot-strapped
MAE	Mean absolute error
GT	Ground truth

NOTATION

$\mathbf{B}$	Blurred image
$\mathbf{L}$	Latent (clean) image
f	Focal length of the camera
$\mathbf{K}$	Intrinsic camera matrix
diag(a, b, c)	Diagonal matrix with diagonal elements a, b, and c.
$H(\cdot)$	Homography mapping
$\mathbf{x}$	Homogeneous (3D) sensor coordinate
$\mathbf{X}$	3D world coordinate
Z	Scene depth
$M \times N$	Row-column dimension of an image
r_b	Row-block size
n_b	Number of row-blocks in an image $\left(= \lceil \frac{M}{r_b} \rceil\right)$
$\mathbf{B_i}$	ith row-block of blurred image $\mathbf{B}$
$\mathbf{L_i}$	ith row-block of Latent image $\mathbf{L}$
$\mathbf{P}$	Continuous camera pose-space
$\mathbb{P}$	Discrete camera pose-space
$\mathbf{p}(t_0)$	Camera pose at time instant t_0
$\mathbf{R}, \mathbf{t}$	Camera rotation matrix and translation vector
$\mathbf{L^p}$	Latent image $\mathbf{L}$ transformed according to the pose $\mathbf{p}$
$\Gamma(r)$	Percentage camera-pose overlap in a row-block of size r
$\mathbf{h}$	Point spread function (PSF)
$\mathbf{w}, w(p)$	Motion density function (MDF)
$\hat{(\cdot)}$	Estimate
t_e	Exposure time (shutter speed)
t_r	Inter-row delay
δ	Impulse function
∇	Gradient operator
$\mathbb{F}(\cdot), \mathbb{F}^{-1}(\cdot)$	Forward and inverse FFT
$\mathbf{LF}_B, \mathbf{LF}_L$	Blurred and latent/clean light fields
$\{k_x, k_y\}$	Axial separations from the lens center
u	Lens-sensor separation
u_s	Focusing point of a scene-point s
k_{xy}	Variable used to denote the subaperture at $\{k_x, k_y\}$
$\mathbf{B}_{k_{xy}}$	Blurred k_{xy}th subaperture image
$\mathbf{L}_{k_{xy}}$	Latent k_{xy}th subaperture image
$\mathbf{K}_{k_{xy}}$	Intrinsic camera matrix for the k_{xy}th subaperture
$\mathbf{h}_{k_{xy}}$	PSF for the k_{xy}th subaperture image
$\mathbf{l_b}$	Baseline vector
$\mathbf{l_c}$	Center-of-rotation vector
$(\cdot)^n$	Quantities of narrow-angle configuration
$(\cdot)^w$	Quantities of wide-angle configuration

$\mathbf{I}$	Identity matrix
$L_{\mathbf{l_c}}$	Cost function for the COR ($\mathbf{l_c}$)
$\{\mathbf{W}, \mathbf{W}'\}$	Bilinear masks ($\mathbf{W}' = 1 - \mathbf{W}$)
$\odot$	Kronecker product
$(\cdot)^L$	Quantities of left-view image
$(\cdot)^R$	Quantities of right-view image
L_{LR} and L_{RR}	Costs for view-consistency
$\downarrow D$	Decimation by a factor D
$\uparrow D$	Interpolation by a factor D
$*$	Convolution operation
$T(\cdot)$	Mapping of encoder-decoder network
$R(\mathbf{h})$	Receptive field of the filter $\mathbf{h}$
σ	Standard deviation
ω	Frequency domain

If you can meet with Triumph and Disaster
And treat those two impostors just the same;
Or watch the things you gave your life to, broken,
And stoop and build'em up with worn-out tools:
If you can make one heap of all your winnings
And risk it on one turn of pitch-and-toss,
And lose, and start again at your beginnings
And never breathe a word about your loss;
If you can force your heart and nerve and sinew
To serve your turn long after they are gone,
And so hold on when there is nothing in you
Except the Will which says to them: 'Hold on!'
If you can fill the unforgiving minute
With sixty seconds' worth of distance run,
Yours is the Earth and everything that's in it ...

(From "If" — *Rudyard Kipling*)

The moon smiles bright as if,
she finds a truth as is,
for at night, souls all raw she sees,
that the Soul in his art,
is none but his Soul apart ...

("Soul" — *mmmr*)

CHAPTER 1

Introduction

Owing to the light-weightedness of today's cameras, motion blur is an ubiquitous phenomenon in hand-held photography. Motion blur is caused by relative motion between camera and scene during the exposure interval. Specifically, a motion blurred image is formed by aggregation of different world-to-sensor projections of the scene, over the exposure interval, onto the image sensor. One solution to reduce blur is by lowering the exposure interval. However, this is not typically preferred due to the inherent noise in imaging; moreover, this solution is seldom practical in low-light scenarios or small-aperture imaging common in mobile-phones and light field cameras. The challenging problem of blind motion deblurring (BMD) deals with estimating a clean image from a motion blurred observation, *without* any knowledge of scene and camera motion. Since most computer vision works are designed for blur-free images and blur derails most of these tasks (Kupyn *et al.*, 2018; Vasiljevic *et al.*, 2016; Dodge and Karam, 2016), BMD is a continuing research endeavour.

Recently, there has been a popular trend in employing cameras beyond conventional cameras (CCs) in order to have additional benefits and functionalities. For instance, most present-day cameras are equipped with rolling shutter sensors, which employ a row-wise world-to-sensor projection of the scene (different from that of concurrent projection in CC), in order to increase frame-rate and to reduce power consumption and cost. Yet another example is that of popular light field cameras and unconstrained dual-lens cameras popularized by today's mobile-phones, which capture multiple images (as opposed to a single image in CC) so as to obtain depth information and to enable post-capture refocusing and f-stoping. Motion blur is a pertinent problem in these non-conventional cameras as well, but it manifests in a different form.

Blind motion deblurring is a well-studied topic in CC, replete with several models and methods. However, these works are not applicable to the non-conventional cameras due to their different imaging mechanism or world-to-sensor projections. Moreover, as the image information in non-conventional cameras is utilized for extended functionalities, a BMD method for these cameras has to ensure that it does not degrade the

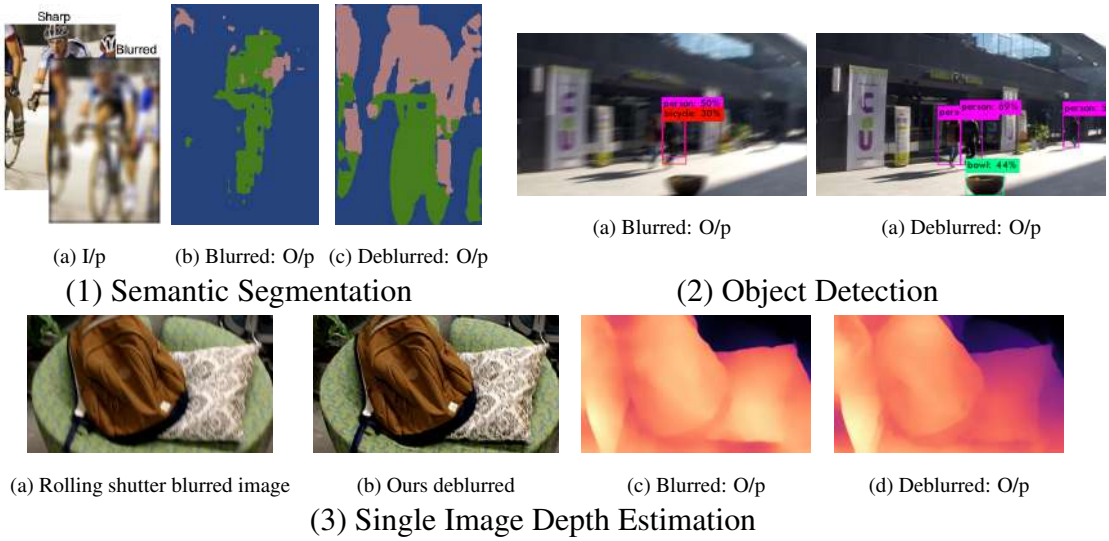


Figure 1.1: Motion Deblurring as a pre-processing for high-level vision tasks: 1(a-c) Semantic segmentation (Vasiljevic *et al.*, 2016), where Fig. 1(c) shows that motion deblurring enables better segmentation of semantic objects (e.g., bicycle and person). 2(a-b) Object classification (Kupyn *et al.*, 2018) where the deblurred result in Fig. 2(b) leads to enhanced detection. 3(a-d) Single image depth estimation using (Poggi *et al.*, 2018) from rolling shutter (RS) blurred image and RS deblurred image using our method (Chapter 3). Comparing Figs. 3(c,d), motion deblurring leads to better preservation of object boundaries (e.g., pillow and chair).

required information (e.g., scene-structure or depth cues). Finally, unlike CCs, light field and dual-lens cameras capture multiple images of a scene; therefore corresponding BMD methods have to tackle associated computational complexity in optimizing for multiple clean images (as compared to only one image in CC-BMD).

1.1 Motivation and Objectives

The terrain of consumer cameras today spans beyond the conventional cameras. Apart from the extended functionalities offered by the non-conventional cameras, the added benefits of being lightweight, portable, and their adoption in standard imaging gadgets (like mobile-phones) have brought these cameras to the forefront in the consumer market. However, motion blur is difficult to avoid completely while capturing scenes using hand-held devices. Motion blur has the detrimental effect of derailing the aesthetic value of the captured images; in addition, most computer vision tasks warrant blur-free inputs. Our work in this thesis attempts to address the problem of motion blur in different non-conventional cameras, such as rolling shutter cameras, light field cameras, and

unconstrained dual-lens cameras. Apart from restoring blurred images, the solutions discussed here can serve as a potential preprocessing step for many computer vision tasks based on these cameras, in order to extend their scope to handle ubiquitous motion blurred observations. This is illustrated in Fig. 1.1 for high-level vision tasks such as semantic segmentation, object detection, and single image depth estimation. Next we discuss the motivation and objectives of the problems addressed in this thesis.

Motion Deblurring for Rolling Shutter Cameras: Today, most cameras employ rolling shutter (RS) sensors for higher frame-rate, extended battery life, and lower cost. As compared to the concurrent exposure of sensor-rows in traditional global shutter cameras, the sensor-rows in RS camera integrate light in a staggered manner. Therefore under the effect of camera motion, each row in an RS sensor perceives different camera motion, and hence different motion blur. As BMD methods for CC assume that motion blur in all the image-rows are due to the same camera motion, those methods are *not* applicable to RS cameras (Su and Heidrich, 2015).

Moreover, the state-of-the art method for RS-BMD (Su and Heidrich, 2015) has several limitations. First, it is effective *only* for narrow-angle settings, whereas wide-angle configuration is a prominent setting in most DSLR cameras, mobile phones and drones. Second, the method warrants precise sensor timings for deblurring, which necessitates camera calibration. Therefore, this method is not effective in deblurring *arbitrary* RS images, e.g., images obtained from internet. Third, this method is limited to parametric ego-motion derived primarily to characterize hand-held trajectories. Hence, it cannot handle blur due to moving or vibrating platforms which are common in robotics, drones, etc., where the ego-motion is typically irregular. Another significant limitation of this method is its heavy computational cost, as compared to typical CC-BMD methods.

To this end, we introduce a motion blur model for RS, which resembles the global shutter blur model but is expressive enough to capture the RS mechanism. Our model acts as a bridge between well-studied CC-BMD and contemporary RS-BMD, in that we propose to extend the scope of efficient techniques developed for the former to the latter. Moreover, we identify a hindrance in readily applying the CC-BMD techniques to RS-BMD; in particular, we show that there exists an ill-posedness in RS-BMD that corrupts scene-information in the deblurred images. We address this ill-posedness using a convex and computationally efficient prior. We show that RS deblurring using our

model and prior achieves state-of-the-art results, and at the same time accommodates narrow- and wide-angle settings, eliminates the need of camera calibration, can handle irregular camera motion, and has a computationally efficient optimization framework.

Full-Resolution Light Field Deblurring: Of late, light field cameras (LFCs) have became popular due to their attractive features over conventional cameras, such as post-capture refocusing, f-stoping, and depth sensing. These features in LFCs are enabled by capturing multiple images, each imaged through a portion of lens-aperture that is open (e.g., 197 images in `Lytro Illum`), as compared to only a single image in CCs. Due to this lens-division, image formation in an LFC is very different from that of a CC, and hence the motion blur model and the BMD methods for CC fail for light fields (Srinivasan *et al.*, 2017). Furthermore, LFCs introduce an additional challenge in deblurring that, unlike in CC or RS-BMD, calls for a method to deblur multiple images using modest computational resources and within a reasonable processing time.

The state-of-the-art method for LF-BMD (Srinivasan *et al.*, 2017) has some major drawbacks. First, the method is too computationally intense that it is limited to handle only down-sampled LFs for practical feasibility. Note that down-sampling the LFs results in an inferior performance of its post-capture capabilities. Moreover, the method considers for optimization a full light field altogether, which warrants GPU-based processing and also leads to convergence issues. Further, this method is limited to narrow-angle configuration and parametric camera motion.

To address these limitations, we introduce a new LF motion blur model which decomposes the LF-BMD problem into independent subproblems (by isolating blur in individual subaperture images or SAIs). Employing this model, we advocate a divide and conquer strategy for LF-BMD; more specifically, we introduce a deblurring scheme such that deblurring a single SAI greatly reduces the complexity in deblurring the remaining SAIs. Due to the independent nature of the subproblems, our methods can deblur full-resolution light fields and eliminates the need of GPU-processing. Further, light field deblurring based on our proposed motion blur model accommodates both narrow- and wide-angle configurations, and irregular camera motions.

Deblurring for Unconstrained Dual-lens Cameras: The world of smartphones today is experiencing a profilteration of dual-lens (DL) cameras so as to enable additional post-capture renderings, which are achieved by utilizing depth cues embedded in the

DL images. The cameras typically employed in DL-smartphones are of unconstrained nature, i.e., the two cameras can have different focal lengths, exposure times and resolutions. The problem of BMD in unconstrained DL cameras has additional challenges over that of CCs. First, a DL set-up warrants deblurring based on depth, whereas CC-BMD is oblivious to depth-cues. Therefore, any errors in depth can adversely affect DL deblurring performance, and hence need to additionally address ill-posedness in depth, if any. Second, any method for DL-BMD has to ensure scene-consistent depth in the deblurred image-pair. We show that naively applying CC-BMD in unconstrained DL set-up easily disrupts this depth consistency, thereby sabotaging the functionalities of DL cameras. Also, the popular trend of including narrow-FOV camera in a DL set-up amplifies the adverse effect of motion blur.

The existing BMD methods for DL and LF cameras are *not* effective for unconstrained DL: The state-of-the-art DL-BMD (Zhou *et al.*, 2019; Xu and Jia, 2012) necessitates a constrained DL set-up, i.e., two cameras need to work in synchronization and share the same settings. Therefore, these methods are not applicable for unconstrained DL cameras. Further, the method of (Xu and Jia, 2012) assumes that blur is primarily caused by inplane camera-translations and warrants a layered depth scene, which is seldom practical (Whyte *et al.*, 2012). The light field BMD directly applied for the problem of unconstrained DL-BMD also fails, as the LFC-BMD method assumes multiple images to share the same setting (which is inherent to LF cameras, but does not hold good for unconstrained DL). Further, most LF-BMD methods warrant more than two images for deblurring, but it *cannot* be supplied by unconstrained DL cameras.

As a first, we address the problem of BMD in unconstrained DL cameras. To this end, we introduce a DL-blur model that seamlessly accommodates both unconstrained and constrained DL configurations with arbitrary center-of-rotation (COR). Second, we reveal an inherent ill-posedness present in DL-BMD that naturally disrupts scene-consistent disparities. We address this using a convex prior on ego-motion. To eliminate the difficulty in deblurring more than one image (as compared to that of CC-BMD), we propose a decomposition of DL-BMD problem while enforcing our DL-prior, which leads to a practical BMD method for today's unconstrained DL cameras.

Dynamic Scene Deblurring for Unconstrained Dual-lens: Apart from camera motion, motion blur happens due to object motion as well. This renders those DL-BMD

methods that restrict to only camera motion induced blur (as discussed in the previous portions) ill-equipped for several practical scenarios. Another important challenge presented by today's unconstrained DL genre is due to its different resolutions and exposure times. This renders feature loss due to blur in the two views different, and hence typical deblurring methods produce binocularly inconsistent deblurred image-pairs. However, almost all computer vision methods for stereoscopic applications require the two views to be binocularly consistent.

The only-existing dynamic scene deblurring method for DL (Zhou *et al.*, 2019) restricts to constrained DL configuration. Therefore, the problem of binocular consistency does *not* arise here and hence has not invoked. The BMD method we discussed before for unconstrained DL (Mohan *et al.*, 2019) also does *not* work for this problem as it is restricted to blur induced by camera motion alone. Moreover, there was *no* attempt to address the problem of view consistency. Typical strategy to address dynamic scene deblurring is via a complex pipeline of segmenting independently moving objects, estimating relative motion in individual segments, and finally, deblurring and stitching individual segments. Due to the presence of large number of unknowns, this approach is computationally very intensive, and hard to optimize.

To alleviate this problem, we propose a deep learning based method for dynamic scene deblurring in unconstrained DL cameras, a first of its kind. Our approach accomplishes this by learning a mapping from unrestricted DL data, that does *not* involve complex pipelines and optimizations while deblurring. We propose three interpretable modules optimized for unconstrained DL that effectively produce binocularly consistent output images and address the space-variant and image-dependent nature of blur, which altogether achieves state-of-the-art deblurring results for unconstrained DL set-up.

1.2 Contributions of the Thesis

The main contributions of this thesis can be summarized as follows:

Chapter 3: We introduce a new rolling shutter motion blur model, and based on the model proposed an RS-BMD method which overcomes some of the major drawbacks of the state-of-the-art method (Su and Heidrich, 2015), including inability to handle wide-angle systems and irregular ego-motion, and the need for sensor data. We also extend the efficient filter flow framework (Hirsch *et al.*, 2010, 2011) to RS deblurring, thereby achieving a speed-up of atleast eight.

Chapter 4: By harnessing the physics behind light field (LF), we decompose 4D LF-BMD to 2D subproblems, which enables the first ever attempt of full-resolution LF-BMD. This formulation bridges the gap between the well-studied CC-BMD and emerging LFC-BMD, and facilitates mapping of analogous techniques (such as MDF formulation, efficient filter flow framework, and scale-space strategy) developed for the former to the latter. Our proposed method dispenses with some important limitations impeding the state-of-the-art (Srinivasan *et al.*, 2017), such as high computational cost and GPU requirement.

Chapter 5: As a first, we formally address BMD problem in unconstrained dual-lens configurations. We introduce a *generalized* DL blur model, that also allows for arbitrary COR. Next, we reveal an inherent *ill-posedness* present in DL-BMD, that disrupts scene-consistent disparities. To address this, we propose a prior that ensures the biconvexity property and admits efficient optimization. Employing our model and prior, we propose a practical DL-BMD method that achieves state-of-the art performance. It ensures scene-consistent disparities, and accounts for the COR issue (for the first time in BMD framework).

Chapter 6: For the first time in the literature, we explore dynamic scene deblurring in today's ubiquitous unconstrained DL camera. First, we address the pertinent problem of view-inconsistency inherent in unconstrained DL deblurring, that forbids most DL-applications, for which we propose an *interpretable coherent-fusion* module. Second, our work reveals an inherent issue that disrupts scene-consistent depth in DL dynamic-scene deblurring. To address this, we introduce an *adaptive multi-scale* approach in deep learning based deblurring. Finally, we extend the widely applicable atrous spatial pyramid pooling (Chen *et al.*, 2017) to address the space-variant and image-dependent nature of dynamic scene blur.

1.3 Organization of the Thesis

The rest of the thesis is structured as follows. Chapter 2 provides some technical background, which covers standard motion blur model and deblurring method of conventional cameras and discuss different optimization techniques. In Chapter 3, we introduce a rolling shutter motion blur model, and based on the model propose an RS-BMD method that also incorporates a prior to alleviate the ill-posedness. Chapter 4 discusses a full resolution light field deblurring method, based on divide and conquer strategy. In Chapter 5, we propose a motion deblurring method for unconstrained dual-lens cameras, which ensures scene-consistent depth while deblurring using a convex prior. Chapter 6 extends motion deblurring for unconstrained dual-lens cameras (in Chapter 5) by accommodating dynamic scenes as well, using a deep learning approach. We conclude the thesis in Chapter 7 with some insights into future directions.

CHAPTER 2

Technical Background

The problem of blind motion deblurring (BMD) deals with estimating camera motion and sharp photograph from a single blurred photograph. This problem can be divided into two different parts: First, a motion blur model is required to relate the sharp photograph to the observed blurred photograph via camera motion parameters. Second, employing this model, camera motion parameters and corresponding sharp photograph have to be estimated. Since the (unknown) sharp photograph has the same number of pixels as the (observation) blurry photograph, and camera motion parameters further increase the unknown dimension, we clearly have more unknowns than observations. Therefore, the problem of BMD is a heavily ill-posed problem, and the associated estimations warrant priors for sharp image and camera motion.

The very first comprehensive study on BMD happened for conventional cameras (CCs). Refer to (Rajagopalan and Chellappa, 2014) for a detailed survey on this area. As our research topic of BMD for non-conventional cameras is not well explored, whereas BMD for CCs is a well-studied area replete with efficient techniques, we invoke some mature concepts from CC-BMD which we discuss in this chapter. Specifically, first we describe a standard motion blur model of CCs, which is followed by a discussion on standard natural image priors and priors on camera motions, and conclude with standard algorithms employed for sharp image and camera motion estimation.

2.1 Motion Blur Model for Conventional Camera

The initial works on BMD assume that motion blur is space-invariant, i.e., a blurred image is modelled as the convolution of sharp image with a convolution kernel (Cho and Lee, 2009; Yuan *et al.*, 2017; Zhang *et al.*, 2013; Wang *et al.*, 2013; Zhu *et al.*, 2012; Sroubek and Milanfar, 2012). However, several later works showed that, in general, motion blur due to 6D motion and 3D scenes are typically space-variant (i.e., convolution kernel at each spatial locations are different). Since the high-dimensional 6D

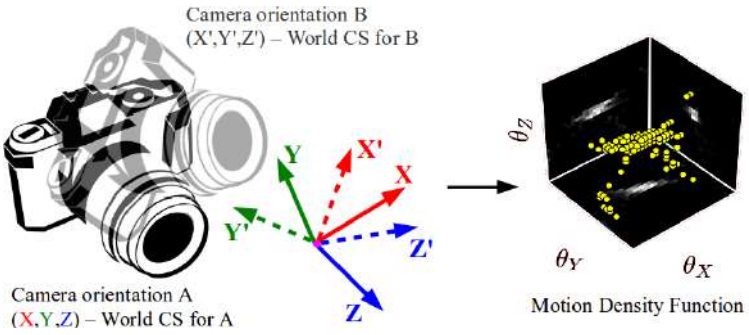


Figure 2.1: Motion Density Function (MDF): Change in camera orientation from A to B is equivalent to the relative change in world coordinate system (CS) from $\mathbf{X}$ to $\mathbf{X}'$. Thus, MDF, which gives the fraction of time the world CS stayed in different poses during the exposure time, *completely* characterizes the camera motion.

camera pose-space leads to higher computational cost and convergence issues, current methods consider a lower dimensional approximation. For instance, Gupta *et al.* (2010) proposed a 3D approximation for general 6D camera pose-space by considering only inplane translations and rotations, while Whyte *et al.* (2012) considered only 3D rotations. Köhler *et al.* (2012) showed that both these 3D models are good approximations to general pose-space. However, the model in (Whyte *et al.*, 2012) is employed in most deblurring algorithms (Pan *et al.*, 2016; Xu *et al.*, 2013) as it requires *no* depth information unlike the model in (Gupta *et al.*, 2010). (Note that estimating depth from a *single* motion blurred image is heavily ill-posed (Hu *et al.*, 2014).)

We now discuss a standard motion blur model for conventional cameras, proposed by Whyte *et al.* (2012). Here, the CC is approximated as a pinhole at lens-center, and camera motion is interpreted as stationary camera but with relative world motion (see Fig. 2.1). Considering full-rotations approximation and a single camera pose change, the relative change in world coordinate is given as

$$\mathbf{X}' = \mathbf{R}\mathbf{X}, \tag{2.1}$$

where $\mathbf{R}$ is rotation matrix, and $\mathbf{X} = [X, Y, Z]^T$ and $\mathbf{X}' = [X', Y', Z']^T$ are the 3D world coordinates with respect to initial and final camera positions, respectively. For the world pose-change in Eq. (2.1), a homography mapping H relates the corresponding displacement in homogeneous image coordinates as

$$\mathbf{x}' = H(\mathbf{K}, \mathbf{R}, \mathbf{x}), \tag{2.2}$$

where $\mathbf{K}$ is the camera matrix, and $\mathbf{x}$ and $\mathbf{x}'$ are 2D image coordinates corresponding to the initial and final camera positions. Note that homography mapping can be different for different cameras in accordance with their imaging principles. For conventional camera, the homography mapping is $\mathbf{x}' = \lambda \mathbf{K}\mathbf{R}\mathbf{K}^{-1}\mathbf{x}$, where $\mathbf{K} = \text{diag}(f, f, 1)$ where f is the focal length, and λ normalizes the third coordinate of $\mathbf{x}'$ to one (Hartley and Zisserman, 2003). Here, the resultant image $\mathbf{L}'$ due to the world pose-change (in Eq. (2.1)) can be related to the initial image $\mathbf{L}$ as

$$\mathbf{L}' = \mathbf{L}\left(\mathbf{K}, \mathbf{R}\right), \tag{2.3}$$

where $\mathbf{L}(\mathbf{K}, \mathbf{R})$ performs warping of image $\mathbf{L}$ in accordance with Eq. (2.2). Thus a general motion blurred image $\mathbf{B}$ (wherein camera experiences multiple pose-changes over its exposure time) can be expressed as

$$\mathbf{B} = \sum_{\mathbf{p} \in \mathbb{P}} w(\mathbf{p}) \cdot \mathbf{L}(\mathbf{K}, \mathbf{R}_{\mathbf{P}}), \tag{2.4}$$

where $\mathbf{R}_{\mathbf{p}}$ spans the plausible camera pose-space $\mathbb{P}$ and $w(\mathbf{p_0})$ is the motion density function (MDF) which gives the fraction of exposure time the camera stayed in the pose $\mathbf{R}_{\mathbf{p_0}}$ (Fig. 2.1-right). Note that the MDF completely characterizes the camera-shake, and can capture regular and irregular camera motion (as *no* particular camera-trajectory path is imposed in MDF) (Whyte *et al.*, 2012). Further, the consideration of full 3D rotations in MDF accommodates both narrow-angle and wide-angle configurations (Su and Heidrich, 2015).

Another important consideration is on how finely the rotational pose-space need to be discretized. Undersampling the set of rotations will affect the ability to accurately reconstruct the blurred image, but sampling it too finely warrants higher computational costs for estimation. For example, as the kernel is defined over the three rotational dimensions, doubling the sampling resolution increases the number of kernel elements by a factor of eight, therefore the choice of sampling is important. It is shown in (Whyte *et al.*, 2012) that, in practice, a good choice of sample spacing is one which corresponds approximately to a displacement of one pixel at the edge of the image. Since images are fundamentally limited by their resolution, reducing further the sample spacing leads to redundant rotations, that are indistinguishable from their neighbours.

2.2 Image and Camera Motion Priors

As discussed earlier, the problem of BMD is inherently ill-posed. Therefore, a proper estimation of unknowns requires additional priors on image and camera motion. Note that the model in Eq. (2.4) admits a linear relation with the sharp image and MDF individually, which is desirable to achieve a least square objective for the data-fidelity term (assuming noise is additive white Gaussian). From Eq. (2.4), the optimization cost for the unknowns clean image and camera motion can be obtained as

$$\begin{aligned} L = \|\mathbf{A}\mathbf{w} - \mathbf{B}\|_2^2 \ + \text{Prior}(\mathbf{L}) + \text{Prior}(\mathbf{w}), \\ \text{where } \|\mathbf{A}\mathbf{w} - \mathbf{B}\|_2^2 \ = \|\mathbf{M}\mathbf{L} - \mathbf{B}\|_2^2. \end{aligned} \tag{2.5}$$

where $\mathbf{L}$ is the clean image (in lexicographical form), and $\mathbf{w}$ is the vectorized form of $w(\mathbf{p})$ (where $\mathbf{p}$ is an element of the pose-space $\mathbb{P}^3$). The terms $\|\mathbf{A}\mathbf{w} - \mathbf{B}\|_2^2$ and $\|\mathbf{M}\mathbf{L} - \mathbf{B}\|_2^2$ are the data fidelity terms, which are obtained from Eq. (2.4) as follows: For MDF $\mathbf{w}$, Eq. (2.4) enforces a linear relation via warp matrix $\mathbf{A}$, wherein its ith column contains the warped version of clean image $\mathbf{L}$, with the pose of $w(\mathbf{p})$. For clean image $\mathbf{L}$, Eq. (2.4) enforces a linear relation via PSF matrix $\mathbf{M}$, wherein its ith column contains the point spread function (PSF) corresponding to the ith coordinate (Whyte *et al.*, 2012; Xu *et al.*, 2013). In what follows, we discuss the standard priors employed for the sharp image and camera motion in order to address the ill-posedness in BMD.

2.2.1 Priors for Sharp Image

One of the most popular regularisers for sharp image is the sparse gradient prior, which penalises the derivatives or gradients of the deblurred image. It is given as

$$\text{Prior}(\mathbf{L}) = \|\nabla\mathbf{L}\|_p \tag{2.6}$$

where $\nabla\mathbf{L}$ is the gradient of the image $\mathbf{L}$. Algorithmically, $\nabla\mathbf{L}$ is obtained by convolving the image with a first order horizontal and vertical filter which is implemented as a matrix multiplication of the sharp image (since the convolution operation is linear (Oppenheim and Schafer, 2014)). Though it is possible to extend the regularisation to higher-order derivatives, this is not generally done in practice due to the added com-

putational cost. Typically, the value of p is selected to be 1 and it is referred to as total variation (TV) prior. The TV prior is convex and is shown to have excellent convergence and efficient solvers (Perrone and Favaro, 2014), e.g., ADMM (Boyd and Vandenberghe, 2004). It is to be noted that p less than one is also employed, but in that case the prior becomes non-convex and difficult to optimize. For example, Krishnan and Fergus (2009) advocate a p between 0.5 and 0.8, which is referred to as hyper-Laplacian prior, whereas Levin *et al.* (2007) advocate the value of p to be 0.8, and Xu *et al.* (2013) considers its value as 0.

2.2.2 Priors for Camera Motion

In the domain of camera pose-space, the camera motion is a 1D path that captures the trajectory of the camera during its exposure interval. Based on this cue, there exist two computationally tractable priors for MDF, i.e.,

$$\text{Prior}(\mathbf{w}) = \lambda_1 \|\mathbf{w}\|_1 + \lambda_2 \|\nabla \mathbf{w}\|_0, \tag{2.7}$$

such that $\mathbf{w}(\mathbf{p}) \geq 0$ (as MDF components indicates the fraction of time, and hence non-negative). The first component in Eq. (2.7) is a sparsity prior on the MDF values. While blur kernels in the 2D image space appears quite dense, it is shown that a 1D camera path represents an extremely sparse population in the higher dimensional MDF space (Gupta *et al.*, 2010). Therefore, the l_1 regularisation combined with non-negativity constraints encourages the optimization to find a sparse MDF and is more likely to choose between ambiguous camera orientations, in contrast with spreading non-zero values across all orientations. The second component is a smoothness prior on the MDF, which incorporates the concept of the MDF representing a path, as it enforces continuity in the space and captures the cue that a given pose is more likely if its nearby pose is likely. As discussed in image priors, the optimization with the second term incurs heavy computational cost for MDF estimation (and there exists no standard optimization framework when both the priors are included). Therefore the current methods predominantly use only the first component (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012). As we will see in the next section, the resultant optimization for MDF leads to a well-studied, computationally efficient solver.

2.3 Motion Deblurring for Conventional Cameras

Blind motion deblurring in conventional cameras typically proceeds by alternating minimization (AM) of MDF $\mathbf{w}$ and sharp image $\mathbf{L}$, i.e., iteratively optimize for the one unknown assuming that the other quantity is known, in an alternating fashion. Also, the AM proceeds in a scale-space manner to accommodate large blurs while keeping the optimization dimension low (Whyte *et al.*, 2012), i.e., MDF estimation starts with a downsampled blurred image where the MDF-dimension is less, and proceeds to finer scale MDF-estimation by leveraging the sparsity of the previous estimate. To be specific, MDF at the original resolution may have thousands or tens of thousands of elements. However, due to the sparse nature of MDF a very few of these should have non-zero values. Solving for the full-dimensional MDF is not desirable as it leads to significant amounts of redundant computation, since most of the MDF entries will correspond to zeros. Instead by proceeding in a scale-space, one can restrict the support of the pose-space of higher scales as the dilated support of the MDF estimated at a lower scale. We now discuss the estimation techniques of MDF and latent image.

2.3.1 Estimation of Camera Motion

In AM, assuming that the sharp image is known, the MDF estimation is given as

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \|\mathbf{A}\mathbf{w} - \mathbf{B}\|_2^2 + \lambda_1 \|\mathbf{w}\|_1 \quad : w(p) \geq 0, \tag{2.8}$$

where we have employed sparsity prior for MDF. This is an instance of the Lasso (least absolute shrinkage and selection operator) problem (Tibshirani, 1996), for which efficient optimisation algorithms exist ((Efron *et al.*, 2004)). More important, this is a convex problem, so that we can be sure of attaining a global minimum. For the coarsest scale, the sharp image is initialized as the shock-filtered blurred image, and in all other levels the previous estimate of sharp image is employed to frame the MDF cost (in Eq. (2.8)). Also, it is a common practice in deblurring algorithms to frame the MDF cost in the gradient domain for faster convergence and to reduce ill-conditionness (Xu *et al.*, 2013; Whyte *et al.*, 2012; Cho and Lee, 2009; Hirsch *et al.*, 2011).

It is to be noted that there exist methods that use l_2 prior on MDF (instead of the

l_1 prior) (Cho and Lee, 2009), but it is shown in (Whyte *et al.*, 2012) that the resultant estimate is highly non-sparse, and the deblurred result with this MDF exhibits many artifacts as compared to the result of l_1.

2.3.2 Estimation of Clean Image

Assuming that the MDF is known (which is considered as the MDF estimate in the previous iteration), the sharp image estimation is obtained using Eq. (2.5) as

$$\hat{\mathbf{L}} = \min_{\mathbf{L}} \|\mathbf{ML} - \mathbf{B}\|_2^2 + \|\nabla \mathbf{L}\|_1, \tag{2.9}$$

where the TV natural image prior is employed (Whyte *et al.*, 2012). Note that the optimization in Eq. (2.9) is a convex problem, and hence guarantees a global minima. Also, Eq. (2.9) is in a standard form encountered in many restoration tasks (such as super-resolution, deblurring, and denoising) and can be effectively solved using ADMM (Boyd and Vandenberghe, 2004). However, as motion deblurring proceeds in a scale-space manner (as discussed earlier), the creation of matrix $\mathbf{M}$ after every update of MDF and optimizing for a high-dimensional sharp image using TV prior for *every* scale and *every* iteration is not computationally efficient. To alleviate this problem, there exists simplified, efficient framework for sharp image estimation, which is discussed next.

Efficient Filter Flow: Hirsch *et al.* (2010) showed that motion blur in practice varies slowly and smoothly across the image. As a result, the PSFs of nearby pixels can be very similar, and hence it is reasonable to approximate spatially-variant blur as being locally-uniform. Following this finding, Hirsch *et al.* (2010) advocated a simplified forward motion blur model, wherein the sharp image is covered with a coarse grid of overlapping patches, each of which is modelled as having a spatially-invariant blur. The overlap between patches enforces the smoothly varying nature of motion blur across the image, rather than blur changing abruptly between neighbouring patches. As each patch has a spatially-invariant blur, the forward model translates to computing N small convolutions as follows:

$$\mathbf{B} = \sum_{k=1}^{N} \mathbf{C}_{\mathbf{k}}^{\dagger} \cdot \left\{ \mathbf{h}^{(k)} * (\mathbf{C}_{\mathbf{k}} \cdot \mathbf{L}) \right\}, \tag{2.10}$$

where N is the total number of overlapping patches in latent image $\mathbf{L}$, k is the patch-index, $\mathbf{C_k} \cdot \mathbf{L}$ is a linear operation which extracts the kth patch from $\mathbf{L}$, and $\mathbf{C_k^\dagger}$ inserts the patch back to its original position with a windowing operation. The $\mathbf{h}^{(k)}$ represents blur kernel or point spread function (PSF) which when convolved with the kth latent image-patch produces the corresponding blurred patch. Given the MDF ($\mathbf{w}$) and the homography mapping, the PSF for the kth patch is obtained as

$$\mathbf{h}^{(k)} = \sum_{\mathbf{p} \in \mathbb{P}} w(\mathbf{p}) \cdot \mathbf{h}_k(\mathbf{p}), \tag{2.11}$$

where $\mathbf{h}_k(\mathbf{p})$ is a shifted impulse obtained by transforming with pose $\mathbf{p}$ an impulse centered at the kth patch-center. Since $\mathbf{h}_k(\mathbf{p})$ is independent of the latent image and the MDF, it needs to be computed *only once*, and can be subsequently used to create the blur kernel in patch k for any image and hence leads to large computational gain.

The blur model in Eq. (2.10) admits a simplified image estimation framework, i.e.,

$$\mathbf{L} = \sum_{k=1}^{N} \mathbf{C_k^\dagger} \cdot \mathbb{F}^{-1} \left(\frac{1}{\mathbb{F}(\mathbf{h}^{(k)})} \odot \mathbb{F}(\mathbf{C_k} \cdot \mathbf{B}) \right), \tag{2.12}$$

where $\mathbb{F}$ and $\mathbb{F}^{-1}$ are the forward and inverse FFT, respectively, and $\odot$ is a point-wise multiplication operator. Note that this approach is computationally efficient, as *no* optimization with costly prior is required. This inversion is typically employed at all scales and iterations except at the finest scale, final iteration. The reason is that Eq. (2.12) does *not* use any prior for sharp images, and as the ego-motion estimation is based on image gradients, only the latent-image gradient information needs to be correctly estimated (which does *not* necessitate computationally expensive priors) (Cho and Lee, 2009; Hirsch *et al.*, 2011; Whyte *et al.*, 2012). However, this is *not* valid for the final scale, final iteration where the image information (and *not* the image-gradient) is important, and hence one typically resorts to optimization-based methods (e.g., TV-based method discussed in Eq. (2.9) or Richarson-Lucy method (Lucy, 1974)).

CHAPTER 3

Motion Deblurring for Rolling Shutter Cameras

3.1 Introduction and Related Works

[1] Complementary metal oxide semiconductor (CMOS) sensor is winning the camera sensor battle as it offers advantages in terms of extended battery life, lower cost and higher frame rate, as compared to the conventional charge coupled device (CCD) sensor (Litwiller, 2001). Nevertheless, the annoying effect of motion blur that affects CCD cameras prevails in common CMOS rolling shutter (RS) cameras too, except that it manifests in a different form (Su and Heidrich, 2015).

The problem of blind motion deblurring (BMD) – i.e., recovery of both the clean image and underlying camera motion from a single motion blurred image – is an extensively studied topic for CCD cameras. In CCD cameras, the standard motion blur method used by the state-of-the-art deblurring methods is that of (Whyte *et al.*, 2012), where the 6D camera pose-space (i.e., 3D translations along **XYZ** directions and 2D out-of-plane rotations (yaw and pitch) and inplane rotations (roll)) is approximated by only 3D rotations. To reduce the ill-posedness of BMD, a recent trend is to introduce novel priors. Some representative works in this direction include natural image priors such as total variation (TV) (Perrone and Favaro, 2014), L_0 sparsity (Xu *et al.*, 2013) , and dark channel prior (Pan *et al.*, 2016). In particular, TV enforces sparsity in the gradient-map of images which is a characteristic of natural images, via a convex cost; Xu *et al.* (2013), instead of explicitly extracting gradients, incorporate a new regularization term consisting of a family of loss functions to approximate the $L0$ cost into the objective, which, during optimization, leads to consistent energy minimization and accordingly fast convergence; in contrast, (Pan *et al.*, 2016) is based on the observation that while most natural clean image patches contain some dark pixels, these pixels are not dark when averaged with neighbouring high-intensity pixels during the blurring

[1] Based on: Going Unconstrained with Rolling Shutter Deblurring *Mahesh Mohan M. R., Rajagopalan A. N., and Gunasekaran Seetharaman.*; ICCV 2017, IEEE Publications, Pages 4010–4018.

process. Ego-motion priors include Tikhonov regularization (Hirsch *et al.*, 2011), and sparsity (Whyte *et al.*, 2012; Gupta *et al.*, 2010) and continuity (Gupta *et al.*, 2010) in pose-space. Another important research direction in BMD is towards reducing computational complexity. Cho and Lee (2009) address this by utilizing the FFT for space invariant blur (Note that FFT-inversion is quite fast and efficient as compared to spatial domain inversion of convolution). Hirsch *et al.* (2011, 2010) extend this to the space-variant case by approximating motion blur as space invariant over small image-patches, and show competitive quality with significant speed up.

However, the aforementioned deblurring methods proposed for CCD cameras are not applicable to CMOS-RS (Su and Heidrich, 2015) since the RS motion blur formation is *strikingly different* as illustrated in Fig. 3.1(left). CCD camera uses a global shutter (GS), whereas CMOS cameras predominantly come with an electronic RS. In contrast to GS in which all sensor elements integrate light over the same time window (or experience the same camera motion), each sensor row in RS integrates over different time window, and thus a single camera motion does *not* exist for the entire image. To the best of our knowledge, *only* three works specifically address motion deblurring in RS cameras – (Tourani *et al.*, 2016) for depth camera videos, (Hu *et al.*, 2016) for hardware assisted deblurring, and the BMD method of (Su and Heidrich, 2015). Tourani *et al.* (2016) use feature matches between *depth maps* to timestamp parametric ego-motion. However, they require *multiple* RGB-depth images as input. Also, blurred RGB images, unlike depth maps, lack sufficient feature matches for reliable ego-motion estimation, which limits their functionality (Tourani *et al.*, 2016). In contrast, Hu *et al.* (2016) use smartphone inertial sensors for timestamping and is thus a non-blind approach. Furthermore, it is device-specific and the cumulative errors from noisy inertial sensors and calibration govern deblurring performance (Hu *et al.*, 2016).

The current state-of-the-art RS-BMD (Su and Heidrich, 2015) eliminates device-specific constrains of (Hu *et al.*, 2016; Tourani *et al.*, 2016), and estimates timestamped ego-motion solely from image intensities. However, the method is limited to parametric ego-motion derived specifically for hand-held blur. This renders it difficult to handle blur due to moving/vibrating platforms, such as in robotics, drones, street view cars etc. Second, wide-angle systems provide a larger field-of-view as compared to narrow-angle lenses, an important setting in most DSLR cameras, mobile phones and drones. This is illustrated in Fig. 3.1(right) using focal-length settings of some popular CMOS

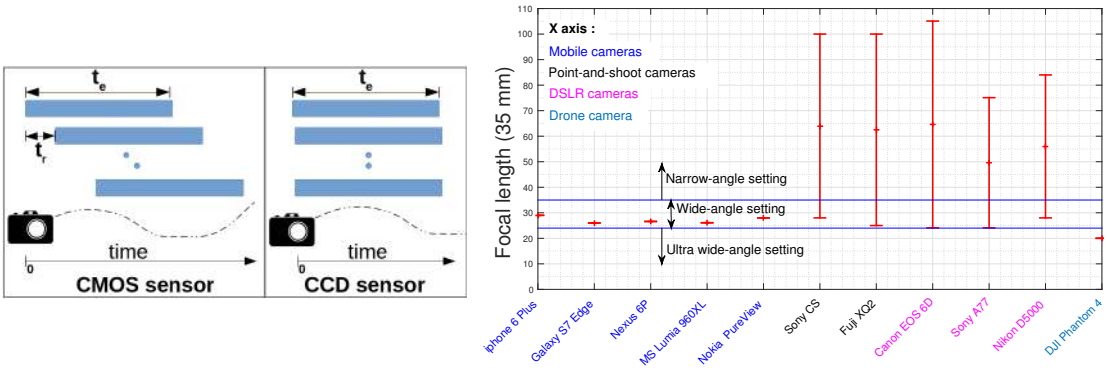


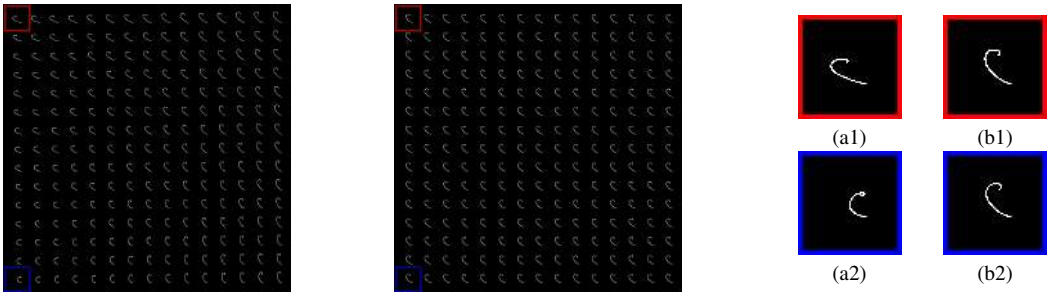
Figure 3.1: (Left) Working principle of CMOS-RS and CCD sensors (i.e., row-wise exposure versus concurrent exposure). (Right) Focal lengths of some popular CMOS devices. Note the wide-angle setting predominant in cell-phone and drone cameras.

imaging devices. It is evident from the figure that wide-angle configurations are indeed important in photography (the primary setting of cell-phones and drone cameras). However, the state-of-the-art RS-BMD (Su and Heidrich, 2015) works *only* for narrow-angle settings. This is so as to provide a good initialization in (Su and Heidrich, 2015) by *discarding* inplane rotations, which precludes it from dealing with wide-angle systems. The reason is illustrated in Fig. 3.2 with two different PSFs generated by a real ego-motion (using (Köhler *et al.*, 2012)) in a wide-angle system with and *without* the inplane rotation, which clearly reveals the inefficacy of their approximation. Another significant limitation of current RS deblurring methods (Su and Heidrich, 2015; Hu *et al.*, 2016; Tourani *et al.*, 2016) is their huge computational load. Moreover, methods (Su and Heidrich, 2015; Hu *et al.*, 2016) require as input *precise* sensor timings t_r and t_e during image capture in order to fragment the estimated ego-motion corresponding to each image-row (see Fig. 3.1(left)). Other RS related works include RS super-resolution (Punnappurath *et al.*, 2015), RS image registration (Rengarajan *et al.*, 2016), RS structure from motion (Ito and Okatani, 2017), etc.

In this chapter, we propose an RS-BMD method that not only delivers excellent deblurring quality but is also computationally very efficient. It works by leveraging a generative model for RS motion blur (different from the one commonly employed), and a prior to disambiguate multiple solutions during inversion. Deblurring with our scheme not only relaxes the constraints associated with current methods, but also leads to an efficient optimization framework.

Our main contributions are summarized below.

- Our method overcomes some of the major drawbacks of the state-of-the-art method



(a) With inplane rotations (b) *Without* inplane rotations (c) Patches of (a) and (b)

Figure 3.2: Effect of inplane rotation for a wide-angle system: (a) Blur kernels (or PSFs) with inplane rotation and (a1-a2) shows its two PSFs magnified (b) Blur kernels *without* inplane rotation and (b1-b2) shows the corresponding two PSFs. Note the variation in shape of the PSFs between (a1-a2) and (b1-b2).

(Su and Heidrich, 2015), including inability to handle full 3D rotations (or wide-angle systems) and irregular ego-motion, and the need for sensor data.

- We extend the computationally efficient filter flow (EFF) framework that is commonly employed in CCD-BMD (Hirsch *et al.*, 2010, 2011) to RS-BMD. Relative to (Su and Heidrich, 2015), we achieve a speed-up by a factor of *at least* eight.
- Ours produces state-of-the-art RS deblurring results for narrow- as well as wide-angle systems and under arbitrary ego-motion, all of these *sans* sensor timings.

3.2 RS Motion Blur Model

In this section, we discuss the generative model for RS motion blur. As mentioned earlier, the entire image in CCD or global shutter (GS) cameras experiences the *same* ego-motion. Thus the motion blurred image $\mathbf{B} \in \mathbb{R}^{M \times N}$ in a GS sensor is generated by integrating the images seen by the camera along its trajectory during the exposure duration $[0, t_e]$ (Su and Heidrich, 2015). It is given by

$$\mathbf{B} = \frac{1}{t_e} \int_0^{t_e} \mathbf{L}^{\mathbf{p}(t)} dt, \tag{3.1}$$

where $\mathbf{p}(t_0)$ represents the general 6D camera pose at time instant t_0, $\mathbf{L}^{\mathbf{p}(t_0)}$ is the latent image $\mathbf{L}$ transformed according to the pose $\mathbf{p}(t_0)$, and t_e is the shutter speed.

In contrast, each RS sensor-row can experience *different* ego-motion due to its staggered exposure windows (Fig. 3.1). Hence, unlike CCD, we cannot associate a global warp for the entire latent image $\mathbf{L}$, but need to consider each row separately. Image row

$\mathbf{B}_i$ (subscript i indicates ith row) of an RS blurred image $\mathbf{B} = [{\mathbf{B}_1}^T\ {\mathbf{B}_2}^T \cdots\ {\mathbf{B}_M}^T]^T$ is given by

$$\mathbf{B}_i = \frac{1}{t_e}\int_{(i-1)\cdot t_r}^{(i-1)\cdot t_r + t_e} \mathbf{L}_i^{\mathbf{p}(t)} dt \quad : i \in \{1, 2, \cdots, M\}, \tag{3.2}$$

where $\mathbf{L}_i^{\mathbf{p}(t_0)}$ is the ith row of the transformed image $\mathbf{L}^{\mathbf{p}(t_0)}$, t_e is the shutter speed or row-exposure time in CMOS sensors, and t_r is the inter-row delay. All the current RS deblurring methods use a discretized form of Eq. (3.2) as the forward model, and we refer to this as temporal model.

An equivalent representation of Eq. (3.2) can be obtained by a weighted integration of the transformed image-rows over camera poses, where the weight corresponding to a transformed image-row with a specific pose determines the fraction of the row-exposure time (t_e) that the camera stayed in the particular pose. This is given by

$$\mathbf{B}_i = \int_{\mathbf{P}} w_i'(\mathbf{p}) \cdot \mathbf{L}_i^{\mathbf{p}} d\mathbf{p} \quad : i \in \{1, 2, \cdots, M\}, \tag{3.3}$$

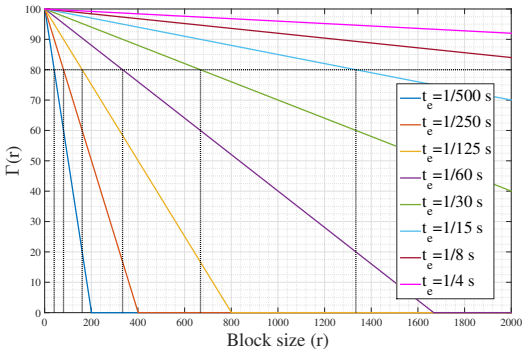
where $\mathbf{P}$ is the continuous camera pose-space and $w_i'(\mathbf{p_0})$ is the weight corresponding to the transformed row $\mathbf{L}_i^{\mathbf{p_0}}$. Unlike existing RS deblurring works, we employ the second model and discretize the pose-space in Eq. (3.3). We consider the discretization step-size to be such that there is less than one pixel displacement between adjacent poses. Thus Eq. (3.3) reduces to

$$\mathbf{B}_i = \sum_{\mathbf{p}\in\mathbb{P}} w_i(\mathbf{p}) \cdot \mathbf{L}_i^{\mathbf{p}} \quad : i \in \{1, 2, \cdots, M\}, \tag{3.4}$$

where $\mathbb{P}$ is the discretized pose-space $\mathbf{P}$, and the discrete weight $w_i(\mathbf{p_0})$ is the summation of all the continuous-weights $w_i'(\mathbf{p})$ for all $\mathbf{p}$ that lie in the half step-size neighbourhood of pose $\mathbf{p_0}$. We identify the weights $w_i(\mathbf{p})$ as the motion density function (MDF), as in (Gupta *et al.*, 2010). We further modify Eq. (3.4) based on an important observation derived from typical CMOS sensor settings.

Observation: *In RS motion blurred images, there exists an* $r_b{:}1 << r_b \leq M$*, such that any block of contiguous rows with size less than or equal to* r_b *will have substantial camera-pose overlap.*

In RS sensors, the fraction of camera-pose overlap in r contiguous rows is equal to the fraction of the time shared among those rows. Thus, from the RS timing diagram in



(a) Pose-overlap vs block-size

(b) Input (c) W/o prior (d) W/ prior

Figure 3.3: (a) Percentage pose-overlap Γ over block-size r for standard CMOS-RS shutter speed (t_e) and an inter-row delay (t_r) of $1/100$ ms, along with optimal block-size. (b) A blurred patch from an RS blurred image (Fig. 3.9); (c & d) Corresponding patch of deblurred results *without* and with our RS prior.

Fig. 3.1, the percentage camera-pose overlap Γ in a block of r rows is obtained as

$$\Gamma(r) = \max\left(\frac{t_e - (r-1)\cdot t_r}{t_e}, 0\right) \cdot 100. \tag{3.5}$$

In Fig. 3.3(a), we plot $\Gamma(r)$ for varying t_e and a fixed t_r of $1/100$ ms – a typical CMOS sensor has standardized t_e as $\{1/1000 \text{ s}, 1/500 \text{ s}, \cdots 1 \text{ s}\}$, and t_r in the range $1/200$ ms to $1/25$ ms (Gu *et al.*, 2010). It is evident from the figure that such a block-wise segregation is possible for these standard settings with camera-pose overlap of almost 80%. We do note that for faster shutter speed (e.g., $t_e < 1/250$ s) r_b can be close to one; but for that setting motion blur will be negligible.

Based on this observation, we approximate each non-intersecting block of r_b rows that have substantial camera-pose overlap to be governed by an individual MDF. We will later show that this approximation is reasonable for RS motion blurred images. Thus our forward RS motion blur model is given by

$$\mathbf{B_i} = \sum_{\mathbf{p}\in\mathbb{P}} w_{\mathbf{i}}(\mathbf{p}) \cdot \mathbf{L_i^p} \quad : \mathbf{i} \in \{1, 2, \cdots, n_b\}, \tag{3.6}$$

where $n_b = M/r_b$ is the total number of blocks, the blurred image $\mathbf{B}$ has structure $[\mathbf{B_1}^T, \mathbf{B_2}^T, \cdots \mathbf{B_{n_b}}^T]$ with $\mathbf{B_i}$ as the $\mathbf{i}$th block (bold subscript represents block), $w_{\mathbf{i}}(\mathbf{p})$ is the approximated MDF of $\mathbf{i}$th block, and $\mathbf{L_i^{p_0}}$ is the $\mathbf{i}$th block of the transformed image $\mathbf{L}$ with pose $\mathbf{p_0}$. Note that for $r_b = M$, Eq. (3.6) reduces to CCD motion blur model (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012; Hirsch *et al.*, 2011; Gupta *et al.*,

2010). We identify Eq. (3.6) as our RS pose-space model. This is unlike the temporal model of (Su and Heidrich, 2015) which constrains the motion model to be parametric. Therefore, our model can accommodate different kinds of motion trajectories including camera shake and vibrations.

3.3 RS Deblurring

We formulate a maximum a posteriori (MAP) framework for estimation of both the latent image and the block-MDFs. In this section, we bring out an ill-posedness in RS-BMD and introduce a new prior to address this.

A direct MAP framework for unknown $\theta = \{\mathbf{L}, \mathbf{w_i} : 1 \leq \mathbf{i} \leq n_b\}$ is given as

$$\hat{\theta} = \min_{\theta} \sum_{\mathbf{i}=1}^{n_b} \|(\mathbf{B_i} - \sum_{\mathbf{p}\in\mathbb{P}} w_{\mathbf{i}}(\mathbf{p}) \cdot \mathbf{L}_{\mathbf{i}}^{\mathbf{p}}\|_2 + \lambda_1 \|\nabla \mathbf{L}\|_1 + \lambda_2 \sum_{\mathbf{i}=1}^{n_b} \|\mathbf{w_i}\|_1, \tag{3.7}$$

where $\mathbf{w_i}$ is the vector containing weights $w_{\mathbf{i}}(\mathbf{p})$ for poses $\mathbf{p} \in \mathbb{P}$ and $\nabla\mathbf{L}$ is the gradient of $\mathbf{L}$. We assume that the optimal block-size r_b is known (we relax this subsequently in section 3.5.1). The first term in the objective in Eq. (3.7) is data fidelity that enforces our forward blur model of Eq. (3.6). To reduce ill-posedness, we too enforce a sparsity prior on the image-gradient following (Whyte *et al.*, 2012; Gupta *et al.*, 2010). We also impose a sparsity prior on the MDF weights since a camera can transit over only few poses in $\mathbb{P}$ during exposure. In the literature on CCD deblurring (i.e., $n_b = 1$) it is well-known that the objective in Eq. (3.7) is biconvex, i.e., it is individually convex with respect to the latent image and MDF, but non-convex overall; and convergence to a local minima is ensured with alternative minimization of MDF and latent image (Whyte *et al.*, 2012; Gupta *et al.*, 2010; Cho and Lee, 2009). However, RS sensors introduce a different challenge if Eq. (3.7) is directly considered (Fig. 3.3(c)).

Claim 1: *For an RS blurred image, there exist multiple solutions for the latent image-MDF pair in each individual image-block. They satisfy the forward model in Eq.* (3.6) *and are consistent with the image and MDF prior in Eq.* (3.7).

Before giving a formal proof, we attempt to provide some intuition. Considering

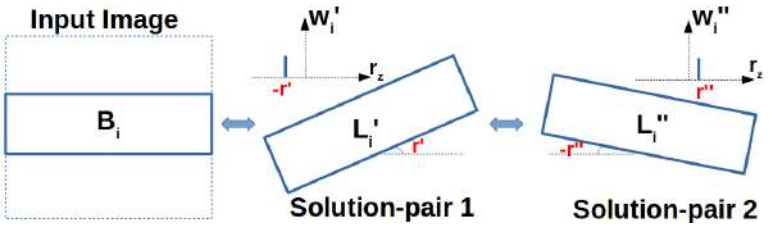


Figure 3.4: Illustration of block-wise latent image-MDF pair ambiguity for a single inplane rotation (only 1D pose-space). Both solution-pairs 1 and 2, though entirely different, result in the same blurred block $\mathbf{B_i}$.

only inplane rotations, Fig. 3.4 illustrates a multiple-solution case where one latent image-block (of solution-pair $\{\mathbf{L_i'}, \mathbf{w_i'}\}$) is rotated anti-clockwise and the second image-block (of solution-pair $\{\mathbf{L_i''}, \mathbf{w_i''}\}$) is rotated clockwise, but both result in the *same* input image-block $\mathbf{B_i}$. This can also be visualized as a natural escalation of the notion of shift-ambiguity in patch-wise PSF estimates (Paramanand and Rajagopalan, 2013) all the way to block-wise MDFs.

Proof: Let $\mathbf{B_i}$ be an RS blurred block formed by latent image $\mathbf{L}$ and MDF $\mathbf{w_i}$ through Eq. (3.6). We form a second RS blurred block $\mathbf{B_i'}$ by considering a *nonzero* pose $\mathbf{p_0} \in \mathbb{P}$ as

$$\mathbf{B_i'} = \sum_{\mathbf{p'} \in \mathbb{P}} w_{\mathbf{i}}(\mathbf{p_0} + \mathbf{p'}) \cdot \mathbf{L_i^{p_0+p'}}, \tag{3.8}$$

where $\mathbf{L_i^{p_0+p'}}$ is the ith block of the transformed version of $\mathbf{L^{p_0}}$ with pose $\mathbf{p'}$, and $w_{\mathbf{i}}(\mathbf{p_0} + \mathbf{p'})$ is obtained by shifting $w_i(\mathbf{p'})$ with a negative offset of $\mathbf{p_0}$. Even though the latent image-MDF pairs for $\mathbf{B_i}$ and $\mathbf{B_i'}$ are different, i.e., $\{\mathbf{L}, w_{\mathbf{i}}(\mathbf{p})\}$ in Eq. (3.6), and $\{\mathbf{L^{p_0}}, w_{\mathbf{i}}(\mathbf{p_0} + \mathbf{p})\}$ in Eq. (3.8), we shall prove that both $\mathbf{B_i}$ and $\mathbf{B_i'}$ are equal. Construct a set $\mathbb{S}_{\mathbb{B}i}$ with elements as *all* individual additive components of Eq. (3.6) that add up to get $\mathbf{B_i}$. Similarly, form set $\mathbb{S}_{\mathbb{B}_i'}$ with *all* additive components of Eq. (3.8). Any element in $\mathbb{S}_{\mathbb{B}i}$ is represented as a singleton $\{w_{\mathbf{i}}(\mathbf{p}) \cdot \mathbf{L_i^p}\}$ with $\mathbf{p} \in \mathbb{P}$. The same element is present in $\mathbb{S}_{\mathbb{B}_i'}$, i.e., at $\mathbf{p'} = -\mathbf{p_0} + \mathbf{p}$ in Eq. (3.8), which implies $\mathbb{S}_{\mathbb{B}i} \subseteq \mathbb{S}_{\mathbb{B}_i'}$. Similarly by considering $\mathbf{p} = \mathbf{p_0} + \mathbf{p'}$ in Eq. (3.6), it follows that $\mathbb{S}_{\mathbb{B}_i'} \subseteq \mathbb{S}_{\mathbb{B}i}$. Since $\mathbb{S}_{\mathbb{B}i} \subseteq \mathbb{S}_{\mathbb{B}_i'}$ and $\mathbb{S}_{\mathbb{B}_i'} \subseteq \mathbb{S}_{\mathbb{B}i}$, both the sets are equal, and so are $\mathbf{B_i}$ and $\mathbf{B_i'}$. Also, as the latent images $\mathbf{L}$ and $\mathbf{L^{p_0}}$ are related by a global warp, the sparsity in gradient domain (i.e., image prior in Eq. (3.7)) is valid for both. Since both the MDFs have equal weight distribution, the sparsity in weights (i.e., MDF prior) is also identical for both. Hence proved. ■

If we consider Eq. (3.7) *alone* for RS deblurring, this ambiguity can cause the latent

image portion of *individual* block to transform *independently* (see Fig. 3.4). This can result in an erroneous estimate of the deblurred image, where the latent image portions corresponding to different blurry blocks are incoherently combined (Fig. 3.3(c)). To address this issue, we introduce an additional prior on the MDFs as

$$\text{prior}(\mathbf{w}) = \sum_{\mathbf{i}=1}^{n_b} \sum_{\mathbf{j}>\mathbf{i}}^{n_b} \|\Gamma\left(r_b(\mathbf{j}-\mathbf{i}+1)\right) \cdot (\mathbf{w_i} - \mathbf{w_j})\|_2^2, \tag{3.9}$$

where $\Gamma\left(r_b \cdot (\mathbf{j}-\mathbf{i}+1)\right)$ is the percentage overlap (Eq. (3.5)) of all groups of blocks between (and including) the **i**th and **j**th block, and $\mathbf{w}$ is a vector obtained by stacking all the unknown MDFs $\{\mathbf{w_i} : 1 \leq \mathbf{i} \leq n_b\}$. This prior restricts drifting of MDFs between neighbouring blocks (i.e., high cost), but allows MDFs to change between distant blocks (i.e., low cost). It also serves to impart an additional dependency between block MDFs which Eq. (3.7) does not possess. This helps to reduce the ill-posedness of ego-motion estimation.

Claim 2: *The prior in Eq.* (3.9) *is a convex function in* $\mathbf{w}$*, and can be represented as a norm of matrix vector multiplication, i.e., as* $\|\mathbf{G}\mathbf{w}\|_2^2$*, with sparse* $\mathbf{G}$*.*

To prove this, we draw from the following well-known properties of convex function (Boyd and Vandenberghe, 2004) which are a linear function is always convex (prop. 1), composition of convex functions with a non-decreasing function is always convex (prop. 2), and non-negative sum of convex functions is convex (prop. 3).

Proof: Considering n_b number of image blocks and each block-MDF $w_\mathbf{i}$ having length l, an individual additive component in our RS prior (in Eq. (3.9)) can be represented as $\|\Gamma\left(r_b(\mathbf{j}-\mathbf{i}+1)\right) \cdot \mathbf{S}_{(\mathbf{i},\mathbf{j})}\mathbf{w}\|_2^2$, where $\mathbf{S}_{(\mathbf{i},\mathbf{j})}$ is a matrix of dimension $l \times n_b \cdot l$, with all zeros except two scaled identity matrices of dimension $l \times l$ corresponding to **i**th TSF (with scale 1) and **j**th TSF (with scale -1). Therefore, the term $\{\Gamma\left(r_b(\mathbf{j}-\mathbf{i}+1)\right) \cdot \mathbf{S}_{(\mathbf{i},\mathbf{j})}\mathbf{w}\}$ is a linear function in $\mathbf{w}$. Since $\|\Gamma\left(r_b(\mathbf{j}-\mathbf{i}+1)\right) \cdot \mathbf{S}_{(\mathbf{i},\mathbf{j})}\mathbf{w}\|_2^2$ is a composite of squared L_2 norm (which is non-decreasing) of a linear function in $\mathbf{w}$, each additive component is convex (props. 1 and 2). Resultantly, the sum of all additive components in Eq. (3.9), i.e., prior($\mathbf{w}$), is a convex function in $\mathbf{w}$ (prop. 3). Further, prior($\mathbf{w}$) can be represented as $\|\mathbf{G}\mathbf{w}\|_2^2$, where matrix $\mathbf{G}$ is obtained by vertically concatenating matrices $\{\Gamma\left(r_b(\mathbf{j}-\mathbf{i}+1)\right) \cdot \mathbf{S}_{(\mathbf{i},\mathbf{j})}\}$ corresponding to the individual additive component in RS prior. Since $\mathbf{S}_{(\mathbf{i},\mathbf{j})}$ is a sparse matrix, $\mathbf{G}$ will also be sparse. Hence proved. ■

Thus inclusion of the prior does *not* alter the biconvexity of Eq. (3.7) (which is necessary for convergence), and paves the way for efficient implementation (as we shall see in section 3.4.2). We identify Eq. (3.9) as our proposed RS prior.

3.4 Model and Optimization

State-of-the-art CCD-BMD methods (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012; Cho and Lee, 2009) work by alternative minimization (AM) of MDF and latent image over a number of iterations in a scale-space manner, (i.e., AM proceeds from coarse to fine image-scale in order to handle large blurs). As we shall see shortly, this requires generation of blur numerous times. Efficiency of the blurring process is a major factor that governs computational efficiency of a method. In this section, we first discuss how our pose-space model allows for an efficient process for RS blurring (analogous to CCD-EFF (Hirsch *et al.*, 2011, 2010)). Next, we elaborate on our AM framework, and eventually relax the assumption of the need for sensor information.

3.4.1 Efficient Filter Flow for RS blur

Following (Hirsch *et al.*, 2010), we approximate motion blur in individual *small* image patches as space invariant convolution with different blur kernels. We represent this as

$$\mathbf{B} = \sum_{k=1}^{R} \mathbf{C}_{\mathbf{k}}^{\dagger} \cdot \left\{ \mathbf{h}^{(k,b(k))} * (\mathbf{C}_{\mathbf{k}} \cdot \mathbf{L}) \right\}, \tag{3.10}$$

where R is the total number of overlapping patches in latent image $\mathbf{L}$, $b(k)$ is a function which gives the index of the block to which the major portion of the k^{th} patch belongs (i.e., $b(k) \in \{\mathbf{1}, \mathbf{2}, \cdots \mathbf{n_b}\}$), $\mathbf{C_k}\cdot\mathbf{L}$ is a linear operation which extracts the kth patch from $\mathbf{L}$, and $\mathbf{C}_{\mathbf{k}}^{\dagger}$ inserts the patch back to its original position with a windowing operation. $\mathbf{h}^{(k,b(k))}$ represents the blur kernel which when convolved with the kth latent image-patch creates blurred patch. Considering $b(k)$ as $\mathbf{j}$, we can write

$$\mathbf{h}^{(k,b(k))} = \sum_{\mathbf{p}\in\mathbb{P}} w_{\mathbf{j}}(\mathbf{p}) \cdot \delta_k(\mathbf{p}), \tag{3.11}$$

where $\delta_k(\mathbf{p})$ is a shifted impulse obtained by transforming with pose $\mathbf{p}$ an impulse centered at the kth patch-center. Intuitively, the blur kernel at patch k due to an arbitrary MDF is the superposition of the $\delta_k(\mathbf{p})$s generated by it. Since $\delta_k(\mathbf{p})$ is independent of the latent image and the MDF, it needs to be computed *only once*, and can be subsequently used to create the blur kernel in patch k for any image. Thus, given a latent image $\mathbf{L}$ and MDF of each block, our blurring process first computes kernels in R patch-centres using the precomputed $\delta_k(\mathbf{p})$ (Eq. 3.11), convolves them with their corresponding latent-image patches and combines them to form the RS blurred image (Eq. (3.10)). We carry out convolution using the *efficient* fast Fourier transform (FFT). Note that the CCD-EFF is a special case of Eq. (3.10) under identical MDFs ($\mathbf{w_i} = \mathbf{w}\ \forall \mathbf{i}$) or the single block case ($n_b = 1$). We next discuss our AM at the finest level. The same procedure is followed at coarser levels and across iterations.

3.4.2 Ego-Motion Estimation

The objective of this step is to estimate the ego-motion at iteration $d+1$ (i.e., $\mathbf{w}^{d+1}$) given the latent image estimate at iteration d (i.e., $\mathbf{L}(d)$). We frame our MDF objective function in the gradient domain for faster convergence and to reduce ill-conditionness (Xu *et al.*, 2013; Whyte *et al.*, 2012; Cho and Lee, 2009; Hirsch *et al.*, 2011). We give it as

$$\mathbf{w}^{d+1} = \arg\min_{\mathbf{w}} \|\mathbf{F}\mathbf{w} - \nabla\mathbf{B}\|_2^2 + \alpha\|\mathbf{G}\mathbf{w}\|_2^2 + \beta\|\mathbf{w}\|_1, \tag{3.12}$$

where the information of the gradient of $\mathbf{L}(d)$ is embedded in blur matrix $\mathbf{F}$, $\nabla\mathbf{B}$ is the gradient of $\mathbf{B}$, and $\|\mathbf{G}\mathbf{w}\|_2$ is the prior we introduce for RS blur (Eq. 3.9). We further simplify the objective in Eq. (3.12) by separating out the sparsity prior as a constraint and taking the derivative (similar to (Whyte *et al.*, 2012)). This yields

$$\begin{aligned} \mathbf{w}^{d+1} = \arg\min_{\mathbf{w}} \|(\mathbf{F}^T\mathbf{F} + \alpha\mathbf{G}^T\mathbf{G})\mathbf{w} - \mathbf{F}^T\nabla\mathbf{B}\|_2^2 \\ \text{such that } \|\mathbf{w}\|_1 \leq \beta'. \end{aligned} \tag{3.13}$$

A major advantage of our pose-space model over the temporal model of Eq. (3.2) is that we can formulate ego-motion estimation as in Eq. (3.13). This is least absolute shrinkage and selection operator (LASSO) (Tibshirani, 1996) of the form $\arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 : \|\mathbf{x}\|_1 \leq \gamma$, and has efficient solvers (least angle regression or LARS (Efron *et al.*,

2004)). Since LASSO needs no initialization, we can account for inplane rotations too. This allows us to handle even wide-angle systems unlike (Su and Heidrich, 2015), which discards it to eliminate the back-projection ambiguity of blur kernels for initialization.

Suppose a blurred image of size $M \times N$ and n_b number of MDFs of length l (i.e., block-size $r_b = M/n_b$). Then the *dense* matrix $\mathbf{F}$ in Eq. (3.13) is n_b times larger compared to the CCD case. This escalates the memory requirement and computational cost for RS deblurring; i.e., a naive approach to create $\mathbf{F}^T\mathbf{F}$ (with size $n_b \cdot l \times n_b \cdot l$) is to form a *large* matrix $\mathbf{F}$ of size $MN \times n_b \cdot l$ (where $MN >> n_b l$), and perform large-matrix multiplication. We avoid this problem by leveraging the block-diagonal structure of $\mathbf{F}$, and thus for $\mathbf{F}^T\mathbf{F}$, that is *specific* for RS blur. The jth column of the $\mathbf{i}$th block-matrix $\mathbf{F_i}$ of $\mathbf{F}$ (of size $r_b N \times l$) is formed by transforming $\nabla\mathbf{L}(d)$ with the pose of $w_{\mathbf{i}}(j)$, and vectorizing its $\mathbf{i}$th block. For this, we employ the RS-EFF. Since each $\mathbf{F_i}$ can be generated independently, we bypass creating $\mathbf{F}$, and instead directly arrive at the block-diagonal matrix $\mathbf{F}^T\mathbf{F}$, one diagonal-block at a time, with the jth block as $\mathbf{F_j}^T\mathbf{F_j}$. A similar operation is also done for $\mathbf{F}^T\nabla\mathbf{B}$. Since $\mathbf{G}$ is sparse, $\mathbf{G}^T\mathbf{G}$ in Eq. (3.13) can be computed efficiently (Yuster and Zwick, 2005).

3.4.3 Latent Image Estimation

Given the ego-motion at iteration $d+1$ (i.e., $\mathbf{w}^{d+1}$), this step estimates the latent image $\mathbf{L}(d+1)$. As discussed in Ch. 2, since ego-motion estimation is based on image gradients (Eq. (3.13)), only the latent-image gradient information needs to be correctly estimated. This eliminates the use of computationally expensive image priors in the alternative minimization step. We obtain the latent image by inverting the forward blurring process in Eq. (3.10), i.e.,

$$\mathbf{L}(d+1) = \sum_{k=1}^{R} \mathbf{C}_{\mathbf{k}}^{\dagger} \cdot \mathbb{F}^{-1}\left(\frac{1}{\mathbb{F}(\mathbf{h}^{(k,b(k))})} \odot \mathbb{F}(\mathbf{C_k} \cdot \mathbf{B})\right), \tag{3.14}$$

where $\mathbf{h}^{(k,b(k))}$ is generated using $\mathbf{w}^{d+1}$, $\mathbb{F}$ and $\mathbb{F}^{-1}$ are the forward and inverse FFT, respectively, and $\odot$ is a point-wise multiplication operator that also suppresses unbounded-values. We combine patches using Bartlett-Hann window that tapers to zero near the patch boundary. It has 70% overlap for patches that span adjacent blocks (to eliminate the effect of sudden changes in MDFs), and 50% for the rest. It is important to note that

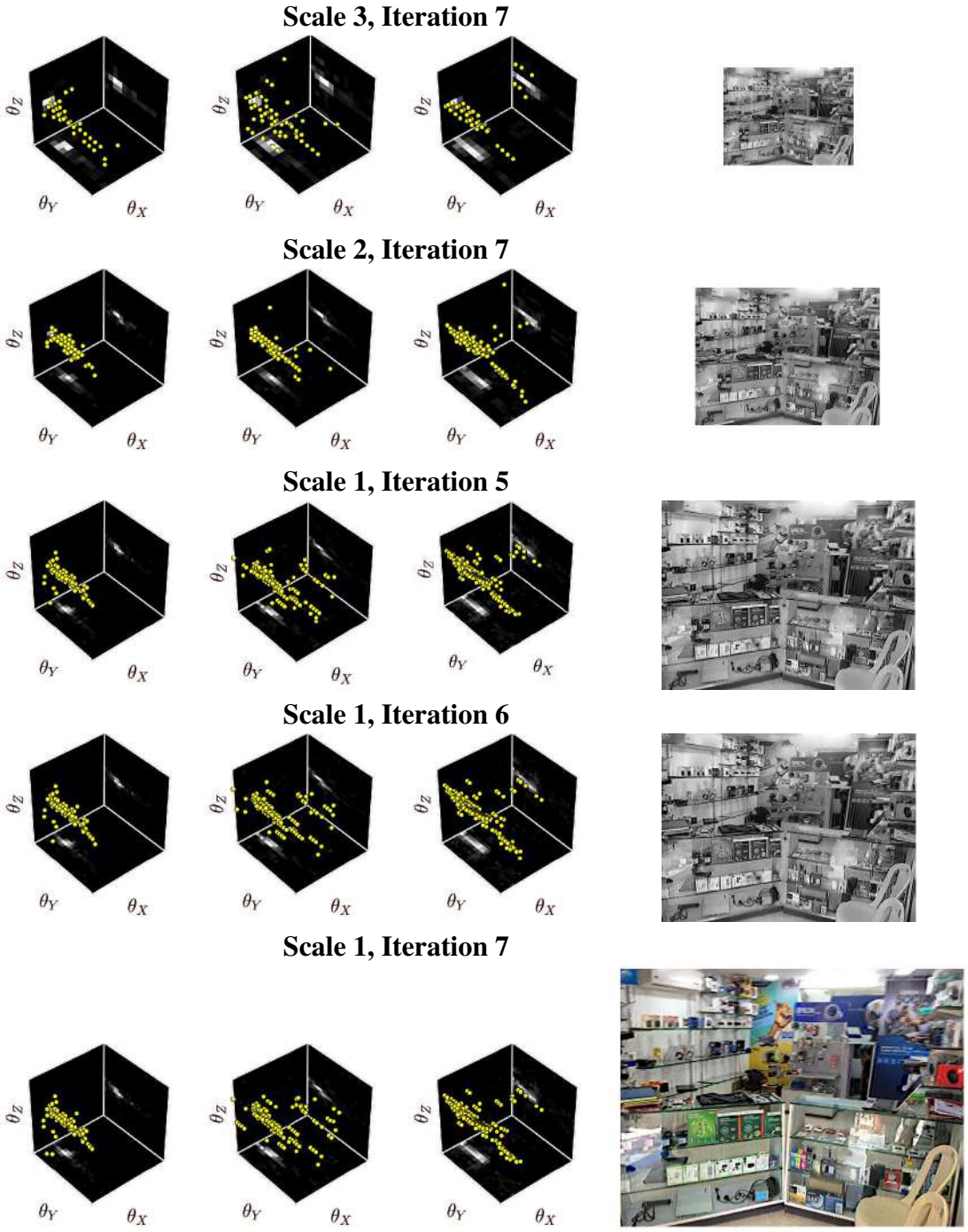
explicit block-wise segregation of blurred image is employed *only* for MDF estimation (to create $\mathbf{F}^T\mathbf{F}$ in Eq. (3.13)), and *not* for latent image estimation where the estimated MDFs are utilized *only* to project PSFs in overlapping patches, akin to CCD-BMD (Whyte *et al.*, 2012; Hirsch *et al.*, 2011, 2010). From a computational perspective, this is equivalent to extracting each patch of the blurred image, deconvolving it with the corresponding blur kernel (created using Eq. (3.11)) with FFT acceleration, and combining the deconvolved patches to form the updated latent image. For the final iteration (in the finest level), instead of FFT inversion (as in Eq. (3.14)), we adopt Richardson-Lucy deconvolution which considers l_2 based TV prior for latent image (Lucy, 1974). (Figure 3.5 illustrates the working of our algorithm using some intermediate results).

3.5 Analysis and Discussions

3.5.1 Selection of Block-Size

In section 3.3, we had assumed the availability of sensor timings t_r and t_e to optimally segregate image-blocks (using $\Gamma(r)$ in Eq. (3.5)) and to derive the RS prior (through Γ in Eq. (3.9)). In this section, we quantify camera-pose overlap and relax the need for sensor timings.

To analyse the effect of block-size r_b, we conducted an experiment using real camera trajectories from the dataset of (Köhler *et al.*, 2012) with CCD blur. Since all the rows would experience a common trajectory, *ideally* all block MDFs should match irrespective of the chosen number of blocks. We estimate MDFs *without* using the RS prior ($\alpha = 0$ in Eq. (3.13)), align their centroids, and use individual MDF to compute PSFs in all R patches. The PSFs are then correlated with the ground-truth PSFs using the kernel similarity metric in (Hu and Yang, 2012) (centroid alignment of MDFs was done since correlation cannot handle arbitrary rotation between PSFs). We plot in Fig. 3.6(a) average kernel similarity and time taken for AM steps at the finest level for different block-sizes. A key observation is that as the block-size r_b falls below 134 (i.e., $n_b \geq 6$ for an 800×800 image dimension), kernel similarity drops significantly. This ineffectiveness for smaller blocks is due to lack of sufficient image gradients *within* individual blocks for MDF estimation. Also, the computation time increases as the block-size is



(a) MDF 1 ($\mathbf{w_1}$) (b) MDF 2 ($\mathbf{w_2}$) (c) MDF 3 ($\mathbf{w_3}$) (d) Latent image ($\mathbf{L}$)

Figure 3.5: Iteration-by-iteration results of the alternative minimization of block-wise MDFs and latent image: (a-c) Estimated block-wise MDFs and (d) Estimated latent image. Notice the variation in block-wise MDFs, which depicts the characteristic of RS blur (as shown in Fig. 3.3). Also, observe the convergence of the block-wise MDFs through iteration 5 to 7 in the finest image scale (last three rows).

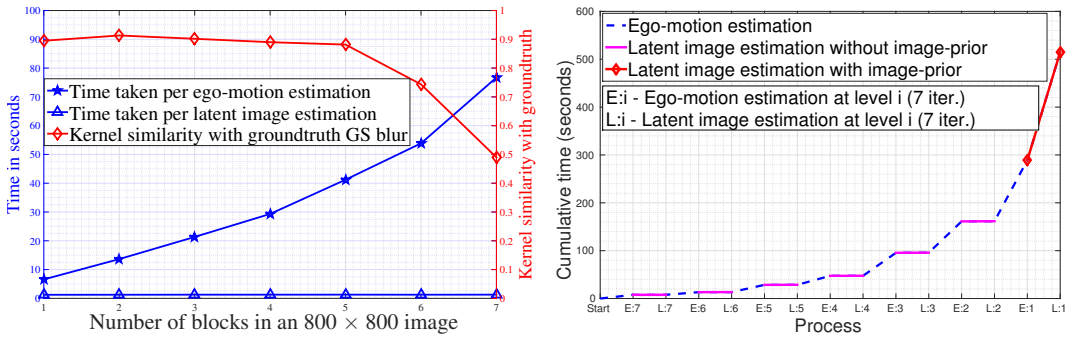
reduced. These factors translate to a pose-intersection of 80%, which gives a reliable block-size for typical CMOS settings (Fig. 3.3(left)).

Apart from sensor timings, camera ego-motion can also influence optimal block-size, e.g., if the camera moves slowly during the exposure of few blocks, merging them as a single block can reduce the number of unknown MDFs. Hence, we allow for variable-sized blocks. We can segregate image-blocks *without* using sensor information as described next. Given a blurred image, we convert the image to a coarse level ($M_0 \times N_0$), and estimate MDFs *without* the RS prior assuming uniform block-size of r_0. For each MDF, we find kernel similarity with the neighbouring MDFs (as described earlier). Adjacent blocks with kernel similarity greater than 0.8 are combined until no such merging is possible. The resultant block-sizes at the coarse level multiplied by the upsampling factor to the finest image-level are considered for final segregation. For the RS prior, since the pose-overlap $\Gamma(r)$ in Eq. (3.5) is parametrised by a single unknown (which is t_r/t_e), we solve for t_r/t_e (*without* requiring it from RS sensor) assuming the number of rows (r) for the smallest segregated block as having 80% Γ.

3.5.2 Computational Aspects

We proceed to analyse the computational gain of our approach against the state-of-the-art methods. First, our pose-space model (Eq. (3.6)) performs RS blurring efficiently as already discussed in section 3.4.1. In contrast, the blurring process adopted in current RS deblurring methods is relatively quite expensive. That is, given a latent image $\mathbf{L}$ and temporal ego-motion (with N_t temporal bins), an RS blurred image is created by N_t *individual* transformations of $\mathbf{L}$ – each using individual warping and bilinear interpolation *over all image-locations* – and the rows are combined using sensor timings (Eq. (3.2) and Fig. 3.1). Second, our pose-space model together with the RS prior is amenable to the very-efficient LARS framework (Eq. (3.13)). In contrast, because of the parametric model (polynomial in (Su and Heidrich, 2015) and splines in (Tourani *et al.*, 2016)) for ego-motion in temporal domain, those methods need to employ non-linear optimization (Eq. (9) in (Su and Heidrich, 2015) and Eq. (8) in (Tourani *et al.*, 2016)), which is much more expensive than LARS (Efron *et al.*, 2004).

Third, our method employs patch-wise deblurring leveraging the very efficient FFT



(a) Analysis of block-size (b) Cumulative time

Figure 3.6: (a) Analysis on the effect of block-size. (b) Cumulative time for different processes. Note the computational gains of the prior-less RS-EFF based image estimation step.

Image dimension	Ego-motion estimation time (s)		Latent image estimation time (s)	
ht. × wd.	Su and Heidrich (2015)	Ours	Su and Heidrich (2015)	Ours
800 × 800	216.01	**29.58**	258.65	**1.44**
450 × 800	122.28	**22.48**	44.23	**1.30**
400 × 400	73.26	**10.34**	23.82	**0.62**

Table 3.1: Time comparisons with state-of-the-art (Su and Heidrich, 2015).

(rather than optimizing the full image with prior). In contrast, as stated under Eq. (15) of (Su and Heidrich, 2015), for *every* iteration it must optimize the high-dimensional latent image with a costly prior as

$$\mathbf{L}(d+1) = \arg\min_{\mathbf{L}} \|\mathbf{X}\mathbf{L} - \mathbf{B}\|_2^2 + \|\nabla\mathbf{L}\|_1, \tag{3.15}$$

where $\mathbf{L}$ and $\mathbf{B}$ are vectorized latent and blurred images of size $MN \times 1$, respectively, and $\mathbf{X}$ is a sparse matrix of size $MN \times MN$. Generating $\mathbf{X}$ using the expensive forward model and optimization with a costly prior is a serious computational bottleneck for (Su and Heidrich, 2015).

To determine the computational gains of our proposal, we conducted experiments on variable-sized images. The average time taken for each AM step at finest level using `MATLAB` is listed in Table 3.1 (in a system with an Intel Xeon processor with 32 GB memory and using the code of (Su and Heidrich, 2015) from the author's website). It is clearly evident that our method offers significant computational gains. Note the improvement of ego-motion estimation from 216 to 30 seconds, and latent image esti-

mation from 258 to 2 seconds for an 800×800 image. Also, observe the steep rise in computational cost for (Su and Heidrich, 2015) with image size, unlike ours. We found that for deblurring an 800×800 RGB image (of maximum blur-length of 30 pixels), our unoptimized `MATLAB` implementation took about 9 minutes. Fig. 3.6(b) provides a detailed break-up of the time taken for each estimation step. Observe that a large fraction of the total time is utilized for latent image estimation in the *final* iteration which involves a costly image-prior (see section 3.4.3). This underscores the importance of our efficient prior-less estimation in the initial iterations derived from RS-EFF.

3.6 Experimental Results

In this section, we demonstrate that our method can handle both wide- and narrow-angle systems, arbitrary ego-motion, and RS as well as GS blurs. We used default parameters for all the competing methods.

Datasets used: For quantitative evaluation, we created RS motion blurred images using hand-held trajectories from the benchmark dataset in (Köhler *et al.*, 2012). We used focal length 29 mm (for wide-angle) and 50 mm (for narrow-angle), and sensor timings of $t_r = 1/50$ ms and $t_e = 1/50$ s (as in (Su and Heidrich, 2015)). Vibration motion was taken from (Hatch, 2000). For real experiments, we considered individually the cases of RS narrow-angle, RS wide-angle and CCD blurs. For the narrow-angle case, we used the dataset in (Su and Heidrich, 2015). Since RS wide-angle configuration has not been hitherto addressed, we created an RS wide-angle blur dataset which contains images captured with iPhone 5S (focal length 29 mm). We also considered drone images from the internet characterizing irregular ego-motion. For CCD blur, we used the dataset in (Pan *et al.*, 2016).

RS deblurring comparisons: We considered mainly the current RS-BMD state-of-the-art method (Su and Heidrich, 2015) for evaluation. Since (Hu *et al.*, 2016) requires inertial sensor data, and (Tourani *et al.*, 2016) uses multiple RGBD images, these techniques do not address BMD and thus have been omitted for comparisons (also their codes are not available). Further, we also include deep learning based single-lens deblurring methods to represent scale-space approach (Tao *et al.*, 2018), generative models (Kupyn *et al.*, 2018), and patch-based approach (Zhang *et al.*, 2019). To analyse the performance of CCD-BMD methods on CMOS data, we also tested with (Xu *et al.*,

2013). Since the space-variant (SV) code of (Pan *et al.*, 2016) (the best CCD-BMD) is *not* available, we evaluated using (Xu *et al.*, 2013) (the second best). For comparisons, we downloaded the codes from the websites of the authors of (Su and Heidrich, 2015) and (Xu *et al.*, 2013).

GS deblurring comparisons: Since the SV code of (Pan *et al.*, 2016) is not available, we report results on the SV examples provided in (Pan *et al.*, 2016). We also give comparisons with other SV-BMD methods (Xu *et al.*, 2013; Whyte *et al.*, 2012; Gupta *et al.*, 2010) in Fig. 3.12.

Quantitative evaluation: As pointed out in a very recent comparative study of BMD (Lai *et al.*, 2016), information fidelity criterion (IFC) (Sheikh *et al.*, 2005) and visual information fidelity (VIF) (Sheikh and Bovik, 2006) are important metrics for evaluating BMD methods (higher values are better).Thus we adopt the same. This is because the correlation between MSE/PSNR and human judgement of quality is not good enough for most vision applications, and hence calls for perceptually consistent metrics. Two popular such metrics are IFC and VIF, which are devised using natural scene statistics in an information-theoretic setting, i.e., natural scenes form a small subspace in the space of all possible signals, and most real world distortion processes (such as the presence of residual blur after deblurring) disturb these statistics and make the image unnatural. These metrics employ natural scene models in conjunction with distortion models to quantify the statistical information shared between the test and the reference images, and posit that this shared information is an aspect of fidelity that relates well with visual quality. IFC is the mutual information between the source and the distorted images between multiple wavelet sub-bands. In contrast, VIF is based on two mutual information: one between the input and the output of the human vision system channel when no distortion channel is present and another between the input of the distortion channel and its output of the human vision system channel. Also, we wish to highlight the observation in (Shan *et al.*, 2008) that ringing artifacts in deblurring are mainly caused by ego-motion estimation error, which can be either due to inaccurate blur/ego-motion model or ineffectiveness of optimization. Hence, we also use ringing as an evaluation tool. For visual comparisons, we show images and their patches from upper and lower image portions.

First, we consider a wide-angle system (29 mm and trajectory #39 (Köhler *et al.*, 2012)) in Figs. 3.7(first-row and a-c). Note that the result of (Su and Heidrich, 2015)

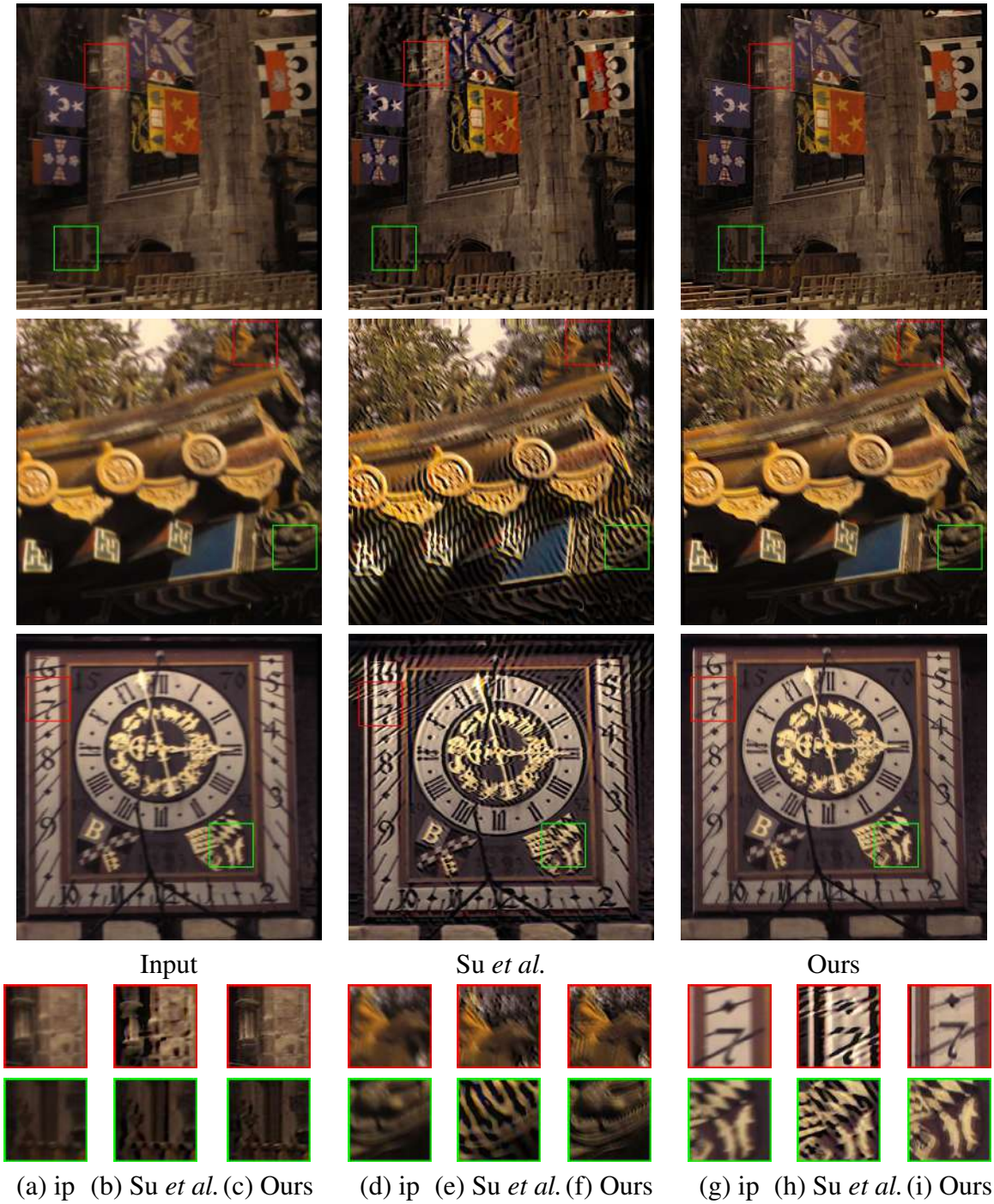
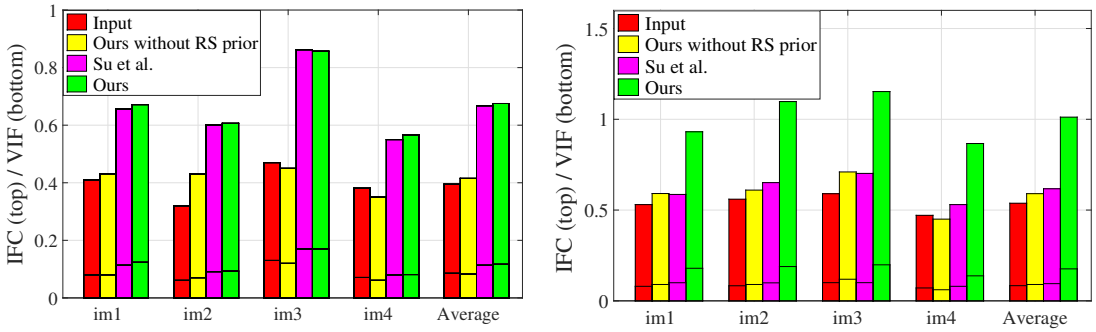


Figure 3.7: Comparison with the state-of-the-art RS deblurring method Su and Heidrich (2015) for different cases: : First row gives a case of wide-angle system, second row gives a case of vibratory motion, and third row provides a case of CCD-blur. (a-i) Two image-patches corresponding to the three rows of different cases. Quantitative evaluation for the three cases is as follows: For an RS wide-angle system (Su and Heidrich (2015) - $\{1.31, 0.23\}$, Ours - $\{\mathbf{1.97}, \mathbf{0.36}\}$), (d-f) For vibratory motion in an RS system (Su and Heidrich (2015) - $\{\ 0.49, 0.076\ \}$, Ours - $\{\mathbf{0.59}, \mathbf{0.086}\}$), and (g-i) For GS blur (Su and Heidrich (2015) - $\{1.25, 0.19\}$, Ours - $\{\mathbf{2.11}, \mathbf{0.32}\}$).



(a) Narrow-angle system (b) Wide-angle system

Figure 3.8: Quantitative evaluation on benchmark dataset Köhler *et al.* (2012) with RS settings. The performance of our method is comparable to that of Su and Heidrich (2015) for narrow-angle systems but outperforms Su and Heidrich (2015) for wide-angle systems; both *sans* RS timings t_r and t_e, unlike Su and Heidrich (2015).

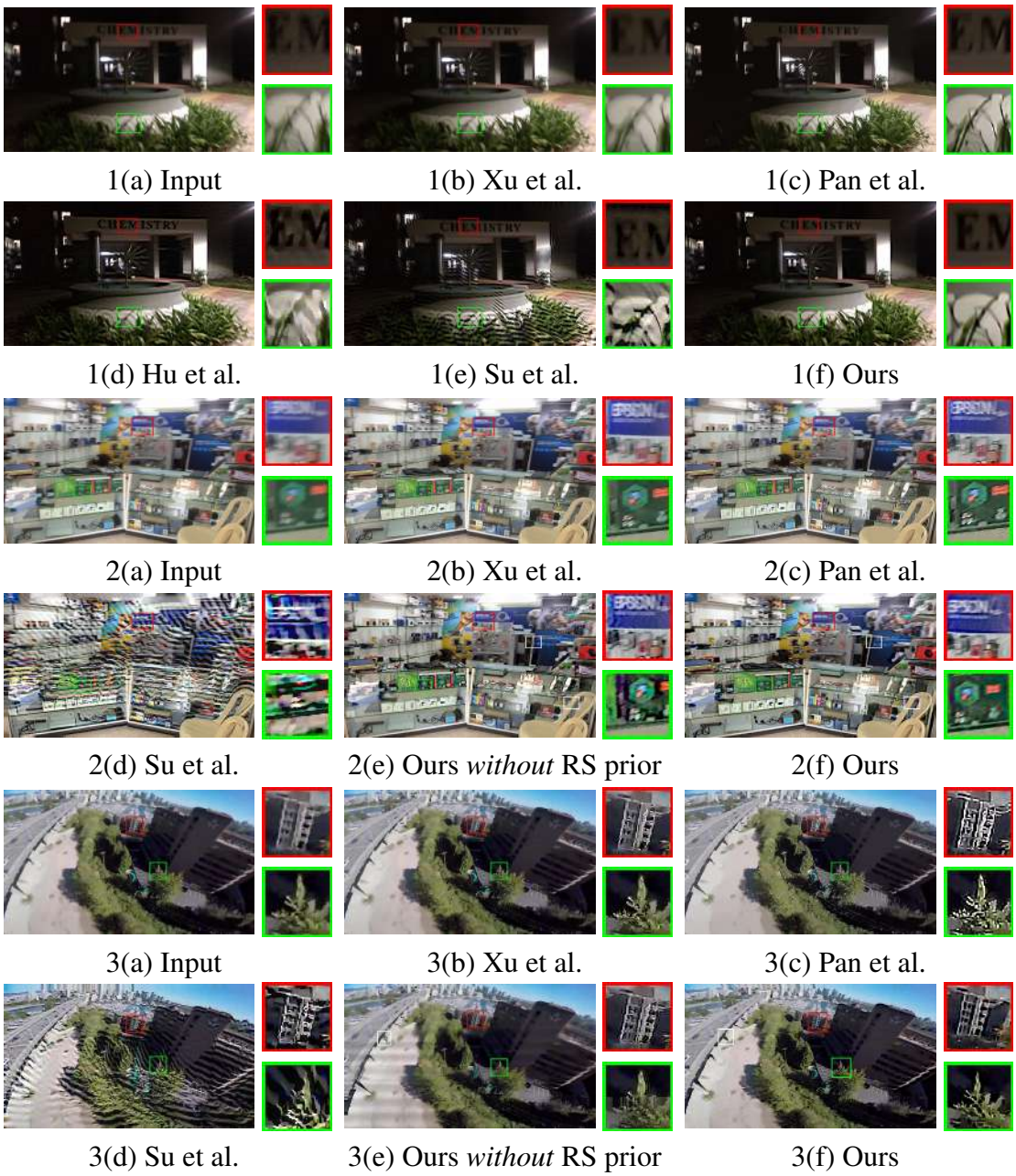
shows moderate ringing artifacts in the top patch (in the wall-linings and lantern), whereas residual blur exists in the lower patch (in the table structure). In contrast, our result recovers fine details in both the patches and with *no* ringing artifacts. This reveals our method's ability to deal with wide-angle systems, unlike (Su and Heidrich, 2015) which is designed for *only* narrow-angle systems.

Next we consider vibration ego-motion in Figs. 3.7(second-row and d-f) that simulates a robotic system with feedback control (Hatch, 2000). Since (Su and Heidrich, 2015) considers *only* narrow-angle systems, we also limit ourselves to narrow-angle setting (50 mm), for a fair comparison. It is evident from the results of Fig. 3.7(e) that the estimated polynomial model fits the initial portion of the trajectory well (top-patch is deblurred), but diverges for the later portion (heavy ringing in bottom patch). In contrast, our method gives good deblurred result uniformly (Fig. 3.7(f)), which underscores the importance of our non-parametric approach to ego-motion.

Finally, we evaluate the performance of RS BMD methods for GS deblurring in Figs. 3.7(third-row and g-i). Here also, we limit to narrow-angle systems (50 mm, trajectory #2 (Köhler *et al.*, 2012)). Either due to the ineffectiveness of polynomial approximation or initialization error, the result of (Su and Heidrich, 2015) has moderate ringing artifacts with residual blurs. Our result reveals that our model generalizes to GS blur well, as compared to (Su and Heidrich, 2015).

A detailed evaluation on dataset (Köhler *et al.*, 2012) for narrow- and wide-angle systems is given in Fig. 3.8. It clearly reveals that our method is either comparable to or

Figure 3.9: Comparisons for RS narrow-angle examples in dataset Su and Heidrich (2015). Our method provides negligible ringing artifacts and fine details, as compared to the state-of-the-art RS-BMD Su and Heidrich (2015). (Table 3.1(450×800 entry) gives the speed-up.) Note the effect of incoherent combination due to the block shift-ambiguity (Section 3.3, claim 1) in (i)-first row, which is successfully suppressed by our RS prior ((i)-second row).



1(a) Input 1(b) Xu et al. 1(c) Pan et al.

1(d) Hu et al. 1(e) Su et al. 1(f) Ours

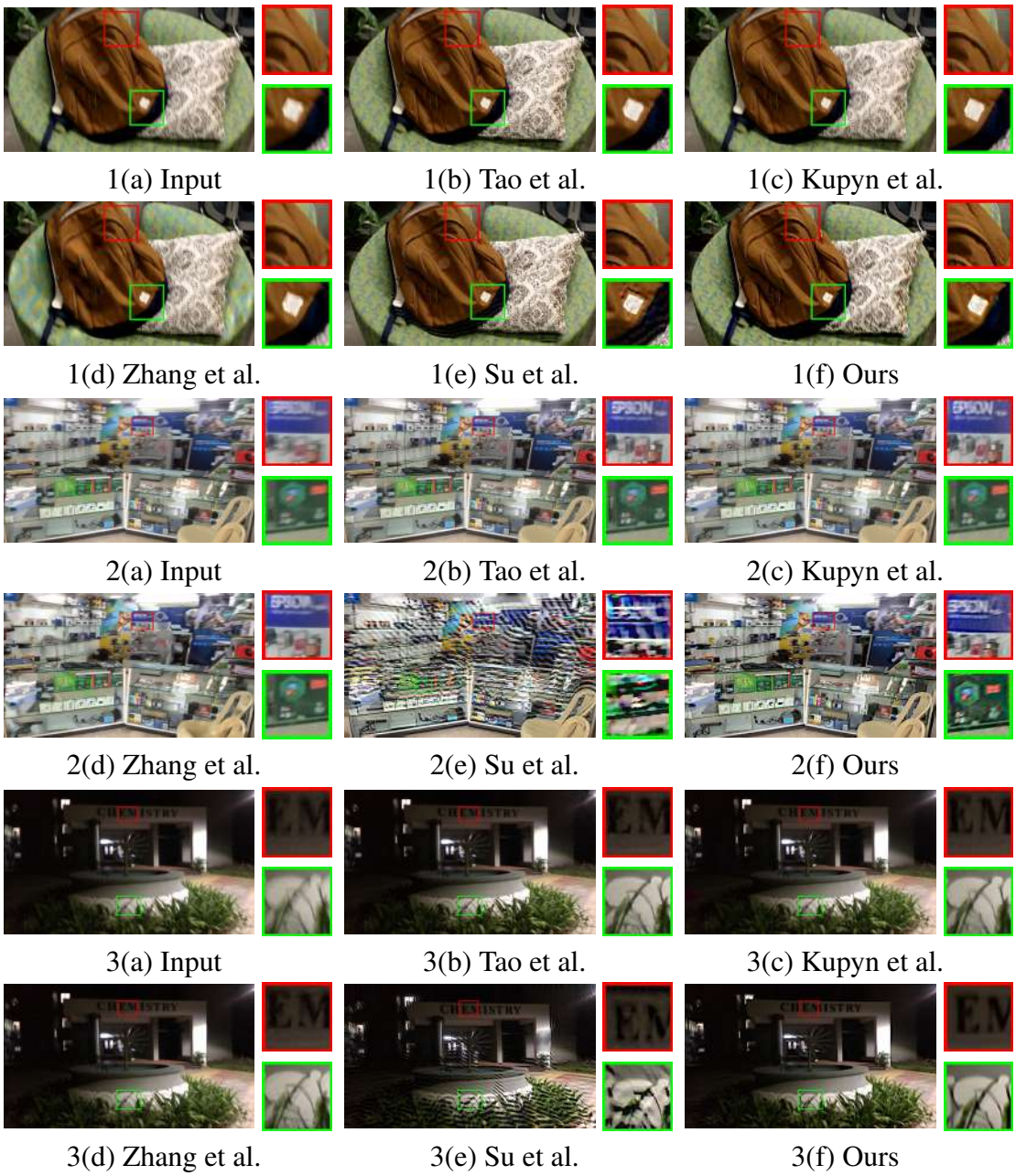
2(a) Input 2(b) Xu et al. 2(c) Pan et al.

2(d) Su et al. 2(e) Ours *without* RS prior 2(f) Ours

3(a) Input 3(b) Xu et al. 3(c) Pan et al.

3(d) Su et al. 3(e) Ours *without* RS prior 3(f) Ours

Figure 3.10: Comparisons for RS wide-angle examples (1 - low-light scenario, 2 - indoor case, and 3 - outdoor case). As compared to the competing methods, our method models the RS ego-motion better and produces consistent results overall.



1(a) Input 1(b) Tao et al. 1(c) Kupyn et al.

1(d) Zhang et al. 1(e) Su et al. 1(f) Ours

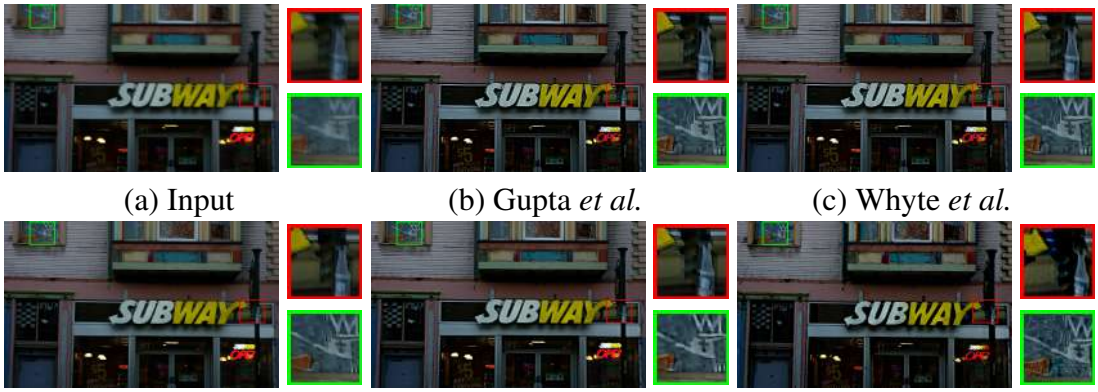
2(a) Input 2(b) Tao et al. 2(c) Kupyn et al.

2(d) Zhang et al. 2(e) Su et al. 2(f) Ours

3(a) Input 3(b) Tao et al. 3(c) Kupyn et al.

3(d) Zhang et al. 3(e) Su et al. 3(f) Ours

Figure 3.11: Comparisons with deep learning methods (Tao *et al.*, 2018; Kupyn *et al.*, 2018; Zhang *et al.*, 2019). As compared to the deep learning methods, our method recovers more details from RS blurred images. This is possibly due to the unique characteristics of RS blur as compared to dynamic scene blur.



(a) Input (b) Gupta *et al.* (c) Whyte *et al.*

(d) Xu *et al.* (e) Pan *et al.* (f) Ours

Figure 3.12: Comparisons for CCD blur example in dataset Pan *et al.* (2016). Our result is comparable with Gupta *et al.* (2010); Whyte *et al.* (2012); Xu *et al.* (2013) and Pan *et al.* (2016).

better than (Su and Heidrich, 2015) for narrow-angle systems. The performance of our method is strikingly superior for wide-angle systems. Note that all these are achieved *without* requiring t_r and t_e, unlike (Su and Heidrich, 2015).

Real examples: In Fig. 3.9, we evaluate our method on real examples for RS narrow-angle systems using the dataset of (Su and Heidrich, 2015). The output of state-of-the-art RS-BMD (Su and Heidrich, 2015) contains residual blur and ringing artifacts compared to ours. Specifically, in the first example, the characters in patch 1 and details in patch 2 are sharper in our output. In the second example, the minute structures of the bag-zipper in patch 1 are restored well, while ringing in patch 2 is negligible. In Fig. 3.10, we evaluate our algorithm for wide-angle systems (including irregular camera motion). The first example depicts the case of low-light imaging, the second example is an indoor case, and the third example is a drone image (outdoor and irregular ego-motion). It is evident from the results of Fig. 3.10 that our method consistently delivers good performance over (Su and Heidrich, 2015) in all the scenarios. The performance degradation of (Su and Heidrich, 2015) in Fig. 3.10 may be attributed to its inability to handle wide-angle systems (first and second examples) and irregular ego-motion (third example). The comparison with deep learning methods (Tao *et al.*, 2018; Kupyn *et al.*, 2018; Zhang *et al.*, 2019) in Fig. 3.11 amply demonstrates the ineffectiveness of those networks for RS deblurring. This can be possibly due to the unique properties of RS blur, which is not present in their training datasets. Figures 3.9 and 3.10 also reveal the inability of CCD deblurring methods to handle blur in RS systems. Also, the effect of prior is qualitatively analysed in Fig. 3.9, which clearly reveals the prior's potential

(in removing aliasing of image-blocks). Finally, Fig. 3.12 considers a GS-blur case, which demonstrates that our method is comparable to state-of-the-art GS-BMD, while applicable for RS-BMD as well (as demonstrated earlier).

3.6.1 Implementation Details

We implemented our algorithm in `MATLAB`. We empirically set 7 scales, each with 7 iterations, in our scale-space framework (Section 3.4). The blurred image in the ith scale is formed by downscaling the input image by a factor of $(1/\sqrt{2})^{i-1}$. For ego-motion estimation (Section 3.4.2), we consistently used the regularization α (in Eq. (3.13)) in level i as 2^{7-i} (so that the RS prior can cope with the increasing image size, and thus the data fidelity magnitude $\|\mathbf{Fw} - \nabla\mathbf{B}\|_2^2$, in finer levels). We used the MDF regularization β' (in Eq. (3.13)) as 0.01. For latent image estimation (Section 3.4.3), we used $R = 48$ such that each image-patch is square, and with 6 patches along the shorter dimension and 8 patches along the longer dimension. For the Richardson-Lucy deconvolution (employed in the last iteration of the finest level), we used a total number of 30 iterations. For the selection of block-size (Section 3.5.1), we selected an initial block-size r_0 as 145, and a downscaling factor of 2 (i.e., $M_0 = M/2$ and $N_0 = N/2$).

3.7 Conclusions

In this chapter, we proposed a block-wise RS blur model for RS deblurring. We provided a detailed analysis of this model, and addressed invertibility issues using a computationally tractable convex prior. We also proposed an efficient filter flow framework that offers significant computational edge. Experiments reveal that our algorithm achieves state-of-the-art results in terms of deblurring quality as well as computational efficiency. Unlike existing RS deblurring methods, it can seamlessly accommodate wide- and narrow-angle systems, blur due to hand-held and irregular ego-motion, and GS as well as RS images; all *without* the need for sensor information.

The motion deblurring method presented in this chapter is pertaining to rolling shutter cameras, which at a given time captures only a single image. Another important imaging modality is light field cameras, which captures multiple images of a scene to

aid post-capture refocusing, f-stopping (Ng *et al.*, 2005) and depth sensing (Tao *et al.*, 2013). Motion blur is a common artifact in light field cameras too. The increasing popularity of light field (LF) cameras necessitates the need for tackling motion blur in this imaging modality as well. But unlike RS deblurring discussed in this chapter, deblurring problem in LF cameras introduces additional challenge due to its multi-image capture and unique imaging model, which we consider in the following chapter.

CHAPTER 4

Full-Resolution Light Field Deblurring

4.1 Introduction and Related Works

[1] Handheld light field cameras (LFCs) are being used in a variety of tasks including wide-angle and hyperspectral imaging, shape recovery, segmentation, etc. (Dansereau *et al.*, 2017; Xiong *et al.*, 2017; Li *et al.*, 2017; Tao *et al.*, 2013). The increase in popularity of LFCs can be attributed to their attractive features over conventional cameras (CCs), including post-capture refocusing, f-stopping, and depth sensing (Tao *et al.*, 2013; Adelson and Wang, 1992; Ng *et al.*, 2005). LFCs achieve this by capturing multiple (subaperture) images instead of a single CC image by segregating the light reaching the CC-sensor into multiple angular components; and synthesize these images post-capture to form an image of desired CC setting (Ng *et al.*, 2005; Adelson and Wang, 1992). However, there is a downside too. The nuisance effect of motion blur becomes exacerbated in LFCs. This is because the light-segregation principle in LFCs reduces the amount of photons that make up individual subaperture images, thereby necessitating higher exposure times relative to CC (under the same setting). This escalates the risk of motion blur in LFC. Moreover, a 4D LF comprising of 2D spatial and 2D angular resolutions can be interpreted as several CC images stacked together. Thus the numerical optimization involved in LF deblurring must deal with *very* large-sized data as compared to that of CC. This poses additional computational challenges (Wu *et al.*, 2017; Srinivasan *et al.*, 2017).

In this chapter, we address the problem of LF blind motion deblurring (LF-BMD), i.e., the problem of estimation of clean LF and underlying camera motion from a *single* motion blurred LF. As discussed in Chapter 2, BMD in CCs is a well-studied topic replete with efficient methodolgies. State-of-the-art CC-BMD methods (Pan *et al.*, 2016; Xu *et al.*, 2013; Su and Heidrich, 2015) are based on the motion density function (MDF)

[1] Based on: Divide and Conquer for Full-Resolution Light Field Deblurring. *Mahesh Mohan M. R. and Rajagopalan A. N.*; CVPR 2018, IEEE Publications, Pages 6421–6429.

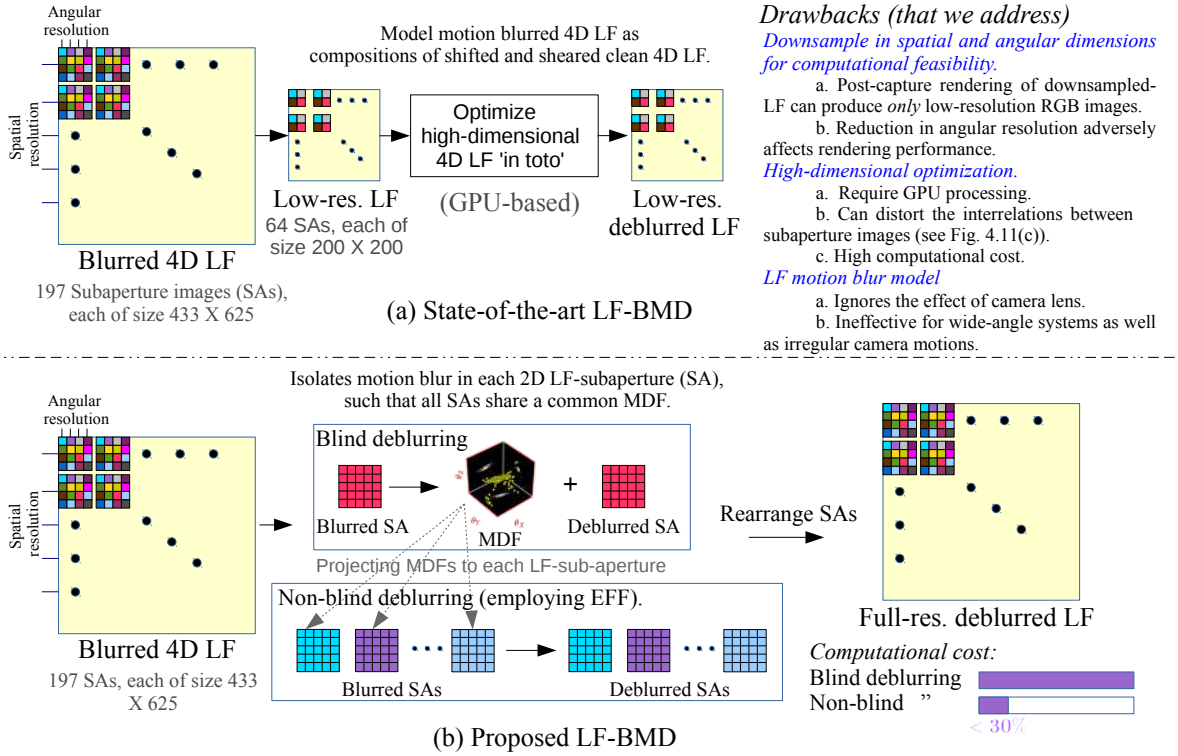


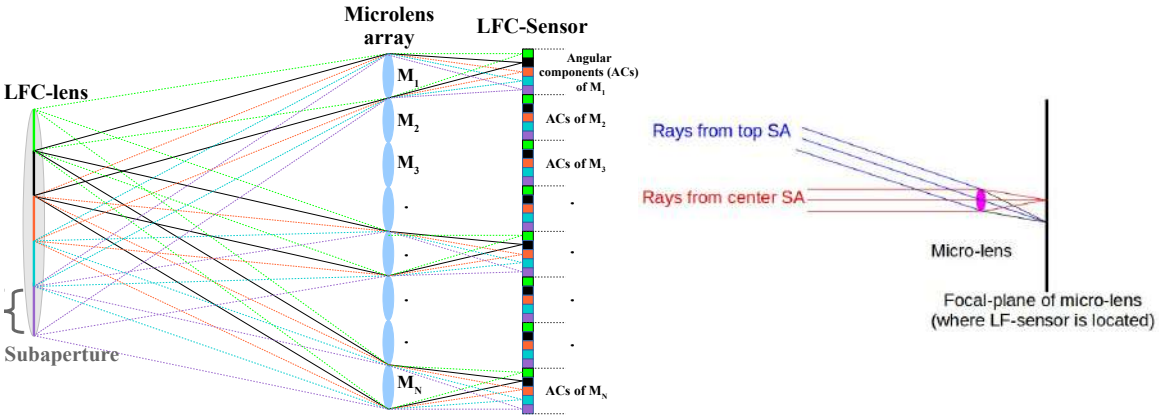
Figure 4.1: (a) Working and drawbacks of the state-of-the-art LF-BMD method (Srinivasan *et al.*, 2017) (b) Outline of our proposed method: Our LF-BMD enables decomposing 4D LF deblurring problem into a set of *independent* 2D deblurring sub-problems, in which a blind deblurring of a *single* subaperture-image enables *low-cost* non-blind deblurring of individual subaperture images in parallel. Since all our sub-problems are 2D (akin to CC-case) and thus cost-effective (as it allows efficient filter flow or EFF (Hirsch *et al.*, 2010) and is CPU-sufficient), our method is able to handle full-resolution LFs, with significantly less computational cost.

(Gupta *et al.*, 2010) which allows both narrow- and wide-angle systems as well as non-parametric camera motion, possess a homography-based filter flow framework for computational efficiency (Hirsch *et al.*, 2010), and employ a scale-space approach to accommodate large blurs. Köhler *et al.* (2012) have shown (albeit for CCs) that general camera motion comprising of 3D translations and 3D rotations can be well approximated by full rotations, or inplane translations and inplane rotation. Inplane rotation common to both the above approximations are necessary to capture wide angle settings (Mohan *et al.*, 2017; Su and Heidrich, 2015). Deblurring methods of Whyte *et al.* (2012); Xu *et al.* (2013); Pan *et al.* (2016) follow the full-rotations approximation and employ efficient filter flow (Hirsch *et al.*, 2010) to yield high quality results in CC-BMD.

In contrast, LF-BMD is an emerging research area and there exists very few works. Jin *et al.* (2015) proposed the first LF-BMD approach, but restrict the scene to be fronto-parallel and bilayer, and limit the camera motion to *only* inplane translations. It works

by reconstructing the support of foreground/background layers, corresponding sharp textures, and motion blurs via an alternating minimization scheme. Further the method shows that the reconstruction of the support of these two layers from a single image of a conventional camera is not possible. A recent LF-BMD work by Srinivasan *et al.* (2017) eliminates the planar scene assumption and even includes full 3D translations. However, there the ego-motion is constrained to be parametric. This *reduces* its effectiveness under irregular ego-motions, which is common when imaging from moving vehicles, robotic platforms, etc. Moreover, since the translational pose cannot model inplane rotation, both (Jin *et al.*, 2015) and (Srinivasan *et al.*, 2017) are *ineffective* for wide angle systems. Dansereau *et al.* (2016) introduce a hardware-assisted deblurring approach using a robotic arm to mount the camera and estimate camera motion (hence non-blind). It generalizes Richardson-Lucy (RL) deblurring to 4D light fields by replacing the convolution steps with light field rendering of motion blur, and introduces a regularization term that maintains parallax information in the light field. The work of (Srinivasan *et al.*, 2017) deals with blind motion deblurring for general 3D scenes. Here, LFC is modeled as an array of pinhole cameras by discarding the effect of LFC-lens, and the motion blurred 4D LF is treated as a composition of shifted and sheared versions of a clean 4D LF. Deblurring with this model proceeds by optimizing for a clean 4D LF at 'one go', using a 4D prior (Srinivasan *et al.*, 2017).

However, (Srinivasan *et al.*, 2017) has some major drawbacks. First, optimization of 4D LF *in toto* brings up new challenges. The computational requirement involved in this optimization restricts (Srinivasan *et al.*, 2017) to handle *only* downsampled LFs – both in spatial and angular resolutions (e.g., a Lytro Illum LF file decoded using (Dansereau *et al.*, 2013) has 197 subaperture images of size 433×625, whereas (Srinivasan *et al.*, 2017) requires downsampling it to 64 images of size 200×200 for computational feasibility). As LF post-capture rendering involves composition of multiple subaperture images (or angular components), angular downsampling can adversely affect rendering performance. On the other hand, spatial downsampling restrics LF-rendering software to produce *only* low-resolution RGB images. Furthermore, optimizing high-dimensional data elevates the computational complexity and the method warrants GPU-based processing. Also, such a high-dimensional optimization can distort the interrelations among subaperture images due to convergence issues, which is an important factor for consistent post-capture rendering of LFs (Ng *et al.*, 2005). Fig-



(a) Formation of 4D light field (b) Working of a micro-lens

Figure 4.2: LF motion blur model: (a) As compared to a conventional camera (CC), a light field camera (LFC) further segregates light in accordance with which portion of the lens the light come from. A micro-lens array in place of CC sensor performs this segregation (b) A micro-lens array focuses light coming from different inclination to different LFC sensor coordinates.

ure 4.1 summarizes these drawbacks. Other very recent LF-BMD works include (Lee *et al.*, 2018) which primarily deals with deblurring *only* the center subaperture image and (Lumentut *et al.*, 2019) which is a deep learning based approach. Notably, both these works assume a parametric ego motion. Lee *et al.* (2018) works by estimating center subaperture image (SAI), depth map, and camera motion from a blurred 4D light field, and uses the estimated center SAI and depth to warp other SAIs (another limitation of this scheme is that occlusion cues in noncenter SAIs will be lost). Lumentut *et al.* (2019) works by generating a LF blur dataset considering 6D motion, and employs an end-to-end network to directly regress the deblurred LF via an MSE loss.

In this chapter, we introduce an MDF-based LF motion blur model which allows for decomposition of LF-BMD into low-dimensional subproblems. This admits an efficient filter flow framework (Hirsch *et al.*, 2010) to remove the computational bottlenecks and several other limitations of the state-of-the-art methods. Specifically, our model *isolates* blur formation in *individual* subaperture images (unlike (Srinivasan *et al.*, 2017)), and imparts a *dependency* among *all* subaperture images through a *common* MDF. Our formulation performs LF-BMD in two efficient steps, as illustrated in Fig. 4.1. First, we estimate the common MDF from the center-subaperture image using BMD (akin to CC-BMD). Second, by invoking the blur-isolation and commonality of MDF properties inherent to LFCs, we perform *independent* (or parallelizable) non-blind deblurring of individual subaperture images using the estimated MDF while simulataneously ac-

counting for the lens-effect and parallax arising from separation of subapertures from the lens-center. Since each of these subproblems is low-dimensional, our method overcomes the drawbacks associated with the high-dimensional optimization, and thus can deblur LFs at full-resolution (Srinivasan *et al.*, 2017). In addition, unlike (Srinivasan *et al.*, 2017), our LF-MDF model captures the effect of camera lens, can cater to both wide- and narrow-angle camera settings, and can handle irregular camera motions.
Our main contributions are summarized below.

- By harnessing the physics behind LF, we decompose 4D LF-BMD to 2D subproblems, which enables the first ever attempt of full-resolution LF-BMD.
- Our work bridges the gap between the well-studied CC-BMD and emerging LFC-BMD, and facilitates mapping of analogous techniques (such as MDF formulation, efficient filter flow framework, and scale-space strategy) developed for the former to the later.
- Our work dispenses with some important limitations impeding the state-of-the-art (Srinivasan *et al.*, 2017), such as high computational cost, GPU requirement, and ineffectiveness in handling wide-angle systems & irregular ego-motions.

4.2 Understanding Light Field Camera

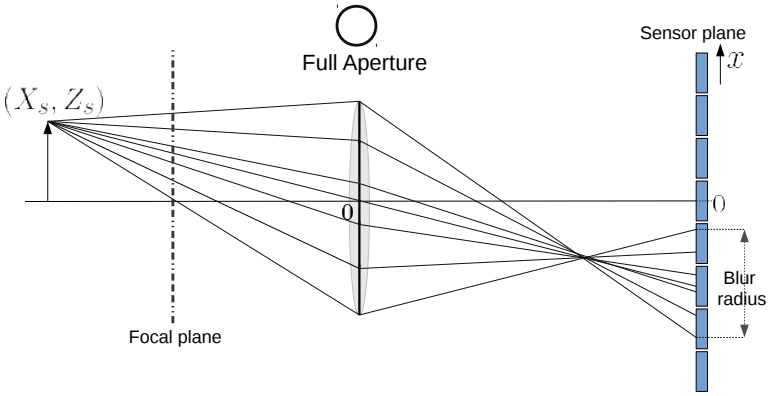
In this section, we describe the working of light field camera, first based on what happens inside the LF camera body, and next in relation to the imaging principles of CC.

A CC with a large-aperture setting spatially resolves light onto a 2D sensor array, but fails to capture information of how the light coming from one part of the lens differs from the light coming from another. In particular, a CC sensor captures in a given sensor element the sum total of light rays coming through the entire lens-aperture to that element. This can be visualized using Fig. 4.2(a), if we consider the micro-lens array as the CC sensor; note that individual bundle of rays from each part of the lens (color-coded in Fig. 4.2(a)) integrates in each CC sensor-element, without allowing them to decompose. These individual bundles of rays, if acquired, address some major problems in conventional photography – enabling post-capture refocusing and f-stopping, and depth sensing (Ng *et al.*, 2005). As compared to a CC, a light field camera further resolves the light coming from different parts of lens-aperture (in 2D angular bins over aperture, referred to as sub-aperture). In other words, if we consider CC as capturing 2D spatial information, then an LFC captures 4D information, containing 2D spatial

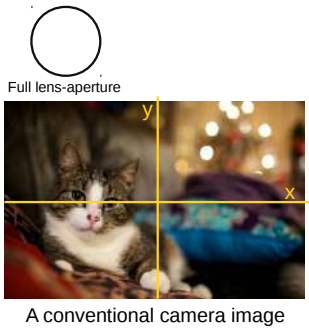
information as well as 2D angular information. Intuitively, an LFC captures *multiple* 2D images, each imaged through a particular subaperture.

Figure 4.2(a) illustrates the LF capturing mechanism inside the camera body: An LFC further segregates the light in each CC sensor-element in accordance with the subaperture through which the light arrives. This is achieved by means of a microlens placed in the position of each CC sensor element, which resolves each spatial component into multiple angular components. The 2D spatially as well as 2D angularly resolved light is stored in a high-resolution LFC-sensor (behind the microlens array) to form a 4D LF. We now proceed to explain the working of a micro-lens. As shown color-coded in Fig. 4.2(a), each microlens maps to a small array of bins in the LFC-sensor. As a micro-lens is orders of magnitude smaller than the lens-aperture (≈ 3), the bundle of rays from each subaperture appears to be parallel to the micro-lens (Ng *et al.*, 2005). Therefore, due to the parallel rays, individual bundle of rays focuses on the focal-plane of micro-lens, where the LFC-sensor is placed, and falls in separate sensor-bins of the micro-lens in accordance with the arriving angle of rays (Fig. 4.2(b)).

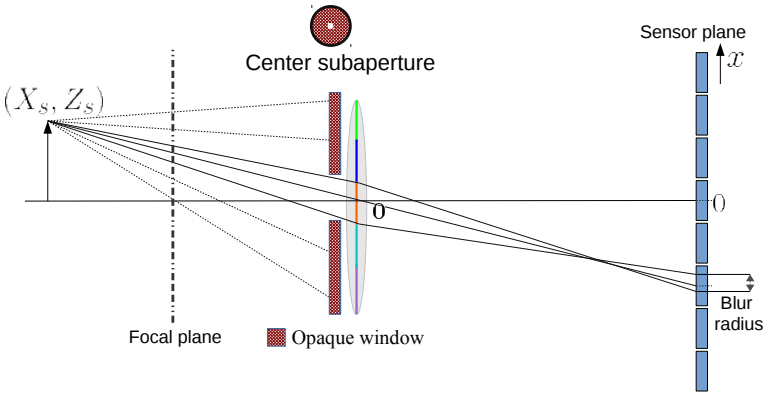
Without focusing on what happens inside camera body (as discussed earlier), the working of a LFC can be closely related to that of a CC. The design of LFC-lens system is such that light from different subapertures do *not* interfere with each other (Ng *et al.*, 2005). From Fig. 4.2(a), it is clear that a subaperture image is equivalent to a 2D image formed in a CC with full-aperture setting but with *only* the respective subaperture open, and the microlens-array is assumed to be a CC-sensor. It is well-known that, for a CC with a large-aperture, the scene-points that are behind or in front of the focal-plane creates defocus blur, with the blur-radius depending on the size of aperture and scene-point depth (Rajagopalan and Chaudhuri, 1999). This is illustrated in Fig. 4.3(a) for a scene-point behind the focal-plane, and a sample defocus blurred image is shown in Fig. 4.3(b). Further, as shown in Fig. 4.3(c-d), if the CC aperture-size is reduced the blur will be less. The center-subaperture image in a LFC forms in a similar way as that of CC with a small aperture and the image formed in the position of micro-lens array. Note that as the center-subaperture is over the lens-center, it incurs negligible refraction and hence the popular pinhole model can be well-employed in this case. A non-centered subaperture image can also be interpreted in the same way, by restricting the rays of the CC through the respective subaperture, projecting onto the CC sensor (Fig. 4.3(e-f)). Due to the restricted entry of rays, as in the previous case, the blur will



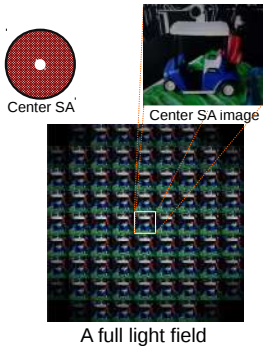
(a) Formation of CC image for a full lens-aperture



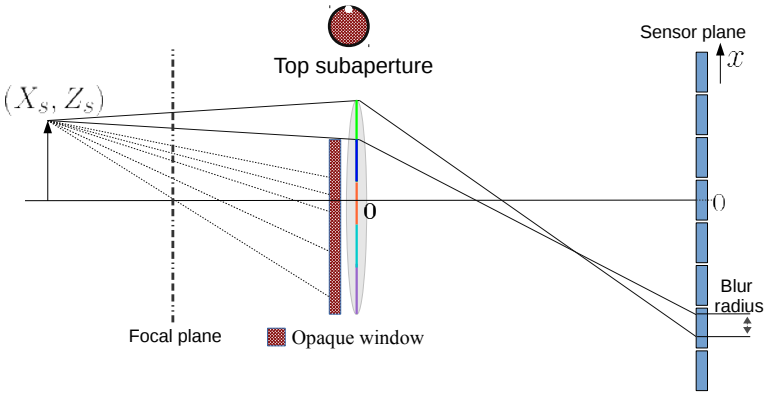
(b) A CC image



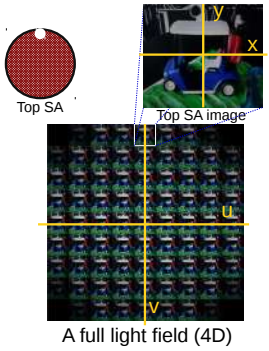
(c) Formation of center-SA image in an LFC



(d) A center SAI in LF



(e) Formation of top-SA image in an LFC



(f) A top SAI in LF

Figure 4.3: Working of a light field camera (LFC) in relation to that of a conventional camera (CC). (a-b) The image formed in a CC with a large-aperture creates defocus blur in accordance with the aperture-size and scene-depth. (c-f) Individual subaperture image in an LFC is equivalent to the image formed in the CC-sensor, but by restricting the light rays to *only* pass through the respective subaperture. Therefore, individual subaperture images contain negligible defocus blur. Also, note the 4D nature of LF (Fig. (f)) as compared to the 2D nature of CC image (Fig. (b)).

be less; however, due to the refraction incurred over the lens' non-centered regions in this case, the pin-hole model is *not* applicable here. More important, *the refraction effect in a subaperture increases as we move farther away from the lens-center*.

4.3 MDF for Light Field Camera

In this section, we discuss the limitations of the LF motion blur model of (Srinivasan *et al.*, 2017). We then proceed to conceptualize (akin to conventional cameras) an MDF based interpretation for motion blur in LFs, that seeks to mitigate the drawbacks of (Srinivasan *et al.*, 2017).

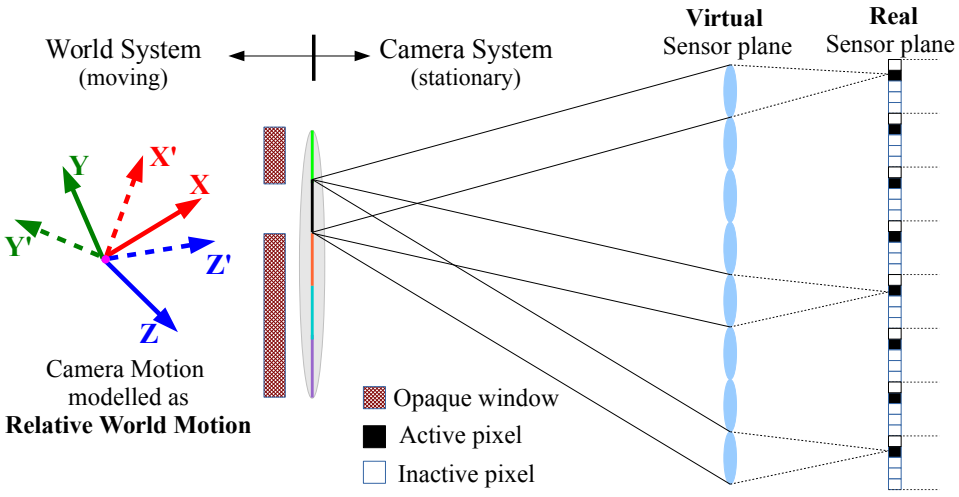
The LF motion blur model of (Srinivasan *et al.*, 2017) discards the effect of LF-lens and approximates LFC as an array of pinhole cameras positioned at the subapertures. The camera motion is interpreted as an associated movement of these pinhole cameras. In effect, a motion blurred LF is modeled as a composition of shifted and sheared versions of the clean 4D LF. Using flatland analysis (i.e., considering a single angular dimension v and a single spatial dimension x), a motion blurred 4D LF ($\mathbf{LF_b}$) can be represented as

$$\mathbf{LF_b}(x, v) = \int_t \mathbf{LF_c}(x, u + \mathbf{p_x}(t) - x\mathbf{p_z}(t))dt, \tag{4.1}$$

where $\mathbf{LF_c}$ is the clean 4D LF and $\{\mathbf{p_x}(t), \mathbf{p_z}(t)\}$ is the camera motion path during the exposure time. As only the angular term v is varying in Eq. (4.1), a motion blurred subaperture image can be interpreted as a composition of multiple clean subaperture images (where the amount depends on camera motion). In the above equation, considering a *single* blurred 2D subaperture image as the observation amounts to solving for *multiple* clean 2D images as unknowns – a heavily ill-posed problem. Instead, (Srinivasan *et al.*, 2017) considers the entire blurred 4D LF as observation and solves for a clean 4D LF as unknown. This reduces the ill-posedness but incurs high-dimensional optimization issues, as discussed in Sec. 4.1.

Motion blur in CC is typically modelled using MDF $\mathbf{w}$ (Chapter 2) as

$$\mathbf{B} = \sum_{\mathbf{p} \in \mathbb{P}} w(\mathbf{p}) \cdot \mathbf{L}(\mathbf{K}, \mathbf{R_P}), \tag{4.2}$$



(a) Formation of blur in a subaperture image

Figure 4.4: LF motion blur model: (a) Interpreting camera motion as relative world motion, each motion blurred 2D subaperture image is obtained as a combination of the projections of moving world (parametrized by a single MDF) through the *respective* subaperture onto a virtual sensor or microlens array. Also, all subapertures experience the *same* world motion (or share a *common* MDF).

where $\mathbf{R_p}$ spans the plausible camera pose-space $\mathbb{P}$ and the MDF for a given pose gives the fraction of exposure time the camera stayed in that particular pose. Along similar lines, it is possible to conceptualize camera shake in LFC to be uniquely characterised by an MDF, but having one-to-many mapping from world to LF-sensor due to LF-capture mechanism. To develop an analogous MDF framework for LF-BMD, we leverage the light field imaging principle in relation to the CC cameras (Fig. 4.3), i.e., a clean subaperture image is equivalent to a 2D image formed in a CC with full-aperture setting and with *only* the respective subaperture open.

Akin to CC, we too interpret motion to be stationary camera and a world moving (Fig. 4.4). It can be shown that the rotations-only approximation in CC is valid for LFCs as well (see Sec. 4.6.1). Each motion blurred subaperture image is thus equivalent to an image formed in a full-aperture CC with *only* the respective subaperture open. Interestingly, note that all subapertures are subjected to same world-motion (i.e., parameterized by a *single* MDF). Intuitively, *each motion blurred subaperture image is formed by a linear combination of the projections of the moving world through the respective subaperture onto a virtual sensor formed by the microlens array*. Thus, a

blurred subaperture image $\mathbf{B}_k$ can be alternatively expressed as

$$\mathbf{B}_k = \sum_{\mathbf{p} \in \mathbb{P}} w(\mathbf{p}) \cdot \mathbf{L}_k(\mathbf{K}_k, \mathbf{R}_\mathbf{p}, \gamma_k) \qquad 1 \leq k \leq N, \tag{4.3}$$

where $\mathbf{L}_k$ is the clean version of the kth subaperture image. Note that the camera matrix $\mathbf{K}_k$ and parameter set γ_k vary with individual subapertures so as to capture one-to-many world-mapping. Their exact forms will be discussed in a later section. Our model in Eq. (4.3) isolates the blur in individual subaperture images, i.e., a *single* 2D blurred subaperture image as observation amounts to solving for the *corresponding* clean subaperture image as unknown (unlike (Srinivasan *et al.*, 2017)). Also, a single MDF $w(\mathbf{p})$ is shared by all subaperture images.

The MDF-based LF motion blur model in Eq. (4.3) provides three distinct advantages. First, because it *isolates* motion blur in individual subaperture images, we can estimate the common MDF from a *single* subaperture image – a low-dimensional optimization (akin to CC-BMD). Second, since all subaperture images share a *common* MDF, we can use the estimated MDF to perform non-blind deblurring of all the other subaperture images. As non-blind deblurring of individual subapertures can be done *independently*, this step is amenable to parallelization. Note that non-blind deblurring methods (which optimize for a clean 2D image *only* once) are quite cost-effective as compared to blind methods (which cumbersomely optimize for MDF and clean 2D image alternately over iterations). These factors drastically reduce the computational cost for LF-BMD and allow full-resolution LF-BMD. Third, since MDF captures both regular and irregular ego-motion, our method can handle unconstrained ego-motion; and consideration of full rotational camera motion accommodates both narrow- and wide-angle systems, unlike (Srinivasan *et al.*, 2017; Jin *et al.*, 2015).

4.4 MDF-based LF Motion Blur Model

In this section, we formulate our MDF-based light field motion blur model. The MDF formulation requires world-to-sensor mapping in each subaperture, so as to derive individual LF homographies.

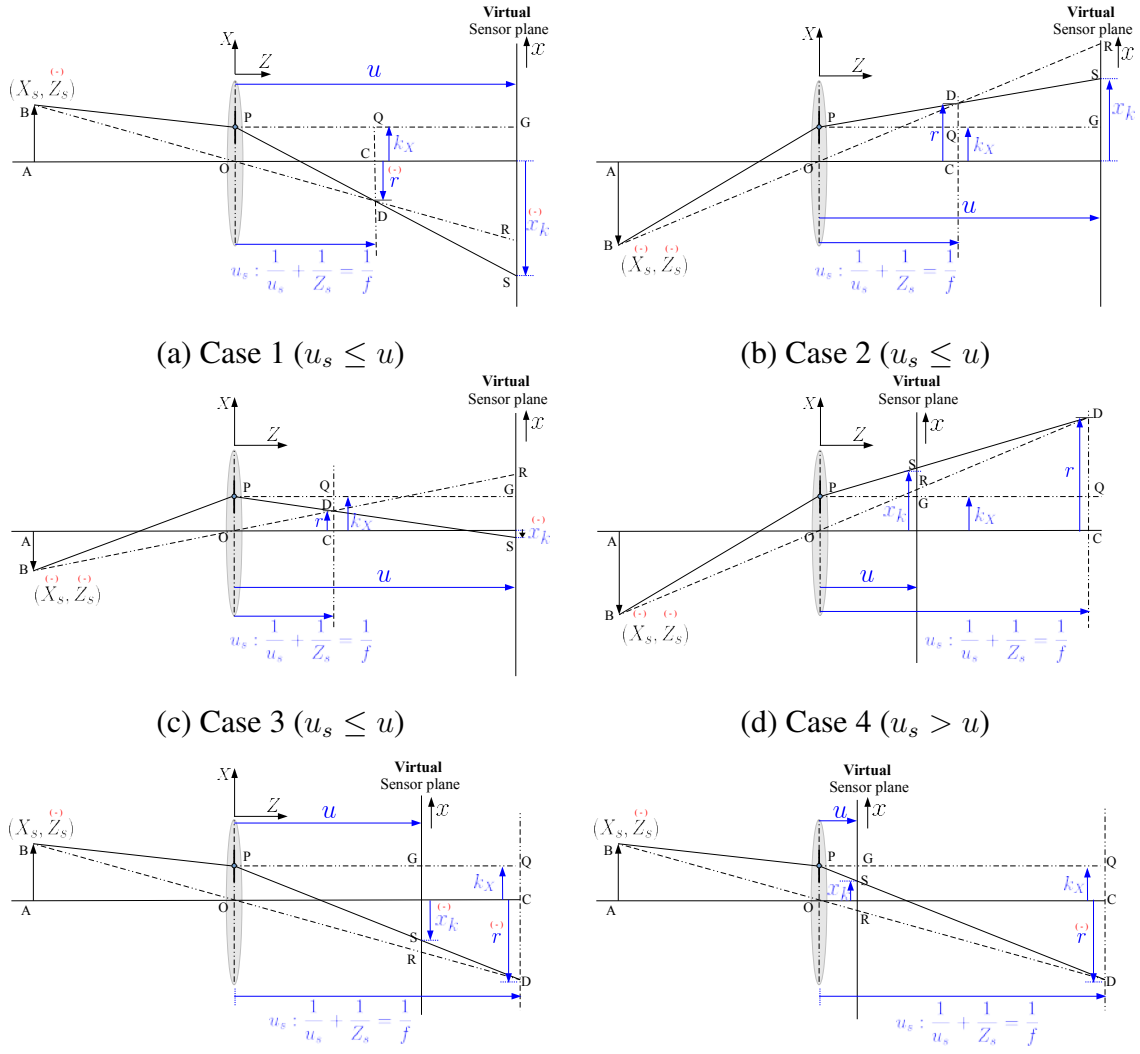


Figure 4.5: LFC Mappings: (a-c) and (d-f) An exhaustive set of world-to-sensor mappings of a scene-point focused before and after the sensor-plane ($u_s \leq u$ and $u_s > u$) for subapertures positioned at positive X axis, respectively. The derived relations are also valid for subapertures at negative X, due to its symmetry about the optical axis.

4.4.1 World-to-Sensor Mapping in a Subaperture

Conventional cameras with a small-aperture setting can be well-approximated by a pinhole centered at the aperture's center. This approximation is widely used in many practical applications (including CC-BMD) (Hartley and Zisserman, 2003; Pan *et al.*, 2016; Xu *et al.*, 2013). In LFCs, the characteristics of light refraction over different subapertures vary in accordance with their positions due to the effect of large-aperture lens (Sec. 4.2). This effect cannot be captured with a pinhole array (as the main lens is not involved); e.g., a beam of parallel rays through LFC-lens *converge* at the focal point, but will pass *parallel* through a pinhole camera array. To account for this effect, we

approximate subapertures as pinholes over subaperture-centers, and yet configured to obey the refractions incurred at that portion.

Figure 4.5(a) shows a flatland ray tracing model for a subaperture positioned above the optical-center and a world point with positive X coordinate. Following the thin-lens equation with focal length f, a light-ray from a world point $\{X_s, Y_s, Z_s\}$ through the subaperture has to pass through the point of intersection of the principal ray (i.e., a ray through the optical center) and a fronto-parallel plane at a distance u_s from the optical center, where u_s is given by

$$\frac{1}{u_s} + \frac{1}{|Z_s|} = \frac{1}{f} \implies u_s = \frac{f|Z_s|}{|Z_s| - f}. \tag{4.4}$$

Note that world coordinate Z_s is negative according to our convention (i.e., $|Z_s| = -Z_s$). From Fig. 4.5(a), similarity of triangles ΔABO and ΔODC gives

$$\frac{-r}{X_s} = \frac{u_s}{-Z_s} \implies r = \frac{u_s X_s}{Z_s}. \tag{4.5}$$

From similarity of ΔPQD and ΔPGS, we get

$$\frac{k - r}{k - x_s} = \frac{u_s}{u} \implies x_s = r \cdot \frac{u}{u_s} - k \cdot \left(\frac{u}{u_s} - 1\right). \tag{4.6}$$

Figures 4.5(b-c) illustrate all cases of the same subaperture but with world point having negative X coordinate. It can be easily verified that Eqs. (4.4)-(4.6) hold good for this situation as well. Moreover, (due to symmetry about the optical axis) these equations are valid even for subapertures positioned below the lens-center.

In Fig. 4.5(d-f), we depict various cases of $u_s > u$ for a subaperture positioned above the optical-center. Following the derivation for $u_s \leq u$ cases, it can be shown that Eqs. (4.4)-(4.6) hold true for various cases of $u_s > u$ as well; i.e., valid irrespective of the scene-point location and the sensor-plane placement ($u > u_s$ or $u \leq u_s$). Due to the symmetry about the optical axis of ray diagrams, these relations are *equally valid for subapertures positioned at negative X axis*. The above discussion establishes that Eqs. (4.4)-(4.6) are quite general in nature.

Substituting in Eq. (4.6), u_s and r from Eqs. (4.4)-(4.5) yields

$$\begin{aligned} x_s &= \frac{uX_s}{Z_s} - k \cdot \left(\frac{u}{f|Z_s|} \cdot (|Z_s| - f) - 1 \right), \\ &= \frac{uX_s}{Z_s} - k \left(\frac{u}{f} - 1 \right) - \frac{ku}{Z_s}, \quad \because |Z_s| = -Z_s. \end{aligned} \tag{4.7}$$

The flatland analysis of Eq. (4.7) can be extended to 3D world coordinate system and a 2D sensor plane as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} u & 0 & k_x(f-u)/f \\ 0 & u & k_y(f-u)/f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \frac{1}{Z} \begin{bmatrix} k_x u \\ k_y u \\ 0 \end{bmatrix}, \tag{4.8}$$

where we have dropped the subscript s for brevity, and k_x and k_y are the distances of subaperture from the optical-center in x and y directions, respectively. Representing $[x, y, 1]^T$ as $\mathbf{x}$, $[X, Y, Z]$ as $\mathbf{X}$, the matrix as $\mathbf{K}_{k_{xy}}$, and the vector as $\mathbf{b}_{k_{xy}}$, Eq. (4.8) can be concisely represented as

$$\mathbf{x} = \frac{1}{Z} \cdot (\mathbf{K}_{k_{xy}} \mathbf{X} - \mathbf{b}_{k_{xy}}), \tag{4.9}$$

where subscript k_{xy} represents individual subapertures in accordance with their separations k_x and k_y. Note that the matrix $\mathbf{K}_{k_{xy}}$ is of full rank or invertible. Further, the world-to-sensor mapping of CC (see Chapter 2) is a special case of Eq. (4.9), i.e., the case that of the center subaperture or $k_x = k_y = 0$.

4.4.2 Homographies for LFC blur

In this section, analogous to the homography mapping in conventional cameras (Sec. 2.1), we derive the homography transformation of LFC for a single camera pose-change and then extend it to our MDF-based blur model. Consider a single world coordinate change from $\mathbf{X}$ to $\mathbf{X}'$ as

$$\mathbf{X}' = \mathbf{R}\mathbf{X}. \tag{4.10}$$

Using world-sensor mapping in Eq. (4.9), the mapping of sensor coordinate from $\mathbf{x}$ to $\mathbf{x}'$ (corresponding to the world coordinate mapping from $\mathbf{X}$ to $\mathbf{X}'$) is given by

$$\begin{aligned} &Z' \cdot \mathbf{K}_{k_{xy}}^{-1}\mathbf{x}' + \mathbf{K}_{k_{xy}}^{-1}\mathbf{b}_{k_{xy}} = Z \cdot \mathbf{R}\mathbf{K}_{k_{xy}}^{-1}\mathbf{x} + \mathbf{R}\mathbf{K}_{k_{xy}}^{-1}\mathbf{b}_{k_{xy}} \\ &\text{or } \mathbf{x}' = \frac{1}{Z'}\left(Z\mathbf{K}_{k_{xy}}\mathbf{R}\mathbf{K}_{k_{xy}}^{-1}\mathbf{x} + (\mathbf{K}_{k_{xy}}\mathbf{R}\mathbf{K}_{k_{xy}}^{-1} - \mathbf{I})\mathbf{b}_{k_{xy}}\right) \end{aligned} \tag{4.11}$$

which gives the homography mapping for subaperture k_{xy}. From Eqs. (4.3)&(4.11), the parameter set γ_i comprises of $\mathbf{b}_i$ and scene depth, to capture the parallax and lens effects. Note that the homography formulation of CC is a special case of Eq. (4.11), which corresponds to that of the center subaperture image (i.e., $k_x = k_y = 0$, which makes $\mathbf{K}_{k_{xy}} = \mathbf{K}$ and $\mathbf{b}_{k_{xy}} = \mathbf{0}$, thereby resulting in $\mathbf{x}' = \frac{Z}{Z'}(\mathbf{K}\mathbf{R}\mathbf{K}^{-1}\mathbf{x})$).

Considering multiple pose-changes, we can represent the motion-blurred subaperture image $\mathbf{B}_{k_{xy}}$ as

$$\mathbf{B}_{k_{xy}} = \sum_{\mathbf{p}\in\mathbb{P}} w(\mathbf{p}) \cdot \mathbf{L}_{k_{xy}}(\mathbf{K}_{k_{xy}}, \mathbf{R}_{\mathbf{p}}, \gamma_{k_{xy}}), \tag{4.12}$$

where $\mathbf{L}_{k_{xy}}(\cdot)$ performs the warping function according to Eq. (4.11) and MDF $w(\mathbf{p_0})$ represents the fraction of time the world stayed in rotational pose $\mathbf{R}_{\mathbf{p_0}}$. Note that the MDF $w(\mathbf{p})$ is *shared* among all the subapertures.

We also throw light on the possibility of individually deblurring subaperture images using CC-BMD. Assuming $\mathbf{K}_{k_{xy}} = \mathbf{K}\ \forall k_{xy}$ and neglecting $\gamma_{k_{xy}}$ necessitates different MDFs for capturing the one-to-many mapping of LF. This distorts their mutual consistencies (e.g., see epipolar image of Figs. 4.12(d-e)), mainly due to shift-ambiguity of latent image-MDF pair (Mohan *et al.*, 2017), and relative estimation-error of different MDFs. This adversely affects the refocusing and f-stopping of LFs (Ng *et al.*, 2005). Furthermore, since blind deblurring is significantly costlier than non-blind deblurring, the computational cost climbs steeply with increase in spatial and angular resolutions.

4.5 Optimization of LF-BMD

In this section, we discuss our divide and conquer strategy for LF-BMD that consists of two steps: First, estimate the *common* MDF from a *single* subaperture image using

regular BMD, and second, employ the estimated MDF to perform low-cost non-blind deblurring of *remaining* subaperture images (in parallel) using respective homography mapping (of Sec. 4.4.2).

4.5.1 LF-MDF Estimation

The homography mapping of the center-subaperture image (i.e., $k_x = k_y = 0$ in Eq. (4.11)) is equivalent to that of a CC-pinhole model (Hartley and Zisserman, 2003), i.e.,

$$\mathbf{x}' = \lambda \cdot \mathbf{K}\mathbf{R}\mathbf{K}^{-1}\mathbf{x} \quad \because \mathbf{K}_{k_{xy}} = \mathbf{K} \text{ and } \mathbf{b}_{k_{xy}} = 0, \tag{4.13}$$

where scalar λ ($= Z/Z'$) normalizes the third coordinate of $\mathbf{x}'$ to unity (Eq. (4.8)). Note that even though depths Z and Z' are present in Eq. (4.13), it is *not* required for homography mapping (and thus for MDF estimation) since it translates to a normalization of the third coordinate of $\mathbf{x}'$ to unity (see structure of $\mathbf{x}$ in Eq. (4.8)) through λ (Hartley and Zisserman, 2003; Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012). Thus, any state-of-the-art CC-BMD method can be employed to find the LF-MDF using the center-subaperture image.

We now analyse the effect of adding more subaperture images (SAIs) to estimate the MDF (instead of one SAI that we followed). Incorporating more SAIs does *not* produce any significant improvement in MDF, while *accentuating* the computational cost. Typically, MDF is estimated as $\hat{\mathbf{w}} = \min_{\mathbf{w}} \|\mathbf{H}_{k_{xy}}\mathbf{w} - \mathbf{B}_{k_{xy}}\|_2 + \lambda\|\mathbf{w}\|_1$. For a maximum 30 pixel blur, 3D rotation space binned by 1 pixel is $\mathbf{29}^3$. Considering a single SAI ($< 1\%$ data), the number of equations (or the number of SAI pixels) will be $\mathbf{10}$X as that of the number of unknowns, which is already an *overdetermined system* and sufficient for MDF estimation (Whyte *et al.*, 2012). Incorporating n more SAIs scales the number of equations by order of n (*but the effect of* more *equations retains the overdetermined nature*), while incurring additional cost for creating *individual* $\mathbf{H}_{k_{xy}}$s and *handling* large matrix (n $\mathbf{H}_{k_{xy}}$s stacked).

4.5.2 EFF for Non-Blind Deblurring of LFs

Since a common MDF is shared among all subaperture images, we utilize the estimated MDF to perform non-blind deblurring of individual subaperture images. For a non-centered subaperture, camera matrix $\mathbf{K}_{k_{xy}}$ varies with subaperture positions and the additive term of Eq. (4.11) is nonzero (which makes it different from the CC-pinhole case). Eventhough $1/Z'$ in Eq. (4.11) can be obtained by normalization (as in CC-case), the depth information Z is required for homography mapping to capture parallax and lens effect. A direct approach for non-blind deblurring involves constructing a large matrix $\mathbf{M}_{kxy}$ using the MDF formulation of Eq. (4.12) for each subaperture k_{xy}, to solve the optimization problem

$$\hat{\mathbf{L}_{k_{xy}}} = \min_{\mathbf{L}_{k_{xy}}} \|\mathbf{M}_{k_{xy}}\mathbf{L}_{k_{xy}} - \mathbf{B}_{k_{xy}}\|_2^2 + \text{prior}(\mathbf{L}_{k_{xy}}), \tag{4.14}$$

where 'prior' is an image regularizer, such as total variation (TV), sparsity in image gradient (Xu *et al.*, 2013), dark channel (Xu *et al.*, 2013), etc. As a full-resolution LF is composed of numerous subaperture images, construction of $\mathbf{M}_{kxy}$ and optimization of individual subaperture images with priors are computationally expensive. To this end, we elegantly extend the efficient filter flow (EFF) employed in CCs (Hirsch *et al.*, 2010) to LFCs.

The EFF approximates space-variant blur in an image to be locally space-invariant over small image patches. Using this approximation, we can simplify the blurring process in a subaperture image as

$$\mathbf{B}_{k_{xy}} = \sum_{i=1}^{R} \mathbf{C}_\mathbf{i}^\dagger \cdot \left\{ \mathbf{h}_{k_{xy}}^i * (\mathbf{C}_\mathbf{i} \cdot \mathbf{L}_{k_{xy}}) \right\}, \tag{4.15}$$

where i iterates over R overlapping patches in clean subaperture image $\mathbf{L}_{k_{xy}}$, $\mathbf{C_i} \cdot \mathbf{L}$ is a linear operation which extracts the ith patch from the image $\mathbf{L}$, $(\mathbf{h} * \mathbf{C_i} \cdot \mathbf{L})$ performs a convolution with kernel $\mathbf{h}$ on ith patch, and $\mathbf{C}_\mathbf{i}^\dagger$ inserts the patch back to its original position with a Barlett windowing operation. The convolution kernel $\mathbf{h}_{k_{xy}}^i$ corresponding to the ith patch center can be derived using Eq. (4.12) as

$$\mathbf{h}_{k_{xy}}^i = \mathbf{C_i} \cdot \left(\sum_{\mathbf{p}} w(\mathbf{p}) \cdot \delta^i(\mathbf{K}_{k_{xy}}, \mathbf{R}_\mathbf{p}, \gamma_{k_{xy}}) \right), \tag{4.16}$$

where δ^i is an image of the same size as that of the subaperture image with only an impulse located at the ith patch center. EFF requires MDF-based motion blur model to be calculated *only* at patch centers and eliminates the need for building large matrices for optimization, as in Eq. (4.14). The EFF allows for an efficient patch-based deblurring:

$$\hat{\mathbf{L}_{k_{xy}}} = \sum_{i=1}^{R} \mathbf{C}_\mathbf{i}^\dagger \cdot \text{deconv}\left(\mathbf{h}_{k_{xy}}^i, (\mathbf{C_i} \cdot \mathbf{B}_{k_{xy}})\right), \tag{4.17}$$

where 'deconv' indicates non-blind deconvolution which is computationally efficient as compared to optimization based deblurring (of Eq. (4.14)).

4.6 Analysis and Discussions

In this section, we elaborate on the validity of rotation-only approximation in LF-BMD, and depth estimation. Further, we consider the effect of noise in our LF-BMD and propound ways to suppress it.

4.6.1 Rotation-only approximation

In this section, we justify our rotation-only model for light field cameras. As discussed in Sec. 4.5.1, the effect of camera motion in center subaperture of light field camera is equivalent to that of a conventional camera. However for a non-centered subaperture, there exists an *inherent* translation component due to subaperture separation from the lens-center, apart from the effect of camera rotation and translation. Note that we have already accounted for the inherent translation component or parallax (via $\mathbf{b}_{k_{xy}}$, as noted in Sec. 4.4.2) together with the camera rotation motion in our blur model, i.e., Eq. (4.11) can be decomposed as:

$$\mathbf{x}' = \frac{1}{Z'}\left(\underbrace{Z\mathbf{K}_{k_{xy}}\mathbf{R}\mathbf{K}_{k_{xy}}^{-1}\mathbf{x} + (\mathbf{K}_{k_{xy}}\mathbf{R}\mathbf{K}_{k_{xy}}^{-1})\mathbf{b}_{k_{xy}}}_{\text{Effect of Camera Rotation}} \quad \underbrace{-\mathbf{b}_{k_{xy}}}_{\text{Inherent translation of SAs}}\right) \tag{4.18}$$

The 3D approximation of general 6D motion (3D translations and 3D rotations) is typically followed to reduce the computational cost for ego-motion estimation. It is shown

in (Pan *et al.*, 2016; Whyte *et al.*, 2012) that 3D camera translations can be neglected in conventional cameras (and hence for center subaperture). In contrast, in non-centered subapertures, as we have already considered their inherent translations, the camera translation will have only an additional effect due to lens-effect (refraction of lens) as compared to CCs. We show here that this lens-effect is negligible, which justifies the assumption of rotation-only model for light field cameras as well. Considering the worst-case *plausible* camera translation as $\widehat{\mathbf{t}_w} = [|r|, |r|, |r|]$, (Whyte *et al.*, 2012) shows for CC that the corresponding worst-case pixel translation $\mathbf{t} = [t_x, t_y, t_z] = K_0\widehat{\mathbf{t_w}}/Z'$ can be *ignored* ($\mathbf{K}_0 = \text{diag}(u, u, 1)$). We claim that pixel translations in our LFC-model is *equivalent* to that of CC for inplane translations, and *approximates* $\mathbf{t}$ for 3D transaltion $\widehat{\mathbf{t}_w}$. Considering $\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{t}_w$ in Eq. (4.10), translation $\mathbf{t}$ for subaperture k_{xy} (using Eq. (4.11)) amounts to $\hat{\mathbf{t}} = \mathbf{K}_{k_{xy}}\mathbf{t}/Z' = (\mathbf{K}_0 + \mathbf{M}_{k_{xy}})\mathbf{t}/Z'$, where $\mathbf{M}_{k_{xy}} = [0, 0, k_x(f-u)/f; 0, 0, k_y(f-u)/f, 0; 0, 0, 0]$. Note that the components of $\mathbf{M}_{k_{xy}}$ constitute the lens-effect (Sec. 4.4.1). For $t_z = 0$, $\mathbf{M}_{k_{xy}}\mathbf{t}/Z' = 0$ (i.e., *equal* effect in CC and LFC). Note that worst-case displacement happens for highest k_x or k_y ($= f/4$). For conventional camera $\widehat{\mathbf{t}}$, LFC $\hat{\mathbf{t}} = [\alpha t_x, \alpha t_y, t_z]$, where $\alpha = 1+|(f-u)/4u|$ or $1+|f|/(4|Z'|)$ (using Eq. (4.4)). As $|Z'|$ is in the order of m and $|f|$ in mm, $1 \approx \alpha < 2$ (i.e. $\hat{\mathbf{t}} \approx \mathbf{t}$). Hence proved.

4.6.2 Depth Estimation

To estimate the convolution kernels for different SAIs (for EFF), our algorithm requires depth information (γ in Eq. (4.16)). We use (Tao *et al.*, 2013) to estimate depth for each patch by picking the most-confident depth estimate within that patch (without final depth refinement). Our consideration of uniform depth within a small image patch is analogous to the flatness and global smoothness priors commonly used for final depth-refinement (Tao *et al.*, 2013; Janoch *et al.*, 2013). Depth estimation method in (Tao *et al.*, 2013) is as follows. Refocusing LF translates to a skew in epipolar images, and their features for a image point will be vertical (or horizontal depending on projection) when it is at focus, and slanted when it is out-of-focus (Ng *et al.*, 2005). (Tao *et al.*, 2013) skews epiolar images corresponding to different depths, and picks among them the depth which makes those features vertical. Motion blurred LFs also possess this

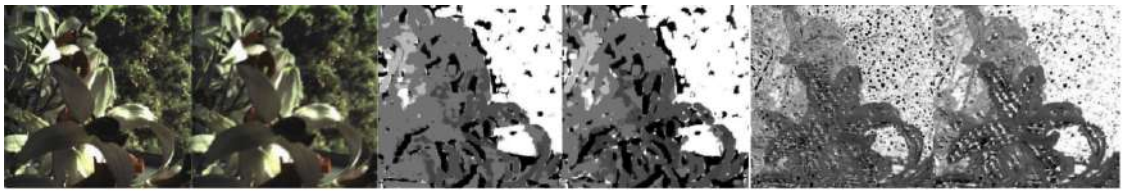
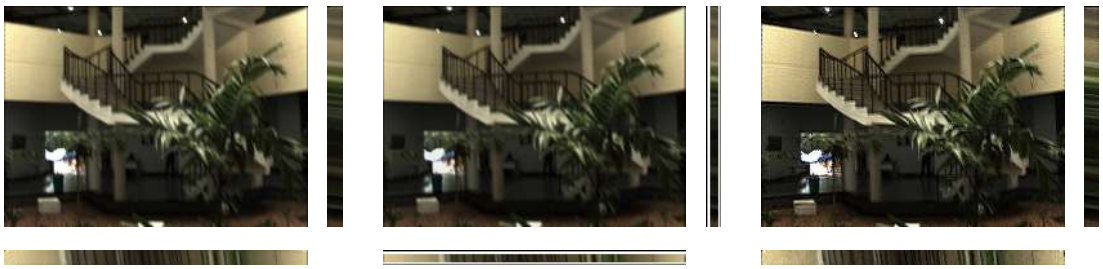


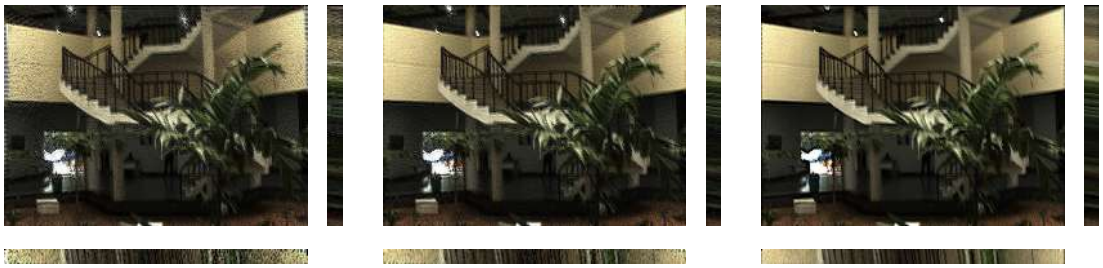
Figure 4.6: Evaluation of depth estimation cues: The first and second entry provides a clean and blurred LF. The third and fourth entries (and fifth and sixth entries) show respective estimated depth using defocus cue (and correspondence cue).

Deconvolution method	Direct (Gaussian)	Fast hyper-Laplacian	0.8 norm on gradients	Richardson Lucy
Time/SA image (**Full-res. LF**)	1.1 second (closed-form)	6.2 seconds (lookup table)	55 seconds (50 iters.)	80 seconds (50 iters.)

Table 4.1: Time per subaperture (SA) image for different LF-EFF deconvolution methods for full-resolution LFs.



(a) Input (b) Srinivasan et al. (c) Gaussian prior



(d) Krishnan et al. (e) Levin et al. (f) Richardson Lucy

Figure 4.7: Qualitative evaluation of different LF-EFF deconvolutions using a full-resolution LF. (a) Input, (b) LF-BMD result of Srinivasan *et al.* (2017) for reference (2X bicubic-interpolated). (c) Direct approach using Gaussian prior, (d) Fast MAP estimation with hyper-Laplacian prior using lookup table Krishnan and Fergus (2009), (e) MAP estimation with heavy-tailed prior ($\alpha = 0.8$) Levin *et al.* (2007), and (f) Richarson Lucy deconvolution Richardson (1972). Note the ringing artifacts in (c) in the saturated regions (e.g., in lights and door exit). Richardson Lucy deconvolution in (f) produces the best result with negligible artifacts.

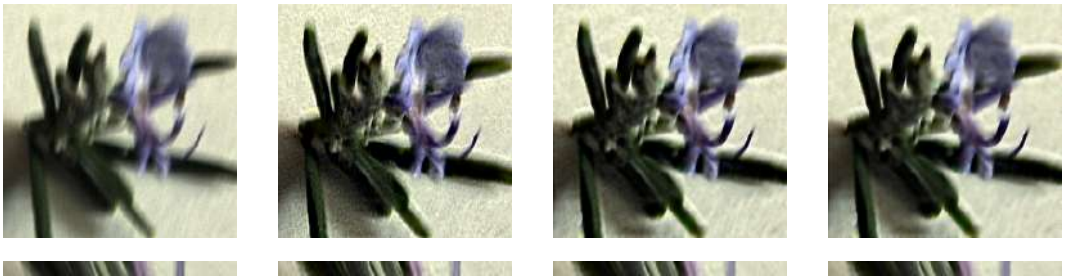
desired characteristics (see epipolar images in Figs. 4.12–4.13(a) and depth cues in Fig. 4.6).

4.6.3 Choice of LF-Deconvolution

In this section, we discuss our choice of deconvolution method employed to perform LF-EFF patch-wise deblurring in Eq. (4.17). A nonblind LF-EFF deconvolution problem, i.e., estimation of a clean image patch given the blur kernel and blurred image patch, possesses multiple solutions due to zero crossings of filter response, saturation or noise effects, etc. Maximum a posteriori (MAP) estimation which imposes prior(s) on clean image patch is typically employed to obtain a single solution from the multiple solution space. A MAP estimation for nonblind deconvolution is given as

$$\hat{\mathbf{L}_{k_{xy}}} = \min_{\mathbf{L}_{k_{xy}}} \|\mathbf{M}_{k_{xy}}\mathbf{L}_{k_{xy}} - \mathbf{B}_{k_{xy}}\|_2^2 + \|\nabla\mathbf{L}_{k_{xy}}\|_\alpha \tag{4.19}$$

where $\mathbf{H}$ captures the blur-kernel information, ∇ is the gradient operator, and $\mathbf{B}_{k_{xy}}$ and $\mathbf{L}_{k_{xy}}$ are blurred and latent image patches of k_{xy}th subaperture, respectively. We considered four different deconvolution approaches: (a) A direct approach which considers Gaussian prior ($\alpha = 2$) and thus has a closed form solution, (b) A fast deconvolution using hyper-Laplacian prior ($0.5 \leq \alpha \leq 0.8$) which is solved using a lookup table (Krishnan and Fergus, 2009), (c) A heavy-tailed prior ($\alpha = 0.8$) which is solved using iterative reweighted least squares process (Levin *et al.*, 2007), and (d) Richardson Lucy deconvolution with smoothness prior which is solved using iterative process (Richardson, 1972). Figure 4.7 provides a representative example of LF deblurring quality (using Fig. 4.13) with different approaches, and Table 4.1 gives the average time per subaperture image; it is evident that there exists a trade-off between visual quality and computational speed. In terms of visual quality, we empirically found out that (Richardson, 1972) is the best, and the direct method comes second but with ringing artifacts (e.g., see Fig. 4.7(c)). In terms of computational time, the direct method is the most efficient, whereas Richardson Lucy method (due to its iterative approach) is less efficient. However, we have selected Richardson Lucy method due to its superior deblurring quality. However, direct deblurring can be selected for computational efficiency, provided one can tolerate minor ringing artifacts.



(a) Input (b) Prior 0.005 (c) Prior 0.009 (d) Prior 0.05

Figure 4.8: Effect of prior in our LF-BMD (using dataset of Srinivasan *et al.* (2017)). (a) Input, (b) Ours with default smoothness regularization (SR) 0.005, (c) Ours with SR 0.009, (d) Ours with SR 0.05. Our result with SR 0.05 prior produces negligible ringing artifacts. Note that our method is CPU-based and yet achieves a speed-up of atleast an order ($\approx$ 17X) as compared to state-of-the-art method of Srinivasan *et al.* (2017) which is GPU-based.

4.6.4 Noise in LF-BMD

LF images captured in low-light scenarios possess higher level of shot noise as compared to that of an analogous CC-camera (due to segregation of photons for angular resolution) (Wu *et al.*, 2017). As deblurring can be interpreted as enhancing the high-frequency content of the scene, LF-BMD also enhances the high-frequency noise (if present). As discussed in Sec. 4.5.1, we consider the center subaperture image to estimate the common LF-MDF using (Whyte *et al.*, 2012). State-of-the-art CC-BMDs frame the objective function in image's gradient space so as to reduce the ill-conditionness (Hirsch *et al.*, 2010; Whyte *et al.*, 2012). Unlike the gradient of scene features which form contiguous segments, the gradients of shot noise form isolated spikes. Harnessing this information, we remove the less-contiguous segments from image-gradient to form the objective function, which reduces the ill-effects of noise in MDF-estimation. For nonblind deblurring (Sec. 4.5.2), we use the estimated MDF to obtain patch-wise kernels for individual subaperture images (Eq. (4.16)), and perform deconvolution using (Richardson, 1972). In case of noisy images, we use a higher smoothness prior (regularization of 0.05) for deconvolution to reduce the noise-effect in deblurred images. Our default regularization value is 0.005. To show how noise can be handled, Fig. 4.8 provides the effect of varying regularization that clearly shows suppression of noise as the prior increases.

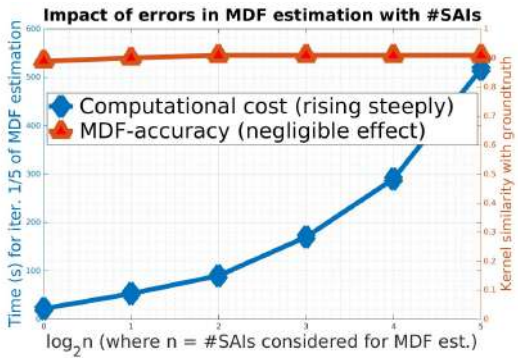


Figure 4.9: Impact of incorporating more subaperture images for camera motion estimation.

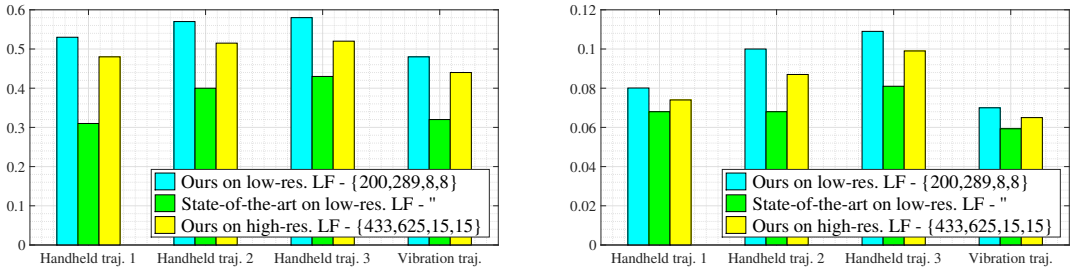
4.6.5 Drawback of decomposing the LF-BMD problem

Our decomposition scheme enables considering the full-resolution LF (which was infeasible by the competing method), and with competitive speed-gain (see Table 2). However, there exists one drawback with respect to accuracy of camera motion estimation. We attempt to experimentally analyse this by considering multiple subaperture images (SAIs) instead of one SAI for camera motion estimation. Figure 6.5 reveals that incorporating more SAIs produces slight improvement in MDF estimation. However, Fig. 6.5 also shows that this *marginal* improvement in accuracy is offset by processing time (in order to optimize with more SAIs as compared to one). The reason for this is explained in Sec.4.5.1.

4.7 Experimental Results

In this section, we provide quantitative and qualitative evaluations to highlight the computational gain of our approach and its ability to deal with full-resolution LFs with competitive performance. We also show that our method can deal with both wide-angle systems and irregular camera trajectories, unlike the state-of-the-art LF-BMD (Srinivasan *et al.*, 2017).

Datasets used: For real experiments on low-resolution LFs, we used the motion blurred LF dataset of (Srinivasan *et al.*, 2017). Since there exist no full-resolution motion blur LF-datasets, we create one with LFs captured using `Lytro Illum`, and decoded raw-LFs to `MATLAB` format full-resolution LFs using (Dansereau *et al.*, 2017). For quantitative evaluation, we synthesized motion blur on clean full-resolution LFs using real



(a) LF-version of IFC (b) LF-version of VIF

Figure 4.10: Quantitative evaluation using the LF-version of VIF and IFC. We use real handheld trajectories (from Köhler *et al.* (2012)) and irregular camera motion using vibration trajectory (from Hatch (2000)). Note that the method of (Srinivasan *et al.*, 2017) *cannot* perform high-resolution LF deblurring.

handheld trajectories from (Köhler *et al.*, 2012) with 29 mm focal-length (wide-angle setting) and $1/50$ s exposure time. For irregular motion, we used real vibratory ego-motion trajectory from (Hatch, 2000).

Comparison methods: We consider mainly the current state-of-the-art LF-BMD (Srinivasan *et al.*, 2017) for evaluation. To demonstrate the *ineffectiveness* of CC methods on LFs, we also use state-of-the-art CC-BMD methods (Krishnan *et al.*, 2011) and (Pan *et al.*, 2016) to perform *independent* deblurring on individual subaperture images. The codes for (Srinivasan *et al.*, 2017; Pan *et al.*, 2016) and (Krishnan *et al.*, 2011) are downloaded from the authors' website, and used their default parameters for all the methods. Note that the code for other LF-BMD is not available for comparison.

Quantitative Evaluation: For LF-BMD benchmarking, we introduce an LF-version of information fidelity criterion (IFC) (Sheikh *et al.*, 2005) and visual information fidelity (VIF) (Sheikh and Bovik, 2006), which are shown to be the best metrics for BMD evaluation for CC in (Lai *et al.*, 2016), by averaging these metric over subaperture images. As processing full-resolution LFs using (Srinivasan *et al.*, 2017) is not feasible, we use a downsampled version (by ≈ 0.5) of our dataset to perform comparisons with (Srinivasan *et al.*, 2017). Using IFC/VIF, Figs. 4.10(a-b) compare with (Srinivasan *et al.*, 2017) for wide-angle scenario (using real trajectories of (Köhler *et al.*, 2012)) and irregular camera motion (using (Hatch, 2000)). It is evident from Fig. 4.10 that our method performs better than (Srinivasan *et al.*, 2017) (performance loss of (Srinivasan *et al.*, 2017) may be attributed to its inability to model these scenarios); ours can also deblur full-resolution LFs (unlike (Srinivasan *et al.*, 2017)). Table 5.2 gives the timing comparisons with the state-of-the-art (Srinivasan *et al.*, 2017). It is evident that,

LF-resolution $\{x, y, u, v\}$	State-of-the-art (**GPU-based**)	Ours (**CPU-based**)
$\{200, 200, 8, 8\}$	2 hrs, 20 mins	8.21 mins (Gain $\mathbf{17.05}\times$)
$\{200, 289, 8, 8\}$ (Low-res. LF)	3 hrs, 17 mins	12.62 mins (Gain $\mathbf{15.61}\times$)
$\{433, 625, 15, 15\}$ (**Full-res. LF**)	Not feasible (Resource allocation error)	38 mins* (**Feasible**)

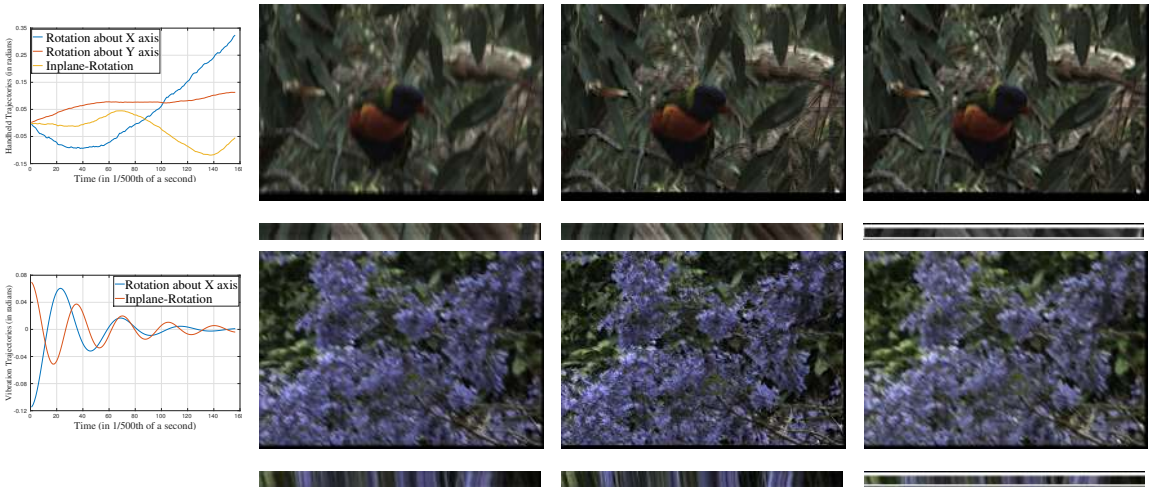
Table 4.2: Time comparisons. *Over 90% of the time is used for *low-cost* 197 non-blind deblurring parallelized in 8 cores of a CPU. Using more cores or GPU further improves the speed significantly. A typical full-resolution LF of consumer LF camera `Lytro Illum` consists of 197 RGB subaperture images of size 433×625.

even though our method uses only CPU, we achieve a gain of atleast an order relative to the GPU-based (Srinivasan *et al.*, 2017). Also, our method performs full-resolution LF-BMD within three-quarters of an hour, which can be further improved using more cores.

Qualitative Evaluation: We qualitatively evaluate our deblurring performance on synthetic and real LFs. Figure 4.11 provides trajectories used for synthetic experiments as well as comparison with competing methods. Figure 4.12 gives an example of real low-resolution LF. Note that the epipolar images of (Srinivasan *et al.*, 2017), (Krishnan *et al.*, 2011) and (Pan *et al.*, 2016) are not consistent with the input. Also, there exists ringing artifacts in Fig. 4.12(c) of (Srinivasan *et al.*, 2017) (especially in upper leaves). In contrast, our result in Fig. 4.12(b) reveals intricate details (see the veins in lower leaf), has negligible ringing artifacts and produces consistent epipolar images. Fig. 4.13 shows comparisons with real full-resolution LFs, where the top and bottom rows depict well-lit and low-lit scenarios, respectively. The LF-BMD of (Srinivasan *et al.*, 2017) processes *only* a downsampled LF (both spatially and angularly) due to computational constraints. In contrast, our method gives superior results in full-resolution and with consistent epipolar images.

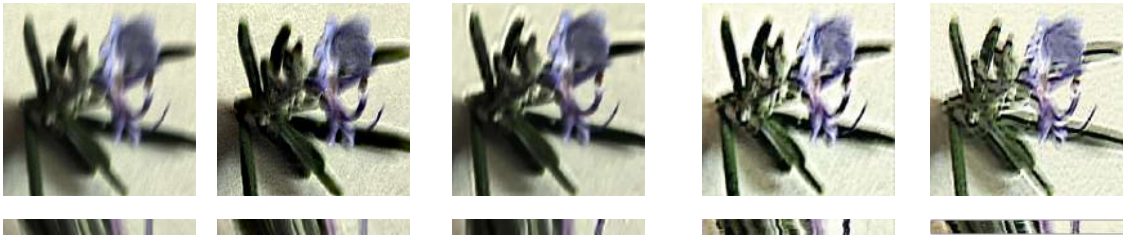
4.7.1 Implementation Details

System Specifications: We used a PC with an Intel Xeon processor and a 16 GB RAM for all CPU-based experiments, and implemented our algorithm in `MATLAB`. The repeatedly used EFF routine is implemented in `C` for computational efficiency. We per-



(a) Trajectories (b) Input (c) Ours (d) Srninivasan et al.

Figure 4.11: Synthetic experiments in dataset (Dansereau *et al.*, 2013) using real handheld (Köhler *et al.*, 2012) and vibration (Hatch, 2000) trajectories. (a) Trajectories, (b) Inputs, (c) Ours, and (d) Bicubic interpolated result of (Srinivasan *et al.*, 2017). Top-row gives a case of handheld trajectory. In d, note that the low-resolution result of (Srinivasan *et al.*, 2017) after interpolation fails to recover intricate details (e.g., feathers in lorikeet's face). Bottom-row gives a case of irregular motion. Deblurring performance of (Srinivasan *et al.*, 2017) in (d) is quite low, possibly due to the inability of its parametric motion model in capturing vibratory motion.



(a) Input (b) Ours (c) Srinivasan et al. (d) Krishnan et al. (e) Pan et al.

Figure 4.12: Comparison using low-resolution LF ($\{200, 200, 8, 8\}$) from dataset of Srinivasan *et al.* (2017). (a) Input, (b) Ours, (c) State-of-the-art LF-BMD Srinivasan *et al.* (2017), (d) State-of-the-art CC-BMD Krishnan *et al.* (2011) (e) State-of-the-art CC-BMD Pan *et al.* (2016). Note the inconsistencies in epipolar image w.r.t input for c (possibly due to convergence issues) and d-e (possibly due to *lack* of dependency among BMD of subaperture images). Also, notice the ringing artifacts in the upper leaves in c. In contrast, ours reveals more details (like veins of lower leaf), has negligible ringing artifacts, and epipolar image is consistent.

(a) Input (b) Ours (c) Srinivasan et al. (d) Pan et al.

Figure 4.13: Comparisons using full-resolution LF ($\{433, 625, 15, 15\}$) of `Lytro Illum`. Top-row shows a well-lit case and bottom row shows a low-light scenario. (a) Input, (b) Ours, (c) State-of-the-art LF-BMD (Srinivasan *et al.*, 2017) and (d) State-of-the-art CC-BMD (Pan *et al.*, 2016). (Srinivasan *et al.*, 2017) can *only* deblur downsampled LF due to computational constraints. Ours produce a superior full-resolution LF with consistent epipolar images in all cases.

form nonblind deblurring of eight subaperture images in parallel. For executing the code of (Srinivasan *et al.*, 2017), we used a GPU-server and employed a Pascal Titan X GPU. Running time reported in Table 5.2 is obtained using these specifications. The camera we used for obtaining full-resolution light field examples is `LYTRO ILLUM 40 Megaray`.

Parameters: We employed Lytro Desktop App to download LF raw images and a publicly available LF toolbox (Dansereau *et al.*, 2013) to decode raw images into LF Matlab file. The camera parameters focal length f and lens-sensor separation u are obtained from Lytro metadata. As Lytro camera has constant aperture setting as $f/2$, we periodically sampled 197 subapertures in a circular disk of the aperture dimension to obtain k_x and k_y. We used camera metadata and a modified source code of (Tao *et al.*, 2013) to produce discrete depth with respect to the center subaperture image in individual patches (as discussed in Sec. 4.5.2).

The sensor coordinate $\mathbf{x}$ corresponding to a scene point varies with subaperture k_{xy} due to parallax and lens effect (e.g., in Fig. 4.5, for the case of $u > u_s$ the depth Z_s of a scene point maps to sensor coordinate at R through the centre pinhole, whereas shifted by RS through the shifted pinhole). As the depth estimate Z obtained using (Tao *et al.*, 2013) is with respect to the center subaperture image, it is required to map this to other non-centered subaperture images for retaining one-to-one correspondence

between x and Z (in Eq. 4.11). This we accomplished by warping the estimated depth (with coordinate x) to subaperture k_{xy} (with coordinate x') as $x' = x - \delta x_{k_{x,y}}$, where $\delta x_{k_{x,y}}$ is derived using similarity of ΔDOP and ΔDRS in Fig. 4.5:

$$\delta x_{k_{x,y}} = k \cdot \frac{u - u_s}{u_s}. \tag{4.20}$$

where u_s is a function of Z. This relation even holds true for the case of $u < u_s$ (which is verifiable using Fig. 4.5).

Development: A pseudo-code is provided in algorithm 1. Our algorithm comprises

Algorithm 1 Light field blind motion deblurring

Require: Decoded motion blurred LF file (LF) (using (Dansereau *et al.*, 2013))
Estimate patch-wise depth using (Tao *et al.*, 2013) (following Sec. 4.6)
$centerSAI \leftarrow LF(0,0)$
Estimate MDF using the $centerSAI$ (employing (Whyte *et al.*, 2012))

for all SAIs (in parallel) **do**
Project blur in SAI patches using the estimated MDF (using Eq. (4.16))
Patch-wise deconvolution using the projected blur (using Eq. (4.17))
Merge individual patches using windowing operation (Sec. 4.5.2)
end for

of two steps: blind deblurring of center subaperture image to estimate the common MDF and project the estimated MDF to other subaperture images to perform nonblind deblurring (in parallel) employing EFF. For the first step, as the MDF-based source code of the best CC-BMD (Pan *et al.*, 2016) is not available and (Xu *et al.*, 2013) provides only an executable code, we used a modified code of (Whyte *et al.*, 2012) to incorporate LF parameters. For the scale-space based alternative minimization for MDF and latent image, we used 5 scales with 6 iterations each. For all experiments, we used MDF regularization as 0.01 and total variation regularization as 0.005. For the second step, we implemented a C-based EFF code to obtain kernels corresponding to the patch centers using Eq. (4.16), and employ Richardson Lucy method in Eq. (4.17).

4.8 Conclusions

We introduced a new interpretation of motion blur in 4D LF as *independent* blurring of multiple 2D images, yet all sharing a *common* motion parametrization. This paved the

way for performing LF deblurring as a single 2D blind deblurring (to estimate the common motion) and parallelizable *low-cost* 2D non-blind deblurring of multiple images. Our approach overcomes several major drawbacks of the state-of-the-art, such as heavy computational cost, ability to deblur *only* low-resolution LFs, and GPU-processing. Unlike the state-of-the-art, our model realistically captures refraction effects of lens, and works for wide-angle scenarios and irregular ego-motion as well. As LF cameras continue to evolve with higher resolutions, our divide and conquer strategy will be invaluable for full-resolution deblurring.

Light field cameras discussed in this chapter capture multiple images, which share the same exposure time, resolution, and focal length due to the micro-lens array set-up. Yet another computational cameras that are popularized by today's smartphones, and provide similar functionality as LF cameras are unconstrained dual-lens cameras. In these cameras, the multiple images can have different exposure times, resolutions and focal lengths. Due to this flexibility, the deblurring method discussed in this chapter is not effective for unconstrained DL configuration, which calls for a different approach; this we discuss in the next chapter.

CHAPTER 5

Deblurring for Unconstrained Dual-lens Cameras

5.1 Introduction and Related Works

[1] Modern cameras come with dual-lens (DL) configuration, that can have *different or identical* focal lengths or field-of-views (FOVs), exposure times, and image resolutions (which we refer to as unconstrained set-up). For instance, the world of smartphones is today experiencing a proliferation of unconstrained DL cameras, wherein almost all devices consider a narrow-FOV camera paired to a conventional wide-FOV camera (for portrait photography), with possibly different resolutions. Also, many of their applications warrant seamless transitions between exposure times, e.g., HDR imaging (Park *et al.*, 2017; Bätz *et al.*, 2014; Sun *et al.*, 2010), low-light photography (Wang *et al.*, 2019*a*), and stereoscopics (Pashchenko *et al.*, 2017) require differently-exposed stereo images in accordance with scene brightnesses, whereas super-resolution (Jeon *et al.*, 2018) and visual odometry (Mo and Sattar, 2018; Iyer *et al.*, 2018) require stereo images with nearly-identical exposure times. All these important applications are marred by motion blur (akin to normal cameras (Hu *et al.*, 2016; Zhang *et al.*, 2010; Lu *et al.*, 2009; Petschnigg *et al.*, 2004)). However, there exists *not* a single BMD method that addresses the current trend of unconstrained DL set-up.

The problem of BMD for DL cameras possess additional challenges over those present in normal cameras. First, a DL set-up warrants deblurring based on scene depth (Xu and Jia, 2012), whereas methods for normal cameras are typically independent of depth (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012; Gupta *et al.*, 2010), as recovering scene depth from a single blurred image is a difficult problem (Hu *et al.*, 2016; Gupta *et al.*, 2010). Second, any method for DL-BMD must ensure scene-consistent disparities in the deblurred image-pair (akin to angular coherence in light fields (Mohan and Rajagopalan, 2018; Srinivasan *et al.*, 2017)), which also opens up many poten-

[1] Based on: Unconstrained motion deblurring for dual-lens cameras. *Mahesh Mohan M. R., Sharath Girsih, and Rajagopalan A. N.*; ICCV 2019, IEEE Publications, Pages 7870–7879.

tial applications (Jeon *et al.*, 2018; Park *et al.*, 2017; Shen *et al.*, 2017; Mo and Sattar, 2018). This is an additional conformity condition in DL-BMD.

In addition, the narrow-FOV genre popularized by the current smartphones admits further issues. The higher focal length of narrow-FOV camera amplifies the effect of camera shake (Whyte *et al.*, 2012), thereby renders motion blur *more* severe. Moreover, the assumption of center-of-rotation (COR) of the camera at the optical center significantly affects ego-motion estimation, and hence the deblurring quality (Hu *et al.*, 2016; Hee Park and Levoy, 2014). In practice, COR may be located at a point far away, such as in the photographer's wrist in case of handheld shake (Sindelar and Sroubek, 2013; Joshi *et al.*, 2010). The higher focal length exacerbates the issues of COR too. It must be noted that, *none* of the existing BMD methods are designed to handle the COR issue.

The works (Lee *et al.*, 2018; Chandramouli *et al.*, 2018; Mohan and Rajagopalan, 2018; Srinivasan *et al.*, 2017; Xu and Jia, 2012) show that BMD methods developed for normal cameras are *seldom* successful for computational cameras. This has necessitated new methods adhering to the modified camera-principles and ensure the coherencies in the computational data (Mohan and Rajagopalan, 2018; Srinivasan *et al.*, 2017; Xu and Jia, 2012). For the case of DL cameras, Xu and Jia (2012) restrict to a constrained set-up, i.e., require two identical cameras to work in synchronization, so that the *same* blur applies to both images. It works by partitioning the blurred image-pair into regions and estimates their PSFs. As small-size regions lack necessary structural information to guide PSF estimation, it proposes region trees to hierarchically estimate them. Importantly, the method imposes strong assumptions on blur that it is *primarily* caused by inplane translations (which does *not* hold good in practice (Whyte *et al.*, 2012)) and that the scene is fronto-parallel with layered depth. Recently, DL video deblurring methods have been proposed (Pan *et al.*, 2017; Sellent *et al.*, 2016), but they address dynamic objects and necessitate as input *multiple* stereo image-pairs. Further, light field deblurring is also not applicable here as those methods constrain all multi-view images to share identical camera settings and ego-motions (Lee *et al.*, 2018; Chandramouli *et al.*, 2018; Mohan and Rajagopalan, 2018; Srinivasan *et al.*, 2017).

Among other closely related works, Hu *et al.* (2014) estimate clean image and layered depth from a *single* blurred image. This work introduces a layer-based approach using matting to partition individual depth layers and an expectation-maximization

scheme to solve this problem. However, (Hu *et al.*, 2014) requires the blur to be primarily due to inplane translations. To reduce the ill-posedness, Pan *et al.* (2019) assume that *accurate* depth is known a priori, but it is difficult to achieve in blur scenarios (Lee *et al.*, 2018; Hu *et al.*, 2016). Further, the method imposes strong assumption of *uniform* ego-motion parameterized by a *single* camera-pose that has *negligible* rotation, which is very unlikely in practice (Köhler *et al.*, 2012; Su and Heidrich, 2015; Whyte *et al.*, 2012). Arun *et al.* (2015) propose a method for multi-shot BMD, but employ four images and restrict to layered depth scenes. It works by estimating PSFs over different spatial locations to arrive at MDF, and employs this to recover latent image and depth by alternate minimization. However, (Arun *et al.*, 2015) requires all the images to be registered within a few pixels (which is typical in ego-motion induced disparities (Sroubek and Milanfar, 2012), but does *not* hold good for baseline induced disparities (Brox *et al.*, 2004)). This constraint is imposed so as to estimate the camera motion with respect to a common pose space.

In this chapter, we address the hitherto unaddressed problem of BMD for *unconstrained* DL set-ups. First, we propose a *DL-blur model* that accounts for arbitrary camera settings and COR. Second, we reveal an *inherent ill-posedness* present in DL-BMD, under the unconstrained exposure scenarios ((Wang *et al.*, 2019*a*; Park *et al.*, 2017; Pashchenko *et al.*, 2017; Sun *et al.*, 2010; Wilburn *et al.*, 2005; Zhang and Chen, 2004)), that disrupts scene-consistent disparities. To this end, we devise a *new prior* that respects consistency of disparities (and also aids ego-motion estimation). Priors that render the resultant cost highly nonconvex or warrant a costly optimization are *not* desirable (Srinivasan *et al.*, 2017; Pan *et al.*, 2016; Xu *et al.*, 2013). We show that our prior is convex and retains the *biconvexity* property (required for convergence (Perrone and Favaro, 2014; Xu *et al.*, 2013; Cho and Lee, 2009)) and allows for the *efficient* LASSO framework. Finally, based on the proposed model and prior, we develop a *practical DL-BMD method*. It eliminates the restrictions of (Mohan and Rajagopalan, 2018; Hu *et al.*, 2014; Xu and Jia, 2012) and also address the COR issue. To eliminate the processing difficulties incur in jointly optimizing multiple images or ego-motions, we propose a divide strategy that decompose a high-dimensional BMD problem into subproblems, while enforcing the proposed prior and convexity. Our main contributions are summarized below:

- This is the first attempt to formally address blind motion deblurring for uncon-

strained camera configurations. To this end, we introduce a *generalized* DL blur model, that also allows for arbitrary COR.

- We reveal an inherent *ill-posedness* present in DL-BMD, that disrupts scene-consistent disparities. To address this, we propose a prior that ensures the bi-convexity property and admits efficient optimization.
- Employing the introduced model and prior, we propose a practical DL-BMD method that achieves state-of-the art performance for a current DL set-up. It ensures scene-consistent disparities, and accounts for the COR issue (for the first time in BMD framework).

5.2 Motion Blur Model for Unconstrained DL

In this section, we introduce a DL motion blur model and its corresponding pixel-wise mapping, considering cameras with different FOVs, exposure times, and resolutions.

In a DL camera set-up, at any instant of time, one camera will perceive a shifted world (by the stereo baseline) with respect to that of a reference camera. Following (Mohan and Rajagopalan, 2018; Pan *et al.*, 2016; Su and Heidrich, 2015; Xu *et al.*, 2013; Whyte *et al.*, 2012), we consider a blurred image as the integration of rotation-induced projections of world over the exposure time, the rotations being caused by camera shake, but do *not* constrain the COR to be *only* at the optical center. Thus, a rotational pose-change translates a world coordinate $\mathbf{X}$ to

$$\mathbf{X}' = \mathbf{R}(\mathbf{X} - \mathbf{l_c}) + \mathbf{l_c} + \mathbf{l_b}, \tag{5.1}$$

where $\mathbf{R}$ is the corresponding rotational matrix (Whyte *et al.*, 2012), $\mathbf{l_b}$ is the baseline vector ($\mathbf{l_b} = \mathbf{0}$ for the reference camera) and $\mathbf{l_c}$ is the unconstrained COR vector (defined in the world coordinate system). We indicate the parameters of the relatively narrow-angle camera by superscript n and the other by superscript w. Thus a DL motion blurred image-pair ($\mathbf{B}^w$ and $\mathbf{B}^n$) (with the COR factored in) can be represented as

$$\begin{aligned} \mathbf{B}^w &= \frac{1}{t_e^w} \int_{t \in t_e^w} P^w\big(\mathbf{R}_t(\mathbf{X} - \mathbf{l_c}) + \mathbf{l_c}\big)\, dt, \\ \mathbf{B}^n &= \frac{1}{t_e^n} \int_{t \in t_e^n} P^n\big(\mathbf{R}_t(\mathbf{X} - \mathbf{l_c}) + \mathbf{l_c} + \mathbf{l_b}\big)\, dt, \end{aligned} \tag{5.2}$$

where the wide-angle camera is considered as reference (without loss of generality). In

practice, the COR ($\mathbf{l_c}$) remains fixed over the exposure time (t_e) (Hu *et al.*, 2016).

For sake of simplicity, with a slight abuse of notation, we use $P^n(\cdot)$ and $P^w(\cdot)$ to denote DL images formed by projecting the world onto the narrow- and wide-angle camera sensors, respectively, that is, by the argument of $P\big(\mathbf{R}_t(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}+\mathbf{l_b}\big)$ we mean a transformation mapping $T_{(\mathbf{R}_t,\mathbf{l_c},\mathbf{l_b})} : \mathbf{X} \rightarrow \mathbf{R}_t(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}+\mathbf{l_b},\ \forall \mathbf{X}$ in world-space. As discussed in Chapter 2, in a conventional camera, a given world coordinate $\mathbf{X_0}$ is mapped to a (homogeneous) sensor coordinate $\mathbf{x_0}$ accordance with $\mathbf{x_0} = \mathbf{K}\mathbf{X_0}/Z_0$, where Z_0 is the scene depth and $\mathbf{K}$ is the intrinsic camera matrix ($\mathbf{K} = \text{diag}(f, f, 1)$, and f is the focal length in pixels). Note that different image resolutions are captured by the scale factors that are used to convert parameters from metres to pixels (Whyte *et al.*, 2012). Resultantly, for a world coordinate $\mathbf{X_0}$, it is evident from Eq. (5.2) that the pixel-displacement due to camera motion (or $\mathbf{R}_t\mathbf{X_0}$) and COR (or $\mathbf{l_c} - \mathbf{R}_t\mathbf{l_c}$) gets relatively amplified in narrow-angle camera by a factor of f^n/f^w. (Typical values of f^n/f^w are around two in portrait-enabled smartphones, and hence exacerbates the issues of motion blur and COR).

To linearize the dual-lens motion blur model, we *equivalently* represent Eq. (5.2) as the integration of image-projections over pose-space (instead of over time) as

$$\mathbf{B}^n = \int_{\mathbf{p}\in\mathbb{P}^3} w^n(\mathbf{p}) \cdot P^n\big(\mathbf{R}_\mathbf{p}(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}+\mathbf{l_b}\big)\, d\mathbf{p}, \tag{5.3}$$

where $\mathbb{P}^3$ is the 3D space of plausible rotational camera poses. The $w^n(\mathbf{p_0})$ gives the fraction of exposure time over which the camera stayed in pose $\mathbf{p_0}$, which defined over the entire $\mathbb{P}^3$ is referred to as motion density function (MDF). The MDF formulation can accommodate both regular and irregular camera motion (unlike (Lee *et al.*, 2018; Srinivasan *et al.*, 2017; Su and Heidrich, 2015)). The consideration of full 3D rotations accommodates *both* narrow- and wide-FOV cameras (Su and Heidrich, 2015).

We now proceed to derive the pixel-mapping in DL set-ups. This is the counterpart of homography-mapping in normal cameras (Chapter 2) or light field cameras (Chapter 4), which is extensively used to create warp matrix for ego-motion estimation and blur-matrix for latent image estimation. Using Eq. (5.1), the transformation of a world coordinate $\mathbf{X}$ for a stationary camera (i.e., $\mathbf{R} = \mathbf{I}$) can be written as

$$\mathbf{X}'' = \mathbf{X} + \mathbf{l_b}, \tag{5.4}$$

where $\mathbf{X}(3)$ $(= Z)$ is the depth of the scene-point $\mathbf{X}$ and $\mathbf{l_b}$ is the baseline vector. The world coordinate $\mathbf{X}''$ maps to the corresponding sensor-coordinate $\mathbf{x}$ (in accordance with Eq. (5.2)) as

$$\mathbf{x} = \mathbf{K}^n \frac{\mathbf{X}''}{Z''} = \mathbf{K}^n \frac{\mathbf{X}''}{Z} = \mathbf{K}^n \frac{(\mathbf{X} + \mathbf{l_b})}{Z} \quad \because \mathbf{l_b}(3) = 0. \tag{5.5}$$

Next, we consider the case of a camera pose-change $\mathbf{R}$ about the COR $\mathbf{l_c}$. That is, the world coordinate $\mathbf{X}$ is transformed as

$$\mathbf{X}' = \mathbf{R}(\mathbf{X} - \mathbf{l_c}) + \mathbf{l_c} + \mathbf{l_b} = \mathbf{R}(\mathbf{X} + \mathbf{l_b}) + (\mathbf{I} - \mathbf{R})\mathbf{l_c} + (\mathbf{I} - \mathbf{R})\mathbf{l_b}. \tag{5.6}$$

Substituting Eq. (5.5) in Eq. (5.6) yields

$$\mathbf{X}' = Z\mathbf{R}(\mathbf{K}^n)^{-1}\mathbf{x} + (\mathbf{I} - \mathbf{R})\mathbf{l_c} + (\mathbf{I} - \mathbf{R})\mathbf{l_b}. \tag{5.7}$$

Using Eq. (5.2), the world coordinate $\mathbf{X}'$ maps to the corresponding sensor-coordinate $\mathbf{x}'$ as

$$\begin{aligned} \mathbf{x}' = \mathbf{K}^n \frac{\mathbf{X}'}{Z'} &= \frac{Z}{Z'}\mathbf{K}^n\mathbf{R}(\mathbf{K}^n)^{-1}\mathbf{x} + \frac{1}{Z'}\mathbf{K}^n(\mathbf{I} - \mathbf{R})\mathbf{l_c} + \frac{1}{Z'}\mathbf{K}^n(\mathbf{I} - \mathbf{R})\mathbf{l_b}, \\ &= \frac{Z}{Z'}\Big(\mathbf{K}^n\mathbf{R}(\mathbf{K}^n)^{-1}\mathbf{x} + \frac{1}{Z}\mathbf{K}^n(\mathbf{I} - \mathbf{R})\mathbf{l_c} + \frac{1}{Z}\mathbf{K}^n(\mathbf{I} - \mathbf{R})\mathbf{l_b}\Big). \end{aligned} \tag{5.8}$$

As the sensor coordinate $\mathbf{x}'$ is in homogeneous system, $\mathbf{x}'(3)$ should be unity. Therefore, the scale Z/Z' in Eq. (5.8) can be considered as a normalization constant (say λ) that normalizes the third coordinate of $\mathbf{x}'$ to 1, which leads to the pixel-mapping of a (homogeneous) coordinate $\mathbf{x}$ as

$$\mathbf{x}' = \lambda\Big(\mathbf{K}^n\mathbf{R}(\mathbf{K}^n)^{-1}\mathbf{x} + \underbrace{\frac{1}{Z}\mathbf{K}^n(\mathbf{I} - \mathbf{R})\mathbf{l_c}}_{\text{center-of-rotation}} + \underbrace{\frac{1}{Z}\mathbf{K}^n(\mathbf{I} - \mathbf{R})\mathbf{l_b}}_{\text{baseline}}\Big). \tag{5.9}$$

Point spread function (PSF) at a spatial coordinate $\mathbf{x}$ is obtained by superimposing the pixel-mappings of $\mathbf{x}$ for all pose-changes undergone during the exposure time. Note that PSFs over spatial coordinates *completely* characterize motion blur (i.e., motion blurred image is obtained by the space-variant convolution of PSFs and latent image) (Whyte *et al.*, 2012; Su and Heidrich, 2015). An important insight from Eqs. (5.2)-(5.9) is that *PSF (and hence motion blur) in a DL set-up is depth-variant due to the baseline and*

COR, with its sensitivity increasing from farther to nearer scene-features (in addition to spatial variance). Wide-angle image can be represented akin to Eqs. (5.3) and (5.9) by enforcing $\mathbf{l_b} = \mathbf{0}$, and with a *different* MDF w^w and projection P^w.

5.3 A New Prior for Unconstrained DL-BMD

In this section, we first attempt to directly formulate a cost using Eqs. (5.3)-(5.9) for DL-BMD. Then we show that this approach is *untenable* for unconstrained DL set-ups, and warrants an additional prior.

The joint cost for DL-BMD is $L = L^n + L^w$:

$$\begin{aligned} L^k = \|\mathbf{A}^k\mathbf{w}^k - \mathbf{B}^k\|_2^2 \ + \lambda_1^k\|\mathbf{w}^k\|_1 + \lambda_2^k\|\nabla\mathbf{L}^k\|_1, \\ \text{where } \|\mathbf{A}^k\mathbf{w}^k - \mathbf{B}^k\|_2^2 \ = \|\mathbf{M}^k\mathbf{L}^k - \mathbf{B}^k\|_2^2. \end{aligned} \tag{5.10}$$

where $k \in \{n, w\}$, $\mathbf{L}^k$ is the clean image, and $\mathbf{w}^k$ is the vectorized form of $w^k(\mathbf{p})$ (where $\mathbf{p}$ is an element of the pose-space $\mathbb{P}^3$). The cost is derived as follows: For MDF $\mathbf{w}^k$, Eq. (5.3) enforces a linear relation via warp matrix $\mathbf{A}^k$, wherein its ith column contains the warped version of clean image $\mathbf{L}^k$, with the pose of $w^k(i)$ (Whyte *et al.*, 2012; Xu *et al.*, 2013), in accordance with Eq. (5.9). For clean image $\mathbf{L}^k$, Eq. (5.9) enforces a linear relation (i.e., space-variant convolution) via PSF matrix $\mathbf{M}^k$, wherein its ith column contains the PSF corresponding to the ith coordinate. The term $\|\mathbf{w}^k\|_1$ enforces a prior on MDF that a 1D camera-path over time represents a sparse population in the 3D pose-space, and $\|\nabla\mathbf{L}^k\|_1$ enforces the total-variation image prior (Perrone and Favaro, 2014; Whyte *et al.*, 2012; Chan and Wong, 1998). *Note that* $\mathbf{A}^k$ *and* $\mathbf{M}^k$ *are depth-dependent and are unique to DL set-up*, via baseline and COR in Eq. (5.9).

As discussed before, the estimated deblurred image-pair $\{\mathbf{L}^n, \mathbf{L}^w\}$ must be related through scene-consistent disparities, i.e., the narrow-angle camera must perceive the *same* scene-orientation, displaced by the baseline $\mathbf{l_b}$, as that by the wide-angle camera (e.g., $\mathbf{L}^n = P^n(\mathbf{X} + \mathbf{l_b})$, if $\mathbf{L}^w = P^w(\mathbf{X})$). However, directly considering the DL-BMD cost for estimating $\{\mathbf{L}^n, \mathbf{L}^w\}$ is *untenable*, as stated below:

Claim 1: **T**here exist *multiple* valid solutions of deblurred image-pairs (or ill-posedness) for the DL-BMD cost (L in Eq. (5.10)) but that produce *scene-inconsistent disparities*.

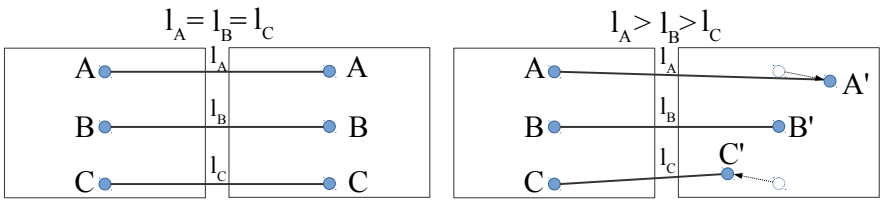
Proof: A desired solution which minimizes Eq. (5.10) is the one involved in the blurring process (Eq. (5.3)), which we refer to as the true image-pair $\{P^n(\mathbf{X}+\mathbf{l_b}), P^w(\mathbf{X})\}$ and true MDFs $\{w^n(p), w^w(p)\}$. Though not characterizing the blur process per se, Eq. (5.3) can be equivalently written as

$$\begin{aligned}\mathbf{B}^n &= \sum_{\mathbf{p}} w^n(\mathbf{p}) P^n\big(\mathbf{R_p}\mathbf{R}_n^{-1}\mathbf{R}_n(\underbrace{\mathbf{X}}_{true} - \mathbf{l_c}) + \mathbf{l_c} + \mathbf{l_b}\big), \\ &= \sum_{\mathbf{p}} w^n(\mathbf{p}) P^n\big(\mathbf{R_p}\mathbf{R}_n^{-1}(\underbrace{\mathbf{R}_n(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}}_{apparent} - \mathbf{l_c}) + \mathbf{l_c} + \mathbf{l_b}\big),\end{aligned} \tag{5.11}$$

where the new scene-orientation of narrow-angle lens is $\mathbf{R}_n(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}$, where $\mathbf{R}_n \neq I$. The quantity $\mathbf{R}_n$ has the effect of shifting all the true poses undergone by the camera ($\mathbf{R_p}, \mathbf{p} \in \mathbb{P}^3$) by an offset of $\mathbf{R}_n^{-1}$, which in turn produces an MDF that is a shifted version of the *true* MDF (and hence the MDF-sparsity cost remains the same). Consequently, a new solution according to Eq. (5.11) is the image-pair $\{P^n(\mathbf{R}_n(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}+\mathbf{l_b}), P^w(\mathbf{X})\}$, which clearly fails the criterion for scene-consistent disparities (i.e., the narrow-angle camera perceives a *different* scene-orientation). Also, as the new narrow-angle image is a warped version of the true narrow-angle image, it adheres to the TV prior, and therefore the new solution minimizes L^n. The cost L^w remains the same (as the wide-angle image or MDF incurs *no* change). Resultantly, the same solution minimizes L, which concludes the proof. ■

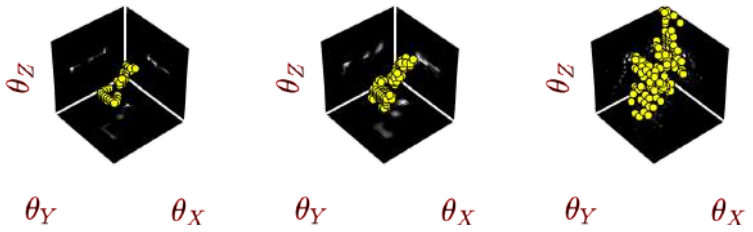
A similar ambiguity also arises for the wide-angle case. This is obtained from Eq. (5.11) by enforcing $\mathbf{l_b} = \mathbf{0}$ and replacing P^n by P^w. As the costs L^n and L^w (in Eq. (5.10)) are *independent*, the pose $\mathbf{R}_n$ need *not* be equal to that of wide-angle ($\mathbf{R}_w$). For unequal $\mathbf{R}_n$ and $\mathbf{R}_w$, the resultant image-pair becomes $\{P^n(\mathbf{R}_n(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c}+\mathbf{l_b}), P^w(\mathbf{R}_w(\mathbf{X}-\mathbf{l_c})+\mathbf{l_c})\}$. Following the similar steps in the proof, we can show that the resultant solution minimizes L, though the image-pairs produce *scene-inconsistent* disparities.

We attempt to provide some insights on the effect of ill-posedness. Consider the case of a positive inplane rotation ambiguity, with COR at the optical center. Figure 5.1(a) shows three image coordinates $\{A, B, C\}$ with *identical* scene-depths (i.e., the same disparities). Fig. 5.1(b) considers the rotational ambiguity, i.e. the coordinates $\{A, B, C\}$ are mapped to $\{A', B', C'\}$, respectively. It is evident from Fig. 5.1(b) that, relative to the scene-feature of B, A's scene-feature appears to be farther and C's scene-feature appears to be nearer, even though all the scene features have identical depths in

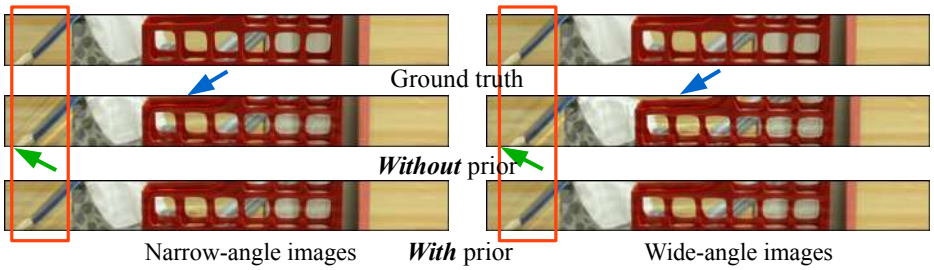


(a) Desired solution (b) An ill-posed solution

Figure 5.1: $\{A, B, C\}$ in Fig. (a) correspond to scene-features at the same depth (i.e., *identical* disparities). Fig. (b) considers an inplane rotational ambiguity, wherein $\{A, B, C\}$ translates to $\{A', B', C'\}$ which clearly leads to *inconsistent* disparities.



(a) GT MDF (b) W/ Prior (c) W/o Prior



(d) Image patches

Figure 5.2: Effect of the proposed prior: (a-d) MDFs and deblurred image patches with (W/) and without (W/o) prior (with all MDFs centroid-aligned with the ground truth (GT) $\mathbf{w}^n$ to align left-images). MDF estimate of the prior-less case has a random offset (Fig. (c)) and the corresponding deblurred image clearly reveals *scene-inconsistent* disparities (Fig. (d)). Also, the deblurred image in the prior-less case exhibits considerable ringing artifacts and residual blur (Fig. (d)). In contrast, the addition of our proposed DL prior successfully curbs the pose ambiguity and improves the MDF accuracy (Fig. (b)) and produces better deblurring quality (Fig. (d)).

the world system.

Note that the ill-posedness *exists* irrespective of the exposure time being identical or different. Moreover, the inconsistent deblurred image-pair shares all the issues associated with the classical problem of stereo rectification (Loop and Zhang, 1999; Xiao *et al.*, 2018) that deals with *physical* misalignment of cameras. These methods work by estimating a pair of homography for rectification (Xiao *et al.*, 2018; Fusiello *et al.*, 2000). However, the ambiguity in DL-BMD is different, in that it necessitates *depth-variant* transformation due to baseline and arbitrary COR (Eq. (5.9)).

We tackle the ill-posedness *within* our deblurring method, by employing a judiciously derived prior. For this, we assume that there exists an overlap between exposure times of different cameras. A DL set-up that violates this assumption has to incur significant ghosting artifacts, and is hence *not* preferred (Park *et al.*, 2017). Note that our assumption is generic as compared to that of the *complete* exposure-time overlap in the only-existing DL-BMD method (Xu and Jia, 2012).

Our prior is motivated by the previous discussion, in that the deblurred image-pair will be consistent if $\mathbf{R}_n = \mathbf{R}_w$. For identical exposure time, this criterion requires that both the MDFs completely intersect over the pose-space. For overlapping exposure time, both MDFs must intersect over the shared poses. Hence, we introduce a DL prior of the form $\|\mathbf{w}^n - \mathbf{w}^w\|_2$. Intuitively, the prior functions as follows: The DL-BMD cost can admit MDF-pairs with significant relative drifts, which severely disrupt scene-consistent disparities (e.g., see Figs. 5.2(c,d)). However, these solutions are not favoured with the inclusion of the prior because it enforces the resultant cost to increase with relative drifts (e.g., see Figs. 5.2(b,d)).

The proposed DL prior has several desirable properties: As shown in (Perrone and Favaro, 2014; Xu *et al.*, 2013; Cho and Lee, 2009; Gupta *et al.*, 2010), the biconvexity property (i.e., the BMD cost is *convex* with respect to MDF for a given clean image, and vice-versa) guarantees convergence via alternating minimization. Our final cost has this property.

Claim 2: **T**he DL-BMD cost L (Eq. (5.10)) is biconvex with respect to image-pair $\{\mathbf{L}^n, \mathbf{L}^w\}$ and MDF-pair $\{\mathbf{w}^n, \mathbf{w}^w\}$. The DL prior is convex, and when added to the cost L retains the biconvexity property.

(Note: For the proofs, we employ two well-known properties of convex functions: (1) Composite of a convex function with a non-decreasing function is a convex function. (2) Non-negative addition of convex (or biconvex) functions is a convex (or biconvex) function (Boyd and Vandenberghe, 2004).)

Lemma 1: **T**he costs L^n (and L^w) are individually biconvex in image $\mathbf{L}^n$ and MDF $\mathbf{w}^n$ (and $\mathbf{L}^w$ and $\mathbf{w}^w$), respectively.

Proof: (Note: A function is biconvex in $\mathbf{L}$ and $\mathbf{w}$ if it is convex in $\mathbf{L}$ for a given $\mathbf{w}$, and vice-versa.) For a given $\mathbf{L}^n$, the cost L^n (in Eq. (5.10)) is given as

$$L^n = \|\mathbf{A}^n\mathbf{w}^n - \mathbf{B}^n\|_2^2 + \lambda_1^n\|\mathbf{w}^n\|_1 + \text{Constant}, \tag{5.12}$$

where the 'Constant' is $\lambda_2^n\|\nabla\mathbf{L}^n\|_1$. The first and second terms are composite of two convex functions (i.e., a linear transformation of $\mathbf{w}^n$) with non-decreasing function (i.e., squared-l_2 or l_1 norm), and hence convex (Property 1). Further, the third term is convex as a constant is a convex function. Therefore, the cost L^n is convex with respect to $\mathbf{w}^n$ (by Property 2 mentioned above).

For a given $\mathbf{w}^n$, the cost L^n (in Eq. (5.10)) can be equivalently represented as

$$L^n = \|\mathbf{M}^n\mathbf{L}^n - \mathbf{B}^n\|_2^2 + \lambda_2^n\|\nabla\mathbf{L}^n\|_1 + \text{Constant}, \tag{5.13}$$

where the 'Constant' is $\lambda_1^n\|\mathbf{w}^n\|_1$. Again, the first and second terms are composite of two convex functions with non-decreasing function (i.e., a linear transformation of $\mathbf{L}^n$ with squared-l_2 or l_1 norm), and the third term is convex. Therefore, the cost L^n is convex with respect to $\mathbf{L}^n$ (by Property 2). Hence L^n is biconvex in $\mathbf{L}^n$ and $\mathbf{w}^n$. Similarly, we can show that L^w is biconvex in $\mathbf{L}^w$ and $\mathbf{w}^w$. ■

Now we proceed to prove the part 1 of Claim that the DL-BMD cost $L = L^n + L^w$ (Eq. (5.10)) is a *biconvex* function in image-pair $\{\mathbf{L}^n, \mathbf{L}^w\}$ and MDF-pair $\{\mathbf{w}^n, \mathbf{w}^w\}$.

Proof: (We denote the function F for a given $\mathbf{w}$ as $F_{\mathbf{w}}$.) From Lemma 1, the biconvexity

of L^n implies

$$\begin{aligned} L^n_{\mathbf{w}^n}\big(\gamma {\mathbf{L_1}}^n + (1-\gamma){\mathbf{L_2}}^n\big) \leq \gamma L^n_{\mathbf{w}^n}({\mathbf{L_1}}^n) + (1-\gamma) L^n_{\mathbf{w}^n}({\mathbf{L_2}}^n), \\ \forall\{{\mathbf{L_1}}^n, {\mathbf{L_2}}^n, \mathbf{w}^n\},\ \forall\gamma \in [0,1]; \\ L^n_{\mathbf{L}^n}\big(\gamma {\mathbf{w_1}}^n + (1-\gamma){\mathbf{w_2}}^n\big) \leq \gamma L^n_{\mathbf{L}^n}({\mathbf{w_1}}^n) + (1-\gamma) L^n_{\mathbf{L}^n}({\mathbf{w_2}}^n), \\ \forall\{{\mathbf{w_1}}^n, {\mathbf{w_2}}^n, \mathbf{L}^n\},\ \forall\gamma \in [0,1]. \end{aligned} \tag{5.14}$$

As the cost L^n is independent of wide-angle parameters $\{\mathbf{L}^w, \mathbf{w}^w\}$, Eq. (5.14) can be *equivalently* written as

$$\begin{aligned} L^n_{\{\mathbf{w}^n,\mathbf{w}^w\}}\big(\gamma\{{\mathbf{L_1}}^n, {\mathbf{L_1}}^w\} + (1-\gamma)\{{\mathbf{L_2}}^n, {\mathbf{L_2}}^w\}\big) \leq \gamma L^n_{\{\mathbf{w}^n,\mathbf{w}^w\}}(\{{\mathbf{L_1}}^n, {\mathbf{L_1}}^w\}) \\ +(1-\gamma) L^n_{\{\mathbf{w}^n,\mathbf{w}^w\}}(\{{\mathbf{L_2}}^n, {\mathbf{L_2}}^w\}); \\ L^n_{\{\mathbf{L}^n,\mathbf{L}^w\}}\big(\gamma\{{\mathbf{w_1}}^n, {\mathbf{w_1}}^w\} + (1-\gamma)\{{\mathbf{w_2}}^n, {\mathbf{w_2}}^w\}\big) \leq \gamma L^n_{\{\mathbf{L}^n,\mathbf{L}^w\}}(\{{\mathbf{w_1}}^n, {\mathbf{w_1}}^w\}) \\ +(1-\gamma) L^n_{\{\mathbf{L}^n,\mathbf{L}^w\}}(\{{\mathbf{w_2}}^n, {\mathbf{w_2}}^w\}). \end{aligned} \tag{5.15}$$

Eq. (5.15) implies that L^n is biconvex in $\{\mathbf{L}^n, \mathbf{L}^w\}$ and $\{\mathbf{w}^n, \mathbf{w}^w\}$. Following similar steps, the same inference can be derived for L^w too. Since the DL-BMD cost L is obtained by the summation of two biconvex function L^n and L^w, it must be biconvex with respect to image-pair $\{\mathbf{L}^n, \mathbf{L}^w\}$ and MDF-pair $\{\mathbf{w}^n, \mathbf{w}^w\}$ (by Property 2). Hence proved. ■

Now we proceed to prove part 2 of the claim that introducing the DL prior in the DL-BMD objective L (Eq. (5.10)) retains the biconvexity property.

Proof: The prior $L^p = \alpha\|\mathbf{w}^n - \mathbf{w}^w\|_2 \ : \alpha > 0$ can be equivalently represented as

$$L^p = \alpha\|\mathbf{S}\mathbf{w}\|_2, \ \text{ where } \mathbf{S} = \text{diag}(I, -I), \text{ and } \mathbf{w} \text{ is } \mathbf{w}^n \text{ and } \mathbf{w}^w \text{concatenated.} \tag{5.16}$$

First, $L^p_{\{\mathbf{w}^n,\mathbf{w}^w\}}(\{\mathbf{L}^n, \mathbf{L}^w\})$ is a constant, and therefore it is convex with respect to dual image-pair $\{\mathbf{L}^n, \mathbf{L}^w\}$. Second, $L^p_{\{\mathbf{L}^n,\mathbf{L}^w\}}(\{\mathbf{w}^n, \mathbf{w}^w\})$ is a composite of a convex function with a non-decreasing function, (i.e., linear transformation of $\{\mathbf{w}^n, \mathbf{w}^w\}$ with l_2 norm). Hence, the function is convex in $\{\mathbf{w}^n, \mathbf{w}^w\}$ (by Property 1). Therefore, the prior L^p is biconvex in $\{\mathbf{L}^n, \mathbf{L}^w\}$ and MDF-pair $\{\mathbf{w}^n, \mathbf{w}^w\}$. Resultantly, the non-negative addition of L^p to a biconvex function L (Claim 1) retains the biconvexity property (by Property 2). ■

Also, our prior serves to impart reinforcement between the dual images (through MDFs), which Eq. (5.10) does *not* possess (as L_n and L_w are independent). It aids in ego-motion estimation, which in turn leads to improved deblurring (e.g., see Fig. 5.2(d)). Further, the prior allows for efficient LASSO optimization (as we shall see in Section 5.4.2).

5.4 A Practical algorithm for DL-BMD

In this section, we propose a practical DL-BMD algorithm for unconstrained camera settings and arbitrary COR (a first of its kind), based on the proposed model and DL prior (Secs. 5.2–5.3). We show that a multi-camera BMD problem can be divided into subproblems (with the same dimension as that of normal camera BMD) while enforcing the DL prior and convexity property.

Our method proceeds in a scale-space manner to handle large blurs (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012; Cho and Lee, 2009). We employ alternating minimization for depth, COR, MDF and latent image, in that order. The convergence of alternating minimization is supported by Sec. 5.3, in that resolving the ill-posedness enforces scene-consistent image-pair, which in turn produces consistent depth and COR (Hu *et al.*, 2016). As 'depth from stereo' is a well-studied problem, we selected an off-the-shelf algorithm for depth estimation (Liu *et al.*, 2009) (owing to its good trade-off between accuracy and speed (Li *et al.*, 2018; Shih and Chen, 2018)).

5.4.1 Center-of-Rotation Estimation

To estimate COR, we consider a cost which is the least squares error between blurred images and synthesized blurry images using the blur model (via Eqs. (5.3)-(5.9)) and current estimates of other unknowns. We frame the cost in the gradient domain of the images to improve the condition number (Cho and Lee, 2009). In order to ensure that all regions of the image constrain COR, the image is split into multiple bins and thresholding is done separately for each bin. The optimization for COR is given as

$\tilde{l}_c = \arg\min_{\mathbf{l_c}} \left(L^w_{\mathbf{l_c}} + L^n_{\mathbf{l_c}}\right)$:

$$L^k_{\mathbf{l_c}} = \|g(\mathbf{B}^k) - g\left(\sum_p \tilde{w}^k(p) P^k(\tilde{\mathbf{L}}^k, \tilde{\mathbf{Z}}, \mathbf{l_c})\right)\|_2, \tag{5.17}$$

where $k \in \{w, n\}$, $g(\cdot)$ produces the first and second-order gradients, and the symbol '$\sim$' denotes the current estimates. A trust region reflective algorithm (Coleman and Li, 1996) is used for optimizing Eq. (5.17), which is initialized with the previous COR estimate. For the first scale and first iteration, we initialize the latent images as the corresponding shock-filtered blurred images, MDFs as Kronecker delta, and COR at the optical center.

5.4.2 Divide Strategy for MDFs and Images

Jointly estimating multiple MDFs or images is computationally inefficient, as the optimization dimension scales-up linearly with each additional camera input. To this end, we decompose the DL-BMD cost with prior, such that convexity is preserved and the optimization dimension remains at par with that of normal camera, irrespective of the number of cameras. The MDF and image estimation are given by

$$\begin{aligned} \arg\min_{\mathbf{w}^n} \|\tilde{\mathbf{A}}^n \mathbf{w}^n - \mathbf{B}^n\|_2^2 \; + \alpha \|\mathbf{w}^n - \tilde{\mathbf{w}}^w\|_2^2 : \; \|\mathbf{w}^n\|_1 \leq \lambda_1'^n, \\ \arg\min_{\mathbf{L}^n} \|\tilde{\mathbf{M}}^n \mathbf{L}^n - \mathbf{B}^n\|_2^2 \; + \lambda_2^n \|\nabla \mathbf{L}^n\|_1, \end{aligned} \tag{5.18}$$

where we have included the DL prior within the objective, but separated out the MDF-sparsity prior as a constraint.

Claim 3: **T**he individual optimizations in Eq. (5.18) are convex. Further, MDF estimation with the DL prior (in Eq. (5.18)) has an *equivalent* LASSO form $\arg\min_{\mathbf{w}^n} \|\mathbf{C}\mathbf{w}^n - \mathbf{b}\|_2^2 : \|\mathbf{w}^n\|_1 \leq \lambda_1'^n$, such that

$$\mathbf{C} = \tilde{\mathbf{A}}^{n^T} \tilde{\mathbf{A}}^n + \alpha I, \text{ and } \mathbf{b} = \tilde{\mathbf{A}}^{n^T} \mathbf{B}^n + \alpha \tilde{\mathbf{w}}^n. \tag{5.19}$$

Proof: MDF optimization problem is given as (Eq. (5.18)):

$$\tilde{\mathbf{w}^n} = \arg\min_{\mathbf{w}^n} \underbrace{\|\tilde{\mathbf{A}}^n\mathbf{w}^n - {\mathbf{I_B}}^n\|_2^2 + \alpha\|\mathbf{w}^n - \tilde{\mathbf{w}}^w\|_2^2}_{G} \; : \|\mathbf{w}^n\|_1 \leq \lambda_1'^n, \tag{5.20}$$

We first prove the convexity property. The first term of G is convex in $\mathbf{w}^n$ (proved in Lemma S1). As the DL prior (the second term) is convex with respect to *both* the MDFs $\{\mathbf{w}^n, \mathbf{w}^w\}$ combined, it should be convex in $\mathbf{w}^n$ for a given $\mathbf{w}^w$. Also, the feasible set is convex (Boyd and Vandenberghe, 2004). Thus the MDF estimation is a convex optimization problem. The convexity of latent image estimation directly follows from the proof of Claim 2 (Eq. (5.13)).

We now proceed to derive an equivalent LASSO form. As Eq. (5.20) is a convex optimization problem, $\hat{\mathbf{w}}^n$ is an optimal solution *iff*

$$\nabla G(\hat{\mathbf{w}}^n) = \mathbf{0} : \|\hat{\mathbf{w}}^n\|_1 \leq \lambda_3^n, \text{ where } \nabla G(\mathbf{w}^n) = 2\cdot\big((\tilde{\mathbf{A}}^{n^T}\tilde{\mathbf{A}}^n + \alpha I)\mathbf{w}^n - (\tilde{\mathbf{A}}^{n^T}{\mathbf{I_B}}^n + \alpha\tilde{\mathbf{w}}^w)\big) \tag{5.21}$$

Leveraging Eq. (5.21), we frame a new optimization problem as follows: As ∇G is multi-dimensional, we consider the cost as the l_2 norm of ∇G (to convert to a single-valued objective function), i.e.,

$$\tilde{\mathbf{w}}^n = \arg\min_{\mathbf{w}^n} \|\nabla G(\mathbf{w}^n)\|_2^2 \; : \|\mathbf{w}^n\|_1 \leq \lambda_3^n \tag{5.22}$$

The new problem in Eq. (5.22) possesses several desirable properties (in addition to the LASSO structure). (I) It is a convex optimization problem, i.e., any local minima need to be a global minima and objective value of all minima should be the same. (II) $\|\nabla G(\mathbf{w}^n)\|_2^2 \geq 0$ and $\|\nabla G(\mathbf{w}^n)\|_2^2 = 0$ *iff* $\nabla G(\mathbf{w}^n) = \mathbf{0}$ (properties of norm). (III) Optimal solution $\hat{\mathbf{w}}^n$ of the problem in Eq. (5.20) has to satisfy $\|\nabla G(\hat{\mathbf{w}}^n)\|_2^2 = 0$, which is a minima of Eq. (5.22) (by Property (II)). Property (III) implies that all solutions of the optimization problem in Eq. (5.20) will also be solutions of the LASSO famework. Also, since Eq. (5.21) is a necessary and sufficient condition, all solutions of the LASSO framework will be solutions of the problem in Eq. (5.20) (by Properties (I-II)), thereby establishing the equivalence of both the frameworks. ■

A similar formulation as that of Eqs. (5.18)-(5.19) applies to the other camera as well. We optimized for MDFs using the standard LASSO solver (Tibshirani, 1996) (following

(Whyte *et al.*, 2012; Cho and Lee, 2009)). Also, our divide strategy converts the latent image estimation to the classic problem of TV-deblurring (Chan and Wong, 1998) (the only difference is that $\tilde{\mathbf{M}}^n$ is now in accordance with DL-model), which has excellent convergence and efficient solvers (Perrone and Favaro, 2014). As image estimators are independent, they can be parallelized for efficiency. This is made possible by our decomposition of the DL-BMD problem while enforcing the DL prior.

5.5 Analysis and Discussions

In this section, we indicate the generalizability of our work to diverse camera set-ups. Then, we analyse the effect of our prior and COR.

5.5.1 Generalizability of our Method

Our theory and method directly apply to DL cameras with entirely different settings. Second, they hold well for *identical* cameras ($f^n = f^w$) or camera arrays (multiple l_b), wherein exposures are different ($\mathbf{w}^n \neq \mathbf{w}^w$ or $\mathbf{w}^n = \mathbf{w}^w$) or identical ($\mathbf{w}^n = \mathbf{w}^w$). Third, they generalize to the mature normal camera methods ($\mathbf{l_b} = \mathbf{l_c} = \mathbf{0}$ and $\mathbf{w}^n = \mathbf{w}^w$) (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012). Further, our method can *seamlessly* address partial and full exposure-overlaps ((Jeon *et al.*, 2018; Park *et al.*, 2017; Wang *et al.*, 2019*a*; Pashchenko *et al.*, 2017; Mo and Sattar, 2018)), *without* any modifications. Based on the previous discussions, we make the following remarks.

Remark 1: The motion blur model of the methods (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012) admits *only* a depth invariant model, whereas motion blur in a DL set-up warrants a depth variant model.

Proof: The pixel-mapping employed in single-lens model is given as (Whyte *et al.*, 2012)

$$\hat{\mathbf{x}}' = \lambda \mathbf{K} \hat{\mathbf{R}} \mathbf{K}^{-1} \mathbf{x} \tag{5.23}$$

where λ normalizes the third coordinate of $\hat{\mathbf{x}}'$ to 1. Note that the mapping in Eq. (5.23) is *invariant* to scene-depth, unlike the depth-variant mapping of DL system due to baseline and COR (from Eq. (5.9)). Consider an arbitrary image-coordinate $\mathbf{x_0}$ with corresponding scene-depth Z_0. Let $\hat{\mathbf{R}}_{\{\mathbf{x_0}, Z_0\}}$ be the optimal pose in Eq. (5.23) that *equates*

to the homography-mapping of the DL system at $\mathbf{x_0}$ (i.e., $\mathbf{x}' = \hat{\mathbf{x}}'$, for $\mathbf{x} = \mathbf{x_0}$). As Eq. (5.23) is depth-invariant, fixing a pose suitable for $\mathbf{x_0}$ will *concurrently* fix the mapping at all other coordinates *irrespective* of their depth values. This clearly violates the inherent *depth-variant* mapping of DL system at those coordinates. Consequently, the PSF optimized for the coordinate $\mathbf{x_0}$, through multiple poses $\hat{\mathbf{R}}_{t\{\mathbf{x_0},Z_0\}}$ for $t \in t_e$, *concurrently* fixes the PSF at other coordinates *irrespective* of their depth values; thereby failing to model the depth-variant PSFs in a DL setup. ■

Remark 2: The blur model of the methods (Pan *et al.*, 2016; Xu *et al.*, 2013; Whyte *et al.*, 2012) modulate the baseline with camera poses, but it must be independent for a DL set-up (for scene-consistent disparities).

Proof: Motion blur model in single-lens system is given as (Whyte *et al.*, 2012)

$$\hat{\mathbf{I}}_B^n = \frac{1}{t_e^n} \int_{t_e^n} P^n\big(\hat{\mathbf{R}}_t(\mathbf{Y})\big)\, dt, \tag{5.24}$$

where the world-coordinate system $\mathbf{Y}$ is defined with respect to the optical center (i.e., $\mathbf{l_b} = \mathbf{l_c} = 0$). In the DL blur model (Eqs. (5.2)-(5.3)), the effect of stereo baseline is *independent* of camera pose-changes, and the disparity relation between stereo image-pair is due to the baseline ($\mathbf{l_b}$). Enforcing the single-lens model in the narrow-angle image leads to $\hat{\mathbf{I}}_B^n = (1/t_e^n) \int_{t_e^n} P^n\big(\hat{\mathbf{R}}_t(\mathbf{X} + \mathbf{l_b})\big)\, dt$ (where the world coordinate $\mathbf{X}$ is defined with respect to wide-angle camera, as followed in Eqs. (5.1)-(5.2)). Evidently, the effect of baseline in this case *varies* with pose-change $\hat{\mathbf{R}}_t$, *unlike* the DL model. Specifically, it characterizes an alien dual-lens set-up with its own physical lens-separation, and in turn the scene disparities, getting modulated by pose-changes over time. ■

Remark 3: The methods (Pan *et al.*, 2016; Whyte *et al.*, 2012; Xu *et al.*, 2013) also admit the ill-posedness that disrupts scene-consistent disparities.

Proof: This is a special case of Claim 1, wherein $\mathbf{l_b} = \mathbf{l_c} = \mathbf{0}$. ■

Next, we show the generalizability of our algorithm to different types of DL set-up. The image PSNR, VIF, IFC metrics and depth PSNR metric are shown in the Table 5.1 for the three DL-configurations: Narrow-Narrow, Narrow-Wide, Wide-Wide. We consider the same exposure time for both cameras, 52mm focal length for narrow angle camera and 26mm focal length for the wide angle camera. The values reported in the Table 5.1 are averaged over three examples. As can be seen, our method performs consistently better than the methods of (Xu and Jia, 2012; Mohan and Rajagopalan, 2018) in

Set-up	Metrics	Blurred	Xu et al.	Mohan et al.	Ours
N-N	Image PSNR/IFC/VIF	27.27 / 1.75 / 0.23	19.90 / 1.08 / 0.22	29.21 / 2.30 / 0.36	31.03 / 3.04 / 0.43
	Depth PSNR	29.22	15.83	29.50	30.35
N-W	Image PSNR/IFC/VIF	27.33 / 1.78 / 0.23	19.86 / 1.13 / 0.22	26.50 / 1.95 / 0.31	30.50 / 3.10 / 0.42
	Depth PSNR	28.51	15.29	28.56	31.11
W-W	Image PSNR/IFC/VIF	27.87 / 1.97 / 0.27	14.56 / 0.94 / 0.17	25.90 / 2.04 / 0.32	30.64 / 4.40 / 0.56
	Depth PSNR	30.15	13.88	28.56	30.62

Table 5.1: Generalizability to diverse DL set-ups (Symbols 'N' and 'W' represent narrow and wide-FOV, respectively.): Our method consistently outperforms the methods of (Xu and Jia, 2012; Mohan and Rajagopalan, 2018) in the PSNR, IFC and VIF metrics for image and the PSNR metric for depth.

all three configurations. Specifically, in terms of Image/Depth PSNR, our method outperforms (Mohan and Rajagopalan, 2018) by 0.82/0.85 dB for Narrow-Narrow setup, 4.00/2.55 dB for Narrow-Wide setup and 4.74/2.06 dB for Wide-Wide setup.

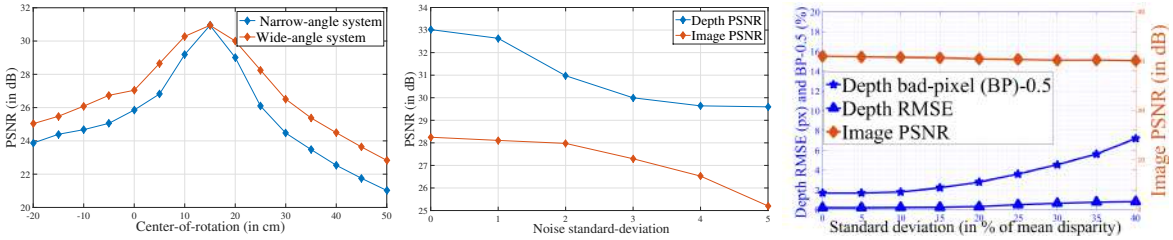
5.5.2 Effectiveness of the DL prior and COR

Table 5.2 summarizes the PSNR results for image/depth (averaged over five examples) by ablating the DL prior and COR estimator. For creating synthetic dataset, exposure overlap and COR are randomly sampled from 10 to 100% and -30 to 30 cm cube, respectively. The unconstrained set-up we employed is narrow- and wide-FOV pair, with $f^n = 52$ mm, $f^w = 26$ mm, and the former having twice the resolution (as in `Samsung S9+`). Observe that for the prior-less case the depth information gets significantly corrupted (i.e., PSNR drops by 7 dB!). This underlines the importance of resolving the pose-ambiguity in dual-lens BMD. Further, the deblurring performance also drops by 2.3 dB in the prior-less case, possibly be due to the loss of reinforcement between the narrow- and wide-angle costs (as discussed earlier). Further, the table reveals that both image and depth accuracies deteriorate when COR issue is *not* addressed, i.e., image and depth PSNRs drop by 1.6 and 1.3 dB, respectively.

To analyze the sensitivity of COR for narrow-angle and wide-angle configurations, we considered images blurred with a common COR , and performed deblurring by perturbing the COR vector and using the true ego-motion (*identically* for both the config-

PSNR (dB)	Blur	W/o Prior W/o COR	W/o Prior W/ COR	W/ Prior W/o COR	W/ prior W/ COR
Image	22.39	25.69	26.59	27.28	28.88
Depth	28.33	23.35	**23.59**	29.12	**30.52**

Table 5.2: Quantitative results of our method with and without the DL prior and COR. In particular, our DL prior reduces the ill-posedness by a good margin (i.e., by 7 dB, as indicated in bold).



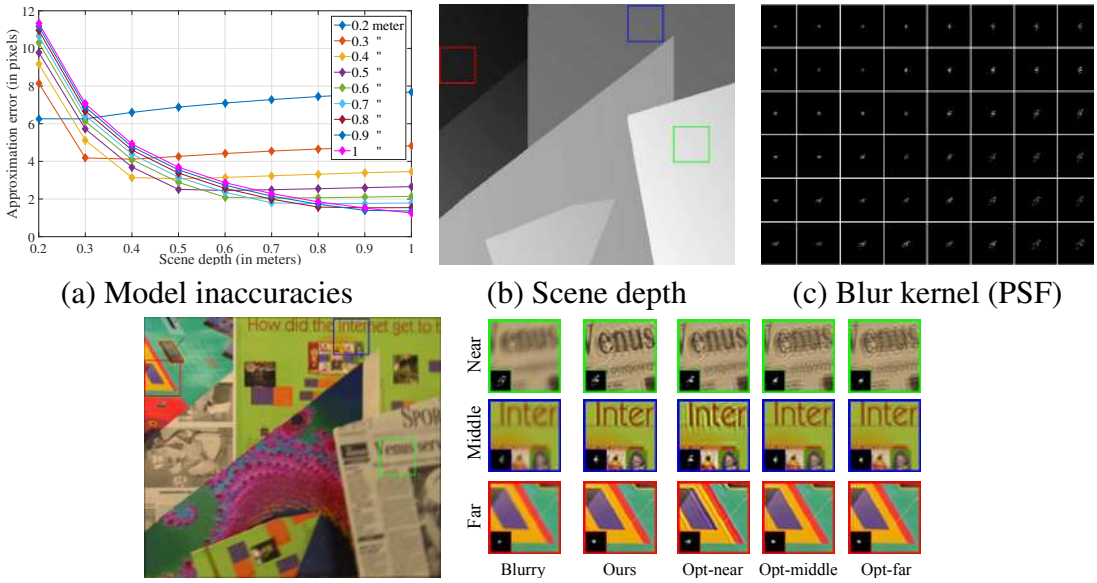
(a) Sensitivity-analysis of COR (b) Image-noise analysis (c) Depth-noise analysis

Figure 5.3: Analysis: (a) Sensitivity of COR: Both narrow-angle and wide-angle configurations are very sensitive to COR, with the former exhibiting relatively more sensitivity. (b-c) Effect of image and depth noise.

urations). Figure 5.3(a) compares the average PSNR of deblurred images for different COR approximations. The figure clearly shows a significant drop in deblurring performance as the approximated COR deviates from the true COR. Also, note the detrimental effect of the common COR approximation about the camera center (that is followed in single-lens BMD methods). The figure also reveals higher sensitivity of COR in narrow-angle configuration as shown by the higher rate of its performance-drop. This is due to higher focal-length, and hence larger blur inherent in narrow-angle setup which is a function of COR (as noted in Sec. 5.2).

5.5.3 Effect of Noise in Image and Depth Estimation

To analyze the effect of noise in our DL-BMD method, we experimented with blurry images corrupted with additive white Gaussian noise. Standard-deviation of noise (in pixels) is varied from 0 (noise-less case) to 5. Fig. 5.3(b) plots the average PSNRs of a deblurred image and depth estimate corresponding to different noise levels. The average PSNRs for deblurred image and depth-estimate is more than 25 dB and 29 dB, respectively, over the entire standard-deviation range; this clearly reveals the noise-robustness of our algorithm. Although we did *not* perform denoising in any examples, for *very* high noisy levels, the blurred image-pair need to be denoised prior to deblurring. This



(a) Model inaccuracies (b) Scene depth (c) Blur kernel (PSF)

(d) Blurred image (e) Image-patches

Figure 5.4: DL configuration warrants a *depth-variant* transformation. (a) Model inaccuracies of the homography model. Note the variation of PSF in Fig. (c) with respect to the scene depth in Fig. (b). As the single-lens motion blur model is *depth-invariant*, the model optimized for a fixed depth can fail for other depths, leading to *ineffective* deblurring across depths (Fig. (e)).

is because noise can deteriorate image-gradients which are required for ego-motion estimation (Sec. 5.4).

The total variation prior in the DL-BMD cost is employed to curb ringing artifacts. We analysed the depth-dependency by adding additive white Gaussian noise in disparity-map (following (Mandal *et al.*, 2016; Liu *et al.*, 2015; Riegler *et al.*, 2016)). The standard deviation (SD) of noise is varied from 0 to 40% of mean disparity, in the disparity-map estimate in *all* iterations (a worst-case scenario). In Fig. 5.3(c), we plot the results averaged over five trials for each SD-unit (for the example in Fig. 5.2), where we utilize the metrics RMSE and bad pixel ratio for depth. Note that over the entire SD-range, image-PSNR and depth-RMSE are reduced by only 0.875 dB and 0.622 pixel, respectively, which clearly reveals our method's robustness.

5.5.4 Uniqueness of the DL pixel-mapping over homography

The uniqueness of DL pixel-mapping is due to the latter's *depth-invariant* nature, which we illustrate with two experiments. For a camera-pose sampled from a real trajectory (Köhler *et al.*, 2012), Fig. 5.4(a) shows the *best* approximation error of the homography-

mapping of normal cameras to the pixel-mapping of Eq. (5.9), optimized for a given depth (under least-squares criteria). Notably, the homography *fails* to model the DL pixel-wise mapping *consistently over different depths*, which clearly illustrates the uniqueness of Eq. (5.9). Further, to analyze the depth-variant nature of PSFs, Figs. 5.4(b-e) consider a camera trajectory and a 3D scene from (Scharstein and Szeliski, 2002). Fig. 5.4(c) shows the corresponding PSFs (projected using Eq. (5.9)), which reveals depth-dependency of blur, with lower depths exhibiting severe blurs relative to the farther ones. Figure 5.4(e) shows the deblurred image-patches for different depths employing the normal camera method (Xu *et al.*, 2013), optimized for a given depth; it is evident that this approach is *not* quite successful due to the *depth-dependency* of the blur, which clearly necessitates a new approach for DL-BMD.

5.6 Experimental Results

In this section, we extensively evaluate our proposed method on both synthetic and real examples.

Comparison Methods: We considered (Pan *et al.*, 2016; Xu *et al.*, 2013) to represent normal camera BMD. For computational cameras, we considered state-of-the-art stereo BMD (Xu and Jia, 2012) and light field BMD (Mohan and Rajagopalan, 2018). For depth-aware case, we considered the single-image BMD (Hu *et al.*, 2014) and multi-image method (Arun *et al.*, 2015). For deep learning, we considered (Tao *et al.*, 2018; Nimisha *et al.*, 2017) which represent recurrent and autoencoder networks, respectively. Note that the publicly available code for (Chandramouli *et al.*, 2018; Srinivasan *et al.*, 2017) require as input 4D light field, whereas the codes for (Pan *et al.*, 2019; Lee *et al.*, 2018) are not available.

Metrics: For quantitative evaluation of image, we employ PSNR, IFC (Sheikh *et al.*, 2005), and VIF (Sheikh and Bovik, 2006). We have selected IFC and VIF because they are shown to be the best metrics for subjective evaluation of BMD (Lai *et al.*, 2016). For qualitative evaluation, we provide the narrow-FOV image and (normalized) depth estimated from deblurred image-pair or by algorithms (Hu *et al.*, 2014; Arun *et al.*, 2015). We consider all methods for four examples and provide sparse comparisons for others.

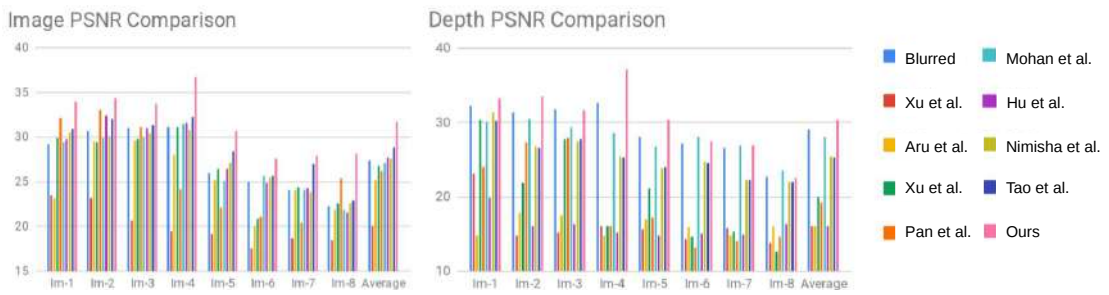


Figure 5.5: Quantitative evaluations using objective measure (PSNR). Our method performs competitively against the state-of-the-art, and produces the least depth errors.

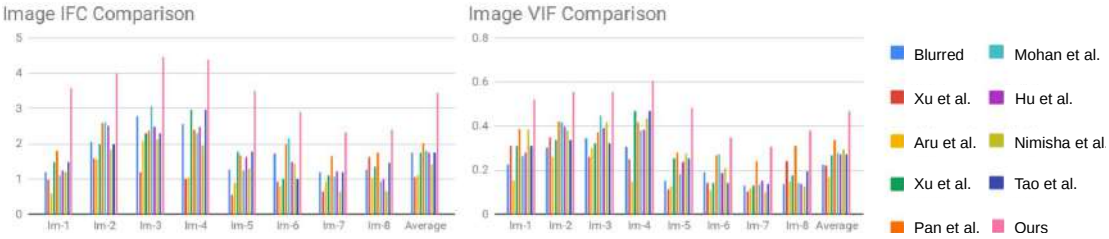


Figure 5.6: Quantitative evaluations using subjective measures (IFC, VIF). Our method performs deblurring with the best aesthetics.

Quantitative Evaluation: Figures 5.5–5.6 provide objective and subjective measures for different methods. First of all, both the measures of the state-of-the-art DL-BMD (Xu and Jia, 2012) clearly reveal its high sensitivity, when it deviates from the assumptions of synchronized and identical cameras, and layered depth scenes. This once again emphasizes the need for an unconstrained DL-BMD method. For normal camera methods (Pan *et al.*, 2016; Xu *et al.*, 2013), there is a perceivable drop in the depth performance (due to Remarks 2-3), which clearly suggests their inadequacy in DL set-up. While the inferior depth performance of (Arun *et al.*, 2015) can be attributed to its assumption of layered depth, for (Hu *et al.*, 2014), it can also be due to its single image restriction. As compared to our method, light field BMD (Mohan and Rajagopalan, 2018) is not quite successful (i.e., image/depth PSNR is less by 2.37/4.47 dB). This can be attributed to its lens effect and assumption of synchronized and identical camera settings. Our method outperforms deep learning methods (Nimisha *et al.*, 2017; Tao *et al.*, 2018) by 3.50 dB and 2.72 dB for image and 4.39 dB and 4.36 dB for depth, respectively. Based on the claims of (Nimisha *et al.*, 2017; Tao *et al.*, 2018) that they generalize well for real-captured images, this performance degradation could be possibly due to the unique characteristics of unconstrained DL blur.

Figure 5.7: Synthetic experiments: The method of (Xu and Jia, 2012; Hu *et al.*, 2014; Arun *et al.*, 2015) exhibits severe ringing artifacts and inaccurate depth estimates. The results of (Pan *et al.*, 2016; Xu *et al.*, 2013) amply underline the shortcomings of normal camera models. As compared to deep learning (Tao *et al.*, 2018; Nimisha *et al.*, 2017) and light field BMD (Mohan and Rajagopalan, 2018), our method retrieves distinct textual information. Also, we compare depth- and space-variant GT and estimated PSFs (inset patches of blurry and our results).

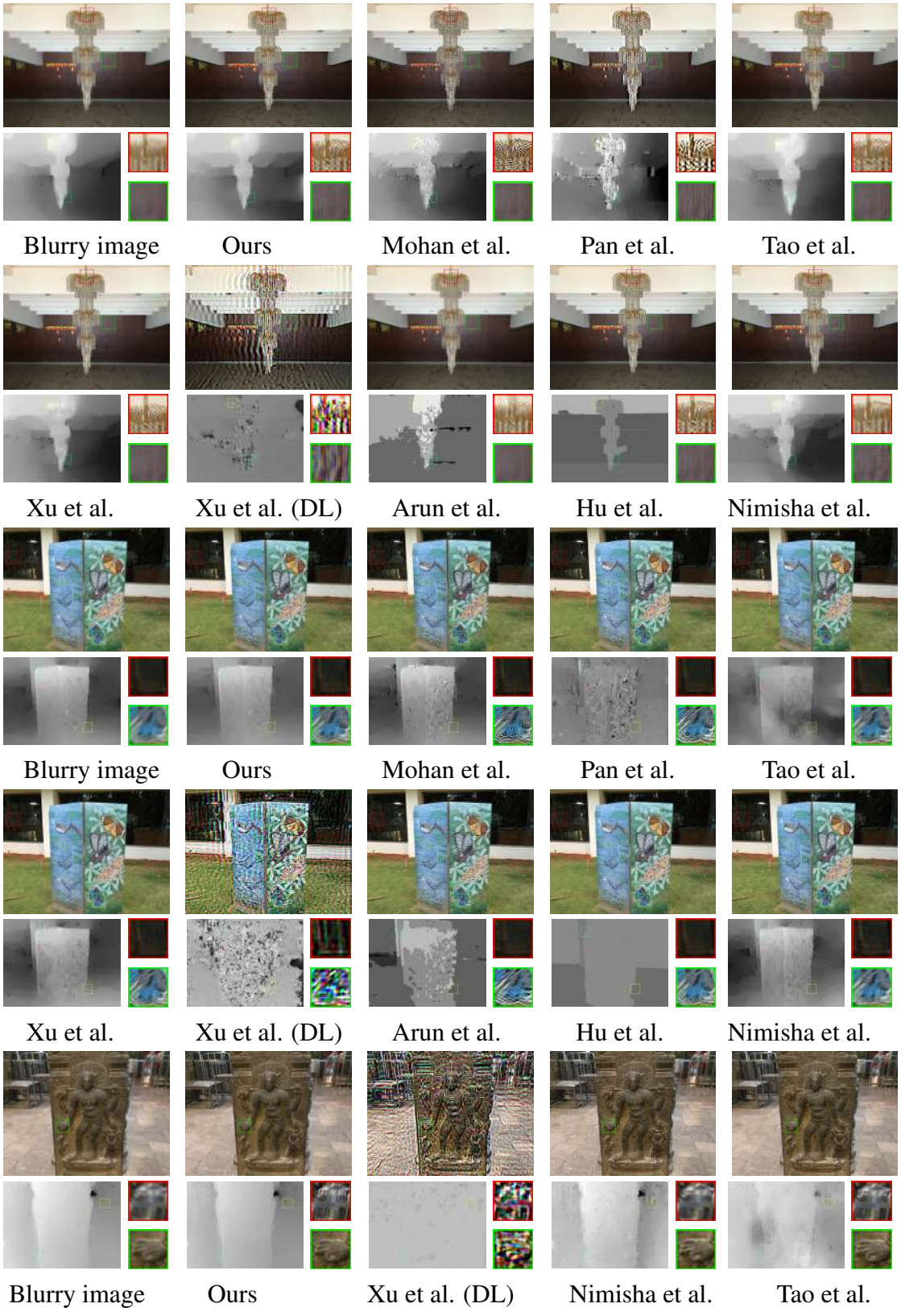


Figure 5.8: Real experiments: (first example - indoor scene, second - outdoor scene, and third - low-light scene). Our method is able to recover finer features at different depth ranges as compared to the competing methods, and is able to faithfully preserve the depth information.

Qualitative Evaluation: Figures 5.7–5.8 provide visual results for synthetic (Scharstein and Szeliski, 2002) and real experiments. We wish to highlight that ringing artifacts in deblurring are mainly caused by ego-motion error, which can be either due to inaccurate blur/ego-motion model or ineffectiveness of optimization. It can be seen that depth estimation is *also* sensitive to ringing artifacts; one reason could be that ringing deteriorates the feature matches required for depth estimation. The deblurred images of (Xu and Jia, 2012; Arun *et al.*, 2015) exhibit severe ringing artifacts (possibly due to the assumptions on scene and ego-motion and capture settings). Also, note that (Hu *et al.*, 2014) produces erroneous layered-depth estimates (e.g., nearer depths appear to be farther, as in Fig. 5.8, first example, chandelier). This is due to its sole restriction to single image cues for depth sensing. The results of (Mohan and Rajagopalan, 2018; Pan *et al.*, 2016; Xu *et al.*, 2013) amply demonstrate the inadequacy of light field and single-lens BMD in the dual-lens setup, where the deblurring is *not* uniform over different depth levels (e.g., in Fig. 5.7, second example, the closer books and farther windows are *not* simultaneously accounted for) and exhibits perceivable ringing artifacts, (e.g., in Fig. 5.8, over the chandelier). The visual results of deep learning methods (Nimisha *et al.*, 2017; Tao *et al.*, 2018) once again prove that they are inadequate to deal with DL blur. When compared with the competing methods on all the examples, it is evident that our DL deblurring method consistently accounts for features at different depths, produces lesser ringing artifacts, and faithfully preserves consistent depth information.

5.6.1 Implementation Details

We used a PC with an Intel Xeon processor and a 16 GB RAM for all experiments, and implemented our algorithm in `MATLAB`. For the scale-space based alternating minimization, we used 5 scales with 6 iterations each. The scaling factor for the ith scale is selected as $\frac{1}{\sqrt{2}}^{(i-1)}$. For estimating COR (following Eq. (5.17)), we have employed the `MATLAB` built-in function `lsqnonlin`. For depth estimation, we adopted the optical-flow algorithm of (Liu *et al.*, 2009) and employed the default parameters (as it provides a good trade-off between speed and accuracy). For optimizing the cost for MDF (Eq. (5.19)), we used the LARS solver of (Efron *et al.*, 2004) (which efficiently solves LASSO problems). The regularization for the proposed MDF-prior α is adapted with the scales as $5^{\frac{(9-i)}{2}}$ (Note that a higher regularization is employed as MDF vectors have

smaller values as compared to image). We have selected the sparsity regularization (λ_3) as 0.01 for both narrow-angle and wide-angle MDFs. We employed ADMM (Boyd *et al.*, 2011) to optimize the cost for latent-image with total-variation prior (Eq. (5.18)), where we used the total-variation regularization as 0.005. For latent image estimation, we consider grey-scale image until the final scale and 5th iteration (to reduce the computational time). We found that for deblurring a 1280×720 RGB narrow-angle image (of maximum blur-length of 30 pixels) and a focal-length ratio of two, our unoptimized MATLAB implementation took about 23 minutes to deblur the dual image-pair. A detailed break-up of the time taken for the final scale, final iteration is as follows: optimizing COR took 49.7s, estimating depth took 14.7s, MDF estimation took 56.4s, and RGB latent image estimation took 39.4s. In contrast, the competing traditional methods which specifically address multi-image deblurring (Xu and Jia, 2012; Arun *et al.*, 2015) took about 36 and 29 minutes, respectively. Even though, the competing deep learning methods for single image deblurring took less than one second (see Table 6.1), those methods are inadequate for dual-lens case.

5.7 Conclusions

In this chapter, we addressed the problem of blind motion deblurring for unconstrained dual-camera set-ups. Our algorithm allows for arbitrary COR in the blurring process and is incorporated in the optimization pipeline. That work revealed an inherent ambiguity in the BMD problem which hampers the scene-consistent depth cues embedded in the image-pair. Towards this end, we introduced a convex and computationally efficient prior. We showed the efficacy of the proposed prior which enforces scene consistent disparities, leading to improved deblurring. Comprehensive comparisons with existing state-of-the-art methods amply demonstrate the superiority and need of our method. As an increasing number of modern cameras are employing dual-lens configurations, our theory and method will be very relevant for steering further research in this field.

We focused in this chapter on restoration of unconstrained DL images blurred due to *only* camera motion. In practice, motion blur happens due to object motion as well. Therefore, a deblurring method that restricts blur due to only camera motion may not work for dynamic scene blur (i.e., blur due to camera motion, object motion or both),

and hence warrants a new method for the problem of dynamic scene deblurring in unconstrained DL cameras. As addressing this problem in a traditional way is computationally expensive as it involves complex pipeline and high-dimensional optimizations, we explore this problem using deep learning (in the next chapter).

CHAPTER 6

Dynamic Scene Deblurring for Unconstrained Dual-lens

6.1 Introduction and Related Works

In practice, apart from camera motion, motion blur happens due to object motion (dynamic scene) as well. This renders those deblurring methods that are restricted to *only* camera motion induced blur ill-equipped for several practical scenarios (Gao *et al.*, 2019; Nah *et al.*, 2017). Consequently, there arises a need to seamlessly tackle blur due to *camera motion, dynamic objects, or both* – the problem so called dynamic scene blind motion deblurring.

As discussed in the previous chapter, there has been a growing trend in modern cameras in employing unconstrained dual-lens (DL) cameras, i.e., two cameras with *same or different* focal lengths, exposures and image resolutions (Mohan *et al.*, 2019). This flexibility supports a plethora of important applications such as capturing narrow and wide field-of-view (FOV) with different focal lengths; HDR imaging (Park *et al.*, 2017; Bätz *et al.*, 2014; Sun *et al.*, 2010), low-light photography (Wang *et al.*, 2019*a*), and stereoscopics (Pashchenko *et al.*, 2017) using different exposure times, whereas super-resolution (Wang *et al.*, 2019*b*; Jeon *et al.*, 2018), visual odometry (Mo and Sattar, 2018; Iyer *et al.*, 2018) and segmentation (Shen *et al.*, 2017) employ nearly-identical exposure times. However, all these applications are meant for input DL images that are blur-free. But motion blur is an ubiquitous phenomenon in unconstrained DL cameras (Mohan *et al.*, 2019; Zhou *et al.*, 2019; Xu and Jia, 2012) and there does *not* exist a single method for dynamic scene deblurring for this popular imaging device.

As compared to single-lens methods, motion deblurring in unconstrained DL cameras involves additional conformity conditions and challenges (Mohan *et al.*, 2019; Zhou *et al.*, 2019). Zhou *et al.* (2019) showed that single-lens BMD methods, due to their obliviousness to stereo cues, are *inadequate* for DL cameras. To enable various DL-camera applications, unconstrained DL deblurring has to leverage stereo cues

and has to ensure scene-consistent disparities in deblurred image-pair. However, Mohan *et al.* (2019) showed that, albeit for the case of static scenes, conventional single-lens deblurring methods applied to unconstrained blurred DL-images can easily violate epipolar constraint (Hartley and Zisserman, 2003). A further challenge stems from the narrow-FOV cameras popularized by today's smartphones, which amplifies the effect of both camera motion and object motion, thereby exacerbating motion blur issues.

More important, a unique challenge presented by today's unconstrained DL genre is due to its different exposure times and resolutions ((Mohan *et al.*, 2019; Wang *et al.*, 2019*a*; Park *et al.*, 2017; Bätz *et al.*, 2014; Sun *et al.*, 2010)). This unconstrained configuration renders feature-loss due to blur in the two images *different*, e.g., image in one view can have more degradation as compared to the other view due to more blur due to large exposure time or low resolution. Consequently, typical methods produce *inconsistent* deblurring performance between the left-right views. However, almost all DL applications, especially those for stereoscopic 3D driven by the emerging demand for augmented/virtual reality (such as stereo super-resolution (Wang *et al.*, 2019*b*; Jeon *et al.*, 2018), style transfer (Chen *et al.*, 2018; Gong *et al.*, 2018), inpainting (Mu *et al.*, 2014; Wang *et al.*, 2008), panorama (Zhang and Liu, 2015), etc.), warrant the input DL images to be binocularly consistent, i.e., the left-right view has to have coherent features as perceived by human eyes (Chen *et al.*, 2013). Therefore, an unconstrained DL deblurring method has to ensure *consistency* between the left-right views, the problem we refer to as *view*-in*consistency*, which is a hitherto unaddressed problem.

For unconstrained DL cameras, there exists only one deblurring method (discussed in Chapter 5), however it restricts itself to *only* camera-motion induced blur. Basically, it models the motion blur in an unconstrained DL set-up, and devises an efficient framework that decomposes the joint DL-BMD deblurring cost in individual images to aid a low-dimensional optimization. This work reveals an inherent ill-posedness in DL-deblurring due to non-identical exposure time that disrupts the desired epipolar constraint, which it addresses using a prior on camera motion. However, that prior is not *effective* when dynamic objects are present (Sec. 6.5.4). Second, as the main focus of the above-mentioned method is to preserve epipolar constraint in DL deblurring, *no* attempts are made to address the problem of view-consistency. Third, it warrants an *iterative* high-dimensional optimization and hence is computationally expensive as compared to deep learning methods.

However, for constrained DL-cameras, wherein both cameras share the same focal length, resolution and exposure time (with full overlap), there exist several dynamic scene deblurring methods. These methods can be broadly categorized into two classes: Model-based optimization and model-agnostic deep learning methods. The model-based optimization class proceeds via a complex pipeline of segmenting different dynamic objects, estimating relative motion in the individual segments, and deblurring and stitching different segments while suppressing possible artifacts in seams (Pan *et al.*, 2017; Sellent *et al.*, 2016; Xu and Jia, 2012). Due to the presence of large number of unknowns, such as segmentation masks of dynamic objects, their depths, their relative motions, etc., these methods either warrant more information or restrict themselves to limited scenarios. For example, (Pan *et al.*, 2017; Sellent *et al.*, 2016) warrant *multiple* stereo image-pairs, whereas (Xu and Jia, 2012) restricts itself to blur due to primarily *inplane* camera or object motions or both, and requires individual objects to have uniform depth. Further, due to the complex pipeline and high-dimensional optimizations involved, methods belonging to this class incurs heavy computational cost. Specifically, (Pan *et al.*, 2017; Sellent *et al.*, 2016; Xu and Jia, 2012) rely on costly optical flow calculations for segmenting different blur regions, and employ a highly non-linear optimization framework to estimate the clean images.

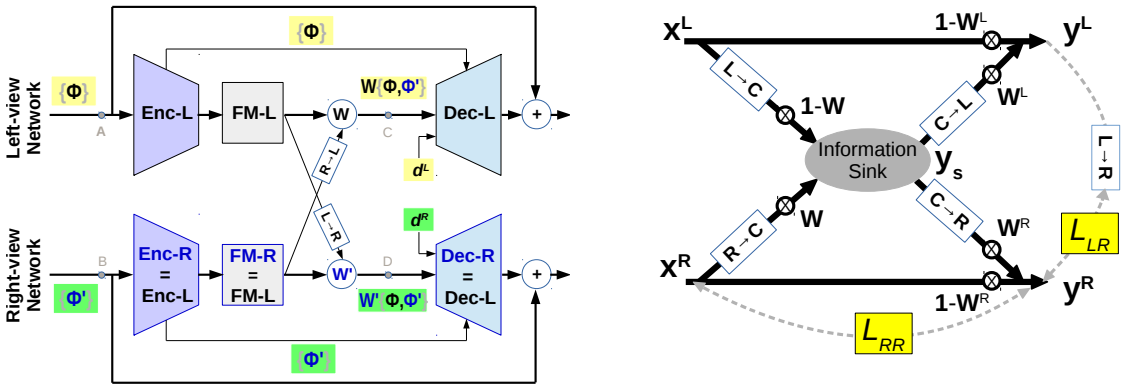
The second class of model-agnostic methods greatly addresses the limitations of the former class, as it learns from *unrestricted* data a mapping, that does *not* involve complex pipelines and optimizations while deblurring. However, this class is an emerging area for DL cameras, with only one existing method (Zhou *et al.*, 2019). It works by employing an identical deblurring network for two views, which leverages scene cues from the depth and varying information from two views to address spatially-varying blur. As the method of (Zhou *et al.*, 2019) restricts itself to constrained set-up, the questions of ill-posedness and view-inconsistency do *not* arise (Zhou *et al.*, 2019); but this is *not* the case for unconstrained DL-cameras wherein two cameras can have different configurations. Yet another issue in dynamic scene deblurring is due to its space-variant and image-dependent nature of blur (Zhang *et al.*, 2018), which is also not at all explored in the only-existing DL dynamic scene deblurring method (Zhou *et al.*, 2019).

For the first time in the literature, this chapter studies the problems of dynamic scene deblurring in today's ubiquitous unconstrained DL configuration. Our work belongs to the less-explored model-agnostic deep learning class. We address three main problems

in dynamic scene deblurring for unconstrained DL cameras, namely, enforcing view-consistency, ensuring scene-consistent disparities, and guaranteeing stability while addressing space-variant and image-dependent nature of blur, all in an interpretable and explainable fashion. In summary: (1). For view-consistency problem, we introduce a coherent fusion module with interpretable costs. Specifically, it works by fusing the unconstrained feature-pair to a single entity, which then sources a constrained feature-pair while retaining useful complementary information. (2). We show that the epipolar constraint in the deblurred image-pair can be enforced using an adaptive scale-space approach. Though adaptive scale-space means directly changing the respective parameters in model-based optimization methods, there is *no* analogous flexibility in deep networks, hitherto, typically owing to their perceivance as a black-box. This we address using signal processing principles. (3). To address the space-variant, image-dependent nature of blur, we extend the widely applicable atrous spatial pyramid pooling (ASPP) (Chen *et al.*, 2017). Basically, it provides freedom for a neural network to produce a 'variety' of space-variant and image-dependent receptive fields and filter-weights. We also contribute an unconstrained DL blur dataset to aid further exploration in this area. Our main contributions can be summarized as:

- As a first, we address the pertinent problem of view-inconsistency inherent in unconstrained DL deblurring, that forbids most DL-applications. To this end, we propose an *interpretable coherent-fusion* module.
- Our work reveals an inherent issue that disrupts scene-consistent depth in DL dynamic-scene deblurring. To address this, we introduce a memory-efficient *adaptive multi-scale* approach for deep learning based deblurring.
- To address the space-variant and image-dependent (SvId) nature of dynamic scene blur, we instil the SvId property in the widely-used deep learning module: atrous spatial pyramid pooling (Chen *et al.*, 2017).
- Our proposed approach based on the above three modules achieves state-of-the-art deblurring results for the popular unconstrained DL set-up, and acts as a potential preprocessing step for further DL applications.

In what follows, we systematically bring out these problems one-by-one, reason the inadequacies of the existing approaches in tackling these issues, and propose solutions to ameliorate these inadequacies.



(a) Standard DL network (b) Coherent Fusion Module

Figure 6.1: View Consistency: (a) Network Architecture of standard DL networks: when identical left-right networks process imbalanced signal, deblurring will be *unidentical*. (c) Coherent module to be placed in nodes $\{A, B\}$ and $\{C, D\}$ to enable feature sharing in order to create a balanced, yet high-feature output-pair.

6.2 View-*in*consistency in Unconstrained DL-BMD

As traditional stereo cameras typically employ a constrained set-up, most applications based on DL-images are designed for symmetric or view-consistent inputs. However, present-day dual-lens cameras employ different resolutions and exposure durations to enable extended applications (Wang *et al.*, 2019*a*; Park *et al.*, 2017; Bätz *et al.*, 2014). An important problem stems from this versatile configuration: as here the feature-loss due to image resolution and motion blur can be different in the left-right views, the constrained-DL BMD methods (e.g., (Xu and Jia, 2012; Zhou *et al.*, 2019)) naively applied to unconstrained case results in view-*in*consistent outputs and hence forbids the use of almost all existing stereo methods. The reason for this view-*in*consistency is due to their assumption that the input stereo images need to have identical resolutions and coherent blur (or fully overlapping exposure-times). In particular, (Xu and Jia, 2012) works by deconvolving the blurred images individually with *identical* PSFs, whereas (Zhou *et al.*, 2019) employ *symmetric, identical* network for left-right images, and hence both methods warrant constrained DL inputs. The only existing deblurring method for unconstrained DL (Mohan *et al.*, 2019) also fails to produce view-consistent output for unconstrained case, because there exists *no* means of feature transfer between the two-views (specifically, it works by deblurring two images *independently*).

To address the view-inconsistency problem, we resort to a Deep Learning based solution. In order to motivate our solution, first we analyse the inadequacy of the

only-existing deep learning DL-BMD method (albeit developed for constrained set-up) (Zhou *et al.*, 2019). Note that there exist *no* deep learning methods for the case of unconstrained DL. We first briefly review the the working of (Zhou *et al.*, 2019). As shown in Fig. 6.1(a), it consists of *symmetrical* networks for left-view and right-view images, with both networks sharing *identical* weights (in order to *not* scale-up trainable parameters as compared to that of single-lens methods (Zhou *et al.*, 2019)). Note that a similar architecture is used in other DL applications, such as style transfer (Gong *et al.*, 2018; Chen *et al.*, 2018) and super-resolution (Wang *et al.*, 2019*b*; Jeon *et al.*, 2018). Here, the mapping from blurred images $\{\mathbf{B}^L, \mathbf{B}^R\}$ to deblurred images $\{\hat{\mathbf{F}}^L, \hat{\mathbf{F}}^R\}$ can be given as

$$\begin{aligned}\hat{\mathbf{F}}^L &= T\big(\mathbf{B}^L_{\Phi}, \mathbf{f}^L_{\Phi,i}, \mathbf{W} \odot \mathbf{f}^L_{\Phi,enc} + \overline{\mathbf{W}} \odot \mathbf{f}^{R\rightarrow L}_{\Phi,enc}, \mathbf{d}^L\big),\\ \hat{\mathbf{F}}^R &= T\big(\mathbf{B}^R_{\Phi'}, \mathbf{f}^R_{\Phi',i}, \mathbf{W}' \odot \mathbf{f}^R_{\Phi',enc} + \overline{\mathbf{W}'} \odot \mathbf{f}^{L\rightarrow R}_{\Phi',enc}, \mathbf{d}^R\big),\end{aligned} \tag{6.1}$$

where the sets Φ and Φ' captures the resolutions and exposures of the left-right views, and $\mathbf{f}_i$ is ith intermediate-features of encoder which are fed-forward to decoder and $\mathbf{f}_{enc}$ is encoder-output. Bilinear mask $\mathbf{W}$ (with $\overline{\mathbf{W}} = \mathbf{1} - \mathbf{W}$) combine left-view and right-view encoder-outputs after registration (denoted by '$\rightarrow$') for view-aggregation, and $\{\mathbf{d}^L, \mathbf{d}^R\}$ are depth-features for depth-awareness.

The primary reason for the success of the generic DL architecture in producing view-consistent output for constrained DL set-up is that mappings in the left- and right-view networks are identical ($T(\cdot)$ in Eq. (6.1)), and more important, in like manner signal flowing in those networks are of identical nature (i.e., $\Phi = \Phi'$). However, as shown in Fig. 6.1(a) using yellow and green highlights, the same architecture leads to view-*in*consistency in unconstrained DL set-up because now signal flowing in those identical networks are of *different* nature (i.e., $\Phi \neq \Phi'$). A similar reasoning of view-inconsistency is valid for single-image deep learning methods as well (the main difference here is that the left- and right-view networks, though can be parallelized as in Fig. 6.1(a), are decoupled, i.e., $\mathbf{W} = \mathbf{W}' = 1$ in Eq. (6.1)).

6.2.1 Coherent Fusion for View-consistency

View-*in*consistency in the generic DL deblurring architecture (Fig. 6.1(a)) occurs because there exist *no* avenues to enforce that the nature of signals flowing in the left- and right-view networks *identical*. Therefore, *a method to address the problem of view-consistency has to ensure signal flowing in left- and right-view networks to be of identical nature 'irrespective' of* $\Phi \neq \Phi'$ *or* $\Phi = \Phi'$. Note that a constrained DL setting is a special case of unconstrained DL settings. As followed in the generic DL architecture, a fusion module inserted only between the final stage of encoder and initial stage of decoder (i.e., node $\{C, D\}$ in Fig. 6.1(a)) is *not* sufficient for unconstrained DL configuration, because even if the nature of signal is made identical in this node, still the imbalance persists because of the standard U-net configuration due to the feed-forward connection from intermediate stages of the encoder to respective stages of the decoder. A careful inspection of the generic DL architecture reveals that the view-inconsistency stems from nodes $\{A, B\}$ and $\{C, D\}$, where the former creates imbalance in the encoder inputs and hence all feed-forward inputs to the decoder and network output, whereas the latter creates imbalance in the decoder inputs.

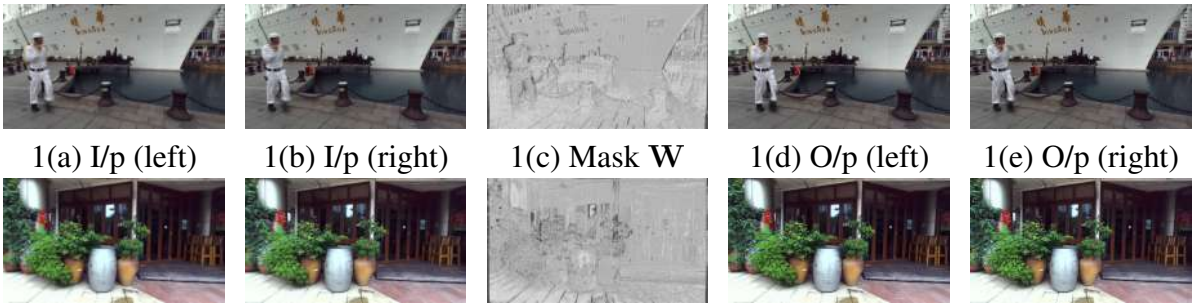
To this end, we introduce a coherent fusion module with two self-supervision costs in those two nodes, which enforce the following conditions: (a) The nature of output signals in the left-right views are *identical*; (b) *Both* the outputs exhibit the properties of the input with *higher* information. We select the high-resolution image as the reference (say, the right-view) since reducing resolutions leads to *irrecoverable* information-loss (e.g., due to anti-aliasing, the basic issue addressed in super-resolution). Without loss of generality, we assume that the right-view has higher resolution. Considering the left-right view input to the module as $\{\mathbf{x}^L, \mathbf{x}^R\}$, the coherent fusion module maps $\{\mathbf{x}^L, \mathbf{x}^R\}$ to left-right view output $\{\mathbf{y}^L, \mathbf{y}^R\}$ as

$$\mathbf{y_s} = \mathbf{W} \odot \mathbf{x}^{R \to C} + \overline{\mathbf{W}} \odot \mathbf{x}^{L \to C}; \tag{6.2}$$

$$\mathbf{y}^L = \mathbf{W}^L \odot \mathbf{y_s}^{C \to L} + \overline{\mathbf{W}^L} \odot \mathbf{x}^L; \tag{6.3}$$

$$\mathbf{y}^R = \mathbf{W}^R \odot \mathbf{y_s}^{C \to R} + \overline{\mathbf{W}^R} \odot \mathbf{x}^R. \tag{6.4}$$

where $\mathbf{x}^{R \to C}$ warps the right-view input $\mathbf{x}^R$ to the center-view, $\odot$ is the Kronecker



1(a) I/p (left) 1(b) I/p (right) 1(c) Mask $\mathbf{W}$ 1(d) O/p (left) 1(e) O/p (right)

2(a) I/p (left) 2(b) I/p (right) 2(c) Mask $\mathbf{W}$ 2(d) O/p (left) 2(e) O/p (right)

Figure 6.2: Visualization of Coherent Fusion Module: Overall high magnitude of mask $\mathbf{W}$ reveals that the view with rich information predominantly sources the information-sink, with exceptions at occlusions or view-changes where information is present only at the other view. In Figs. 1-2(b), observe the relatively rich information in right-view inputs where $\mathbf{W}$ has high magnitudes overall (Figs. 1-2(c)). Also, compare the coat behind the sailor in Figs. 1(a-b) or the specularity-difference in the pillar or bright-window in Figs. 2(a-b) where only the left-view contains the information and hence $\mathbf{W}$ magnitudes in those regions are low (Figs. 1-2(c)). The coherent-fusion costs $L_{LR} + L_{RR}$ aid this phenomenon, which results in a high view-consistent deblurring performance in both the left- and right views (see Figs. 1-2(d-e)).

product, and $\{\mathbf{W}, \mathbf{W}^L, \mathbf{W}^R\}$ are image-dependent bilinear masks produced by a simple mask-generation network (as in (Zhou *et al.*, 2019; Gong *et al.*, 2018; Chen *et al.*, 2018)), e.g., $\mathbf{W}$ is a function of the error between $\mathbf{x}^{L\to C}$ and $\mathbf{x}^{R\to C}$, where $0 \preceq \mathbf{W} \preceq 1,\ \mathbf{W} + \overline{\mathbf{W}} = \mathbf{1}$. Also, the two self-supervision costs are

$$L_{LR} = \|\mathbf{y}^{L\to R} - \mathbf{y}^R\|_2^2 \ \text{ and } \ L_{RR} = \|\mathbf{y}^R - \mathbf{x}^R\|_2^2. \tag{6.5}$$

In words, Eq. (6.2) fuses the input left-view and right-view features warped to the center-view (using $\mathbf{W}$) to form intermediate feature $\mathbf{y_s}$; Eqs. (6.3)-(6.4) create the output left-view (and right-view) features by merging the inverse-warped intermediate feature and the input left-view (and right-view) features using $\mathbf{W}^L$ (and $\mathbf{W}^R$). In Eq. (6.5), the cost L_{LR} minimizes the means square error (MSE) between the output right-view feature and the output left-view feature warped to the right-view, whereas the cost L_{RR} minimizes the MSE between the output and input right-view features.

We now attempt to provide a high-level interpretation of how this approach ensures view-consistency. As shown in Fig. 6.1(b), the first part of the coherent fusion module acts as an information sink, which accumulates information from the (possibly asymmetric) input left-right views to form a *single* center-view information source. Note

that the information taken from different views are image-dependent (through mask $\mathbf{W}$). This is illustrated using an example in Fig. 6.2, where the overall high magnitudes of mask $\mathbf{W}$ reveal that the input-image in the view having relatively rich information (here, the right-view) predominantly sources the information-sink, with exceptions at occlusions or view-changes where information is present in only the other view (e.g., observe the overall high-magnitudes of mask for the right-view image which contains more information, except at the regions of coat behind the sailor in Fig. 6.2(a) or the difference in specularity behind the pillar or in the bright-window in Fig. 6.2(a), where right-view image does *not* have sufficient information). Finally, the single center-view information sources the output left-right views symmetrically, i.e., two signals with identical nature, with a provision to fill the occlusion in left-right views which is not present in the center-view (but present in the input left-right view information) through masks $\{\mathbf{W}^L, \mathbf{W}^R\}$. Note in the Fig. 6.2(d-e) that the deblurred output have identical, rich information, with *no* holes due to occlusion. We now proceed to discuss the importance of the two costs. (For the sake of simplicity, we assume that occlusions and specularities in stereo images are negligible.)

Remark 1: The mapping of the coherent fusion module (Eq. (6.2)) *minimizes* individual costs L_{LR} and L_{RR}. Further, both costs L_{LR} and L_{RR} are necessary to satisfy the conditions: (A) The nature of outputs in the left-right views *identical*; (B) The two outputs exhibit the properties of the input with *higher* information.

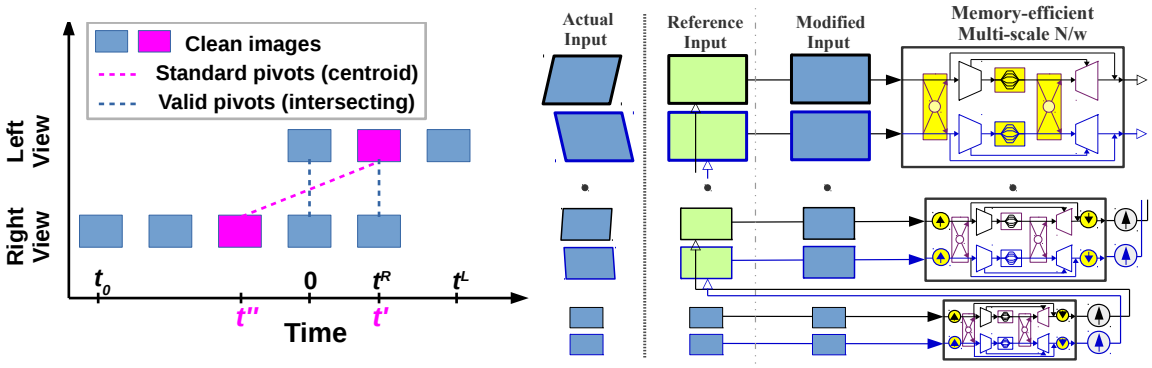
Justification: To justify the first part, it is sufficient to show that there exists atleast a case where the mapping in Eq. (6.2) attains the least objective (zero) for the costs L_{LR} and L_{RR} (as $L_{LR}, L_{RR} \geq 0$). It is evident from Eq. (6.2) that the cost $L_{LR} = 0$ when $\mathbf{W}^L = \mathbf{W}^R = \mathbf{1}$, and $L_{RR} = 0$ when $\mathbf{W} = \mathbf{W}^R = \mathbf{1}$. To justify the second part, we show that criteria that minimize L_{LR} satisfies the Condition A but *not* necessarily Condition B; similarly, criteria of L_{RR} satisfies the Condition B but *not* necessarily Condition A, and finally, the common criterion of L_{LR} and L_{RR} satisfy both Conditions A and B. For brevity, we refer to "first-view is sourced by second-view" if the first-view's output is formed by the second-view's input. From Eq. (6.5), the cost $L_{LR} = 0$ implies $\mathbf{y}^R = \mathbf{y}^L$, (i.e., Condition A). For non-identical left-right inputs in general, the cost $L_{LR} = 0$ when the left- and right-views are sourced by only right-view (i.e., $\mathbf{W} = \mathbf{W}^L = \mathbf{1}$ in Eq. (6.2)), or only left-view (i.e., $\mathbf{W} = \mathbf{0}, \mathbf{W}^R = \mathbf{1}$), or a combination of left- and right-view (i.e., $\mathbf{0} \prec \mathbf{W} \prec \mathbf{1}, \mathbf{W}^L = \mathbf{W}^R = \mathbf{1}$). Clearly from Eq. (6.5), $L_{RR} > 0$ for

the last two cases, as $\mathbf{y}^R \neq \mathbf{x}^R$ (thereby violating Condition B). Similarly, $L_{RR} = 0$ implies $\mathbf{y}^R = \mathbf{x}^R$, (i.e., condition B). The cost L_{RR} can be zero when right-view, but not left-view, is sourced by right-view (i.e., $\mathbf{W} = \mathbf{W}^R = \mathbf{0}$) or both right and left view are sourced by the right view (i.e., $\mathbf{W} = \mathbf{W}^L = \mathbf{1}$). For the first case $L_{LR} > 0$, as $\mathbf{y}^R \neq \mathbf{y}^L$ (thereby violating Condition A). Resultantly, the common criterion that minimizes L_{LR} and L_{RR} is when *both* the left- and right-views are sourced by only right-view (which satisfies both Conditions A and B), which proves the remark. (Relaxing the assumption on occlusions and specularities, the word "sourced" becomes "predominantly sourced", wherein occlusion and specularity information will be passed to left- and right-view outputs by the left-and right-view inputs, respectively.)

The implication of this module spans beyond the deblurring problem. In particular, existing deep learning based DL applications are designed for constrained set-up (e.g., style transfer (Gong *et al.*, 2018; Chen *et al.*, 2018) and super-resolution (Wang *et al.*, 2019*b*; Jeon *et al.*, 2018)), and hence fail to produce view-consistent results for today's popular unconstrained set-up (due to the same reason as that of deblurring). Our Coherent Fusion Module can potentially aid in removing this limitation and can serve as a basic block for view-consistency in future deep learning works for unconstrained DL.

6.3 Scene-*in*consistent depth in Unconstrained DL-BMD

Though the generic DL network with coherent fusion (in Sec. 6.2) enforces view-consistency, it need *not* encode scene-consistent depth while deblurring. Note that scene-consistent depth is important for further DL applications, such as augmented reality, 3D reconstruction, and scene understanding (Innmann *et al.*, 2019; Lv *et al.*, 2018; Zhang *et al.*, 2011). A DL image encodes scene-consistent depth if the epipolar constraints are satisfied (Hartley and Zisserman, 2003), i.e., in a typical stereo camera, horizontal disparities of image-features are consistent with scene-geometry and vertical disparities are negligible (Xiao *et al.*, 2018; Fusiello and Irsara, 2011; Fusiello *et al.*, 2000; Loop and Zhang, 1999). For the scenario of dynamic objects, in general, a clean DL image with scene-consistent depth is obtained when both images are captured at the *same* time-instant; otherwise, world-position of dynamic object(s) in one-view need *not* be the same in the other-view and hence violates epipolar constraints. Next, we



(a) Ambiguity in depth UDL-BMD (b) Adaptive scale-space network

Figure 6.3: Scene-consistent Depth: (a) As centroid of blurred images need not align for unconstrained case, deblurring *violates* epipolar constraint. (b) The discrepancy in unconstrained DL deblurring can be solved using a scale-space approach, where networks at lower scales can be derived from the top-most one.

show that standard deep learning based deblurring methods developed for single-lens and constrained DL, directly applied to unconstrained DL results in a similar depth-*in*consistency issue as that in the case of dynamic objects.

A motion blurred image encodes a video sequence over its exposure time (Jin *et al.*, 2018; Purohit *et al.*, 2019); in particular, blurred image is formed by the summation of clean frames of that video sequence (Whyte *et al.*, 2012; Mohan *et al.*, 2019). Considering an unconstrained DL exposure setting, i.e., exposures need *not* be identical and fully-overlapping (as in (Park *et al.*, 2017; Bätz *et al.*, 2014; Wang *et al.*, 2019*a*; Pashchenko *et al.*, 2017)), blurred image-pair $\{\mathbf{B}^L, \mathbf{B}^R\}$ in the left-right views is given as

$$\mathbf{B}^L = \frac{1}{t_L}\int_0^{t^L} \mathbf{L}_t^L \, dt, \qquad \mathbf{B}^R = \frac{1}{t^R - t^0}\int_{t^0}^{t^R} \mathbf{L}_t^R \, dt, \tag{6.6}$$

where $\{\mathbf{L}_t^L, \mathbf{L}_t^R\}$ is the clean DL image-pair at time-instant t, and $[0, t^L]$ and $[t^0, t^R]$ are respectively the exposure times in the left-right views. Note that the constrained DL setting (in (Zhou *et al.*, 2019; Pan *et al.*, 2017; Sellent *et al.*, 2016; Xu and Jia, 2012)) is a special case of Eq. (6.6), where $t^0 = 0$ and $t^L = t^R$ which implies identical, fully-overlapping exposures.

Standard deep learning based deblurring methods works by learning a mapping from blurred image-pair to a clean image-pair located at a *particular* time-instant (which we refer to as pivot). As discussed earlier, left- and right-view pivots for dynamic scenes should match at the same time instant for scene-consistent depth. In deep learning based

approaches, pivots are typically selected at the middle of exposure time (Zhou *et al.*, 2019; Tao *et al.*, 2018; Jin *et al.*, 2018) (otherwise ill-posedness exists in learning as reversing the arrow of time produces the same blurred image but maps to a different clean image). This scheme is apt for constrained setting, as it *automatically* results in a clean image-pair with scene-consistent depth, i.e., $\{\mathbf{F}^L_{t'}, \mathbf{F}^R_{t''}\}$ where $t' = t''(= t^R/2 = t^L/2)$. However, for partially overlapping exposures, it causes serious binocular *in*consistency as t' deviates from t'' (as illustrated in Fig. 6.3(a)). Further, even if the pivots are chosen as some Mth and Nth fraction of exposure times, the deblurred image-pair (in general) will still exhibit binocular *in*consistency, with severity increasing with the separation between the pivots $\big(M \cdot t^R$ and $N \cdot (t^L - t^0)\big)$. In fact, for an unconstrained exposure where timings $\{t^R, t^0, t^L\}$ can *freely* vary, there does *not* exist a unique choice of pivots (or M and N) which will produce scene-consistent depth.

Therefore, *a method to address the problem of scene-inconsistent depth has to* adaptively *select pivots in accordance with input blurred image-pair (via exposure timings). In particular, it has to establish a mutual agreement between the left- and right-view images to arrive at an intersecting pivot.* Since single-image methods for DL operate by independently reusing the same network for the two views (Mohan *et al.*, 2019; Zhou *et al.*, 2019), a mutual agreement *cannot* be established between the views. Even though the generic DL architecture (in Fig. 6.1(a)) promotes a signal-flow between the views, i.e., by adding *registered* encoder-output of one view to encoder-output of the other view (in node $\{C, D\}$), this registration hinders the control on pivots; but, registration is indispensable for coherently adding the two encoder-outputs (Zhou *et al.*, 2019; Chen *et al.*, 2018; Gong *et al.*, 2018). Further, the prior developed in (Mohan *et al.*, 2019) to tackle scene-inconsistent depth problem is confined to only camera-motion induced blur and is inadequate for dynamic scenes (justification is provided in Sec. 6.5).

We first provide an outline of how we address this problem of scene-*in*consistent depth. We show in Sec. 6.3.1 that if DL deblurring is performed in lower scales (i.e., on decimated DL blurred images), the problem of depth discrepancy becomes less severe. This motivates our *adaptive* scale-space approach for unconstrained DL deblurring where we employ multiple network-levels (that correspond to increasing image-scales, as shown in Fig. 6.3(a)). For a given DL blur input, we start deblurring from an *appropriate* lower scale where the depth discrepancy is negligible in order to produce scene-consistent deblurred results in that scale. Then, the deblurred results in lower

scales are employed to progressively correct depth inconsistency in subsequent higher scales till the fine scale is reached. For a constrained DL case, only one network-level is sufficient (as in (Zhou *et al.*, 2019)), but for an unconstrained DL case, as depth discrepancy becomes higher more network-levels are required.

However, existing scale-space approach (Gao *et al.*, 2019; Tao *et al.*, 2018; Nah *et al.*, 2017) in this regard has several major limitations. First, as it is typically designed for a pre-determined network-levels (exactly three) it imposes an upper limit on allowable depth discrepancies, and hence becomes *ineffective* for a large class of unconstrained DL inputs. Second, simply increasing network-levels is *not* desirable as it calls for *independent* network-weights for different image-scales which escalates the memory requirement. Third, as the existing approach employs the same network-levels for all inputs, it increases the computational cost (with respect to both FLOPs and processing time) due to sequentially processing through *all* levels *irrespective* of constrained and various unconstrained cases. Our adaptive scale-space addresses these limitations as follows: We address the first two limitations using signal processing principles in Sec. 6.3.2, where we show how to optimally convert a fine-scale network to a multi-scale network by reusing the same weights, thereby allowing any desired number of multi-levels while *not* escalating the memory requirement. To optimize the computational cost, we devise a training and testing strategy in Sec. 6.6.1 which *appropriately* selects the lower scale depending on the input case, and employing the technique noted before, we employ our single-scale network to derive the required multi-scale network. In the following, we elaborate our adaptive scale-space approach in detail.

6.3.1 Adaptive Scale-space for Scene-consistent Depth

Following (Zhou *et al.*, 2019; Tao *et al.*, 2018; Jin *et al.*, 2018), we consider the standard choice of pivot. i.e., at the center of exposure time or the centroid of blurred images. As discussed earlier, this choice of pivot for an unconstrained DL set-up results in deblurred left-right images at different time-instants, e.g., time-instants $\{t', t''\}$ in Fig. 6.3(a). As a result, a scene-point with respect to one view can undergo different pose-changes in the other view due to object motion or camera motion or both, thereby leading to disparities that are scene-*in*consistent. We now attempt to quantify this disparity error.

Let the world coordinate of a scene-point at time-instant t' is $\mathbf{X}$; then its corresponding image-coordinates at the same pivot t' in the left- and right-views are respectively $\{\mathbf{K}^L\big(\frac{\mathbf{X}}{Z}\big), \mathbf{K}^R\big(\frac{\mathbf{X}+\mathbf{l_b}}{Z}\big)\}$, where $\mathbf{K}^L$ and $\mathbf{K}^R$ are the intrinsic camera matrices of left- and right-views, $\mathbf{l_b}$ is the stereo baseline, and Z is the actual scene-depth (Mohan *et al.*, 2019). The matrix $\mathbf{K}$ is of the form $\text{diag}(f, f, 1)$, where f is the focal length in pixels which is proportional to the number of image-rows or columns (Whyte *et al.*, 2012; Mohan *et al.*, 2019). Note that this case produces scene-consistent depth as the right-view sees the same world-coordinate as the left-view, displaced by the baseline. Constrained DL deblurring belongs to this category of intersecting pivots.

Next, suppose that the scene-point $\mathbf{X}$ has undergone a rotation and translation $\mathbf{R}$ and $\mathbf{t}$ at time-instant t'' (i.e., $\mathbf{RX} + \mathbf{t}$, with corresponding depth Z'), and the left and right-views have pivots at $\{t', t''\}$ (as shown in Fig. 6.3(a)). In this case, corresponding image-coordinates become $\{\mathbf{K}^L\big(\frac{\mathbf{X}}{Z}\big), \mathbf{K}^R\big(\frac{\mathbf{RX}+\mathbf{t}+\mathbf{l_b}}{Z'}\big)\}$. Clearly, the latter case exhibits a scene-inconsistent offset in the right-view as compared to the previous case, which is given as

$$\Delta\mathbf{x}^R = \mathbf{K}^R\left(\frac{\mathbf{X}+\mathbf{l_b}}{Z} - \frac{\mathbf{RX}+\mathbf{t}+\mathbf{l_b}}{Z'}\right), \tag{6.7}$$

where $\Delta\mathbf{x}^R$ is the image-coordinate discrepancy which contributes to scene-inconsistent depth. Note that unlike camera motion induced blur (Mohan *et al.*, 2019), $\mathbf{R}$ and $\mathbf{t}$ for dynamic scenes can vary with scene-points due to independent object motion. Now consider that we decimate the left-right blurred images by a factor of $D(> 1)$, i.e., image resolutions are scaled-down by D and hence the focal lengths (in pixels) will be scaled by $1/D$. Therefore the resultant image-coordinate discrepancy in Eq. (6.7) becomes (Hartley and Zisserman, 2003; Whyte *et al.*, 2012)

$$\Delta\mathbf{x}_D^R = \mathbf{D}\Delta\mathbf{x}^R, \text{ where } \mathbf{D} = \text{diag}\left\{\frac{1}{D}, \frac{1}{D}, 1\right\}. \tag{6.8}$$

An important insight from Eqs. (6.7)-(6.8) is that *image-coordinate discrepancies get scaled down in accordance with decimation factors*. This motivates our adaptive scale-space approach (Fig. 6.3(b)). First, we judiciously select a decimation factor that reduces the maximum discrepancy within a sensor-pitch, so that the epipolar constraints hold good in the discrete image-coordinate domain (Innmann *et al.*, 2019; Hartley and Zisserman, 2003). Next, we consider the coherent deblurred image-pair from

the selected scale as the reference to centroid-align the binocularly inconsistent blurred image-pair in the higher scale (via registration), which similarly produces a coherent deblurred image-pair. This process is repeated till the fine-scale. Note that our registration approach is similar to the video deblurring method (Su *et al.*, 2017) where a blurred frame is used as the reference to centroid-align its neighbouring blurred frames, which together produce a coherent deblurred frame. Further, employing deblurred image from a coarse scale as the reference for higher scale is standard practice in conventional deblurring methods (Whyte *et al.*, 2012; Pan *et al.*, 2016; Mohan *et al.*, 2019).

The scale-space approach has *not* been explored in deep learning based DL deblurring. In addition, existing scale-space methods for single image deblurring (Gao *et al.*, 2019; Tao *et al.*, 2018; Nah *et al.*, 2017) restrict themselves to *fixed* decimation scales and *limited* network-levels owing to memory consideration. The main reason is that a single-level network trained for fine image-scale is seldom effective for lower image-scales, and hence necessitates independent trainable parameters in individual network-levels (Gao *et al.*, 2019). More important, how to adapt a network trained for an image-scale to be applicable for lower image-scales *without* accentuating the memory requirement is still an open question and has significant potential; e.g., it allows a single-level network to be extended as a multi-scale network with *diverse* decimation scales and *unconstrained* multi-levels (as required for our adaptive scale-space approach), and it enables a network trained with high-resolution images to effectively accommodate inputs of lower-resolutions (possibly due to camera-constraints). To this end, we reveal an untapped potential in our DL deblurring architectures which allows the above-mentioned memory-efficient scheme via a simple yet effective transformation based on signal processing principles (which we discuss next).

6.3.2 Memory-efficient Adaptive Scale-space Learning

In this section, we analytically reason the well-known empirical observation in scale-space literature that a network trained for fine image-scale is *not* effective for lower image-scales (Gao *et al.*, 2019; Tao *et al.*, 2018; Nah *et al.*, 2017). Based on it, we alleviate in our deblurring network the aforementioned issue to a large extent via a suitable transformation which warrants *no* new parameters. This paves the way for memory-efficient adaptive multi-scale approach by appropriately stacking networks suitable for

various decimation scales (derived from the fine-scale network).

We briefly review some signal processing concepts used here (Oppenheim and Schafer, 2014). For sake of simplicity, we consider one-dimensional signal representation. We denote the frequency spectrum of a discrete signal $\mathbf{x}(n)$ by $\mathbf{X}(\omega)$ (where ω is the frequency domain). The convolution of $\mathbf{x}(n)$ and $\mathbf{y}(n)$ (denoted by $\mathbf{x}(n) * \mathbf{y}(n)$) results in a frequency spectrum $\mathbf{X}(\omega)\mathbf{Y}(\omega)$. Decimating a signal $\mathbf{x}(n)$ by a factor D first removes its high-frequency content (via anti-aliasing filter) and then *expands* the frequency spectrum by D. In contrast, interpolating a signal $\mathbf{x}(n)$ first *compress* the frequency spectrum by D and then removes its high-frequency content (via anti-imaging filter). We denote the decimation and interpolation by $\mathbf{X}_{\downarrow D}$ and $\mathbf{X}_{\uparrow D}$, respectively. In general, decimation followed by an interpolation (i.e., $(\mathbf{X}_{\downarrow D})_{\uparrow D}$) is *not* an inverse operation due to anti-aliasing operation. For brevity, we refer to "a particular feature of $\mathbf{X}(\omega)$ matches $\mathbf{Y}(\omega)$" if that feature of $\mathbf{X}$ is present in $\mathbf{Y}$ as it is or as in a decimated form.

Before going into details, we provide an outline of this section. In a scale-space network, the output of lower-scale network should be the decimated version of the fine-scale network-output (Gao *et al.*, 2019; Tao *et al.*, 2018; Nah *et al.*, 2017). This implies that frequency spectrum of lower-scale network-output must *match* to that of fine-scale network-output (except at those high-frequency features lost due to decimation). Our DL deblurring network (as in Fig. 6.1) maps the input via a composition of individual functions, which are realized using a cascade of convolutions, non-linearities (e.g., ReLu or Sigmoid), decimations in encoder and interpolations in decoder, etc. Resultantly, for output features of lower-scale networks to match that of fine-scale network, all those individual functions must map to *matching* features for *all* image-scales. However, we show (in Remark 3) that directly employing the fine-scale network in lower-scales maps to *complimentary* features for convolutions, and hence fails to produce matching features in lower-scales. Further, we show that our proposed transformation alleviates the aforementioned issue of convolutions. Also, we show that this property of transformation generalizes to other network-functions as well, as required in a scale-space network. Finally, we demonstrate some practical utilities of our approach.

Assume that our DL deblurring network is optimally trained for the fine-scale. Since the coherent fusion (Sec. 6.2.1) creates symmetric networks for the two views, the inference derived from the network for a particular view is valid for the other as well. In

frequency domain, the overall network-mapping for a view is

$$\mathbf{Y}(\omega) = \mathbf{X}(\omega) + T\big(\mathbf{X}(\omega)\big), \tag{6.9}$$

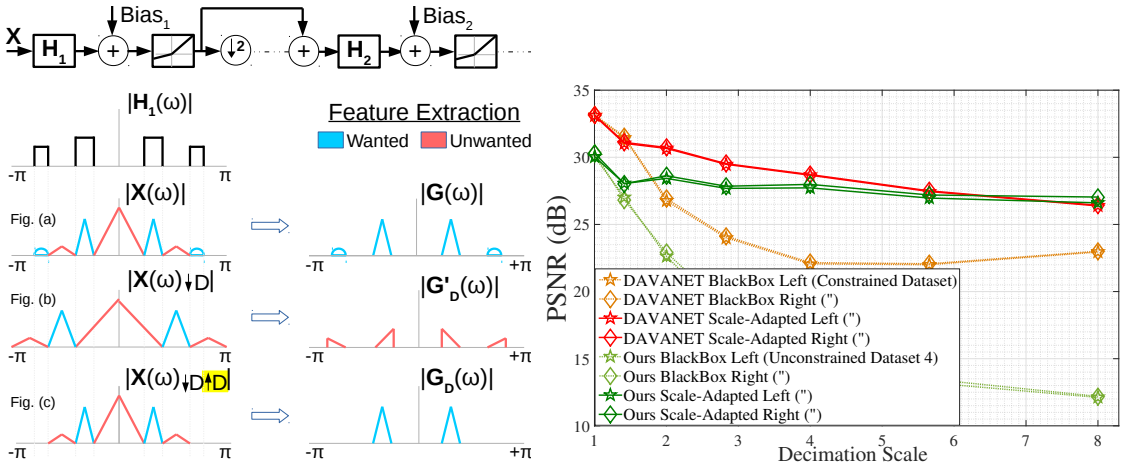
where $\mathbf{Y}$ is the output of the network and $\mathbf{X}$ is the respective output of the first coherent-fusion module (at node $\{A, B\}$ in Fig. 6.1(a)), and $T(\cdot)$ is the mapping of encoder-decoder network which involves a series of convolutions and other non-linear operations. Next, consider that the network optimized for the fine-scale (Eq. (6.9)) is directly employed for a lower image-scale ($D > 1$). For decimated input images, the costs of the first coherent fusion module (Eq. 6.5) enforces its new output as a decimated version of its fine-scale output ($\mathbf{X}(\omega)$). Therefore, the network-mapping becomes

$$\mathbf{Y}'_{\mathbf{D}}(\omega) = \mathbf{X}(\omega)_{\downarrow D} + T\big(\mathbf{X}(\omega)_{\downarrow D}\big), \tag{6.10}$$

We claim that considering the fine-scale network as a black-box for other lower-scales (i.e., Eq. (6.10) is *not* optimal as it maps to complimentary features, whereas the following network-mapping addresses this limitation:

$$\mathbf{Y}_{\mathbf{D}}(\omega) = \Big(\mathbf{X}(\omega)_{\downarrow D\uparrow D} + T\big(\mathbf{X}(\omega)_{\downarrow D\uparrow D}\big)\Big)_{\downarrow D}. \tag{6.11}$$

We assume that anti-aliasing and anti-imaging filters in decimation and interpolation (Oppenheim and Schafer, 2014) are ideal, but our training procedure discussed at the end of this section relaxes this. As compared to Eq. (6.10), basically Eq. (6.11) interpolates the network-input before feeding to the network, and finally decimate the network-output. We attempt to provide some intuitions. Suppose that the transformation $T(\cdot)$ in the fine-scale (in Eq. (6.9)) extracts/maps features in a particular frequency band (wanted features), whereas suppresses features in the remaining frequency band (unwanted features). Then to leverage the same transformation $T(\cdot)$ for lower-scales and yet retain those wanted features (as required in a scale-space network), the domain of the wanted features for lower scales should be *identical* as that of fine-scale. However, decimation of inputs in lower-scales (as followed in Eq. (6.11)) alters the domain of frequency features by expanding the frequency spectrum, and hence it extracts/maps complementary features instead of wanted features. To this end, Eq. (6.11) initially interpolates the decimated inputs in lower scales in order to make the domain of wanted



(a-c) Feature matching for different cases (d) Potential of the transformation

Figure 6.4: Memory Efficient Scale-space Learning: (a-c) If a filter is optimized for a particular signal, then the same signal scaled will not produce a similar response, unless the signal is matched to the original version. (b) Feature-matching is performed in a standard network (Zhou *et al.* (2019)) and ours. Albeit a simple technique, both networks yield superior performance.

features identical for all scales. Finally, Eq. (6.11) decimates the network-output which scales down the discrepancies in accordance with Eq. (6.8). This technique, though a simple one, extends the applicability of typical deep networks for different resolutions other than the one trained for (as shown by experiments in Fig. 6.7(a)).

In what follows, we justify our claim that the transformation in Eq. (6.11) is better than Eq. 6.10 for lower image-scales. We first consider the case of convolution, which maps an input tensor $\mathbf{f}$ with depth d to a tensor $\mathbf{g}$ with depth d' as

$$\mathbf{g}^j = \sum_{i=1}^{d} \mathbf{f}^i * \mathbf{h}^{\{i,j\}} \quad : \; 1 \leq j \leq d', \tag{6.12}$$

where $\mathbf{g}^j$ (the jth layer of $\mathbf{g}$) is produced by aggregating the convolution of $\mathbf{f}^i$ and filters $\mathbf{h}^{\{i,j\}}\ \forall i$. In particular, $\mathbf{f}$ in a decoder stage is obtained by concatenating feed-forward encoder features and decoder features (Zhou *et al.*, 2019), whereas the standard residual block (i.e., $\mathbf{g}^j = \sum_{i=1}^{d} \mathbf{f}^i * \mathbf{h}^{\{i,j\}} + \mathbf{f}^i$) and atrous spatial pyramid pooling (i.e., $\mathbf{g}^j = \sum_{i=1}^{d}\sum_{k=1}^{p} \mathbf{f}^i * \mathbf{h}^{\{i,j,k\}}$) (Chen *et al.*, 2017) has equivalent convolution filter as $\overline{\mathbf{h}}^{\{i,j\}} = \mathbf{h}^{\{i,j\}} + \delta(i,j)$ and $\overline{\mathbf{h}}^{\{i,j\}} = \sum_{k=1}^{p} \mathbf{h}^{\{i,j,k\}}$, respectively.

Remark 2: Convolution filters optimized to map certain frequency features in the fine-scale (Eq. (6.9)) need *not* map to matching features in lower scales (Eq. (6.10)), whereas with the transformation in Eq. (6.11) map to matching features.

Justification: The mappings in both Eqs. (6.10) and (6.11) employ identical convolution filter in fine-scale as well as lower scales (via the same $T(\cdot)$), but the difference is that for lower-scales the former gets a decimated input, whereas the latter gets an interpolated version of the decimated input (i.e., with the same resolution of fine-scale). In frequency domain, convolution mapping (Eq. (6.12)) in the fine-scale is

$$\mathbf{G}^{j}(\omega) = \sum_{i=1}^{d} \mathbf{F}_{i}(\omega)\mathbf{H}^{\{i,j\}}(\omega), \tag{6.13}$$

where $\mathbf{H}_{\{i,j\}}(\omega)$ is the frequency spectrum of convolution filter. The spectrum $\mathbf{H}_{\{i,j\}}(\omega)$ is typically non-uniform (Xu *et al.*, 2014) and hence maps/extracts frequency features in a particular way (e.g., in Fig. 6.4(a), the filter extracts wanted features while suppressing unwanted features). Now we consider the case of using the same operation of Eq. (6.13) for lower scales (i.e., Eq. (6.10)). This results in

$$\mathbf{G'_D}^{j}(\omega) = \sum_{i=1}^{d} \mathbf{F}_{i}(\omega)_{\downarrow D}.\mathbf{H}^{\{i,j\}}(\omega), \tag{6.14}$$

Note that the spectrum $\mathbf{H}_{\{i,j\}}(\omega)$ now maps/extracts frequency features in a different way as compared to Eq. (6.13) (due to expanded input spectrum). Resultantly, it can map to *non*-matching features in lower scales (e.g., see Fig. 6.4(b) where the wanted features get suppressed). Next, we consider the proposed transformation (Eq. (6.11)):

$$\mathbf{G_D}^{j}(\omega) = \sum_{i=1}^{d} \mathbf{F}_{i}(\omega)_{\downarrow D\uparrow D}.\mathbf{H}^{\{i,j\}}(\omega). \tag{6.15}$$

Note that we do not consider the overall decimation of Eq. (6.11) as it is *not* present after each convolution stage (but *only once* at the network-output). In Eq. (6.15), frequency features of input coincide with that of the fine-scale input in frequency domain (due to inverse-scaling or $\uparrow D$). Resultantly, the spectrum $\mathbf{H}_{\{i,j\}}(\omega)$ maps/extracts frequency features in the same way as compared to Eq. (6.13), and hence map to matching features (except at those high-frequency features lost due to anti-aliasing) as warranted by scale-space network. This is illustrated in Fig. 6.4 (compare Figs. (b) and (c)).

Remark 2 establishes that our transformation in lower-scales maps to matching features for convolution stages. We next show that this transformation is favourable in other network-stages as well, which ensures its applicability for lower-scales.

Generalization of Remark 2: The non-linearities, such as bias, leaky ReLU, etc., optimized for fine-scale is applicable to lower scales under the input-feature transformation of Eq. (6.11) (i.e., interpolating the decimated input).

Justification: The non-linear transformations such as bias, leaky ReLU, etc., are point-wise operations and these functions are continuous (in particular, bias is a constant point-wise offset and leaky ReLU is continuous though *not* differentiable at origin). Further, as input images have predominantly low-frequency components, which is supported by the natural image priors such as total variation (Perrone and Favaro, 2014), l_0 in image-gradients (Xu *et al.*, 2013), dark-channel (Pan *et al.*, 2016), etc., input features to these non-linearities are predominantly low-pass. Therefore, the point-wise values of decimated (by D) and then interpolated (by D) version will have closer intensity values as that of the fine-scale image (as shown in Fig. 6.4(b)). Hence, due to closer point-wise values and continuous characteristic of non-linear functions, the response of the non-linearities to decimated-and-then-interpolated input must be closer to the corresponding response of the fine-scale input. Finally, under the assumption that anti-aliasing and anti-imaging filters are ideal, intermediate stages of decimation (in encoder) and interpolation (in decoder) map to matching features as that of fine-scale network. Therefore, with the proposed transformation of Eq. (6.11), fine-scale network in lower scales map to matching features as that of fine-scale (as required in a scale-space network).

We next experimentally validate our proposed transformation. First, we evaluate the state-of-the-art DL deburring network (Zhou *et al.*, 2019) for different image-scales over its dataset (in Fig. 6.4(b)). It is clear from the plot that the performance drastically reduces with lower scales (validating ineffectiveness of Eq. (6.10)). Next we introduce our transformation (Eq. (6.11)) to the same network and repeat the same experiment. It is evident from Fig. 6.4(b) that, though a simple modification, our proposed approach significantly improves the performance over different scales. Also note that the PSNR curves in the latter case is decaying quite slowly as compared to the former; this decay can be attributed to our assumption of ideal anti-aliasing and anti-imaging filter, i.e., a network trained only for the fine-scale need not optimize the filters to perform ideal anti-aliasing and anti-imaging in decimation and interpolation stages. To this end, we train our network in a scale-space approach by employing the fine-scale network with the transformation for lower scales with *tied* network-weights (see Sec. 6.6.1), and optimize deblurring performances averaged over *all* image-scales (Eq. (6.19)-(6.20)). This

training strategy demands the fine-scale network to realize effective anti-aliasing and anti-imaging through trainable filters, in order to improve the performances in lower scales as well. Figure 6.4(b) plots the performance of our network for different image-scales for the same dataset, which reveals a significant improvement in performance drop over lower scales and hence, the importance of our training strategy.

6.4 Image-dependent, Space-variant Deblurring

In this section, we focus on yet another issue that stems from directly employing the network-modules of the state-of-the-art DL deblurring method (Zhou *et al.*, 2019) for coherent-fusion (Sec. 6.2.1) and adaptive scale-space (Sec. 6.3.1). An effective dynamic-scene deblurring network requires mapping that varies with spatial locations (with varying receptive fields), and that adaptively varies with different blurred images (Zhang *et al.*, 2018; Purohit, 2020). For instance, consider a scenario of static camera, and two dynamic objects at different depths moving with the same velocity. Here, the static background exhibits no motion blur, whereas the nearer object exhibits more blur than the farther one (due to parallax (Zhou *et al.*, 2019)). Therefore, an effective deblurring network warrants an identity mapping for the background and non-identity mapping for the dynamic objects, but with a relatively larger receptive fields for the nearer object. Also, positions of those dynamic objects can vary for different images, and hence the mappings need to be image-dependent. However, the DL deblurring network of (Zhou *et al.*, 2019) does *not* admit such a space-variant image-dependent (SvId) mapping.

One key component that leads to the performance improvement in the state-of-the-art DL deblurring network (Zhou *et al.*, 2019) is the context module (used as feature mapping in the DL network in Fig. 6.1), which is a slightly modified version of atrous spatial pyramid pooling (ASPP) (Chen *et al.*, 2017). This improvement is because the ASPP offers a good trade-off between accurate localization (small receptive field) and context assimilation (large receptive field). Note that ASPP has also been adopted for a broader set of tasks, such as semantic segmentation, object detection, visual question answering, and optical flow; however it does not support SvId mapping. Owing to the presence of both small and large receptive fields in ASPP, we extend the ASPP module to instil the SvId mapping in our deblurring network.

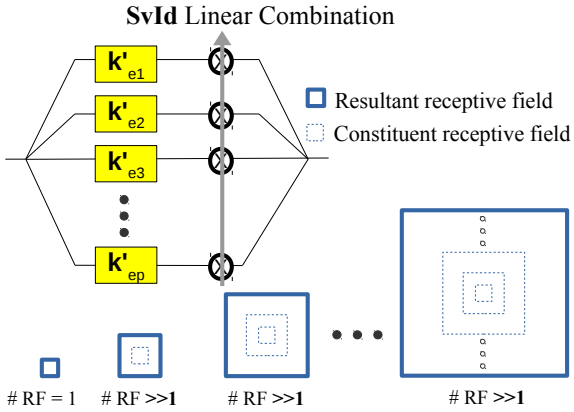


Figure 6.5: Space-variant, image-dependent (SvId) atrous spatial pyramid pooling (ASPP): The ASPP Chen *et al.* (2017) produces *only one* resultant filter (RF) with receptive field as that of the constituent filter with maximum field-of-view (in Fig., RF in the far-right). As this filter realization is *same* for all spatial coordinates *irrespective* of input, it does *not* admit SvId property. SvId-ASPP has the freedom to produce numerous RFs with receptive field as that of any constituent filter through SvId linear combinations of filtered outputs in individual branches.

First we briefly discuss about the ASPP module (Chen *et al.*, 2017). As shown in Fig. 6.5(a), ASPP probes an input with filters at multiple sampling rates and different fields-of-views. This is efficiently implemented using multiple parallel atrous convolutional layers with different sampling rates. As in today's convolutional neural networks, atrous convolutional layer also employs spatially small convolution kernels (typically 3×3) in order to keep both computation and number of parameters contained. But the difference in atrous convolution layer is that its filter is associated with a rate $r \geq 1$, which introduces $r-1$ zeros between consecutive filter values, thereby effectively enlarging the kernel size of a $k \times k$ filter to $(k-1)(r-1) \times (k-1)(r-1)$ without increasing the number of parameters or the amount of computation. Mathematically, the mapping of an input $\mathbf{x}$ in an ASPP module is given as

$$\mathbf{y} = \mathbf{x} * (\mathbf{k}_{e1} + \mathbf{k}_{e2} + \cdots + \mathbf{k}_{ep}); \qquad \mathbf{k}(m,n) = \mathbf{k}_{e1} + \mathbf{k}_{e2} + \cdots + \mathbf{k}_{ep}, \tag{6.16}$$

where $\mathbf{k}_{ei}$ is the filter at ith branch with sampling rates ei such that $e1 = 1$ and $ep > \cdots > e2 > e1$, and $\mathbf{k}(m,n)$ is the resultant filter at spatial coordinate (m,n). Clearly, even though the individual filters $\mathbf{k}_{ei}$ possess diverse receptive fields, there exists only one resultant filter realization in Eq. (6.16) (i.e., $\mathbf{k}(m,n)$) and hence a single receptive field (as that of the filter $\mathbf{k}_{ep}$). Also, the filter realization is identical in all spatial coordinates and is irrespective of input. Therefore, the ASPP module (Chen *et al.*,

2017) *cannot* admit an SvId mapping, which is desirable for motion deblurring.

Next we propose a simple but effective modification to ASPP to enable the SvId property. As shown in Fig. 6.5(b), we introduce in each parallel branch of ASPP a space-variant, input-dependent bilinear mask, which modifies the mapping in Eq. (6.16) as

$$\mathbf{y} = (\mathbf{x} * \mathbf{k'}_{e1}) \odot \mathbf{W}_{e1} + (\mathbf{x} * \mathbf{k'}_{e2}) \odot \mathbf{W}_{e2} + \cdots + (\mathbf{x} * \mathbf{k'}_{ep}) \odot \mathbf{W}_{ep} \tag{6.17}$$

where $\mathbf{k'}_{ei}$ is the filter at ith branch and $\mathbf{W}_{ei},\ 1 \leq i \leq p$ are non-negative SvId masks which sum to unity ($0 \preceq \mathbf{W}_{ei} \preceq 1$ and $\sum_{i=1}^{p} \mathbf{W}_{ei}(m,n) = \mathbf{1}$). The masks are produced by a mask-generating network similar to the one employed for view-aggregation (Zhou *et al.*, 2019; Gong *et al.*, 2018; Chen *et al.*, 2018) with slight modifications to allow for more than two masks, as discussed in Sec. 6.6. Taking into consideration some desired properties for dynamic scene deblurring, we slightly modify the SvId as follows. First, inverse filters for deblurring may require a very large receptive fields (Zhang *et al.*, 2018; Purohit, 2020), for which we cascade multiple stages (exactly three) of the given module. Second, as discussed previously, deblurring may also require unity receptive field for identity mapping, for which we consider the identity mapping as the first branch of each stage (instead of 3×3 filter in ASPP).

Remark 3: The modified ASPP admits space-variant, image-dependent (SvId) mapping with *diverse* receptive fields wherein each receptive field (other than 1, i.e., the trivial identity mapping) admits *numerous* filter realizations or mappings.

Justification: The resultant filter in Eq. (6.17) can be represented as

$$\mathbf{k'}(m,n) = \mathbf{W}_{e1}(m,n) \cdot \mathbf{k'}_{e1} + \mathbf{W}_{e2}(m,n) \cdot \mathbf{k'}_{e2} + \cdots + \mathbf{W}_{ep}(m,n) \cdot \mathbf{k'}_{ep} \tag{6.18}$$

As the masks $\mathbf{W}_{ei}(m,n),\ 1 \leq i \leq p$ (in Eq. (6.17)), that linearly combine filter basis $\mathbf{k'}_{ei}$, can vary with spatial-coordinates (m,n), our modified ASPP can produce different filters at different spatial locations. Further, as those masks are function of input blurred image (via mask-generating network), the modified ASPP becomes image-dependent as well, and hence admits SvId mapping. Denoting the receptive field of a filter $\mathbf{k'}$ as $R(\mathbf{k'})$, the receptive field for the modified ASPP at a coordinate (m,n) becomes $\max R(\mathbf{k'}_{ei})$ if $\forall j > i, \mathbf{W}_{ej}(m,n) = 0$, and hence the image-dependent masks diversi-

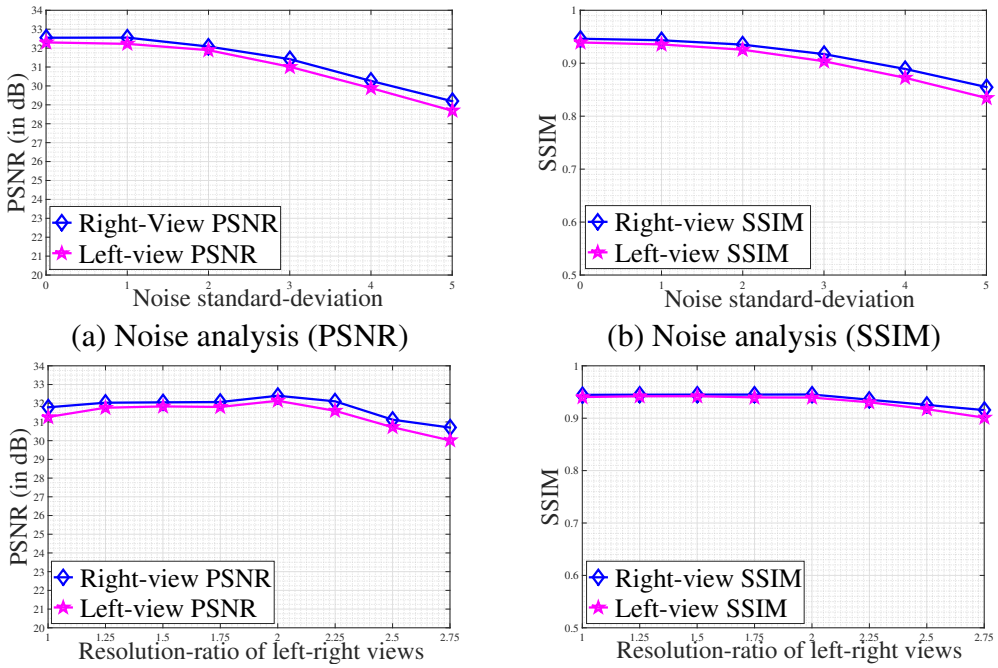
fies the receptive field (Fig. 6.5(b)-bottom). Further, a resultant filter with receptive field $R(\mathbf{k'}_{ei})$ can be realized by numerous linear combination of filters $\mathbf{k'}_{ej} : j \leq i$, specifically, in Eq. (6.18) filters obtained by all possible combinations for $\mathbf{W}_{ek}(m,n) : k < i$ with $\mathbf{W}_{ei}(m,n) > 0$ and $\mathbf{W}_{ek}(m,n) = 0 : k > i$. Also, note that even though the individual filter-basis ($\mathbf{k}_{ei}$) is sparse, this linear combination of multiple filter-basis can produce dense filter for a given receptive field. Finally, a cascade of M such modules for deblurring retains the SvID property, and further increases the receptive field to $\{\sum_{i=1}^{M} R(\mathbf{k'}^{(i)})\} - M + 1$, where $\mathbf{k'}^{(i)}$ is the effective filter (Eq. (6.18)) at the ith stage (because $R(\mathbf{k_1} * \mathbf{k_2}) = R(\mathbf{k_1}) + R(\mathbf{k_2}) - 1$ (Oppenheim and Schafer, 2014)).

6.5 Analysis and Discussions

6.5.1 Sensitivity to Image-noise and Resolution-ratio

To analyse the performance dependence due to image noise, we introduce *independent* additive white Gaussian noise ($0 \leq \sigma \leq 5$ pixels) to blurry images in the two views. Figs. 6.6(a-b) respectively plot the mean PSNRs and SSIMs of left-right view deblurred images for different noise levels. Note that over the entire standard-deviation range the mean PSNRs for deblurred image is over 29 dB, and difference between the PSNRs is within 0.7 dB. This clearly reveals the noise-robustness of our method, which can be primarily attributed to the process of noise-addition during training.

We next analyse the performance of our network for different resolution-ratios. We have considered resolution-ratios from 1:1 to 1:2.75, which span a wide range of today's unconstrained DL-smartphones. The PSNRs and SSIMs of left-right view deblurred images for different resolution-ratios are plotted in Figs. 6.6(c-d). Note that resolution-ratios and focal length ratios are synonymous, and therefore the findings of this analysis holds good for diverse focal length ratios as well. It is evident from the figure that though our network is only trained for 1:2 case (where both the metrics in Fig. 6.6 have a maxima), the performance degradation is quite less over the other resolution-ratios, and further, the view-consistency is well-preserved over the entire range. This reveals the generalization capability of the coherent fusion module in channelling rich complementary information for deblurring for diverse resolution and focal length ratios.



(a) Noise analysis (PSNR) (b) Noise analysis (SSIM)

(c) Scale analysis (PSNR) (d) Scale analysis (SSIM)

Figure 6.6: Analysis: (a-b) Performance dependence with respect to image noise. (c-d) Effect of resolution-ratio on deblurring performance.

6.5.2 Ablation Studies

To study the effectiveness of the three proposed modules, we replace them with analogous existing modules and retrain using the same strategy. Table 6.1 reveals that our method performs best when all our modules are present. To analyse the effect of coherent fusion module, we considered only the view aggregation network at node {C,D} in Fig. 6.1 (as employed in constrained DL-BMD network (Zhou *et al.*, 2019)). For this case, note that the deblurring performances (in terms of PSNR and SSIM in Table 6.1) of left- and right-views deviate by a large margin, and hence fails to preserve view-inconsistency. This implies that information fusion seldom happens *without* coherent fusion module, and in this case network tries to primarily improve the right-view PSNR (which is easy to accomplish due to its rich features), but neglects the left-view (where PSNR improvement is difficult to achieve due to more degradation). To study the effect of adaptive scale-space, we considered only our fine-scale network (as in (Zhou *et al.*, 2019)) for unconstrained DL configuration as well. Note in table 6.1 that the mean absolute error (MAE) for disparity for different unconstrained cases are greater than one pixel which reveals the inadequacy of single-scale network in preserving the epipolar geometry in the deblurred image-pair. Also, note that our multi-scale network

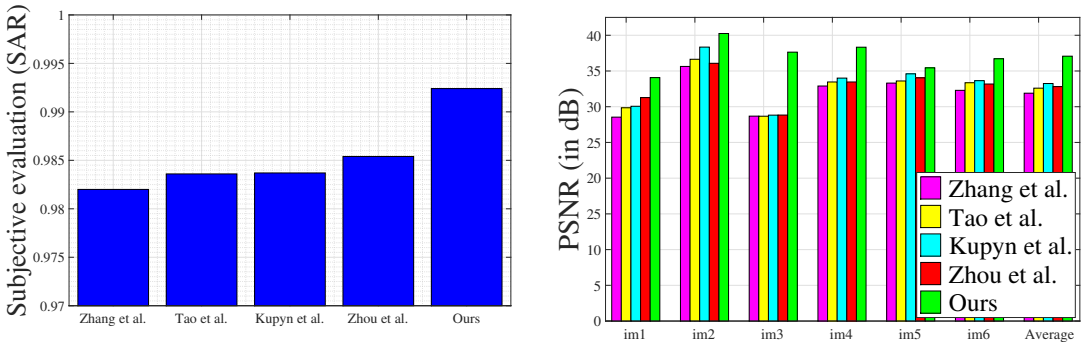
Table 6.1: Quantitative evaluations: SA - Scale adaptive; CF - Coherent fusion; BS- Bootstrap. (First/Second)

Method	Xu et al. Xu and Jia (2012)	Mohan et al. Mohan *et al.* (2019)	Tao et al. Tao *et al.* (2018)	Kupyn et al. Kupyn *et al.* (2018)	Zhang et al. Zhang *et al.* (2019)	Zhou et al. Zhou *et al.* (2019)	Ours	Ours (BS)	Ours W/o SA	Ours W/o CF	Ours W/o SvID
Unconstrained DL Case 2: Exposure 4:3											
MAE	1.929	2.273	1.9704	1.923	2.0488	1.9328	**0.8465**	0.8533	1.9718	0.8572	0.8318
PSNR	16.167	25.169	26.536	26.743	26.406	26.437	**30.581**	**30.560**	30.118	27.11	28.32
PS:Offset	0.8390	1.0600	6.4970	5.7810	5.6060	5.6360	**0.8450**	**5.1810**	0.8971	5.2181	0.8677
SSIM	0.471	0.816	0.860	0.862	0.858	0.863	0.917	0.913	0.915	0.894	0.899
SS:Offset	0.0630	0.0130	0.0860	0.0720	0.0730	0.0740	0.0070	0.0350	0.0081	0.0581	0.0083
Unconstrained DL Case 3: Exposure 3:5											
MAE	1.894	3.021	2.2524	2.2204	2.3518	2.2444	**1.0043**	1.0066	2.2731	1.0076	1.0068
PSNR	17.465	26.348	26.593	26.536	26.431	26.040	**28.801**	**28.724**	28.402	26.181	27.65
PS:Offset	1.1230	1.9870	4.1550	4.1280	3.2460	3.3640	**1.0050**	**4.1380**	1.1139	3.254	1.0178
SSIM	0.559	0.876	0.862	0.858	0.858	0.868	0.904	0.901	0.898	0.885	0.891
SS:Offset	0.0790	0.0140	0.0570	0.0590	0.0470	0.0440	0.0090	0.0310	0.0131	0.0413	0.0454
Unconstrained DL Dataset 7: Exposure 1:1											
MAE	0.941	1.215	0.8869	0.8794	0.9921	0.8672	**0.7380**	0.7718	0.7802	0.7591	0.7328
PSNR	17.047	26.815	26.984	26.906	25.854	29.198	**32.052**	**31.006**	31.047	29.187	28.180
PS:Offset	1.4260	1.8090	1.2520	1.3060	1.2960	3.9320	**0.2580**	**2.3430**	0.2577	3.897	0.357
SSIM	0.477	0.854	0.861	0.855	0.828	0.892	0.905	0.904	0.905	0.889	0.857
SS:Offset	0.0940	0.0300	0.0330	0.0310	0.0330	0.0480	0.0070	0.0350	0.0065	0.0493	0.0073
Time (S)	2160	1630	0.507	0.39	0.5237	0.31	0.34/scale	0.34/scale	0.34/scale	0.33/scale	0.31/scale
Size (M)	-	-	8.06	5.09	21.69	4.83	5.98	5.98	5.98	5.90	5.93

performs quite well for unconstrained DL set-up, and our single-scale network is adequate for constrained DL configuration. Finally, we analyse our SvId ASPP module for deblurring by replacing it with analogous ASPP module (Chen *et al.*, 2017), i.e., three cascaded stages of ASPP but *without* SvId mapping. It is evident from Table 6.1 that our SvId ASPP module significantly boost the deblurring performance as compared to the standard ASPP, which highlights the importance of SvId mapping in motion deblurring.

6.5.3 View-consistency Analysis

Subjective Evaluation of View-consistency: To quantify view-consistency in DL deblurring, we consider the metric (Chen *et al.*, 2013) which quantify binocular rivalry, i.e., a perceptual effect that occurs when the two eyes view mismatched images at the same retinal location(s). It describes the quality of stereoscopic images that have been



(a) Quality assessment of stereo (b) Post-processing capability

Figure 6.7: Analysis: (a) Subjective evaluation using "Full-reference quality assessment of stereo-pairs accounting for rivalry (SAR)" Chen *et al.* (2013). (b) DL super-resolution Wang *et al.* (2019*b*) is performed on different deblurred results. Clearly, the performance significanly drop for view-*in*consistent inputs.

afflicted by possibly asymmetric distortions, where a higher score is attained for good DL-images, with the highest score unity. Figure 6.7(a) compares this metric for mature deblurring methods on the unconstrained DL blur dataset (Sec. 6.6), which yet again proves the effectiveness of our proposed method for stereoscopic-vision applications.

View-consistency for Extended DL application: Motion deblurring is an important preprocessing step in many computer vision applications that are not designed to work well in the presence of motion blur. Here, we employ various deblurring methods as a preprocessing stage for the state-of-the-art DL super-resolution method (Wang *et al.*, 2019*b*). In Fig. 6.7(b), we quantify the performance of (Wang *et al.*, 2019*b*) for various deblurred results. (As super-resolved ground truth is not available, we use the super-resolved DL clean images as the reference for all the methods). Notably, there is a significant performance drop in competing deblurring methods. In particular, our method outperforms the second-best approach by an average PSNR of 4 dB. Figure 6.8 provides super-resolved left-right patches for visualization. This performance drop is possibly because the super-resolution work (Wang *et al.*, 2019*b*) is designed for view-consistent inputs as is in most DL-based works (Jeon *et al.*, 2018; Chen *et al.*, 2018; Gong *et al.*, 2018) (but their codes are not available); however, as evaluated previously, the competing deblurring methods fails to produce view-consistent inputs.

View-*in*consistency via Bootstrapping: There exist DL applications that are *not* meant for stereoscopic vision (e.g., monocular-based), and in this case, instead of balancing the deblurring performance in the two-views for view-consistency, what is desirable is maximum deblurring performance in individual views. But due to training with the

(a) Kupyn et al. (b) Tao et al. (c) Zhang et al. (d) Zhou et al. (e) Ours

Figure 6.8: Qualitative Results: Applicability of different deblurring methods for DL super-resolution (Wang *et al.*, 2019*b*). As compared to the competing deblurring methods, our method is able to produce the desired view-consistent super-resolution results.

self-supervision costs $L_{RR} + L_{LR}$ in the coherent fusion module, our method seldom raises the performance in one-view *unconditionally* while ignoring the other-view. So as to enable our network to allow those kind of applications, we propose a bootstrapping approach where our trained network is *fine-tuned without* the self-supervision cost. Since we are starting with a good identical-performance in the two-views, there exists a superior view-aggregation (Zhou *et al.*, 2019) provided by the already improved highly-degraded input image. This allows unconditionally improving the deblurred image with more image features, while maintaining the superior performance of the other. Table 6.1 and Fig. 6.11) provides our bootstrapped results as well, which clearly reveals that our approach outperforms all other methods in this aspect too.

6.5.4 Inadequacy of DL Prior for depth-consistency

Remark 1:Dual-lens prior in (Mohan *et al.*, 2019) is *not* applicable for scene-consistent dynamic-scene motion deblurring.

Justification: First, we summarize the working of DL-prior. According to Claim 1 in (Mohan *et al.*, 2019), there exist multiple valid solutions of MDF-pairs (that quantify camera-motion) for image-pair in the left-right views, wherein some solutions produce *scene-inconsistent* disparities and others produce scene-consistent disparities. Assuming the motion blur is due to *only* camera motion, the DL prior in (Mohan *et al.*, 2019) promotes a valid MDF-pair, but which can only resolve the deblurred image upto an unknown pose-variation of the actual scene (which is denoted by $\mathbf{R}_n$).

To aid a fair comparison, we allow the following relaxations in (Mohan *et al.*, 2019): (a) Despite (Mohan *et al.*, 2019) restricts to only 3D rotation-changes due to only

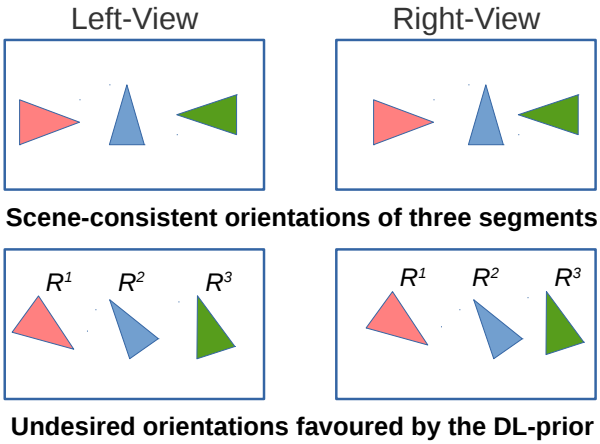


Figure 6.9: Effect of DL-prior of (Mohan *et al.*, 2019) on dynamic scenes: Due to possibly different relative-motions in individual dynamic objects, the pose-ambiguity of DL-prior (Mohan *et al.*, 2019) need not be identical in different objects. The figure shows the case of different in-plane rotation ambiguity ($\{\mathbf{R}^1, \mathbf{R}^2, \mathbf{R}^3\}$) in three different objects, which clearly derails the scene-consistency as required for most DL applications.

camera-motion assumption, we assume that (Mohan *et al.*, 2019) handles 3D translations as well, which is required to model dynamic scenes (Pan *et al.*, 2017; Sellent *et al.*, 2016). (b) Even though extending (Mohan *et al.*, 2019) for dynamic scene deblurring is non-trivial, as it involves complex pipeline which include coherently segmenting dynamic objects in the two views and stitching different segments with negligible artifacts in seams (Pan *et al.*, 2017; Sellent *et al.*, 2016; Xu and Jia, 2012), we assume that such a pipeline exists for (Mohan *et al.*, 2019). The ambiguity due to $\mathbf{R}_n$ causes a relative change in scene-orientation in both the views. Though it does *not* produce any issues for the case of static scene (as it renders the entire scene to have an arbitrary pose-change), this is not the case for dynamic scenes where each dynamic object can have (relative) independent motion. Let there be x dynamic objects, then the DL-prior on individual segments produces n independent pose-ambiguity (say $\mathbf{R}_n^i,\ 1 \leq i \leq x$) as the MDFs in each segments can be independent and hence unrelated. Resultantly, it renders individual objects in the scene to have different pose-changes, e.g., as illustrated in Fig. 6.9, a horizontally moving object, with respect to background, can be rendered moving diagonally due to an in-plane rotational ambiguity (as considered in Fig. (2) of (Mohan *et al.*, 2019)), which clearly distorts the scene-consistent disparities.

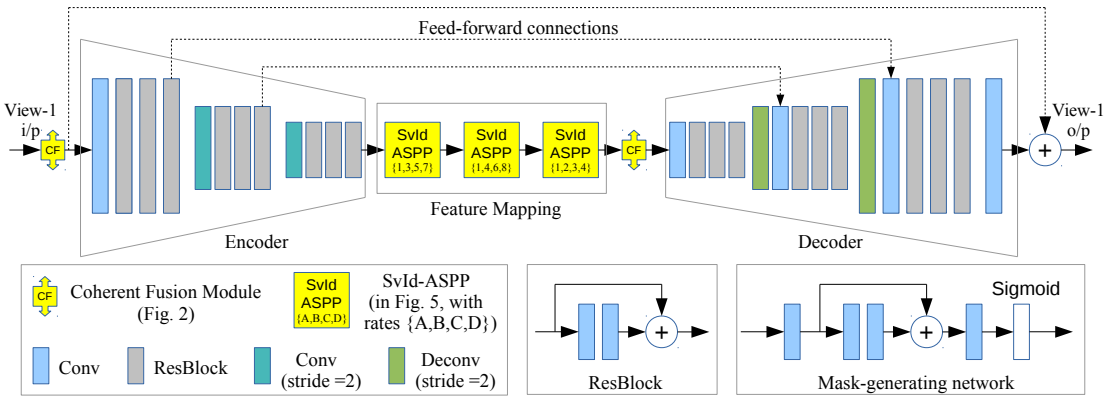


Figure 6.10: Network Architecture: Our fine-scale network consists of a three-stage encoder/decoder, with SvId for feature mapping and coherent fusion module to balance signals in the two-views. The same network is shared for both views.

6.6 Experiments

In this Section, we provide more details of our network (Fig. 6.10) and dataset, and demonstrate our method's effectiveness in diverse unconstrained DL settings.

Network Details: We consider the standard DL-deblurring encoder and decoder architectures of (Zhou *et al.*, 2019), i.e., three stages each for encoder and decoder where each stage consists of 3×3 convolution layer and three Resblocks. We employ the disparity estimation network of (Zhou *et al.*, 2019). We do note that since our method requires *only* disparity-maps (unlike (Zhou *et al.*, 2019) which warrants in addition network-specific disparity-features), any disparity estimation methods, whether conventional or deep learning based, can be employed. Our SvId ASPP consists of three stages, where receptive field of filter-kernels at individual stages is selected as $\{1, 3, 5, 7\}$, $\{1, 4, 6, 8\}$, and $\{1, 2, 3, 4\}$. To control possible intensity variations between the left- and right-view images, we normalize the mean and standard deviation of left-view input of the coherent fusion module to that of its right-view input. To create bilinear masks for coherent fusion and SvId filter module, we consider a light-weight network as similar to (Zhou *et al.*, 2019); the difference from (Zhou *et al.*, 2019) is that, as our method requires *multiple* output masks for the filter-module, we additionally consider three Resblocks (instead of one) at the middle and soft-max layer at the end (instead of sigmoid) to normalize multiple masks.

For training our deblurring network, apart from the self-supervision costs of coherent fusion module (Eq. (6.5)), we consider two supervision costs to measure the

Table 6.2: Data distribution

Type	Case 1 (1:3)	Case 2 (4:3)	Case 3 (3:5)	Case 4 (3:1)	Case5 (3:4)	Case 6 (5:3)	Case 7 (1:1)	Total DL-images
Training	4,159	4,403	4,600	4,159	4,403	4,600	17,319	43,643
Testing	837	803	805	837	803	805	3,318	8,208

difference between the deblurred images ($\{\hat{\mathbf{F}}^L, \hat{\mathbf{F}}^R\}$) and sharp images ($\{\mathbf{F}^L, \mathbf{F}^R\}$) for the left-right views. The first one is an objective cost based on the standard MSE loss:

$$L_{mse} = \frac{1}{S}\sum_{l=1}^{S}\frac{1}{2CMN}\sum_{k\in\{L,R\}}\|\hat{\mathbf{F}}_l^k - \mathbf{F}_l^k\|_2, \tag{6.19}$$

where S is the number of scale-space network-levels, $\mathbf{F}_l$ denotes images at level l, and C, M, and N are dimensions of image. The second cost is the perceptual loss employed in (Zhou *et al.*, 2019), which is the l_2-norm between conv3-3 layer VGG-19 (Simonyan and Zisserman, 2015) features of deblurred images and sharp images:

$$L_{vgg} = \frac{1}{S}\sum_{l=1}^{S}\frac{1}{2C_vM_vN_v}\sum_{k\in\{L,R\}}\|\Phi_{vgg}(\hat{\mathbf{F}}_l^k) - \Phi_{vgg}(\mathbf{F}_l^k)\|_2, \tag{6.20}$$

where Φ_{vgg} is the required VGG mapping, and C_v, M_v, and N_v are dimensions of VGG features. Denoting the normalized coherent fusion cost in Eq. (6.5) as L_{cf}, our overall loss function is empirically selected as $0.4L_{cf} + 0.5L_{mse} + 0.1L_{vgg}$.

Dataset Generation: Since unconstrained dynamic scene blur has not been hitherto addressed, we created a dataset with diverse exposure (like the constrained case in (Zhou *et al.*, 2019)). We follow a similar procedure as that of (Zhou *et al.*, 2019) in creating DL blur dataset, which we briefly summarize (to highlight our differences). As typically followed in single-lens dynamic scene blur generation (Nah *et al.*, 2017), a blurry image is generated by averaging a sharp high frame rate sequence to approximate a long exposure. The dataset in (Zhou *et al.*, 2019) consists of a wide variety of scenarios, both indoor and outdoor, which include diverse illumination, weather, and motion patterns. To increase the video frame rate, it employs a fast and high-quality frame interpolation method (Niklaus *et al.*, 2017) and generate different blur-sizes by: (a) *considering equal number of synchronized set of DL image-pairs in both left- and right-views* (i.e., identical and fully-overlapping exposures) (b) *ground truth frame is*

temporally centered on the exposure time. The major difference of our data generation is regarding the points (a) and (b). For point (a), following (Mohan *et al.*, 2019), we do *not* consider synchronization between DL image-pairs, but allow a random non-zero exposure-intersection (as shown in Fig. 6.1(a)). Further we allow different number of DL image-pairs in the two-views in the ratio $1:1$, $1:3$, $3:1$, $3:4$, $4:3$, $3:5$, and $5:3$ (totalling seven exposure-cases). For all cases, following (Mohan *et al.*, 2019), exposure-overlap is randomly sampled from 10-100% with standard resolution 1:2. Table 6.2 provides the distribution of data samples. For point (b), we consider the ground truth frame to be temporally centered on the intersection of the left-right view exposure times to ensure ground truth is view-consistent (unlike the case of (Zhou *et al.*, 2019), as illustrated in Fig. 6.1(a)).

Comparisons: We consider all standard DL BMD methods, i.e., (Zhou *et al.*, 2019) that handle unconstrained DL for static scenes and (Mohan *et al.*, 2019; Xu and Jia, 2012) that handle constrained DL for dynamic scenes. We also include state-of-the-art single-lens methods to represent scale-space approach (Tao *et al.*, 2018), generative models (Kupyn *et al.*, 2018), and patch-based approach (Zhang *et al.*, 2019). We also considered for evaluation constrained DL dataset of (Zhou *et al.*, 2019) and unconstrained DL static-scene dataset of (Mohan *et al.*, 2019). For quantitative evaluation, we consider the metrics mean absolute error (MAE) for disparity, and for deblurring, PSNR and SSIM in the view with relatively more degradation and respective offsets in the other view. For qualitative evaluation, we provide images with left-right patches.

Evaluations: Our method can seamlessly address unconstrained DL dynamic scene deblurring under diverse exposures, exposure-overlap, and resolution-ratio; unconstrained static scene deblurring as in (Mohan *et al.*, 2019), and constrained DL deblurring as in (Zhou *et al.*, 2019) (where the last two are special cases of the first problem). Table 6.1 provides the quantitative evaluation of deblurring performance for different unconstrained DL settings. Scene-consistent disparities in unconstrained DL deblurring can be judged by MAE (lower values are better). From Table 6.1, it is evident that the competing methods produce a large discrepancy (e.g., MAE above two pixels in the exposure-case 3:5). Further, a higher PSNR with a small offset implies good view-consistent deblurring. It is clear from the table as well as qualitative examples (in Fig. 6.11) that our method exhibits good performance. Averaging over all the seven cases, our method has a PSNR of 30.653 dB with an offset 0.706, whereas the next-best

1(a) Input 1(b) Xu et al. 1(c) Mohan et al.

1(d) Tao et al. 1(e) Kupyn et al. 1(f) Zhang et al.

1(g) Zhou et al. 1(h) Ours 1(i) Ours bootstrapped

2(a) Input 2(b) Xu et al. 2(c) Mohan et al.

2(d) Tao et al. 2(e) Kupyn et al. 2(f) Zhang et al.

2(g) Zhou et al. 2(h) Ours 2(i) Ours bootstrapped

Figure 6.11: Comparisons for unconstrained DL exposure-cases 3:5 and 4:3. Our method is able to produce view-consistent results as compared to the competing methods. After bootstrapping in (i), our method produces good view-*in*consistent result as well (see patches from *both* views).

1(a) Input 1(b) Xu et al. 1(c) Mohan et al.

1(d) Tao et al. 1(e) Kupyn et al. 1(f) Zhang et al.

1(g) Zhou et al. 1(h) Ours 1(i) Ours bootstrapped

2(a) Input 2(b) Xu et al. 2(c) Mohan et al.

2(d) Tao et al. 2(e) Kupyn et al. 2(f) Zhang et al.

2(g) Zhou et al. 2(h) Ours 2(i) Ours bootstrapped

Figure 6.12: Comparisons for unconstrained DL exposure-cases 5:3 and 3:4. Note that, as compared to the competing methods, our method produces superior deblurring results with good view-consistency.

1(a) Input 1(b) Xu et al. 1(c) Mohan et al.

1(d) Tao et al. 1(e) Kupyn et al. 1(f) Zhang et al.

1(g) Zhou et al. 1(h) Ours 1(i) Ours bootstrapped

2(a) Input 2(b) Xu et al. 2(c) Mohan et al.

2(d) Tao et al. 2(e) Kupyn et al. 2(f) Zhang et al.

2(g) Zhou et al. 2(h) Ours 2(i) Ours bootstrapped

Figure 6.13: Comparisons for constrained DL dynamic blur case (from Zhou *et al.* (2019)) and unconstrained DL static scene case (from Mohan *et al.* (2019)). Our method is comparable with respect to the state-of-the-art methods.

competitor, i.e., (Zhou *et al.*, 2019) has only 27.718 dB with an offset 6.074. Figures 6.11–6.12 provides extensive evaluation on all comparison methods. It is evident from the results that existing methods are *not* adequate for unconstrained DL dynamic scene deblurring, which calls for a new approach (like ours). We also evaluate our method on unconstrained DL static scene blur examples (from (Mohan *et al.*, 2019)) and constrained DL blur examples (from (Zhou *et al.*, 2019)) in Fig. 6.13. In all the cases, our method proves superior over all the competitive methods.

6.6.1 Implementation Details

Our method is implemented using `Pytorch 1.1.0` in a server with Intel Xeon processor and an Nvidia RTX 2080 TI GPU. For training our model, we use Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and set the batch-size as four. Following (Zhou *et al.*, 2019; Tao *et al.*, 2018), we consider 256×256 patches for training, and to aid generalization we perform random chromatic transformation (brightness, contrast and saturation sampled uniformly from [0.15,0.85]) and Gaussian noise-addition ($\sigma = 0.01$). The decimation step-size in our adaptive scale-space approach is selected as $\frac{1}{\sqrt{2}}$ (following traditional scale-space deblurring methods (Mohan *et al.*, 2019; Whyte *et al.*, 2012)). The learning rate is decayed from 0.001 with a power of 0.3, and convergence is observed for our network within 4,00,000 iterations.

Since each network-level in a scale-space method adds to the computational cost, an optimal scale-space approach for our problem has to *adaptively* select the number of levels according to the input, e.g., a constrained case does *not* require decimation, whereas decimation scales for unconstrained case can be optimized in accordance with the maximum extent of disparity error (i.e., $\Delta \mathbf{x}^R$ in Eq. (6.7)). Note that a priori knowledge of DL exposure setting (or pivot-separation $t' - t''$ in Fig. 6.3(a)) is *not* sufficient to decide on the optimal decimation factor because the disparity error depends on the relative motion undergone in this interval. To this end, we optimize the computational cost for training and testing as follows. As noted earlier, exposure-overlap for each training-sample is randomly sampled from 10-100%. Therefore, we first classify our dataset according to the optimal number of network levels each training-sample requires. For this, we find the registration error between the left- and right-view input images (following (Su *et al.*, 2017)). The ninety-quartile of the vertical displacement error (in pixels) is

considered as the estimate of discrepancy, which is selected empirically so as to provide a good data-classification accuracy. We do *not* consider horizontal displacement error because it can be primarily due to stereo parallax. Then an optimal decimation scale is chosen such that it reduces the maximum discrepancy of those training samples within one pixel. For training, we confine all images in a batch to have a particular number of network levels, thereby allotting optimal multi-scale network for each batch (which is derived from the single-scale network following Sec. 6.3.2). The same procedure is employed to estimate optimal scales during testing. For multi-scale case, weights in each scale are shared (Sec. 6.3.2), and hence are updated together.

6.7 Conclusions

In this chapter, we proposed the first dynamic scene deblurring method for present-day unconstrained DL cameras. We identified and addressed its three major issues, namely, ensuring view-consistency using a coherent fusion module, preserving the epipolar constraint using an adaptive scale-space approach, and space-variant image-dependent nature of dynamic scene blur using an advanced ASPP filter module. We also built a new dataset for the current problem. Comprehensive evaluations with the existing DL and state-of-the-art monocular techniques clearly reveal the necessity of our method. Our proposed modules can be easily adapted to future deep learning methods that have to handle unconstrained DL cameras.

CHAPTER 7

Conclusions

In this thesis, we explored the problem of blind motion deblurring (BMD) in rolling shutter cameras, light field cameras and unconstrained dual-lens cameras. Specifically, we undertook a principled study to develop convenient, appropriate motion blur model for each imaging modality, and proposed effective algorithms with modest computational considerations and processing requirements.

First we addressed the BMD problem in rolling shutter (RS) cameras. Here, an RS motion blur model is developed which resembles a *block-wise* conventional camera blur model. We showed that the mature deblurring methods of conventional cameras *cannot*, in any straightforward manner, be extended for RS deblurring. This is because of an important ill-posedness present in RS-BMD which distorts scene-structures. To tackle this, we next proposed an effective prior which is convex and can be easily incorporated in the BMD cost. We also demonstrated how the computationally efficient filter flow can be extended to RS-BMD problem to achieve a significant speed-up. Extensive evaluations were conducted to validate the ability of our proposal in dealing with narrow- and wide-angle RS settings as well as arbitrary camera motions.

Next we turned our attention to the BMD problem in light field (LF) cameras. Existing LF-BMD methods have to optimize for clean LF 'in toto', which brings in severe computational constraints such as requirement of GPU and inability to deal with full-resolution LFs. We showed that it is possible to isolate motion blur in individual subaperture images, and relate different blurred subaperture images to a single camera motion. This model allowed estimating camera motion from a single subaperture image, as opposed to the full LF. Once the camera motion is estimated, we devised a strategy to independently deblur the remaining subaperture images (in parallel) by an efficient non-blind deblurring technique, thereby greatly reducing the computational cost for LF-BMD. We experimentally showed that our method performs full-resolution LF-BMD *without* GPU-requirement, and leads to significant computational gain.

Our subsequent endeavour was to study the deblurring problem in unconstrained

dual-lens (DL) cameras that have became increasingly commonplace in present-day smartphones. A deblurring method for this problem has to ensure scene-consistent depth-cues in deblurred images. We first introduced a motion blur model for unconstrained DL that also explicitly accounts for arbitrary center-of-rotation. Next, we revealed an inherent ill-posed in DL-BMD which easily disrupts scene-consistent disparities. We addressed this issue using an effective prior on camera motion. Based on our model and prior, we built an alternating minimization framework to recover center-of-rotation, camera motion and deblurred image-pairs. We demonstrated the practical utility of our proposed method using both synthetic and real examples.

Finally, we addressed the problem of dynamic scene deblurring in unconstrained DL cameras. Due to the large number of unknowns and associated complexity involved in tackling this problem via traditional methods, we resorted to a deep learning solution backed by signal processing principles. We showed that the existing DL-BMD methods fail to produce a view-consistent image-pair for today's unconstrained DL cameras. To this end, we brought out the main reason for this limitation and addressed it using a coherent fusion module. Further, to ensure scene-consistent disparities in unconstrained DL dynamic scene deblurring, we proposed an adaptive scale-space approach. Also, we addressed the space-variant image-dependent nature of blur by extending the widely-applicable ASPP module. An extensive evaluation of our algorithm demonstrated the potential of the proposed method for unconstrained DL deblurring.

7.1 Some directions for future work

Our rolling shutter motion blur model with its computational capability (Chapter 3) has extra potential to be tapped. An active research area in conventional camera BMD is in formulating effective natural image priors, which typically are highly non-linear and non-convex (e.g., (Pan *et al.*, 2016; Xu *et al.*, 2013)). For optimization with these priors, BMD methods have to efficiently create blur matrix ($\mathbf{X}$ in Eq. (3.15), Sec. 3.5.2) numerous times, for which they primarily rely on efficient filter flow (EFF). But there exists *no* EFF scheme for RS, apart from what we proposed in Sec. 3.4.1. Therefore, our RS-EFF can be extended to incorporate these priors developed for conventional cameras in ubiquitous rolling shutter cameras as well.

Present-day light field cameras are not amenable to wide field-of-view (FOV) configuration due to various practical design constraints. Nevertheless, wide-FOV 4D light field can be synthesized by merging multiple narrow-FOV LFs; but this is challenging as it warrants a motion transformation criterion that is consistent with the light field principles and scene-geometry. Though we have derived a motion transformation in Eq. (4.11), it is limited to rotation-only motion (which is satisfactory for motion deblurring (Sec. 4.6.1), but not necessarily holds good for LF merging (Szeliski and Shum, 1997)). One way forward is to extend Eq. (4.11) considering general 6D motion to arrive at a general motion transformation for LFs. Yet another research topic that can be derived from this chapter is dynamic scene LF deblurring, which has *not* been attempted hitherto. A major challenge would then be to segment multiple dynamic segments from a blurred LF, in order to individually assign them independent relative camera motions. One way forward is to use LF depth information as well (as compared to confining only to photometry information) to segment dynamic objects at different depth. Once it is addressed, our divide and conquer approach can be effectively utilized for individual dynamic segments, which paves the way for LF dynamic scene deblurring.

The recent growing popularity of unconstrained DL cameras, especially in today's smart phones, calls for several problems to be addressed. For instant, rolling shutter effects are pertinent problems in well-lit scenarios, but they have *not* been addressed for unconstrained DL cameras. They also require a 'homography-like' warping (such as Eq. (5.9)), admit the *same* ill-posedness, and hence necessitate an analogous prior. An exciting pursuit of research would be to extend the ideas in Chapter 5 for tackling RS effects in DL cameras, including RS deblurring, RS super-resolution, and RS change detection. Further for deep learning, the DL motion blur model in Eqs. (5.2)-(5.9) can potentially aid in generating training datasets.

Finally, there exists several deep learning based augmented/virtual reality applications using DL cameras, e.g., super-resolution (Wang *et al.*, 2019*b*; Jeon *et al.*, 2018) and style-transfer (Chen *et al.*, 2018; Gong *et al.*, 2018); but these methods are designed for constrained DL set-up. Directly using this methods in unconstrained DL configurations naturally leads to view-inconsistency. Our coherent fusion module in Chapter 6 can potentially extend these deep learning methods to tackle view-inconsistency in those methods. In addition, the adaptive scale-space approach, though a simple technique, can potentially allow existing deep-learning based deblurring works in accom-

modating lower image-scales (e.g., extending the state-of-the-art DL deblurring work (Zhou *et al.*, 2019) as in Fig. 6.7(a)). Further, the atrous spatial pyramid pooling (ASPP) (Chen *et al.*, 2017) is widely applicable in semantic segmentation, object detection, visual question answering, and optical flow. We have extended the ASPP (Chen *et al.*, 2017) to instil the space-variant image-dependent (SvId) property primarily to address motion blur. But one can evidently see SvId nature in other ASPP-based applications as well, e.g., in semantic segmentation, spatial positions and scales of an object in an image can freely vary, and these attributes are image-dependent. Therefore, an exciting research direction is to explore the potential of SvId-ASPP for those applications. Finally, many of the ideas developed here can be extended for dynamic scene deblurring for rolling shutter and light field cameras using deep learning (which has *not* been explored hitherto).

As the trend of consumer cameras going beyond conventional cameras continues, the methodologies developed in this thesis will be invaluable for motion deblurring.

REFERENCES

1. **Adelson, E. H.** and **J. Y. Wang** (1992). Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(2), 99–106.

2. **Arun, M.**, **A. N. Rajagopalan**, and **G. Seetharaman**, Multi-shot deblurring for 3d scenes. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*. IEEE, 2015.

3. **Bätz, M.**, **T. Richter**, **J.-U. Garbas**, **A. Papst**, **J. Seiler**, and **A. Kaup** (2014). High dynamic range video reconstruction from a stereo camera setup. *Signal Processing: Image Communication*, **29**(2), 191–202.

4. **Boyd, S.**, **N. Parikh**, **E. Chu**, **B. Peleato**, and **J. Eckstein** (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, **3**(1), 1–122.

5. **Boyd, S.** and **L. Vandenberghe**, *Convex optimization*. Cambridge university press, 2004.

6. **Brox, T.**, **A. Bruhn**, **N. Papenberg**, and **J. Weickert**, High accuracy optical flow estimation based on a theory for warping. *In European conference on computer vision (ECCV)*. Springer, 2004.

7. **Chan, T. F.** and **C.-K. Wong** (1998). Total variation blind deconvolution. *IEEE transactions on Image Processing*, **7**(3), 370–375.

8. **Chandramouli, P.**, **M. Jin**, **D. Perrone**, and **P. Favaro** (2018). Plenoptic image motion deblurring. *IEEE Transactions on Image Processing*, **27**(4), 1723–1734.

9. **Chen, D.**, **L. Yuan**, **J. Liao**, **N. Yu**, and **G. Hua**, Stereoscopic neural style transfer. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

10. **Chen, L.-C.**, **G. Papandreou**, **I. Kokkinos**, **K. Murphy**, and **A. L. Yuille** (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **40**(4), 834–848.

11. **Chen, M.-J.**, **C.-C. Su**, **D.-K. Kwon**, **L. K. Cormack**, and **A. C. Bovik** (2013). Full-reference quality assessment of stereopairs accounting for rivalry. *Signal Processing: Image Communication*, **28**(9), 1143–1155.

12. **Cho, S.** and **S. Lee**, Fast motion deblurring. *In ACM Transactions on Graphics (TOG)*, volume 28. ACM, 2009.

13. **Coleman, T. F.** and **Y. Li** (1996). An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, **6**(2), 418–445.

14. **Dansereau, D. G.**, **A. Eriksson**, and **J. Leitner** (2016). Richardson-lucy deblurring for moving light field cameras. *arXiv preprint arXiv:1606.04308*.

15. **Dansereau, D. G.**, **O. Pizarro**, and **S. B. Williams**, Decoding, calibration and rectification for lenselet-based plenoptic cameras. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013.

16. **Dansereau, D. G.**, **G. Schuster**, **J. Ford**, and **G. Wetzstein**, A wide-field-of-view monocentric light field camera. *In Proccedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

17. **Dodge, S.** and **L. Karam**, Understanding how image quality affects deep neural networks. *In Proceedings of the International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2016.

18. **Efron, B.**, **T. Hastie**, **I. Johnstone**, **R. Tibshirani**, *et al.* (2004). Least angle regression. *The Annals of Statistics*, **32**(2), 407–499.

19. **Fusiello, A.** and **L. Irsara** (2011). Quasi-euclidean epipolar rectification of uncalibrated images. *Machine Vision and Applications*, **22**(4), 663–670.

20. **Fusiello, A.**, **E. Trucco**, and **A. Verri** (2000). A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, **12**(1), 16–22.

21. **Gao, H.**, **X. Tao**, **X. Shen**, and **J. Jia**, Dynamic scene deblurring with parameter selective sharing and nested skip connections. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019.

22. **Gong, X.**, **H. Huang**, **L. Ma**, **F. Shen**, **W. Liu**, and **T. Zhang**, Neural stereoscopic image style transfer. *In Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

23. **Gu, J.**, **Y. Hitomi**, **T. Mitsunaga**, and **S. Nayar**, Coded rolling shutter photography: Flexible space-time sampling. *In Proceedings of the International Conference on Computational Photography (ICCP)*. IEEE, 2010.

24. **Gupta, A.**, **N. Joshi**, **C. L. Zitnick**, **M. Cohen**, and **B. Curless**, Single image deblurring using motion density functions. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2010.

25. **Hartley, R.** and **A. Zisserman**, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

26. **Hatch, M. R.**, *Vibration simulation using MATLAB and ANSYS*. CRC Press, 2000.

27. **Hee Park, S.** and **M. Levoy**, Gyro-based multi-image deconvolution for removing handshake blur. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014.

28. **Hirsch, M.**, **C. J. Schuler**, **S. Harmeling**, and **B. Schölkopf**, Fast removal of non-uniform camera shake. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2011.

29. **Hirsch, M.**, **S. Sra**, **B. Schölkopf**, and **S. Harmeling**, Efficient filter flow for space-variant multiframe blind deconvolution. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010.

30. **Hu, Z.**, **L. Xu**, and **M.-H. Yang**, Joint depth estimation and camera shake removal from single blurry image. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014.

31. **Hu, Z.** and **M.-H. Yang**, Good regions to deblur. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2012.

32. **Hu, Z.**, **L. Yuan**, **S. Lin**, and **M.-H. Yang**, Image deblurring using smartphone inertial sensors. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

33. **Innmann, M.**, **K. Kim**, **J. Gu**, **M. Niessner**, **C. Loop**, **M. Stamminger**, and **J. Kautz** (2019). Nrmvs: Non-rigid multi-view stereo. *arXiv preprint arXiv:1901.03910*.

34. **Ito, E.** and **T. Okatani**, Self-calibration-based approach to critical motion sequences of rolling-shutter structure from motion. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

35. **Iyer, G.**, **J. Krishna Murthy**, **G. Gupta**, **M. Krishna**, and **L. Paull**, Geometric consistency for self-supervised end-to-end visual odometry. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. IEEE, 2018.

36. **Janoch, A.**, **S. Karayev**, **Y. Jia**, **J. T. Barron**, **M. Fritz**, **K. Saenko**, and **T. Darrell**, A category-level 3d object dataset: Putting the kinect to work. *In Consumer Depth Cameras for Computer Vision*. Springer, 2013, 141–165.

37. **Jeon, D. S.**, **S.-H. Baek**, **I. Choi**, and **M. H. Kim**, Enhancing the spatial resolution of stereo images using a parallax prior. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

38. **Jin, M.**, **P. Chandramouli**, and **P. Favaro**, Bilayer blind deconvolution with the light field camera. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*. IEEE, 2015.

39. **Jin, M.**, **G. Meishvili**, and **P. Favaro**, Learning to extract a video sequence from a single motion-blurred image. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

40. **Joshi, N.**, **S. B. Kang**, **C. L. Zitnick**, and **R. Szeliski** (2010). Image deblurring using inertial measurement sensors. *ACM Transactions on Graphics (TOG)*, **29**(4), 30.

41. **Köhler, R.**, **M. Hirsch**, **B. Mohler**, **B. Schölkopf**, and **S. Harmeling**, Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2012.

42. **Krishnan, D.** and **R. Fergus**, Fast image deconvolution using hyper-laplacian priors. *In Proceedings of the Advances in Neural Information Processing Systems (NIPS) 22*. 2009.

43. **Krishnan, D.**, **T. Tay**, and **R. Fergus**, Blind deconvolution using a normalized sparsity measure. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011.

44. **Kupyn, O.**, **V. Budzan**, **M. Mykhailych**, **D. Mishkin**, and **J. Matas**, Deblurgan: Blind motion deblurring using conditional adversarial networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

45. **Lai, W.-S.**, **J.-B. Huang**, **Z. Hu**, **N. Ahuja**, and **M.-H. Yang**, A comparative study for single image blind deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

46. **Lee, D.**, **H. Park**, **I. Kyu Park**, and **K. Mu Lee**, Joint blind motion deblurring and depth estimation of light field. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2018.

47. **Levin, A.**, **R. Fergus**, **F. Durand**, and **W. T. Freeman** (2007). Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics (TOG)*, **26**(3), 70.

48. **Li, B.**, **C.-W. Lin**, **B. Shi**, **T. Huang**, **W. Gao**, and **C.-C. J. Kuo**, Depth-aware stereo video retargeting. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

49. **Li, Z.**, **Z. Xu**, **R. Ramamoorthi**, and **M. Chandraker**, Robust energy minimization for brdf-invariant shape from light fields. *In Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

50. **Litwiller, D.** (2001). CCD vs. CMOS. *Photonics Spectra*, **35**(1), 154–158.

51. **Liu, C.** *et al.* (2009). *Beyond pixels: exploring new representations and applications for motion analysis*. Ph.D. thesis, Massachusetts Institute of Technology.

52. **Liu, L.-K.**, **S. H. Chan**, and **T. Q. Nguyen** (2015). Depth reconstruction from sparse samples: Representation, algorithm, and sampling. *IEEE Transactions on Image Processing*, **24**(6), 1983–1996.

53. **Loop, C.** and **Z. Zhang**, Computing rectifying homographies for stereo vision. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1. IEEE, 1999.

54. **Lu, P. Y.**, **T. H. Huang**, **M. S. Wu**, **Y. T. Cheng**, and **Y. Y. Chuang**, High dynamic range image reconstruction from hand-held cameras. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009.

55. **Lucy, L. B.** (1974). An iterative technique for the rectification of observed distributions. *The Astronomical Journal*, **79**, 745.

56. **Lumentut, J. S.**, **T. H. Kim**, **R. Ramamoorthi**, and **I. K. Park** (2019). Deep recurrent network for fast and full-resolution light field deblurring. *IEEE Signal Processing Letters*, **26**(12), 1788–1792.

57. **Lv, Z.**, **K. Kim**, **A. Troccoli**, **D. Sun**, **J. M. Rehg**, and **J. Kautz**, Learning rigidity in dynamic scenes with a moving camera for 3d motion field estimation. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2018.

58. **Mandal, S.**, **A. Bhavsar**, and **A. K. Sao** (2016). Depth map restoration from undersampled data. *IEEE Transactions on Image Processing*, **26**(1), 119–134.

59. **Mo, J.** and **J. Sattar** (2018). Dsvo: Direct stereo visual odometry. *arXiv preprint arXiv:1810.03963*.

60. **Mohan, M. R. M.**, **S. Girish**, and **A. N. Rajagopalan**, Unconstrained motion deblurring for dual-lens cameras. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2019.

61. **Mohan, M. R. M.** and **A. N. Rajagopalan**, Divide and conquer for full-resolution light field deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

62. **Mohan, M. R. M.**, **A. N. Rajagopalan**, and **G. Seetharaman**, Going unconstrained with rolling shutter deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

63. **Mu, T.-J.**, **J.-H. Wang**, **S.-P. Du**, and **S.-M. Hu** (2014). Stereoscopic image completion and depth recovery. *The Visual Computer*, **30**(6-8), 833–843.

64. **Nah, S.**, **T. Hyun Kim**, and **K. Mu Lee**, Deep multi-scale convolutional neural network for dynamic scene deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

65. **Ng, R.**, **M. Levoy**, **M. Brédif**, **G. Duval**, **M. Horowitz**, and **P. Hanrahan** (2005). Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR*, **2**(11), 1–11.

66. **Niklaus, S.**, **L. Mai**, and **F. Liu**, Video frame interpolation via adaptive separable convolution. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017.

67. **Nimisha, T. M.**, **A. K. Singh**, and **A. N. Rajagopalan**, Blur-invariant deep learning for blind-deblurring. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017.

68. **Oppenheim, A.** and **R. Schafer**, *Discrete Time Signal Processing*. Prentice-Hall, 2014.

69. **Pan, J.**, **D. Sun**, **H. Pfister**, and **M.-H. Yang**, Blind image deblurring using dark channel prior. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

70. **Pan, L.**, **Y. Dai**, and **M. Liu**, Single image deblurring and camera motion estimation with depth map. *In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019.

71. **Pan, L.**, **Y. Dai**, **M. Liu**, and **F. Porikli**, Simultaneous stereo video deblurring and scene flow estimation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

72. **Paramanand, C.** and **A. N. Rajagopalan**, Non-uniform motion deblurring for bilayer scenes. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013.

73. **Park, W.-J.**, **S.-W. Ji**, **S.-J. Kang**, **S.-W. Jung**, and **S.-J. Ko** (2017). Stereo vision-based high dynamic range imaging using differently-exposed image pair. *Sensors*, **17**(7), 1473.

74. **Pashchenko, N.**, **K. Zipa**, and **A. Ignatenko** (2017). An algorithm for the visualization of stereo images simultaneously captured with different exposures. *Programming and Computer Software*, **43**(4), 250–257.

75. **Perrone, D.** and **P. Favaro**, Total variation blind deconvolution: The devil is in the details. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014.

76. **Petschnigg, G.**, **R. Szeliski**, **M. Agrawala**, **M. Cohen**, **H. Hoppe**, and **K. Toyama** (2004). Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics (TOG)*, **23**(3), 664–672.

77. **Poggi, M.**, **F. Aleotti**, **F. Tosi**, and **S. Mattoccia**, Towards real-time unsupervised monocular depth estimation on cpu. *In IEEE/JRS Conference on Intelligent Robots and Systems (IROS)*. 2018.

78. **Punnappurath, A.**, **V. Rengarajan**, and **A. N. Rajagopalan**, Rolling shutter super-resolution. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015.

79. **Purohit, K.**, **A. Shah**, and **A. Rajagopalan**, Bringing alive blurred moments. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

80. **Purohit, R. A. N., Kuldeep**, Region-adaptive dense network for efficient motion deblurring. *In Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*. AAAI, 2020.

81. **Rajagopalan, A. N.** and **S. Chaudhuri** (1999). An MRF model-based approach to simultaneous recovery of depth and restoration from defocused images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **21**(7), 577–589.

82. **Rajagopalan, A. N.** and **R. Chellappa**, *Motion Deblurring: Algorithms and Systems*. Cambridge University Press, 2014.

83. **Rengarajan**, **Vijay**, **Rajagopalan**, **A. N.**, **R. Aravind**, and **G. Seetharaman** (2016). Image registration and change detection under rolling shutter motion blur. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

84. **Richardson, W. H.** (1972). Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America (JOSA)*, **62**(1), 55–59.

85. **Riegler, G.**, **M. Rüther**, and **H. Bischof**, Atgv-net: Accurate depth super-resolution. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2016.

86. **Scharstein, D.** and **R. Szeliski** (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)*, **47**(1-3), 7–42.

87. **Sellent, A.**, **C. Rother**, and **S. Roth**, Stereo video deblurring. *In European Conference on Computer Vision (ECCV)*. Springer, 2016.

88. **Shan, Q.**, **J. Jia**, and **A. Agarwala**, High-quality motion deblurring from a single image. *In ACM Transactions on Graphics (TOG)*, volume 27. ACM, 2008.

89. **Sheikh, H. R.** and **A. C. Bovik** (2006). Image information and visual quality. *IEEE Transactions on Image Processing (TIP)*, **15**(2), 430–444.

90. **Sheikh, H. R.**, **A. C. Bovik**, and **G. De Veciana** (2005). An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on Image Processing (TIP)*, **14**(12), 2117–2128.

91. **Shen, X.**, **H. Gao**, **X. Tao**, **C. Zhou**, and **J. Jia**, High-quality correspondence and segmentation estimation for dual-lens smart-phone portraits. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017.

92. **Shih, K.-T.** and **H. H. Chen** (2018). Generating high-resolution image and depth map using a camera array with mixed focal lengths. *IEEE Transactions on Computational Imaging*.

93. **Simonyan, K.** and **A. Zisserman**, Very deep convolutional networks for large-scale image recognition. *In Proceedings of the International Conference on Learning Representation (ICLR)*. IEEE, 2015.

94. **Sindelar, O.** and **F. Sroubek** (2013). Image deblurring in smartphone devices using built-in inertial measurement sensors. *Journal of Electronic Imaging*, **22**(1), 011003.

95. **Srinivasan, P. P.**, **R. Ng**, and **R. Ramamoorthi**, Light field blind motion deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

96. **Sroubek, F.** and **P. Milanfar** (2012). Robust multichannel blind deconvolution via fast alternating minimization. *IEEE Transactions on Image Processing*, **21**(4), 1687–1700.

97. **Su, S.**, **M. Delbracio**, **J. Wang**, **G. Sapiro**, **W. Heidrich**, and **O. Wang**, Deep video deblurring for hand-held cameras. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

98. **Su, S.** and **W. Heidrich**, Rolling shutter motion deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015.

99. **Sun, N.**, **H. Mansour**, and **R. Ward**, HDR image construction from multi-exposed stereo LDR images. *In Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2010.

100. **Szeliski, R.** and **H.-Y. Shum**, Creating full view panoramic image mosaics and environment maps. *In Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 1997.

101. **Tao, M. W.**, **S. Hadap**, **J. Malik**, and **R. Ramamoorthi**, Depth from combining defocus and correspondence using light-field cameras. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2013.

102. **Tao, X.**, **H. Gao**, **X. Shen**, **J. Wang**, and **J. Jia**, Scale-recurrent network for deep image deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

103. **Tibshirani, R.** (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288.

104. **Tourani, S.**, **S. Mittal**, **A. Nagariya**, **V. Chari**, and **M. Krishna**, Rolling shutter and motion blur removal for depth cameras. *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016.

105. **Vasiljevic, I.**, **A. Chakrabarti**, and **G. Shakhnarovich** (2016). Examining the impact of blur on recognition by convolutional networks. *arXiv preprint arXiv:1611.05760*.

106. **Wang, F.**, **T. Li**, and **Y. Li**, Dual deblurring leveraged by image matching. *In Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2013.

107. **Wang, J.**, **T. Xue**, **J. Barron**, and **J. Chen** (2019*a*). Stereoscopic dark flash for low-light photography. *arXiv preprint arXiv:1901.01370*.

108. **Wang, L.**, **H. Jin**, **R. Yang**, and **M. Gong**, Stereoscopic inpainting: Joint color and depth completion from stereo images. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008.

109. **Wang, L.**, **Y. Wang**, **Z. Liang**, **Z. Lin**, **J. Yang**, **W. An**, and **Y. Guo**, Learning parallax attention for stereo image super-resolution. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019*b*.

110. **Whyte, O.**, **J. Sivic**, **A. Zisserman**, and **J. Ponce** (2012). Non-uniform deblurring for shaken images. *International Journal of Computer Vision (IJCV)*, **98**(2), 168–186.

111. **Wilburn, B.**, **N. Joshi**, **V. Vaish**, **E.-V. Talvala**, **E. Antunez**, **A. Barth**, **A. Adams**, **M. Horowitz**, and **M. Levoy**, High performance imaging using large camera arrays. *In ACM Transactions on Graphics (TOG)*, volume 24. ACM, 2005.

112. **Wu, G.**, **B. Masia**, **A. Jarabo**, **Y. Zhang**, **L. Wang**, **Q. Dai**, **T. Chai**, and **Y. Liu** (2017). Light field image processing: An overview. *IEEE Journal of Selected Topics in Signal Processing*.

113. **Xiao, R.**, **W. Sun**, **J. Pang**, **Q. Yan**, and **J. Ren**, Dsr: Direct self-rectification for uncalibrated dual-lens cameras. *In Proceedings of the International Conference on 3D Vision (3DV)*. IEEE, 2018.

114. **Xiong, Z.**, **L. Wang**, **H. Li**, **D. Liu**, and **F. Wu**, Snapshot hyperspectral light field imaging. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

115. **Xu, L.** and **J. Jia**, Depth-aware motion deblurring. *In Proceedings of the IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2012.

116. **Xu, L.**, **J. S. Ren**, **C. Liu**, and **J. Jia**, Deep convolutional neural network for image deconvolution. *In Proceedings of the Advances in Neural Information Processing Systems (NIPS)*. 2014.

117. **Xu, L.**, **S. Zheng**, and **J. Jia**, Unnatural l0 sparse representation for natural image deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013.

118. **Yuan, X.**, **Z. Xu**, **H. Wang**, **Y. Liu**, and **L. Fang**, Cascaded image deblurring with combined image prior. *In Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2017.

119. **Yuster, R.** and **U. Zwick** (2005). Fast sparse matrix multiplication. *ACM Transactions on Algorithms (TALG)*, **1**(1), 2–13.

120. **Zhang, C.** and **T. Chen**, A self-reconfigurable camera array. *In Eurographics conference on Rendering Techniques*. Eurographics Association, 2004.

121. **Zhang, F.** and **F. Liu**, Casual stereoscopic panorama stitching. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

122. **Zhang, G.**, **J. Jia**, and **H. Bao**, Simultaneous multi-body stereo and segmentation. *In Proceedings of the International Conference on Computer Vision (ICCV)*. IEEE, 2011.

123. **Zhang, H.**, **Y. Dai**, **H. Li**, and **P. Koniusz**, Deep stacked hierarchical multi-patch network for image deblurring. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

124. **Zhang, H.**, **D. Wipf**, and **Y. Zhang**, Multi-image blind deblurring using a coupled adaptive sparse prior. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013.

125. **Zhang, J.**, **J. Pan**, **J. Ren**, **Y. Song**, **L. Bao**, **R. W. Lau**, and **M.-H. Yang**, Dynamic scene deblurring using spatially variant recurrent neural networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

126. **Zhang, L.**, **A. Deshpande**, and **X. Chen**, Denoising vs. deblurring: HDR imaging techniques using moving cameras. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010.

127. **Zhou, S.**, **J. Zhang**, **W. Zuo**, **H. Xie**, **J. Pan**, and **J. S. Ren**, Davanet: Stereo deblurring with view aggregation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

128. **Zhu, X.**, **F. Šroubek**, and **P. Milanfar**, Deconvolving psfs for a better motion deblurring using multiple images. *In Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2012.

LIST OF PAPERS BASED ON THESIS

Ch. 3 Going Unconstrained with Rolling Shutter Deblurring.
Mahesh Mohan M. R., Rajagopalan A. N., and Gunasekaran Seetharaman.
In Proceedings of the International Conference on Computer Vision
(**ICCV 2017**), IEEE Publications, Pages 4010–4018.

Ch. 4 Divide and Conquer for Full-Resolution Light Field Deblurring.
Mahesh Mohan M. R. and Rajagopalan A. N.
In Proceedings of the Conference on Computer Vision and Pattern Recognition
(**CVPR 2018**), IEEE Publications, Pages 6421–6429.

Ch. 5 Unconstrained motion deblurring for dual-lens cameras.
Mahesh Mohan M. R., Sharath Girsih, and Rajagopalan A. N.
In Proceedings of the International Conference on Computer Vision
(**ICCV 2019**), IEEE Publications, Pages 7870–7879.

Ch. 6 Dynamic Scene Deblurring for Unconstrained Dual-lens cameras.
Mahesh Mohan M. R., Nithin G. K., and Rajagopalan A. N.
IEEE Journal of Selected Topics in Signal Processing, Issue on Deep Learning for Image Restoration (Submitted after Major Revision).