

**FACULTY
OF MATHEMATICS
AND PHYSICS**
Charles University

DOCTORAL THESIS

Jan Kotera

Image Deblurring in Demanding Conditions

Institute of Information Theory and Automation,
Czech Academy of Sciences

Supervisor of the doctoral thesis: Doc. Ing. Filip Šroubek, Ph.D., DSc.

Study program: Computer Science

Study branch: Computer Graphics and Image Analysis

Prague 2020

I declare that I carried out this doctoral thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

Prague, April 15, 2020

Jan Kotera

I would like to express my utmost gratitude to my supervisor, Filip Šroubek, for his invaluable guidance during my Ph.D. studies, for always having time for me, for countless insightful discussion and observations, and seemingly never-ending patience.

Title: Image Deblurring in Demanding Conditions

Author: Jan Kotera

Department: Institute of Information Theory and Automation, Czech Academy of Sciences

Supervisor: Doc. Ing. Filip Šroubek, Ph.D., DSc., Institute of Information Theory and Automation, Czech Academy of Sciences

Abstract: Image deblurring is a computer vision task consisting of removing blur from image, the objective is to recover the sharp image corresponding to the blurred input. If the nature and shape of the blur is unknown and must be estimated from the input image, image deblurring is called blind and naturally presents a more difficult problem. This thesis focuses on two primary topics related to blind image deblurring. In the first part we work with the standard image deblurring based on the common convolution blur model and present a method of increasing robustness of the deblurring to phenomena violating the linear acquisition model, such as for example intensity clipping caused by sensor saturation in overexposed pixels. If not properly taken care of, these effects significantly decrease accuracy of the blur estimation and visual quality of the restored image. Rather than tailoring the deblurring method explicitly for each particular type of acquisition model violation we present a general approach based on flexible automatic relevance determination prior of the observation error and variational Bayes method for the blur estimation, which allows us to identify arbitrary outliers of the acquisition model automatically.

Most blind methods require estimation of the image blur as initial step of the image restoration. As another topic in the first part of the thesis we propose a method of assessing accuracy of the blur estimation independently of the image restoration.

In the second part of the thesis we address the scenario of a motion-blurred object in an otherwise sharp image, sometimes called dynamic scene deblurring. We are interested in situations when the object moves so fast, with respect to the background, that partial transparency of the object image caused by the motion blur is not negligible and must be taken into consideration in the deblurring process. We formulate the problem of recovering the object shape, appearance, and motion from the blurred input and present a solution of a suitably simplified case. We further study properties of the problem and its relationship to standard image deblurring. In the final chapter of the thesis we present a particular practical application of the presented method, visual tracking of fast, motion-blurred objects in videos, and we show how the method can be used to computationally increase the framerate of video sequences containing such objects and simulate blur-free acquisition by a high-speed camera. The thesis is maximally self-contained and we put great emphasis on meticulous and complete presentation of all the introduced topics.

Keywords: image deblurring, deconvolution, boundary artifacts, blur model violation, outliers, saturation, variational Bayes, ARD, ADMM, fast moving objects, FMO deblurring, object motion deblurring, dynamic scene deblurring, deblatting, visual object tracking, temporal super-resolution, blur estimation accuracy, PSF similarity

Contents

1	Introduction	1
1.1	Image deblurring	2
1.2	Difficulties of image deblurring	5
1.3	Goals and contributions	9
1.4	Contents of the thesis	10
2	Related work	15
2.1	Maximum a posteriori estimation	16
2.2	Variational Bayes	22
2.3	Dynamic scene deblurring	24
2.4	Deep learning	25
I	Blind image deblurring with acquisition model violations	27
3	Maximum a posteriori deblurring of images with overexposure	28
3.1	Problem formulation	30
3.2	Problem solution	32
3.3	Experimental evaluation	39
3.4	Conclusion	43
3.A	Alternating direction method of multipliers	44
4	Variational Bayesian deblurring of images with arbitrary model violations	48
4.1	Automatic relevance determination priors	49
4.2	Model definition	51
4.3	Variational Bayesian inference	58
4.4	Posterior approximation	62
4.5	Algorithm summary	70
4.6	Experimental evaluation	75
4.7	Conclusion	82
5	Blur estimation accuracy assessment	84
5.1	Common image similarity metrics	84
5.2	Motivation and goals	87
5.3	Blur accuracy measure	88
5.4	Experimental validation	91
5.5	Conclusion	92

II	Fast moving objects	94
6	Deblurring of fast moving objects	95
6.1	Problem formulation	98
6.2	Problem solution	101
6.3	Uniqueness of solutions	105
6.4	Experimental evaluation	112
6.5	Conclusion	120
7	Tracking by deblatting	121
7.1	General overview	122
7.2	Deblatting for 2D tracking	124
7.3	Deblatting for 3D tracking	129
7.4	Conclusion	136
8	Thesis conclusion	139
	Bibliography	141

Chapter 1

Introduction

Digital image processing can be characterized as a discipline of transforming one or more images into another image that is in some aspects better than the original – either for direct human consumption or as a input to another computer-based processing. Examples of image processing tasks include e.g. color and intensity adjustment, image enhancement, denoising or deblurring, resampling, super-resolution, image registration, compression, and many others. Closely related is the field of computer vision. The boundary between image processing and computer vision is not well defined and the overlap is huge, but usually computer vision task takes one or more images as input from which it extracts some higher-level information. Typical computer vision tasks include object or action recognition, object detection, segmentation, object tracking in video, 3D scene reconstruction, pose estimation, motion estimation, and countless others.

Trillions of pictures or videos are taken every year by consumer cameras alone, accompanied by an even larger number of images taken in industry and medical environment, drone and satellite images, videos from traffic and security cameras, and numerous other sources. Such amount of data cannot be even looked at, let alone meaningfully analyzed and processed manually by humans. Semi-automated or fully automated processing is necessary and is one of the reasons for the massive growth of interest in image processing and computer vision in recent years. In some cases, fully automatic processing is required by definition, for example in self-driving cars or other self-operating machines. The traditional image analysis pipeline is composed of several stages. Image acquisition is followed by initial preprocessing, which aims to remove potential errors or artifacts introduced during the acquisition, enhance salient structures or features, crop and resample image to preselected region or size, and overall normalize the image for subsequent processing. After that, numerical features are extracted from the image. These features are passed to a high-level stage for final processing. This stage is highly application-dependent, in image recognition or segmentation it is a pre-trained classifier, in 3D reconstruction it is image registration and scene geometry computation, in visual object tracking it is motion estimation etc., but the general structure remains very similar. More recently, with the advance of learning-based methods, clear separation of the feature extraction and high-level processing stages is phasing out and is being replaced by a more autonomous, combined processing.

Image restoration can be viewed as a part of the initial preprocessing, though it is also a standalone task on its own – the restored output image being the final desired product. The primary goal of image restoration is to computationally and retroactively mitigate various shortcomings of the acquisition process and improve visual quality of the image while maintaining fidelity to the photographed scene. In contrast, image enhancement methods and “filters” commonly available in photo editing software usually attempt to make the image look better and may easily produce unrealistic results. Typical tasks of image restoration include for example image denoising (removal of pixel-wise errors of random nature), image deblurring (removal of blur due to camera motion, object motion, lens aberration, media turbulence etc.), super-resolution (increase of image resolution from one or more input image or a video sequence), compression artifacts removal, image de-raining or raindrop removal (restoration of images taken during rain or through glass with raindrops), image dehazing and fog removal (restoration of images corrupted by atmospheric or underwater light scattering), image inpainting (completion of missing areas of the image), and numerous others.

Image restoration is one of the standard disciplines of image processing, with initial approaches formulated in 1950s as an application of techniques from signal processing. Popularity and demand for image restoration methods have been increasing ever since and are now at their peak due to omnipresence of imaging devices in industry, medicine, science, or everyday life. As a famous historical example, successful application of image restoration was deblurring of initial images received from the Hubble Space Telescope that were blurred due to a warped mirror in the optical system. All acquired images had to be computationally rectified until an optical correction was installed years later. The need for computational image restoration stems from multitude of reasons including physical impossibility of perfect acquisition (noise in short-time photography, blur in long-time photography, optical aberration in many imaging devices, physical properties of the scene) or cost reduction (low-resolution chips, standard consumer cameras used in demanding conditions etc.). Some form of image restoration is nowadays integrated in practically every imaging device as a part of the acquisition pipeline.

The primary interest of this work is image deblurring as a standalone task. We will address several less frequently studied aspects or formulations of the problem.

1.1 Image deblurring

Image deblurring is the process of removing blur from image – recovering the underlying sharp image from blurred input. There are numerous reasons for the presence of blur in images. The most obvious and familiar to everyone is camera being out of focus or blur due to motion – either of the camera or the photographed object. Illustrative examples of these types of image blur can be seen in Fig. 1.1. Another principally similar cause of blur is motion of the optical medium between the camera and the scene, e.g. hot air turbulence. Imperfections of the optical system such as camera lens aberration or finite tip size in scanning microscopes form a whole other class of causes of image blur.



Figure 1.1: Illustration of the most common types of blur encountered in photography.

Common blur models

Blur is commonly modeled as a sort of local averaging process – pixel value in the blurred image is a weighted sum of pixel values from the sharp image in the neighborhood of the target pixel. Let u and g denote the sharp and blurred image, respectively, and let u_i and g_i denote particular values of the i -th pixel. Then the blurring process can be modeled as

$$g_i = \sum_{j \in N_i} h_{ij} u_j, \quad (1.1)$$

where N_i is a neighborhood of the i -th pixel and h are positive weights of the linear combination. More concisely, we can express the blur process simply as an application of a linear operator

$$g = Hu, \quad (1.2)$$

where H is the operator (matrix) containing in each i -th row all the non-negative weighting coefficients h_{ij} and zeros for pixels outside the local neighborhood N_i . Image deblurring is therefore a typical inverse problem – from a known output g of a certain operation H we attempt to reconstruct the input u .

Note that the presented model is merely an approximation of the true blur process, the first hint being the use of a sum instead of an integral in (1.1). We conveniently work with discretized digital images, even though camera exposure and therefore blur are undoubtedly continuous processes. Another inaccuracy is in the assumption of a single underlying sharp image u . In case of a general camera motion and non-planar scene, slightly different image of the scene is visible at each infinitesimal time interval during the exposure and these images are then summed into the blurred observation g . Attempting to reconstruct single sharp u is an ill-defined problem. Nevertheless, we will use the introduced and customary approximation throughout the rest of the work because it presents a good compromise between fidelity and tractability.

In some cases, the weighting coefficients h_{ij} are identical for every neighborhood N_i (i.e. independent of i) and such blur is then called *spatially invariant*. In this case, the blurred image can be expressed in terms of a convolution as

$$g = h * u. \quad (1.3)$$

The convolution kernel h is commonly called *point spread function* (PSF), as it is equivalent to an image the device would acquire after capturing an ideal point source $u = \delta$ (Dirac delta function, convolution identity). Examples of approximately spatially-invariant blur in the real world are camera out-of-focus blur or blur introduced by

scanning imaging devices, because in atomic force microscopy or scanning tunneling microscopy the resulting images are convolved with a PSF whose shape is related to the shape of the scanning probe tip. In medical imaging, e.g. magnetic resonance perfusion, pharmacokinetic models consist of convolution with an unknown arterial input function. Motion blur is generally *spatially variant*, mainly due to rotation of the camera around the optical axis and effects of scene projection, though it is very often modeled as spatially invariant for simplicity.

Deblurring tasks

In some practical scenarios, the blur H is known because it can be measured separately – for example by using calibration images to determine optical aberrations of the device, motion sensors to track camera motion during exposure, or from point sources in astronomy or microscopy. In this case image deblurring effectively consists of inverting H and estimating the sharp image u . This is called *non-blind deblurring*, because knowledge of H can be used to determine u . More often, though, the blur is different for each acquired image and unknown, as in the case of general motion blur or blur due to motion in the scene. In this case the so-called *blind deblurring* requires estimating both the blur H and the sharp image u , which is understandably more difficult and, strictly speaking, not even a well-defined task. Some recent learning-based methods circumvent the blur estimation and directly estimate the sharp image u , but majority of deblurring methods require estimating the blur as a necessary intermediate step and then estimate the sharp image in the non-blind setting.

Special situation arises when multiple blurred images $g^{(i)}$ depicting the same sharp scene u are available,

$$g^{(i)} = H^{(i)}u, \quad \forall i. \quad (1.4)$$

The corresponding problem of estimating u is called *multi-channel* or *multi-frame* deblurring and enjoys certain advantages to the previously introduced *single-channel* deblurring, provided the blurs $H^{(i)}$ are sufficiently different from each other. Inverting H can be problematic due to its poor condition number, as we will see later. When multiple images are available, one essentially needs to invert the sum of the corresponding blurs which is typically much better conditioned, resulting in greater robustness to image noise or errors in the acquisition model. The greatest advantage of multi-frame deblurring is that in the blind setting it is much less susceptible to ending up in a local spurious optimum which does not correspond to the true image u .

Though the possibilities of (de)blurring are far from exhausted, the last scenario we will mention is blur due to motion in the scene. The situations covered so far model global blurring of a static scene, but quite often blur occurs due to change of the scene itself, in literature this is sometimes called *dynamic scene deblurring*. We will limit ourselves to an object moving in the scene – the object is blurred due to its motion while the rest of the scene is static and sharp. Example of such scenario is simply an image of a street with a car driving by, see Fig. 1.1-right. The task is to estimate the sharp appearance of the foreground object from the composite input image containing both the blurred foreground and the sharp background.

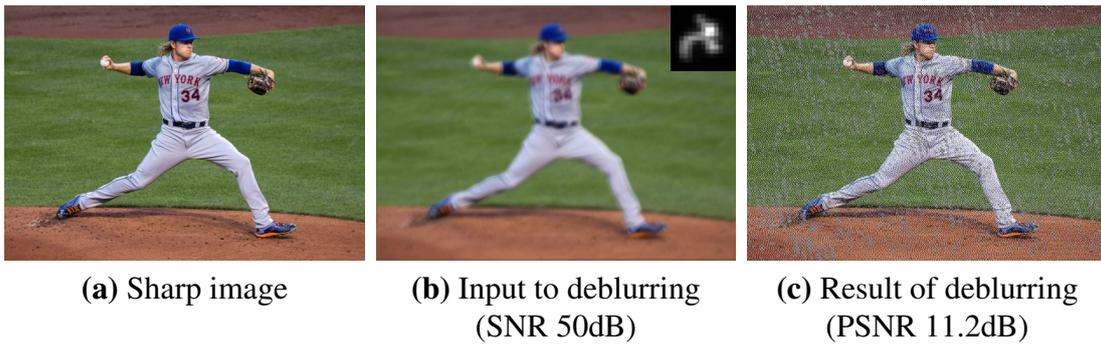


Figure 1.2: Effect of noise on image deblurring. (a) Sharp image used for deblurring examples. (b) Synthetically blurred image (a) with mild Gaussian noise added; PSF of blur H overlaid in top right (enlarged). (c) Result of non-blind deblurring using inverse of the blur H with only minimal regularization.

1.2 Difficulties of image deblurring

Let us outline the main obstacles faced when attempting to remove blur from image. Non-blind deblurring, in which the blur is known and fixed, is generally easier than blind deblurring and as a result works better in practice, although it has its share of difficulties.

Sensitivity to noise

One of the main problems encountered when inverting (1.2) is high condition number of H , which is typical for blur operators and causes severe noise amplification resulting in visual artifacts; Fig. 1.2 shows an example. Imperceptible amount of noise was added to a synthetically blurred image in Fig. 1.2b, which was then deblurred by inversion of H using just minimal regularization to avoid division by zero. Although the SNR¹ of the blurred image is over 50dB, which is roughly the upper limit of current high-end consumer cameras, the PSNR of the reconstruction is only 11dB (compare with Fig. 1.2a). The problem lies in small eigenvalues of H , which cause that many frequencies of the noise in the input image get massively amplified during deblurring and the restored image is substantially noisier than the input.

Convolution boundary problem

Another source of problems in deblurring is the image boundary. In real world, the blurred image g is a crop of a larger blurred scene. Due to blurring, however, it functionally depends on pixels from a larger viewport. In other words, the blur matrix H is rectangular (with fewer outputs than inputs) and cannot be readily inverted, estimation of u requires estimating an image that is larger than the input and the corresponding problem is under-determined. It is naturally more convenient, if the size of the image undergoing blur is retained. This is reflected in different definitions of convolution frequently appearing in image processing as models of image blur; some are illustrated in Fig. 1.3. Convolution $g = h * u$ is called valid (using terminology common in

¹Review of common image quality metrics is available in Sec. 5.1.

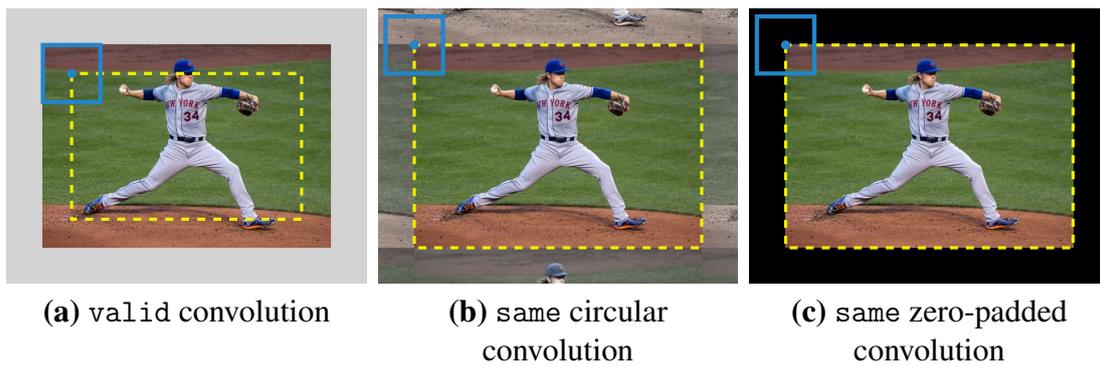


Figure 1.3: Illustration of different definitions of image convolution $g = h * u$. Blue square represents the blur PSF of h placed at the top left corner of the output image g , size of the output image is indicated by the dashed yellow box. (a) Output image is smaller than the input, no extrapolation is necessary. (b) Image size is retained, input is extrapolated periodically (highlighted by desaturation). (c) Image size is retained, input is extrapolated by zeros.

popular scripting languages), if the output g is appropriately smaller than the input u , so that at each pixel of the output, the sliding window with PSF of h fits inside the source image u and the output is therefore well defined without the need of any extrapolation of u (see Fig. 1.3a). Convolution is called *same* if the output image has the same size as the input image and the corresponding blur matrix H is square. In this case, some form of extrapolation of the input is necessary, because for pixels close to the boundary of the output g , the convolution sliding window exceeds the dimensions of the source image u . Common type of extrapolation is periodic extension (circular convolution), in which the input u is extended periodically (see Fig. 1.3b), and zero-padding, in which the input u is extrapolated by zeros (Fig. 1.3c). Circular convolution is numerically convenient because in that case the blur matrix H is diagonalized by the Fourier transform and it is therefore much faster to calculate inversions. Arguably the only realistic blur model, though, is the `valid` convolution in Fig. 1.3a. The dashed yellow box shows the smaller size of the output, the blue square represents the blur PSF of h placed at top left corner of the output image g ; the corresponding blur matrix H is rectangular.

Though computationally advantageous, approximation of the blur operator H by a square matrix by assuming some form of trivial extension beyond the image boundary as in Fig. 1.3bc leads to visible artifacts in the deblurring result (sometimes called *ringing*), as seen in Fig. 1.4. The noiseless version of Fig. 1.2b was deblurred with appropriate regularization. On the left is the result computed with the assumption of periodic extension beyond image boundaries as in Fig. 1.3b (which is coincidentally almost fitting for this particular image) and on the right is the result computed using zero-padding as in Fig. 1.3c. Unpleasant artifacts are visible in both cases and it is apparent that more sophisticated treatment of the boundary problem is necessary.

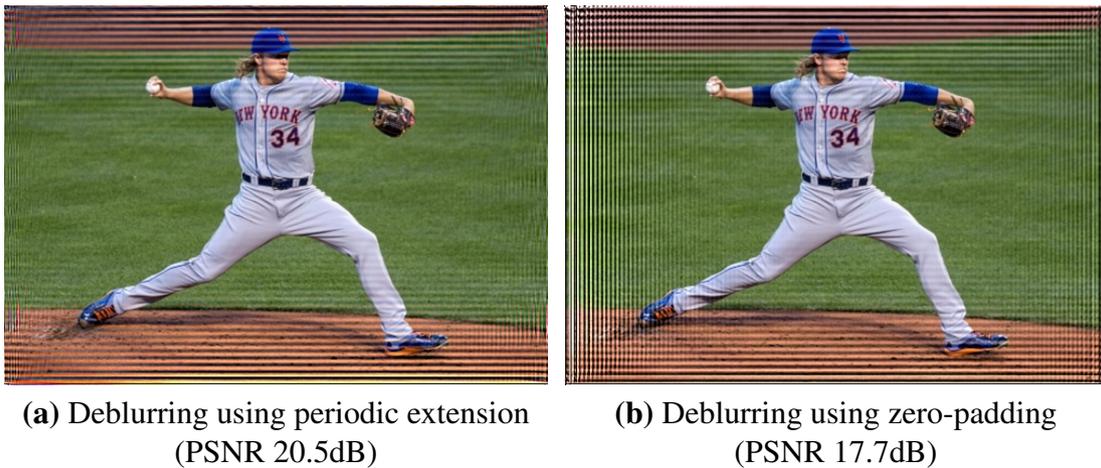


Figure 1.4: Problem of unknown image boundary during deblurring. Input image is as in Fig. 1.2b without noise. **(a)** Result of regularized deblurring with the assumption of periodic extension outside image boundaries (as in Fig. 1.3b). **(b)** Result of regularized deblurring with the assumption of zeros outside image boundaries (as in Fig. 1.3c).

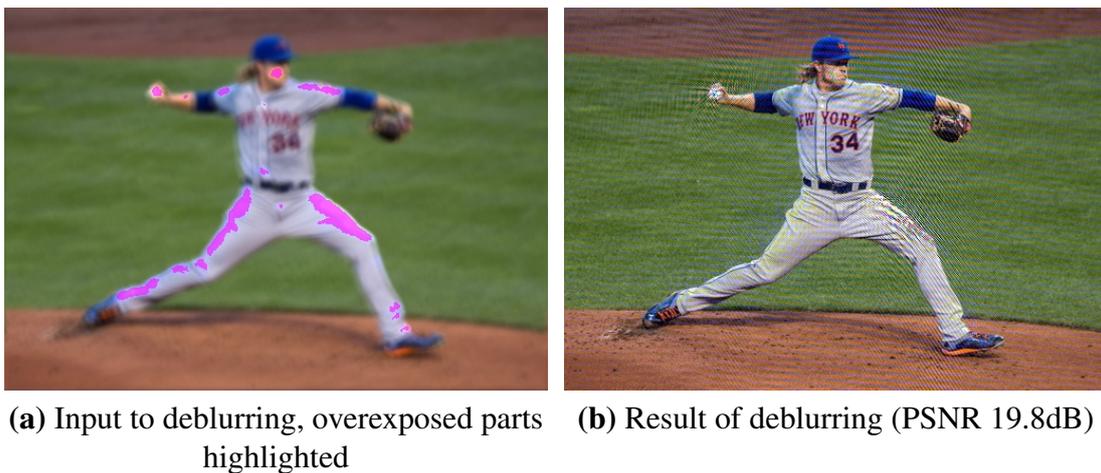


Figure 1.5: Example of how violations of the assumed acquisition model negatively impact image deblurring. **(a)** Input blurred image with some parts overexposed and clipped (highlighted by magenta). **(b)** Result of regularized deblurring without modeling of the nonlinearities caused by intensity clipping with visible ringing near the clipped regions.

Blur model infidelity

Similarly, if the blur model does not hold in certain parts of the image (e.g. due to nonlinear effects like intensity clipping of bright, overexposed regions or slight spatial-variance of the blur), restoration without special care for these problems gives visually poor results; see Fig. 1.5. An image with certain parts overexposed was blurred and the overexposed regions clipped to maximum intensity (Fig. 1.5a). This image was deblurred with appropriate regularization, but without any extra handling of the non-linearity caused by the intensity clipping, which is not modeled by eq. (1.3). Visible artifacts appear near the clipped regions (Fig. 1.5b).

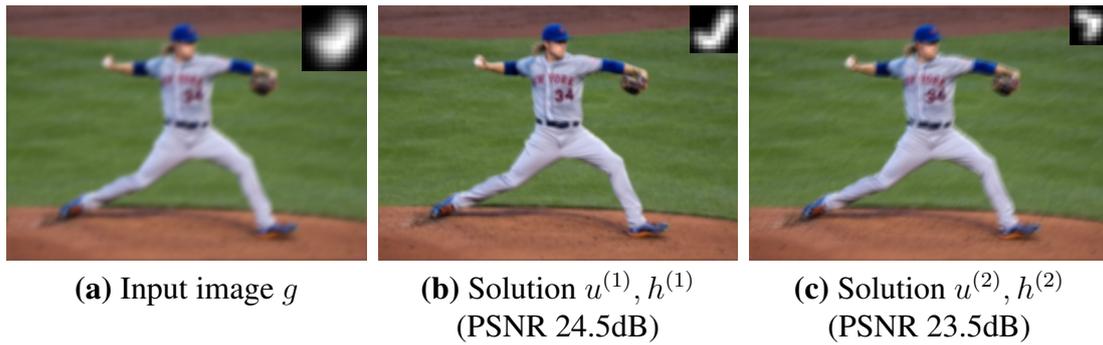


Figure 1.6: Non-uniqueness of blind image deblurring. (a) Input image g blurred according to eq. (1.3) with the true blur h superimposed (enlarged). (b) One of the possible solutions $u^{(1)}$ for which $g = h^{(1)} * u^{(1)}$ with $h^{(1)}$ superimposed. (c) Another possible solution $u^{(2)}$ for which $g = h^{(2)} * u^{(2)}$.



Figure 1.7: Sensitivity of deblurring to accuracy of blur estimation. (a) Input image g blurred according to eq. (1.3) with the true blur h superimposed (enlarged). (b,c) Solution of non-blind deblurring using incorrect $h^{(1,2)}$ containing visible ringing (PSF superimposed, compare with (a)).

Blind deblurring ambiguity

All of these characteristics are inherited by blind deblurring. By far the most troublesome property of blind deblurring, though, is the non-convexity of the inverse problem, resulting in non-uniqueness of solution. Considering e.g. the convolutional spatially-invariant formulation in eq. (1.3), there exist multiple pairs $(u^{(i)}, h^{(i)})$ such that $g = h^{(i)} * u^{(i)}$; this is illustrated in Fig. 1.6. On the left is the input blurred image g while the two images $u^{(i)}$ on the right are both perfect solutions to (1.3) with corresponding $h^{(i)}$ pictured in top-right. The problem of inferring u from g is then obviously ill-defined. Most notable spurious solution is the trivial *no-blur solution*, in which $h = \delta$ (convolution identity) and $u = g$, the output of deblurring is trivially the blurred input. Naturally, the deblurring algorithm has a tendency to get trapped in this local optimum. Many methods have been proposed to overcome the non-uniqueness with varying degree of rigor and success, finding the correct h is nevertheless problematic and rarely managed exactly right.

Similarly to the discussed sensitivity of deblurring to model fidelity, inaccuracy of blur estimation propagates to the final non-blind image estimation and leads to visual artifacts, as demonstrated in Fig. 1.7. The input blurred image in Fig. 1.7a was de-

blurred using a blur PSF that is similar but not exactly the same as the correct one to simulate unsuccessful blind blur estimation. Despite regularization and apparent similarity of the PSFs, the deblurring results in Fig. 1.7bc suffer from ringing artifacts.

Dynamic scene deblurring exhibits similar high sensitivity to noise and model infidelity. Major difference to regular image deblurring, however, is the necessity to consider the shape of the blurred object, which is usually unknown in practice and must be estimated together with the object appearance. Implications and similarities to as well as differences from standard blind image deblurring are one of the topics studied in this work.

The thesis focuses primarily on two topics related to blind image deblurring. In the first part we work with the standard blind image deblurring based on the convolution blur model (1.3) and present methods of increasing its robustness to phenomena violating this linear acquisition model, such as the already discussed intensity clipping caused by sensor saturation in overexposed pixels. Since most blind methods require estimation of the image blur as initial step of the image restoration, we also propose a method of assessing accuracy of the blur estimation independently of the image restoration.

In the second part we address the scenario of motion-blurred objects in an otherwise sharp image. We define the image formation model and formulate the problem of recovering the object shape, appearance, and motion from the blurred input and study its properties, solvability, and relationship to standard image deblurring. We present a solution of a suitably simplified case and show a particular practical application in the form of visual tracking of a motion-blurred object.

1.3 Goals and contributions

The primary goals of the thesis are two less frequently studied problems related to blind image deblurring that appear in deblurring of images captured in real-world conditions. The first is local violation of the convolution blur assumption, the second is proper modeling of the acquisition process in case of motion-blurred objects in the image.

Blur estimation and image restoration are very sensitive to the fidelity of the assumed degradation model; we have seen some examples above. In practice, the true image degradation often deviates from the ideal model and success of image deblurring is compromised. We want to address this problem by formulating a general approach in which the model-violating pixels are identified as part of the deblurring process and are removed from estimation of the blur and sharp image. The proposed approach will augment a general blind deblurring method and as a result will perform well also for images captured in demanding conditions.

Deblurring of motion-blurred objects in the image is sometimes viewed as a special case of spatially-variant blur, but this does not accurately model the true nature of the degradation. When the object moves in front of the background, different parts of the background are visible during image exposure depending on current position of the object. The acquired image is then a blend of the blurred foreground and sharp background, the foreground object appears semi-transparent and the sharp background is partially visible through the blurred image of the object. Such composite image cannot be described as a result of spatially-variant blur. We want to formulate the correct

model of such acquisition process and propose corresponding deblurring method to recover the sharp appearance of the object.

Contributions of the thesis are aligned with the goals. In the thesis we present a general blind image deblurring method based on variational Bayes that includes automatic identification and rejection of pixels violating the blur model. The method uses very unified probabilistic framework with the same class of prior for all primary unknown variables. As a result, our method does not require any explicit steps to identify potential model violations. Its ability to handle such violations stems from sufficient flexibility of the noise prior alone, which sets it apart from other methods proposed in literature, where extra processing steps are included for the purpose of outlier handling. As a prelude to this general approach we present a slightly simpler method based on maximum a posteriori estimation that also serves as a foundation establishing the theoretical principles of blind image deblurring used in the thesis. We believe that the thorough and complete presentation of the studied problems accompanied by numerous comments and implementation suggestions also constitutes one of the contributions of this work because it can serve as an ideal starting point or reference material for anyone working on similar topics.

The initial step of most blind methods is estimation of the image blur. During our work on image deblurring we frequently needed to assess the accuracy of the estimated blur. Since there were no established methods other than those based on visual similarity of the blur PSFs, we developed our own and it is presented in the thesis. The method is simple to use and, unlike other widely used metrics of blur similarity, it faithfully reflects how inaccuracy of blur estimation affects subsequent non-blind image restoration.

Object motion deblurring with proper modeling of semi-transparency due to blur is much less frequently studied in literature and, to our knowledge, such model has never been used in the context of blind deblurring of the object appearance. Our main contribution therefore lies in formulating the problem with proper blur model, proposing a solution (albeit to its simplified variant), and identifying its theoretical properties. The problem differs from standard image deblurring in several ways, most notably by the presence of unknown shape of the object that has to be estimated during the deblurring process. This has some theoretical implications on the solvability of the problem, which we study in this work. Lastly, we apply the developed method to visual tracking of fast, motion blurred objects in a video sequence. Such approach is unique in its principle, because other tracking methods either ignore motion blur or are designed to be sufficiently robust to it, and at best they predict the location of the object in the frame. In contrast, our method actively estimates motion blur and uses it to determine the whole intra-frame motion trajectory of the object, in certain simplified cases including the 3D rotation.

1.4 Contents of the thesis

The rest of the thesis is structured as follows.

Chapter 2 presents a brief review of other literature related to image deblurring in general and the topics discussed in this work in particular. It provides an overview of the state of the art in image deblurring as of the beginning of 2020 in a comprehensive and relatively compact form and is organized by the topics addressed in the rest of the work to emphasize relevance to particular chapters.

Chapter 3 studies the standard blind image deblurring based on the convolution blur model (1.3). To account for observation error, it is common practice to augment this model by the assumption of additive Gaussian noise. We will focus on situations when such simple approach is not adequate because the acquisition model is locally violated in a very non-Gaussian manner. Without special treatment, this causes poor accuracy or complete fail of the blur estimation and correspondingly low quality of the image restoration, as we have demonstrated above. This chapter will address a particular kind of model violation – image intensity clipping due to sensor saturation. In low-light photography, bright parts are often overexposed and intensities higher than the acquisition device can capture are clipped. This clipping occurs after the blurring, which is a violation of the linear model (1.3). We will present a method that automatically identifies the saturated pixels and removes them from the blur estimation. The same technique is used to solve the convolution boundary problem without sacrificing model fidelity or computational efficiency. The solution is based on maximum a posteriori estimation of blur and image simultaneously, which is generally susceptible to getting trapped in spurious local minima of the non-convex problem. We employ several measures to overcome this problem, namely using heavy-tailed distribution of image gradients as prior, gradual adjustment of the data-fidelity term weight, and multiscale optimization scheme. We also use this chapter to establish some general concepts used in image deblurring, so we start with the very basics and provide solid foundation for the rest of the work. To make the thesis more self-contained, we include a quick primer on the ADMM optimization method, which is used extensively in this and several other chapters.

The chapter is based on the following papers:

- Jan Kotera, Filip Šroubek, and Peyman Milanfar, “Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors,” in *Computer Analysis of Images and Patterns (CAIP)*, Richard Wilson, Edwin Hancock, Adrian Bors, and William Smith, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 59–66

The paper received *Outstanding contribution* award.

- Jan Kotera and Filip Šroubek, “Blind deconvolution of images with model discrepancies using maximum a posteriori estimation with heavy-tailed priors,” in *Digital Photography XI*, Nitin Sampat, Radka Tezaur, and Dietmar Wüller, Eds., International Society for Optics and Photonics, vol. 9404, SPIE, 2015, pp. 64–75

The paper received *Best student paper* award.

Chapter 4 largely addresses the same topic as the previous chapter, standard blind image deblurring in the presence of non-Gaussian violations of the convolution blur

model (1.3), but the class of admissible model violations as well as our approach to the deblurring problem are substantially generalized. We will present a blind image deblurring method based on variational Bayesian inference with a unified formulation that uses the same class of prior for each of the primary unknowns – image, blur, and observation error. Using sufficiently flexible noise prior allows us to automatically deal with arbitrary violations of the image acquisition model without having to tailor the method for each particular kind of model error separately, as in the previous chapter, and at the same time the noise model can again be used to solve the convolution boundary problem. As the prior distribution we use non-identical Gaussian with automatically estimated precision, also called the Automatic Relevance Determination model. We also investigate a slightly modified formulation with explicit modeling of the global Gaussian and local non-Gaussian component of the noise and automatic estimation of the degree of non-Gaussianness of the noise distribution. Our presentation is accompanied by a thorough analysis of both the image and the noise prior model and an experimental validation of their fitness for the studied problem. Blind image deblurring is then formulated as inference on the variational approximation of the posterior distribution. Corresponding solution is derived and presented in maximum detail and accompanied by an in-depth discussion and implementation suggestions. For reader's convenience the chapter also includes essential introduction to variational Bayesian inference and automatic relevance determination priors.

The chapter is based on the following papers:

- Filip Šroubek, Václav Šmídl, and Jan Kotera, “Understanding image priors in blind deconvolution,” in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct. 2014, pp. 4492–4496
- Jan Kotera, Václav Šmídl, and Filip Šroubek, “Blind deconvolution with model discrepancies,” *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2533–2544, May 2017

Chapter 5 addresses a subproblem that arises in evaluation of blind image deblurring methods following the traditional concept of first estimating the blur and then restoring the image, such as the methods presented in the preceding chapters. When evaluating or benchmarking a blind deblurring method, it is beneficial to evaluate not only the final output image but also the intermediate result of the blur estimation, as that is usually the critical part that determines potential success of the whole deblurring. In this chapter we propose a similarity metric between the estimated blur kernel and the known ground truth that can be used as a full-reference measure of accuracy of the blur estimation. The metric can be calculated using a simple, direct formula, for which we provide derivation and rationale. Unlike other commonly used and equally simple approaches such as direct mean squared error or cross-correlation of the blur PSFs, the proposed metric faithfully gauges the relative error of the image restoration caused by inaccuracy of the evaluated blur, without having to actually perform the image restoration. This approach has the benefit that different blur estimation methods are directly and absolutely comparable since the comparison is not dependent on the process of image restoration, which is not standardized.

The chapter is based on the paper

- Jan Kotera, Barbara Zitová, and Filip Šroubek, “PSF accuracy measure for evaluation of blur estimation algorithms,” in *2015 IEEE International Conference on Image Processing (ICIP)*, Sep. 2015, pp. 2080–2084

The paper received *Top 10% best papers* mention.

Chapter 6 is devoted to restoration of motion-blurred objects in an otherwise sharp image, a topic much less frequently addressed in literature compared to general image or video deblurring. We are interested in a particular phenomenon called *fast moving objects* (FMO) – objects that travel large distance during single frame exposure time and are therefore heavily blurred. In this chapter we formulate a problem called *FMO deblurring* (or alternatively *deblatting*) of recovering the object shape, unblurred appearance, and motion from a single blurred input. In its full generality such problem is quite difficult and we consider its suitable simplification that is general enough to encompass some useful real-world cases and interesting enough to reveal the characteristics of the general problem. In the formal theoretical analysis we limit ourselves to 2D objects undergoing arbitrary 2D motion and rotation, though we also present some experimental results on 3D objects. FMO deblurring is then formulated as a constrained optimization problem, we state necessary conditions for solution uniqueness, and present a solution accompanied by an extensive discussion of the theoretical properties of the problem as well as similarities and differences between FMO deblurring and standard blind image deblurring. As a practical application we show how the presented deblurring method can be used to computationally increase the acquisition framerate of video sequences containing fast moving objects.

The chapter is based on the following papers:

- Jan Kotera and Filip Šroubek, “Motion estimation and deblurring of fast moving objects,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct. 2018, pp. 2860–2864
- Jan Kotera, Jiří Matas, and Filip Šroubek, “Restoration of fast moving objects,” *IEEE Transactions on Image Processing*, vol. 29, pp. 8577–8589, 2020

Chapter 7 builds on the results of the previous chapter and explores how image deblurring can be utilized in the task of visual object tracking, in particular determining location and pose of fast moving objects with sub-frame frequency. Our premise is that if fast moving objects travel significant distance within a single frame, they must be localized by their intra-frame trajectory, instead of position. We focus mainly on balls and similar objects and make adjustments to the previously presented solution to FMO deblurring to make it suitable for the task of object tracking. This becomes the central component of the *tracking by deblatting* (TbD) method presented in this chapter. We provide its general high-level overview and then present in detail two versions of the TbD method differing in their target scenario, corresponding modifications of the deblurring step, and post-processing of the result of deblurring. In the first version we work with the assumption that any perceived 3D effects of the environment and motion of the object can be neglected and we formulate the tracking task as identification

of the perceived 2D intra-frame trajectory of the tracked object. The second version naturally builds on the first one and extends its non-blind deblurring step by partially relaxing the assumption of constant object appearance during the whole exposure time of the frame. This allows us to estimate the object appearance with sub-frame sampling rate and extend the method to objects undergoing 3D rotation. At this point we limit ourselves to tracking of balls and use the recovered appearance snapshots to determine the ball 3D rotation and relative distance from the camera, which in effect allows us to track the object with sub-frame rate in full six degrees of freedom. In each case we present the appropriate modifications of the general FMO deblurring problem in full detail and provide corresponding solution, traditionally accompanied by discussion and experimental demonstration of the efficacy of the presented method.

The chapter is based on the following papers:

- Denys Rozumnyi, Jan Kotera, Filip Šroubek, Lukáš Novotný, and Jiří Matas, “The world of fast moving objects,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 4838–4846
- Jan Kotera, Denys Rozumnyi, Filip Šroubek, and Jiří Matas, “Intra-frame object tracking by deblatting,” in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct. 2019
- Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas, “Non-causal tracking by deblatting,” in *Pattern Recognition, DAGM GCPR 2019*, Gernot A. Fink, Simone Frintrop, and Xiaoyi Jiang, Eds., Cham: Springer International Publishing, 2019, pp. 122–135

The paper received *Honorable mention* award.

- Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas, “Sub-frame appearance and 6D pose estimation of fast moving objects,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 6777–6785
- Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas, “Tracking by deblatting,” *International Journal of Computer Vision*, 2020, (revision under review)

Chapter 2

Related work

Image deblurring has been an active topic of research for decades and finds its applications in many practical fields, such as astronomy, medical imaging, remote sensing, microscopy, optics, forensic science, cultural heritage preservation, or photography among others. Early techniques have their roots in classical signal processing, such as inverse filtering or Wiener filtering. Specialized image methods began appearing in 1950s [107], though the topic started receiving considerable attention in 1960s and onwards [172, 62, 159, 105, 145, 71, 161].

Significant boost in popularity as well as performance of the proposed methods occurred around the year 2000 with the works of Miskin and MacKay [111] and later Fergus *et al.* [47] based on heavy-tailed distribution of image derivatives and variational Bayes method, accompanied by a plethora of maximum a posteriori methods proposing various techniques to overcome the problem of getting trapped in the no-blur solution [29, 157, 191]. Surprisingly, some of these methods are very good and to this day keep frequently appearing in evaluation of novel works in the latest computer vision conferences and journals, performing well against the state of the art. Interest in image deblurring keeps increasing and many survey papers or dedicated monographs have been compiled [5, 76, 89, 61, 140, 181, 19]. Nowadays, with ubiquitous presence of digital cameras and massive growth of the use of images and videos in all aspects of everyday life, this scientific discipline is at its prime. For example, Scopus database returns approx. 2800 unique scientific papers with the keyword “image deblurring” published in the last three years alone and the numbers are growing each year.

In this chapter we briefly review the most important works in image deblurring with emphasis on methods focusing on similar problems that we address in the thesis. The reviewed methods are roughly categorized by their primary principle or focus, although many works employ hybrid approaches that borrow from several categories while other methods do not belong to any defined category (e.g. [53]). The amount of literature on image deblurring is overwhelming and we are under no illusion that the list of works provided here is in any way exhaustive. We do not further mention works by the author of the thesis listed in Sec. 1.4 as the foundation of the thesis.

2.1 Maximum a posteriori estimation

By far the most common approach to blind image deblurring is maximum a posteriori (MAP) joint estimation of blur and sharp image from the blurred input. To be able to talk about the principles employed by different deblurring methods, we first need to establish the essential basics of MAP image deblurring, which are then revisited in more detail in the main body of the thesis.

In vast majority of works, image blur is modeled according to (1.3) as a convolution of the latent sharp image u with an unknown blur h ,

$$g = h * u + n, \quad (2.1)$$

where g is the resulting observed image (input to deblurring) and n is observation noise, usually considered zero-mean Gaussian, though less frequently also more general observation error model appears. Some methods work with the spatially-variant blur model (1.2)

$$g = Hu + n, \quad (2.2)$$

where the blur operator H is expressed as a weighted combination of a finite number of fixed linear operators $H^{(i)}$, corresponding e.g. to discretized space of camera poses,

$$H = \sum_i h_i H^{(i)} \quad (2.3)$$

and the unknown weights h are then subject to blind estimation. The model remains linear and efficient optimization of such formulation was proposed e.g. by Hirsch *et al.* [65] and at similar time by other authors [60, 69, 186]. As a result, spatially-variant blur formulation can be regarded as a relatively straightforward extension of most methods based on the convolution formulation (2.1).

By considering the unknown sharp image u and blur h as random variables with probability distributions denoted $p(u)$ and $p(h)$, respectively, MAP inference dictates that the estimates of u and h are maximizers of the posterior probability,

$$(u, h) = \underset{u, h}{\operatorname{argmax}} p(u, h | g) \quad (2.4)$$

and by Bayes rule we have

$$p(u, h | g) \propto p(g | u, h) p(u) p(h). \quad (2.5)$$

By minimizing its negative log, we arrive at the general structure of the objective of MAP estimation methods,

$$(u, h) = \underset{u, h}{\operatorname{argmin}} L(u, h), \quad (2.6)$$

$$L(u, h) = D(u, h) + R_u(u) + R_h(h).$$

The minimized loss L consists of a *data term* D , corresponding to the likelihood $p(g | u, h)$, and *regularizers* R_* , corresponding to the prior distributions $p(u)$ and $p(h)$. Note that we use the term MAP very loosely in this section and call many method “MAP-based” if their objective has approximately the form of eq. (2.6), even though it does not correspond to any probabilistic formulation.

Blind image deblurring based on MAP then consists of two steps:

1. Alternating minimization of L with respect to u and h in order to establish the blur estimate h . The image estimate u at this step is called *intermediate image* and serves only to estimate h , at the end of this step it is discarded.
2. Possibly change parameters or slightly modify L and find the final image estimate u as the minimization of L with h from the previous step fixed.

The notion of the intermediate image, which is a deviation from the basic MAP paradigm, and its conceptual difference from the final restored image will become clear in the upcoming discussion.

No-blur solution

As already discussed in the introductory chapter, major problem of all MAP-based blind deblurring methods is their tendency to converge to the dreaded no-blur solution, for which $h = \delta$ and $u = g$. This is unavoidably strong, global minimum of the data-term D and in many cases also a local minimum of both R_u and R_h . Primary contribution of different MAP deblurring methods often lies in the proposed techniques of avoiding the no-blur solution and converging to the correct minimum. Although the list of proposed solutions is long, some dominant and recurring trends can be identified. Theoretical analysis and some interesting observations on MAP deblurring and the no-blur solution can be found in Perrone and Favaro [130], [131] and from a different perspective in Cho and Lee [30].

Unnatural priors

One of the first successful blind deblurring methods is the well-known total variation method by Chan and Wong [23] with image prior based on image gradients as $R_u(u) = \|\nabla u\|_1$. Using priors defined in terms of image gradients and enforcing their sparsity has since become extremely popular, though other sparsifying linear transforms have been proposed as well, such as for example wavelets and their siblings [17, 18, 48, 135, 178] or dictionaries [20, 46, 93, 104, 204].

To force the optimization leave the no-blur solution, more aggressively sparsity-promoting, nonconvex priors were proposed, such as the log prior of the form

$$R_u(u) = \sum_i \log((\nabla u_i)^2 + \epsilon^2) \quad (2.7)$$

by Perrone and Favaro [132]. More popular is the ultimately sparse ℓ^0 prior $R(u) = \|\nabla\|_0$ proposed in its relaxed, continuous form

$$R_u(u) = \sum_i \min((\nabla u_i)^2/\epsilon^2, 1) \quad (2.8)$$

by Xu *et al.* [193]. This prior was adopted by many other papers [127, 128, 124, 197], though some use ℓ^0 directly without relaxation.

The general idea is that by using “unnatural” priors that far exaggerate the true sparsity of natural image gradients, the intermediate image is forced to develop strong

edges connected by nearly constant regions. Such image is then very different from the input g and the trivial no-blur solution $h = \delta$ no longer minimizes the data term. This is the first example of a recurring pattern that stepping outside the procedure corresponding to strict MAP optimization is embraced and considered necessary to avoid the no-blur solution or improve quality of the result. Levin *et al.* [96] pointed out that using regularizers corresponding to true priors of natural images is insufficient and for blur estimation they suggested marginalization of the latent image in the posterior (see introduction to Chapter 3). Perrone and Favaro [130] observed that natural priors work when combined with adjustment of the regularizer weight during optimization. Since the no-blur solution is often the global minimum of the optimized loss L , this implies that even if blind deblurring empirically works, one cannot rely on any theoretical convergence guarantees.

It was proposed by Krishnan *et al.* [88] to normalize gradient-based natural image priors to compensate for the fact that they favor images with low local variance, e.g. blurred. Their proposed image prior has the form $R_u(u) = \|\nabla u\|_1 / \|\nabla u\|_2$. In each iteration of the optimization such prior is convexified by approximating the denominator by its value evaluated at previous iteration. In effect, this reduces to the on-the-fly adjustment of the regularizer weight used previously by other methods.

New regularization terms were proposed to actively fight the no-blur solution by favoring sharp images. One of them is the so-called *dark channel* prior, originally proposed for image dehazing [63] and adopted for blind image deblurring by Pan *et al.* [128]. The image dark channel u_i^{dark} of a pixel i is defined as a minimum intensity value of all pixels (incl. color channels) in its fixed neighborhood N_i ,

$$u_i^{\text{dark}} = \min_{j \in N_i} \left(\min_{k \in \{R,G,B\}} u_{j,k} \right). \quad (2.9)$$

For natural sharp image the values of u_i^{dark} are low and sparse, but increase (in sum) and lose sparsity with blur. This is exploited in [128] where regularizer of the form $R_u(u) = \|u^{\text{dark}}\|_0$ is shown to aid in avoiding the no-blur solution. This idea was quickly adopted by other authors. Yan *et al.* [197] replaced min with max, call the result *bright channel* and use regularizer of the form $R_u(u) = \|u^{\text{dark}}\|_0 + \|1 - u^{\text{bright}}\|_0$. In Chen *et al.* [27] authors use the exact same expression expressed in gradient domain and propose corresponding *local maximum gradient* prior.

Unfortunate property of all regularizers of this type is that for the purposes of optimization, the min function (analogously max) is linearized in each iteration by fixing the position (index) of the pixel which attains minimum based on the current value of the operand. More explicitly, in each iteration the min function is approximated as $\min u \approx m^T u$ where m is a one-hot vector with the “1” in the position where u currently attains minimum, i.e. $m_i = 1$ if $i = \operatorname{argmin}_i u_i$, where in this example u is the local image neighborhood on which the proposed regularizer operates. The approximated functional is then minimized while making very large steps between iterations (using exact solutions of linear systems instead of small gradient-descent steps) and it is not at all clear that the optimization actually converges to the minimum of the original objective. “Convergence is verified empirically,” [128].

Intermediate image

Another technique frequently used to avoid the no-blur solution is exploiting the fact that during the blur estimation phase, the current image estimate does not necessarily have to correspond to the original sharp image that is the ultimate goal of the deblurring. By tampering with the intermediate image estimate, the optimization can be steered away from the no-blur solution and the blur estimated correctly. This is another example of deviation from the MAP probabilistic paradigm and one that lacks solid theoretical backing, but admittedly works well in practice.

The usual technique, introduced by Cho and Lee [29] and further popularized by Xu and Jia [191], works as follows. During the kernel estimation phase, the current image estimate is further processed and in each iteration, small texture is removed while salient edges are enhanced (both [29] and [191] use shock filtering, though [191] employs different strategy in the initial edge selection scheme based on spatial distribution of edges in the image). When the optimization loss L in eq. (2.6) is then minimized with respect to h , the current blur estimate is forced to leave the $h = \delta$ solution, as it no longer presents a minimum of the data term (in this case, the data term is usually expressed in filter domain, though it is only a numerical advantage rather than principal difference). This process is repeated until the blur estimation converges, then the final image restoration is performed with realistic parameter setting and without additional tampering. Unnatural priors discussed above achieve the similar effect without the need for external filtering.

Near convergence of the blur estimation (when between-iteration difference is negligible), if u is current intermediate image and \hat{u} the same image after the edge-enhancing filtering, then it should hold that $Hu \approx g$ and also $H\hat{u} \approx g$, which implies that $H(u - \hat{u}) \approx 0$, which contradicts the energy preservation of H , unless $u \approx \hat{u}$. In other words, this technique converges only if the edge-enhancement is identity for some non-trivial (i.e. not constant) class of images, which is unfortunately rarely satisfied in practice. Nevertheless, the technique has its merit and keeps reappearing even in newly proposed methods. It could possibly be exploited more, if its mechanism were better understood.

In Sun *et al.* [163] authors used a dictionary of learned as well as hand-crafted priors of sharp edge patches to identify the most salient edges in the intermediate image and remove the rest. Lai *et al.* [92] instead use the color-line model of image patches [121] to determine salient edges. Gong *et al.* [54] base the edge selection scheme on a combinatorial task of optimally choosing a fixed number of salient edge pixels. The authors admit that early stopping is necessary, otherwise the method diverges. Different approach was taken by Zhou and Komodakis [212], who propose extra prior terms for the intermediate edge image based on assumed geometry of the scene, for example that edges can be grouped into lines. In Xu *et al.* [195], convolutional neural network (CNN) is trained to predict a suitable intermediate image containing only salient edges. No claims of optimality (with respect to blur estimation) of the predicted image are made and the network is trained simply on filtered pairs of blurred-sharp images. The predicted intermediate image is then used to initialize a regular MAP optimization with ℓ^0 image prior. Edge-enhancing in the intermediate image to facilitate blur estimation was utilized in various mutually similar forms in other works [125, 126, 209]. Zuo *et al.* [215] use an approach that lies on the boundary of intermediate image enhancement

and use of unnatural priors and partially addresses the problem of convergence. They use image regularizer of the type $R_u(u) = \alpha \|\nabla u\|_p^p$ and learn how to optimally vary the values of α and p per iteration to go from unnatural prior, that provides good blur estimate, to natural prior such that the process converges. They extend their method in [216] where they work with negative p values and learn the optimal parameter choice on a training set. On a somewhat related topic, Hu and Yang [70] trained a conditional random field to predict which regions of the input image are most likely to yield good blur estimate.

Natural priors

Another area of research that is actively pursued is improving the visual quality of the restored image. Results of image deblurring often suffer from unpleasant ringing artifacts caused by inaccurate blur estimation or other infidelity of the image formation model. This can be remedied either by making the deblurring more resistant to model violations (discussed later), by better image priors that favor naturally looking images, or by active suppression of ringing artifacts.

Zoran and Weiss [214] proposed a general MAP framework for image deblurring with priors formulated in terms of image patches (as their expected log likelihood). They learn a Gaussian mixture model of natural image patches and show that such prior outperforms hard-coded state of the art and produces better-looking results. Deblurring of class-specific images (e.g. faces) was addressed by Anwar *et al.* [6], who proposed an image prior formulated in the Fourier domain, where the frequency bands of the restored image are expressed as a sparse combination of the training set examples.

Frequently exploited property of natural images is their self-similarity (recurrence of image patches) across scales, used previously e.g. for super-resolution [52] or denoising [213]. Michaeli and Irani [110] show that this property diminishes with blur and propose a blind deblurring method with image regularizer that penalizes ℓ^2 difference between each patch of the latent image estimate and its nearest neighbor in a downsampled version of this image. The operator performing patch-matching is linearized in each iteration. Similarly, Peyré *et al.* [134] propose using similarity of non-local patches within a single scale as a regularizer, an idea exploited in different forms by other authors [43, 10, 175, 200, 208]. For example, instead of ℓ^2 difference Ren *et al.* [143] form matrices from non-local image patches and penalize their rank (nuclear norm) as another way of enforcing image self-similarity.

One of the first sufficiently general attempts to improve the visual quality of the restored image and actively suppress ringing artifacts were made by Shan *et al.* [157] and Yuan *et al.* [199]. In [157] authors detect smooth regions in the input image g and in these regions enforce smoothness of the restored image by regularization term of the form $R_u(u) = \|\nabla u - \nabla g\|_2^2$. In [199] ringing suppression is the primary focus, authors extend the Richardson-Lucy algorithm by a regularization term enforcing image smoothness in a local Gaussian-weighted window and then propose multiscale non-blind deblurring algorithm which further enforces similarity of the restored image in the new scale to upsampled restoration from the previous scale. Mosleh *et al.* [113] propose removing ringing artifacts by postprocessing of the restored image. They analyze the Fourier spectrum of the blur and construct filters corresponding to zeros in

this spectrum. Then they filter the restored image by, effectively, a denoising method such that response to these filters is minimized. More recently, Mosleh *et al.* [114] propose a no-reference metric for ringing detection based on comparison of filtered blurred input and the restored latent image. They incorporate this metric directly into the deblurring loss as a regularizer. Still in non-blind setting, Schuler *et al.* [153] perform simple and fast deblurring by direct inverse and then learn a CNN to remove the remaining restoration artifacts. The same idea was later revisited by Wang and Tao [182] or Vasu *et al.* [176] who predict the residual between deblurred image and true sharp image. Several works explicitly addressed ringing caused by inaccuracy of the blur estimation [72, 133, 142].

Plug and play priors

The deblurring optimization procedure often splits into several subproblems, each corresponding to one of the terms in the minimized loss. This motivated Venkatakrisnan *et al.* [177] to propose a framework where different priors or regularizers can be easily incorporated into existing restoration methods (hence the name “plug and play”), often in the form of denoisers, since the update step corresponding to the minimization of the regularization term can be viewed as a denoising operation. Using general denoisers for image deblurring is not a novel idea per se and was proposed e.g. by Danielyan *et al.* [37], who proposed image regularization using the well-known BM3D denoiser [33].

General formulations similar to [177] soon followed and unleashed the trend of using also priors that cannot be explicitly expressed in the original formulation of the objective and are present only via their proximal mapping (analogous to the kernel trick in SVMs). Chan *et al.* [22] or Teodoro *et al.* [167] base their approach on ADMM, provide convergence analysis, and use BM3D and Gaussian mixture models of patches, respectively. Ljubenić and Figueiredo [102] extend the method of [167] to the blind case. Tirer and Giryes [170] (see also references therein) use slightly modified formulation and use BM3D denoiser or a pretrained CNN denoiser. Romano *et al.* [146] provide approach that does not rely on ADMM. Brifman *et al.* [15] used ADMM and the denoiser by [43] for super-resolution. Xiao *et al.* [189] work with half-quadratic splitting and learn a recurrent network to act as proximal operator for fixed number of iterations in a general image restoration task. Similar methods with minor differences (e.g. different splitting or network architecture) were proposed by other authors [79, 205, 207, 100, 42]. Instead of a denoiser, Li *et al.* [97] learn a binary classifier between blurred and sharp image patches, which they incorporate into a MAP method. Pan *et al.* [122] extend the plug and play idea also to the likelihood term and propose a corresponding blind deblurring method; the same idea was pursued by others [2, 41].

Outlier handling

It is well known that outliers to the linear blur model present in the input image significantly decrease accuracy of the blur estimation and cause ringing artifacts in the deblurring result. On top of using more robust data term, such as ℓ^1 instead of quadratic (e.g. [191]), efforts have been made to further increase robustness of deblurring methods by explicit identification and rejection of outliers. At the end, virtually all MAP

approaches to outlier handling are effectively based on weighting (either binary or continuous) of pixels in the quadratic data term, although some methods are initially formulated differently.

Cho *et al.* [31] propose a non-blind method based on expectation-maximization, in which the M-step corresponds to the latent image estimation and E-step corresponds to inlier mask estimation. Outliers are effectively identified based on the blur model residual $r_i = (h * u - g)_i^2$. Whyte *et al.* [185] modify the non-blind Richardson-Lucy algorithm to consider saturated pixels, identified by direct thresholding of the input. Hu *et al.* [68] proposed a blind method for pictures captured in low light and containing streaks of saturated pixels (e.g. light sources in the scene). These streaks are used to estimate the blur PSF, outliers are found very similarly to [31] and for the final non-blind image estimation authors use slightly modified method [185]. Blind method very similar to [191] (e.g. with shock-filtered intermediate image) was proposed by Pan *et al.* [125] in which pixels in the data term are weighted according to the sigmoid of their residual, $w_i = 1/(1 + \exp(r_i^2))$. Minimally modified approach was proposed later by similar author team [40], this time they do not use shock-filtering and propose a robust non-quadratic and non-convex data term, but for optimization they use reweighted least-squares, which the method effectively reduces exactly to [125]. Gong *et al.* [55] formulate outlier detection as a graph problem and proceed in a coarse-to-fine manner by decreasing the threshold for certainty of which pixels are considered inliers. Xu *et al.* [192] train a relatively small, 5-layer CNN for non-blind deblurring that contains outlier-rejection module. They exploit singular-value decomposition of the known inverse kernel to decrease the network size, but the network has to be re-trained for each new blur kernel. Nonlinearity of the camera response function was studied by Tai *et al.* [164].

Several methods were proposed to cope with strong, albeit Gaussian, noise in the input image. See e.g. Zhong *et al.* [211], who use directional filtering of the input image, or Jin *et al.* [75], who use interesting formulation which includes CNN.

Boundary problem of the convolution formulation was addressed by several works, initially by Liu and Jia [99] who propose working with mirror extension of the image, which is rather inefficient, and later more efficiently in a series of nearly identical methods based on variable splitting by Matakos *et al.* [109] or Almeida and Figueiredo [3], [4]. Different approach was proposed by Portilla [136] where the input image is extended in such a way that its autocorrelation is retained. Despite this effort, the boundary problem often remains ignored and new methods are being proposed with direct inversion of the blur operator in the Fourier domain without any care for the model fidelity near boundaries, e.g. [27, 129].

2.2 Variational Bayes

Result of MAP is the maximizer (i.e. point estimate) of the intractable posterior distribution $p(u, h|g)$. In contrast, Variational Bayes (VB) seeks the optimal approximation of the whole posterior by some tractable distribution, the restored image is then e.g. expectation calculated on this distribution. VB is often viewed as an alternative to MAP with two primary advantages: automatic estimation of model parameters, that are nearly always user-specified in MAP, and feasible marginalization of the posterior.

Marginalization was popularized by the famous paper by Levin *et al.* [96] where authors suggested that in order to escape the no-blur solution in the blur estimation, one should maximize the posterior $p(h|g)$, which is the joint posterior marginalized over latent images (see introduction to Chapter 3). The same author team then proposed a solution [95] which uses Gaussian mixture of image derivatives as image prior and in the alternating minimization the latent image is estimated in terms of its Gaussian distribution (i.e. including covariance), which is then used in the blur estimation step for said marginalization. Different analysis of joint MAP and marginalized MAP is available in Wang *et al.* [179], where authors discuss sharpening of the intermediate image in blur estimation phase and its analogue in marginalized MAP.

For other than Gaussian distributions, though, direct marginalization is intractable analytically. Variational Bayes is widely considered as a viable approach because it approximates the posterior by a factorizable distribution (if the predominantly popular mean-field approximation is used) and the marginalization is then trivial and cost-free. However, this is possible only due to relatively drastic approximations. It is questionable how much the theoretical advantages of the marginalization are then retained [187].

The mathematical apparatus required for VB formulation is more complex than its MAP counterpart and the iterative update equation are less easily derived and solved if the model contains non-trivial terms intended to improve the performance of the method, such as most of the techniques reviewed in the previous section. This is likely the reason why VB formulation of blind deblurring has not recently enjoyed the popularity of MAP or deep learning based methods, although in all likelihood the decline is only temporary. Review of the general principles of VB inference is provided in Chapter 4, Sec. 4.3. Review of VB in the context of image deblurring is available in Ruiz *et al.* [151].

Successful VB methods for blind image deblurring appeared not long after the first MAP attempts [111, 98, 47]. Fergus *et al.* [47] proposed using Gaussian mixture as image prior to capture statistics of natural images and mixture of exponential distributions as blur prior to enforce sparsity and their method arguably marked the boom of VB in image deblurring. Due to the nature of VB framework, all VB-based deblurring methods are very similar to each other and differ mostly in details such as particular definition of prior distributions or formulation in intensities or filter space.

Molina *et al.* [112] use Gaussian distributions of second derivatives as priors of image and blur. Babacan *et al.* [8], [9] use i.i.d. total variation image prior and automatically estimate its global precision (weight). Chantas *et al.* [24], [25] use locally-weighted (non-i.i.d.) sparse image priors (student- t or total variation, respectively). Tzikas *et al.* [173] proposed locally-weighted prior of the noise prior (likelihood). Framework for general sparse image priors was formulated by Babacan *et al.* [7]. Spatially-variant blur under VB framework with locally-weighted priors is addressed by Zhang and Wipf [202] accompanied by a discussion on the VB approach in general, which later resulted in their theoretical work [187]. Sparsity of groups of wavelet features was used as a prior in [201]. More recently, Ševčík *et al.* [156] extend the spatially-weighted Gaussian image prior, commonly formulated in derivative space, and consider the sparsifying filters as unknowns which are also estimated from the data. Serra *et al.* [155] use mixture of a delta and a flat Gaussian (sort of an ℓ^0 prior) as

image prior and interestingly, their posterior calculation is not based on the prevalent mean-field approximation. VB approach was used also for deblurring under mixed Poisson-Gaussian noise [108] or for blind color deconvolution [64]. Interesting approach was proposed by Yang and Ji [198], which combines standard VB formulation with intentional sharpening and selection of salient edges in the intermediate image during blur estimation phase, which is routinely used in MAP approaches to facilitate blur estimation, as discussed above.

Critical analysis and a fresh look at VB and its relation to MAP was provided by Wipf and Zhang [187]. They show that under certain conditions, the VB framework can be reformulated as a MAP problem with an extra regularization term connecting the image and the blur. They further show that the widely believed reason for superiority of VB regarding no-blur solution is unsubstantiated and vanishes due to the requisite approximations in the VB optimization. For other conclusions, such as analysis of natural and unnatural image priors and their suitability for blur estimation, we refer the reader to [187].

2.3 Dynamic scene deblurring

So far we reviewed works addressing mostly uniform image blur modeled by convolution, though some of the work explicitly work with spatially-variant blur expressed in terms of basis blurs (2.3). In the second part of this thesis we address the problem of deblurring a motion-blurred object in otherwise sharp image. This is an example of non-uniform motion blur, i.e. blur that is particular only to the one image component (do not confuse with spatially-variant blur of the whole image). In literature, similar scenario can be found under different names such as non-uniform motion, heterogeneous motion, object motion, or dynamic scene deblurring. Although our scenario is a bit different than what is usually considered in published works, we now briefly review other works focusing broadly on this type of problem.

Existing works could be categorized into three predominant categories. Fully classical methods based on forward blur model [77, 78, 123, 139], mixed classical-learned methods [162, 56], and fully end-to-end deep learning methods [115, 166, 206, 50]. After early initial attempts e.g. by [34, 35] who estimated motion blur from pre-computed alpha matte of the blurred object, the main boom of dynamic scene deblurring began with the work of Kim *et al.* [77], who use weighted sum of the traditional data terms $\|h * u - g\|_2^2$ to represent a fixed number of image layers (e.g. moving objects and background) and estimate weights of these layers, effectively performing a segmentation of the input image. They relaxed this approach in their subsequent work [78] where they got rid of the segmentation and estimate a dense motion field. Segmentation of the image was used also by Pan *et al.* [123], where binary segmentation is augmented by its real-valued confidence map, which is estimated using similar equations that appear in image matting. The method also assumes fixed number of objects and needs to be manually initialized by an approximate segmentation of the objects (e.g. by convex hull). Qiao *et al.* [139] use image segmentation represented in terms of super-pixels and estimate blur in these regions. Interesting approach to estimate motion blur field was proposed by Bahat *et al.* [11], who analyze the spectral content of image patches before and after blur to determine blur direction of each patch.

Motion field is used also by Sun *et al.* [162], who estimate patch-wise motion vectors by a classification CNN. Dense motion field is then obtained via Markov random field. Gong *et al.* [56] use similar approach but they estimate dense motion field directly using a CNN. Both methods use the classical non-blind method by Zoran and Weiss [214] for the final image restoration. Classification network to determine patch blur was used also by Yan and Shao [196], who included also defocus blur. Many of these methods assume that the blur model is separable and the input consists of independently blurred image regions (objects), they do not properly consider the semi-transparency and blending of image layers at boundaries of objects in their forward model. Scenario that is quite similar to what we address in this work was recently addressed also by Dong *et al.* [39], albeit formulated for out-of-focus blur instead of object motion.

End-to-end deep learning approaches differ in particular details of the used architecture a corresponding training procedure. Nah *et al.* [115] simulate the traditional multiscale approach using CNN trained with loss that mimics the conventional coarse-to-fine approach. This approach was improved upon by Tao *et al.* [166] by introducing weight sharing between scales. Multiscale restoration approach was used also by Noroozi *et al.* [119]. Zhang *et al.* [206] used complex pipeline of networks for end-to-end image restoration. Refined network architecture was recently proposed by Gao *et al.* [50]. Other proposed end-to-end methods include e.g. encoder-decoder network (Liu *et al.* [101]), or GANs ([58]) to generate good intermediate edge image (Zheng *et al.* [210]).

Visual tracking of motion-blurred objects was explicitly addressed by several works [36, 188, 165, 38, 190]. Estimation of local linear motion from blur was proposed e.g. in [51]. The reverse task of determining camera motion from a blurred image was considered e.g. in [118].

As an application of our object deblurring method presented in the second part of the thesis we introduce the concept of *temporal super-resolution* of a single image or a sequence – reconstructing the object motion by splitting its trajectory into several short segments and resynthesizing a blur-free sequence corresponding to acquisition by a high-speed camera. The same idea was considered in literature quite recently in the work of Jin *et al.* [74], who learned an end-to-end deep CNN with loss invariant to temporal order of the reconstructed video frames. The same authors augment their approach in [73] by another deep network that performs interpolation between recovered frames and improves smoothness of the generated sequence, the presented results are impressive. Different approach is proposed by Purohit *et al.* [138], who extract motion information from the blurred frame by a deep network and pass this to the neck of a pre-trained video autoencoder.

2.4 Deep learning

After the explosive success of deep learning in image recognition in 2012, convolutional networks (CNN) naturally entered also the world of image deblurring. Since deep learning based methods are of relatively minor relevance to this work and the

literature is vast (especially if considering relevant super-resolution methods), we will only briefly mention some selected works and approaches. Many deep learning methods were already mentioned in the preceding sections.

One of the earliest and pre-AlexNet approaches was done by Aizenberg *et al.* [1] (see also references therein to author's previous work) who used multi-valued neurons to determine blur type and estimate its parameters. Many other initial approaches followed similar path and use CNNs only for blur estimation, such as already discussed works [56, 162, 195, 196]. Chakrabarti [21] learn to estimate the inverse blur filter in each image patch and then re-estimate one global blur for the whole image. For the final image restoration they use the classical non-blind method [214]. Other authors used deep learning for the non-blind image restoration [153, 192, 182]. The drawback of e.g. [153, 192] is that the network has to be retrained for each new blur PSF, this was addressed by Ren *et al.* [144] who generalize the idea of blur decomposition introduced by [192] by working with a training set of general blurs. Entirely different approach is proposed by Gong *et al.* [57], they train a CNN to predict gradient steps of a traditional optimization rather than the direct result image.

Arguably the most popular is the end-to-end approach, where the sharp image is predicted by the network directly from the blurred input [67, 119, 115, 166, 206, 50, 137]. Schuler *et al.* [154] proposed a network that mimics the traditional blind deblurring approach. The network consists of two iterations of a feature extraction, blur estimation, and latent image estimation modules and convolution blur model is enforced in the optimized loss. Nimisha *et al.* [117] train an encoder-decoder network to learn representation of sharp image patches and separately a GAN generator ([58]) that predicts such representation directly from a blurred patch. The representation is then supplied to the decoder. GANs are commonly used as implicit image priors in other end-to-end approaches, such as in the well received DeblurGAN by Kupyn *et al.* [90], [91], where generator is learned to predict sharp image directly from the blurred input, or previously in a similar manner by Ramakrishnan *et al.* [141].

Another popular and promising way of utilizing deep learning in image deblurring is to train the neural network to act as an explicit or implicit image prior that replaces hand-crafted regularization terms in traditional optimization methods. Several of such approaches were already mentioned in preceding section on plug and play priors [42, 79, 97, 100, 170, 189, 205, 207]. We will mention the work by Ulyanov *et al.* [174] which created quite a buzz and ignited further research. The authors show that even an untrained generator network can successfully serve as an effective image prior. They attribute this property to the structure of the network rather than its ability to learn from training data. For more information see also the analysis by Cheng *et al.* [28].

Part I

Blind image deblurring with acquisition model violations

Chapter 3

Maximum a posteriori deblurring of images with overexposure

In the introductory Chapter 1 we repeatedly stressed and demonstrated that image deblurring formulated as an inverse problem is over-reliant on the accuracy of the acquisition model. If the actual image acquisition deviates from this assumed model, visible artifacts appear in the restored image. The same holds for blind blur estimation – if the assumed model is violated in some parts of the image, the blur estimation is typically either inaccurate or fails entirely, which further negatively propagates to the subsequent non-blind image restoration. A seemingly possible solution is to generalize the model and include all possible phenomena that occur during the image acquisition, but inverse of such process and blind estimation of the blur soon become computationally intractable. In this and the next chapter we analyze the problem at greater depth and propose an extension of the conventional approach to blind image deblurring that automatically identifies such local model violations and mitigates their adverse impact while retaining the simple yet practically useful convolution blur model.

Let us assume the standard spatially-invariant image blur model, in which the observed image g is a result of convolution of a latent image u with an unknown blur h and an additive perturbation n ,

$$g = h * u + n. \quad (3.1)$$

The observation error n is customarily called “noise”, but we emphasize that a more general error than purely random noise with fixed distribution is expected. Our goal is to recover u and h solely from the given blurry image g and as independently of n as possible (i.e. output the same (u, h) for $\|n\| = 0$ and $\|n\| > 0$).

One of the most common approaches is to formulate blind image deblurring as a maximum a posteriori probability (MAP) estimation of u and h for the given g . Let all the variables under consideration be 2D discrete random fields with corresponding prior distributions denoted (with slight abuse of notation) by $p(u)$, $p(h)$, and $p(n)$. Maximum a posteriori inference dictates that estimating the pair (u, h) amounts to maximizing the posterior probability distribution $p(u, h|g) \propto p(g|u, h)p(u)p(h)$. It has been advocated by Levin *et al.* [96] that to avoid trivial local solutions, the better approach is to estimate the blur alone from the posterior marginalized over all possible

images, i.e. maximize

$$p(h|g) = \int p(u, h|g) du, \quad (3.2)$$

and then proceed with the non-blind image estimation. In practically interesting cases this is directly intractable and requires some approximation, but both joint estimation as well as approximated marginalization are viable approaches and are used in literature. In this chapter we will focus on the former and formulate image deblurring as a simultaneous (u, h) estimation. The next chapter will present a method that effectively achieves the more mature marginalized formulation.

The acquisition model (3.1) is used by vast majority of blind image deblurring methods (optionally generalized to spatially-variant blur, see Sec. 2.1), but unless explicitly addressed, it is virtually always assumed that the observation error n is an i.i.d. zero-mean Gaussian noise,

$$p(n) = \prod_i \mathcal{N}(n_i|0, \sigma^2). \quad (3.3)$$

In real-world scenarios, the error comes from multiple different sources, for example sensor noise, digital quantization, sensor saturation (overexposure), dead pixels, blur spatial-variance (e.g. optical aberrations around image boundary), scene variation (objects moving during exposure) to name a few. Many of these are local phenomena so the total observation error will not be identically distributed. Most blind deblurring methods do not take any extra measures to deal with such model violation and the blur estimation typically fails when this effect becomes non-negligible.

Depending on the exact formulation, non-Gaussian model errors may actually appear without any actual physical cause due to the convolution boundary problem discussed on page 5 in the introductory chapter. If the acquisition model (3.1) sacrifices fidelity and for the sake of numerical convenience uses the circular convolution definition illustrated in Fig. 1.3a, it is usually not correct around image boundary, which may lead to poor deblurring results as demonstrated in Fig. 1.4.

In this chapter we will focus on a particular kind of non-Gaussian observation error – image intensity clipping due to sensor saturation in overexposed image areas. Photographs captured in low-light conditions are susceptible to motion blur due to long exposure time but at the same time, bright parts are often overexposed. Intensities higher than the acquisition device can capture are clipped to maximum measurable value. This clipping, however, occurs *after* the blurring, which is a violation of the linear model (3.1). We will present a method that automatically identifies the overexposed regions and removes them from the blur estimation. The same technique is used to solve the convolution boundary problem without sacrificing model fidelity or computational efficiency.

The presented method is a simultaneous h and u MAP estimation from the given blurred input g , formulated as a minimization of $-\log p(u, h|g)$ with respect to u and h . Such direct approach has the unfortunate property of easily getting trapped in a local minimum which does not correspond to the true sharp image u . As already discussed in the previous two chapters, most troublesome is the no-blur solution $h = \delta$ and $u = g$, in which the output of deblurring is trivially the blurred input. Many

published methods address this problem by performing additional steps on top of the basic MAP procedure and these steps are designed to steer the optimization away from the trivial minima (see “No-blur solution” on page 17 in Sec. 2.1). Partial contribution of our approach is demonstrating that these empirical steps are not strictly necessary and if well formulated and carried out, even direct alternating MAP approach results in an effective blind deconvolution algorithm that is able to outperform more complex methods.

Our approach is based on image prior $p(u)$ that models the distribution of image gradients and is more heavy-tailed than Laplace distribution. This better approximates statistics of natural images as well as reduces the problem of getting trapped in the no-blur solution. In addition, data term masking inspired by Matakos *et al.* [109] or Almeida and Figueiredo [4] is included in the formulation which elegantly solves the problem of image boundaries in convolution. This masking approach is extended to deal with overexposed regions and we evaluate experimentally that this increases robustness of blind deconvolution to these model violations. The resulting algorithm is efficient and elegant. For the numerical optimization we employ additional measures to avoid the no-blur solution, namely multiscale optimization scheme and adjustment of the weight of the data-fidelity term during optimization.

The rest of this chapter is structured as follows. In the first two sections we formulate the image deblurring task as an optimization problem and present its solution in full detail. We use this chapter to establish some general concepts used in image deblurring, so we start with the very basics and provide a foundation for the rest of the thesis. In the following experimental section the performance of the presented method is evaluated both qualitatively and quantitatively on real blurred images with and without overexposure. We conclude with a discussion of the presented method. As an appendix to this chapter we provide a quick primer on the ADMM optimization method (alternating direction method of multipliers). ADMM is used extensively in this chapter as well as in Chapters 6 and 7, so a brief reminder of the most important principles and formulas is included to make the thesis more self-contained.

3.1 Problem formulation

Maximizing the posterior $p(u, h|g)$ is equivalent to minimizing its negative logarithm (loss). Assuming the convolution blur model (3.1) and i.i.d. Gaussian noise prior for $n = g - h * u$, we formulate blind image deblurring as the problem

$$\min_{u,h} L(u, h), \quad L(u, h) = \frac{\gamma}{2} \|h * u - g\|_2^2 + Q(u) + R(h), \quad (3.4)$$

where $\gamma = 1/\sigma^2$ is the (squared) noise precision and $Q(u) = -\log p(u)$ and $R(h) = -\log p(h)$ are regularizers corresponding to prior distributions of u and h , respectively.

Commonly used image priors are super-Gaussian distributions of the coefficients of some sparsifying linear image transform, most commonly image derivatives, for which the corresponding regularizer is the ℓ^p norm of said coefficients. For example, Laplace distribution of image derivatives results in the well-known total variation (TV)

$$Q_{\text{TV1}}(u) = \|\nabla u\|_1 = \sum_i |(\nabla_x u)_i| + |(\nabla_y u)_i|, \quad (3.5)$$

where ∇_x and ∇_y are partial derivative operators defined as forward differences and $\nabla = [\nabla_x^T, \nabla_y^T]^T$. The formulation in (3.5) is the so-called *anisotropic* TV, as opposed to *isotropic* TV defined as

$$Q_{\text{TV}_2}(u) = \|\nabla u\|_{2,1} = \sum_i \sqrt{(\nabla_x u)_i^2 + (\nabla_y u)_i^2}, \quad (3.6)$$

The less frequently used notation $\|\nabla u\|_{q,p}$ means ℓ^p norm of the ℓ^q norm of sub-groups (e.g. gradients). We will use it only with $q = 2$ for image derivatives and define isotropic generalized total variation as the sum of p -th powers of image gradient magnitudes

$$\|\nabla u\|_{2,p}^p = \sum_i ((\nabla_x u)_i^2 + (\nabla_y u)_i^2)^{\frac{p}{2}}. \quad (3.7)$$

In contrast, the anisotropic generalized total variation is defined simply as the sum of p -th powers of partial derivatives

$$\|\nabla u\|_p^p = \sum_i (|\nabla_x u|_i^p + |\nabla_y u|_i^p). \quad (3.8)$$

The isotropic formulation is generally preferable as an image prior, as it does not exhibit a grid bias. The term ‘‘generalized’’ refers to using general $p > 0$ rather than conventional $p = 1$ for which the term total variation is commonly used. Let us state at this point that when talking about image priors we use the term ‘‘norm’’ loosely without distinguishing from seminorms etc., as we are not interested in any metric-inducing properties.

It has been reported (e.g. Simoncelli [158]) that the distribution of gradients of natural images is more heavy-tailed than Laplace distribution, i.e. $p < 1$. We therefore use the isotropic generalized total variation as our image prior

$$Q(u) = \alpha_u \|\nabla u\|_{2,p}^p, \quad (3.9)$$

where α_u is the regularizer weighting factor, corresponding to precision of the distribution of gradient magnitudes.

For the blur kernel we use Laplace distribution on the positive kernel values to force sparsity and zero on the negative values, which results in

$$R(h) = \begin{cases} \alpha_h \|h\|_1 & \text{if } h_i \geq 0 \forall i, \\ +\infty & \text{otherwise.} \end{cases} \quad (3.10)$$

What remains to be addressed is to specify how to interpret the convolution in (3.4) and sacrifice neither numerical efficiency nor fidelity near image boundary and how to extend the method to deal with image overexposure, which is not taken care of by the assumed Gaussian noise prior.

Recall that convolution $h * u$ is linear (in both h and u separately) and therefore can be written as a matrix-vector multiplication with a suitable matrix H , $h * u \equiv Hu$. Suppose that the sought sharp image u has size $n_1 \times n_2$, then the blurred image g is of size $m_1 \times m_2$ where $m_i = n_i - s_i + 1$ and $s_1 \times s_2$ is size of the point spread function (PSF) of the blur h . This corresponds to the correct valid convolution definition

illustrated in Fig. 1.3c in which the sizes of the two images differ. Let us denote by \hat{H} the $m_1 m_2 \times n_1 n_2$ matrix which performs valid convolution with h . Equation (3.1) then reads $g = \hat{H}u + n$. This is significant for the practical numerical solution. Image recovery requires repeatedly solving a linear system with matrix containing $\hat{H}^T \hat{H}$, which can be very time consuming. If instead we consider circular convolution as in Fig. 1.3a represented by a square matrix H , inverting $H^T H$ is in comparison much faster because it can be done using 2D Fourier transform. Matrix H corresponding to 2D circular convolution is circulant (more precisely, block-circulant with circulant blocks) and if F denotes the linear operator of 2D Fourier transform, then $F H F^*$ is diagonal. Convolution $h * u$ is therefore equivalent to pixel-wise multiplication of the Fourier transforms of blur and image while matrix inversion is pixel-wise division. Circular convolution, however, treats u as though it were periodic and such blur model is thus typically violated near boundary of real blurred images, which leads to deblurring artifacts visible in Fig. 1.4.

Unless processing time is immaterial, it has been almost a standard practice to use the circular formulation with H and seek remedy e.g. by using the edgetaper function in MATLAB, which pre-blurs boundary regions of g and reduces the severity of ringing artifacts. A better approach has been proposed by Matakos *et al.* [109] which exploits the numerical properties of circular convolution and yet keeps the valid convolution in the model. The trick is in splitting the valid convolution into circular part and cropping part as $\hat{H} = M H$, where M is $m_1 m_2 \times n_1 n_2$ cropping matrix with at most one “1” in each row, all other elements zero. Inverting the Hessian $H^T M^T M H$ is not any easier than inverting $\hat{H}^T \hat{H}$ but by utilizing variable splitting in the minimization, the action of the two operators is separated and they can be inverted individually. We present the details in the next section.

The operator M also gives us the possibility to ignore regions of the input image where the convolution blur model does not hold by treating them as outliers and excluding them from the data term in (3.4). Here we focus explicitly on intensity clipping due to overexposure. To deal with this problem, we set to zero such entries in M that correspond to potentially clipped pixels in g , i.e. pixels with maximum intensity. These pixels are then effectively removed from the data term. We will show in the experimental section how this modification significantly helps the blur estimation in the presence of overexposure.

To summarize, we solve the following problem

$$\min_{u,h} L(u,h), \quad L(u,h) = \frac{\gamma}{2} \|M H u - g\|_2^2 + \alpha_u \|\nabla u\|_{2,p}^p + \alpha_h \|h\|_1 \quad \text{s.t. } h \geq 0. \quad (3.11)$$

3.2 Problem solution

We minimize the joint loss $L(u,h)$ in a coordinate-descent manner by minimizing alternately with respect to u and h , respectively, while keeping the other variable constant. To solve the particular subproblems $\min_u L(u,h)$ and $\min_h L(u,h)$ we use the ADMM optimization method, which is summarized at the end of this chapter in App. 3.A.

Minimization with respect to u

To problem to solve is

$$\min_u \left(\frac{\gamma}{2} \|MHu - g\|_2^2 + \alpha_u \|\nabla u\|_{2,p}^p \right), \quad (3.12)$$

where H denotes a (fixed) circular convolution operator corresponding to the h estimate from the previous u - h alternation, i.e. $h * u \equiv Hu$. Utilizing ADMM variable splitting we introduce substitutions $z_1 = Hu$ and $z_2 = \nabla u$. The first substitution z_1 allows us to split the awkward inversion of the valid convolution matrix $\hat{H}^T \hat{H} = H^T M^T M H$ into two much more convenient subproblems, each of which can be solved efficiently and non-iteratively. The second substitution z_2 decouples the data term $\|MHu - g\|_2^2$ and the non-smooth regularizer $\|\nabla u\|_{2,p}^p$ and allows us to minimize them separately since they now depend on different variables, which is preferable to simultaneous minimization. The complete Lagrangian of our problem is then

$$\begin{aligned} L(u, z_1, z_2, w_1, w_2) \\ = \frac{\gamma}{2} \|Mz_1 - g\|_2^2 + \alpha_u \|z_2\|_{2,p}^p + \frac{\rho_1}{2} \|Hu - z_1 + w_1\|_2^2 + \frac{\rho_2}{2} \|\nabla u - z_2 + w_2\|_2^2, \end{aligned} \quad (3.13)$$

where the new variables w_1 and w_2 correspond to the Lagrange multipliers of the constraints caused by the introduced substitutions and $\rho_{1,2}$ are user-defined parameters; see App. 3.A for details.

According to the ADMM procedure outlined in eqs. (3.34) in App. 3.A, minimizing (3.12) from the initial state $u^0, w_{1,2}^0$ consists of the following iteration steps for $k = 1, \dots, N$

$$z_1^k = \operatorname{argmin}_{z_1} \left(\frac{\gamma}{2} \|Mz_1 - g\|_2^2 + \frac{\rho_1}{2} \|Hu^{k-1} - z_1 + w_1^{k-1}\|_2^2 \right), \quad (3.14a)$$

$$z_2^k = \operatorname{argmin}_{z_2} \left(\alpha_u \|z_2\|_{2,p}^p + \frac{\rho_2}{2} \|\nabla u^{k-1} - z_2 + w_2^{k-1}\|_2^2 \right), \quad (3.14b)$$

$$u^k = \operatorname{argmin}_u \left(\frac{\rho_1}{2} \|Hu - z_1^k + w_1^{k-1}\|_2^2 + \frac{\rho_2}{2} \|\nabla u - z_2^k + w_2^{k-1}\|_2^2 \right), \quad (3.14c)$$

$$w_1^k = w_1^{k-1} + Hu^k - z_1^k, \quad (3.14d)$$

$$w_2^k = w_2^{k-1} + \nabla u^k - z_2^k. \quad (3.14e)$$

Note that we switched the order of u and z update equations, which allows us to initialize the optimization with u^0 rather than $z_{1,2}^0$. Update equations of the Lagrange multipliers $w_{1,2}$ are straightforward, but let us go over the preceding non-trivial updates of z_1, z_2 , and u individually and in more detail.

Problem (3.14a) amounts to solving the linear system which arises from differentiating the objective with respect to z_1 and setting the derivative to zero (iteration indices omitted for brevity)

$$(\gamma M^T M + \rho_1 I) z_1 = \gamma M^T g + \rho_1 (Hu + w_1). \quad (3.15)$$

The left-hand side matrix $M^T M$ is diagonal (containing 1s for valid pixels, 0s for pixels excluded due to model violation or boundary cropping), the inversion of the

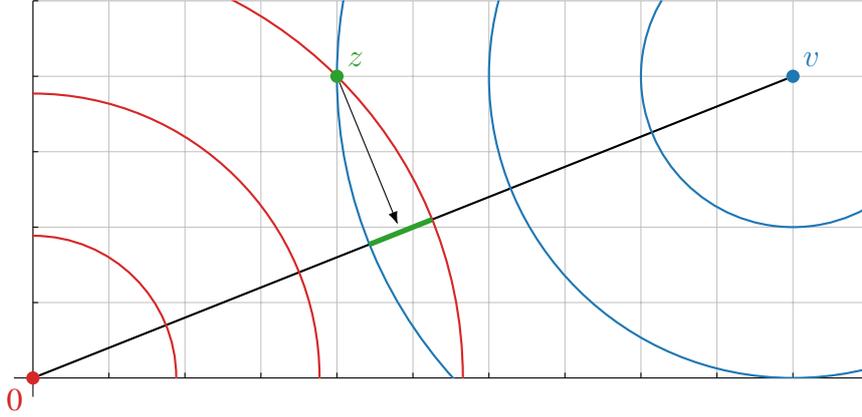


Figure 3.1: Reduction of the vector version of (3.16) for $v \in \mathbb{R}^2$ to the scalar version for $\|v\|_2$. Isolines of the two terms in the 2D version of (3.16) are depicted in red and blue, respectively. Loss of any potential minimizer z that does not lie on the segment connecting v and the origin can be improved, any point lying on the green part of the segment attains better loss value than z , as can be seen on the isolines. The minimizer must therefore be a scalar multiple of v .

left-hand side is therefore straightforward. The right-hand side term $M^T g$ is the input image zero-padded and with the clipped pixels set to zero. Notice that due to z_1 substitution the compound matrix MH is inverted in two separate steps, one containing only the circulant part while the other containing only the diagonal part, and both can be inverted non-iteratively.

ℓ^p proximal mapping

The update equation for z_2 (3.14b) warrants thorough discussion, as it will be beneficial for future reference. Both terms in the objective are summations of mutually independent terms over all image pixels, the minimization can therefore be carried out pixel by pixel independently and for each pixel it is a problem of the kind

$$\mathcal{P}_p(v) = \operatorname{argmin}_z \left(\frac{\alpha}{\rho} \|z\|_2^p + \frac{1}{2} \|z - v\|_2^2 \right). \quad (3.16)$$

The function \mathcal{P}_p which maps v to the minimizer z of (3.16) is called a *proximal mapping* of the function $\frac{\alpha}{\rho} \|\cdot\|_2^p$. For the anisotropic image prior $Q \sim \|\nabla u\|_p^p$ (3.8) the corresponding subproblem (3.16), which arises in solving (3.14b), is a scalar problem (the $\|\cdot\|_2$ norms are simple absolute values) at each pixel with $v = \nabla u + w_2$. Note that both z_2 and w_2 have twice as many pixels as the image, because ∇u outputs both x - and y - derivative for each pixel, so we have in fact two independent scalar problems at each pixel. For the isotropic image prior $Q \sim \|\nabla u\|_{2,p}^p$ (3.7), where the x - and y -derivatives are coupled, the corresponding subproblem (3.16) is a vector problem in \mathbb{R}^2 with $v = (v_x, v_y) = (\nabla_x u + w_{2,x}, \nabla_y u + w_{2,y})$ for each pixel.

The problem (3.16) is easy to interpret. We seek a vector $z \in \mathbb{R}^2$ (or number in \mathbb{R} in the scalar version) with as small p -power of its magnitude (absolute value) as possible while being as close to the given vector $v \in \mathbb{R}^2$ (number) as possible; the ratio of

these two objectives given by α/ρ . Intuitively, the vector version is more difficult, as it contains multiple unknowns tied in square roots. But by a simple geometric argument involving the isotropic levelsets of both terms in (3.16) (regardless of $p > 0$) it can be seen that the minimizer z must in fact be a scalar multiple of v , see illustration in Fig. 3.1. The optimized loss (3.16) consists of two terms whose isolines are depicted in red and blue, respectively. Loss value of any potential minimizer z that does not lie on the segment connecting v and the origin can be improved – any point lying on the green part of the segment attains better loss value than z , as can be seen on the isolines. The vector problem therefore effectively reduces to the scalar problem for $\|z\|_2$ as a function of $\|v\|_2$, from which the solution is reprojected back to \mathbb{R}^2 . Explicitly, if the solution $\mathcal{P}(\|v\|)$ of the scalar problem is known, then the solution $\mathcal{P}(v)$ of the vector problem is

$$\mathcal{P}(v) = \mathcal{P}(\|v\|) \frac{v}{\|v\|}. \quad (3.17)$$

Solving (3.16) for $z, v \in \mathbb{R}$ is therefore sufficient for solving the primary problem (3.14b). It is a simple exercise of differentiating (3.16) with respect to z and considering the non-smooth point $z = 0$ separately to see that for $p = 1$ the solution is the soft-thresholding (see Fig. 3.2-left)

$$\mathcal{P}_1(v) = \text{sign}(v) \max\left(|v| - \frac{\alpha}{\rho}, 0\right). \quad (3.18)$$

Consequently, for the isotropic TV image prior $Q(u) = \alpha_u \|\nabla u\|_{2,1}$ we get that for each pixel the iteration step (3.14b) is

$$z_{2,x} = \frac{\sigma (\nabla_x u + w_{2,x})}{\sqrt{(\nabla_x u + w_{2,x})^2 + (\nabla_y u + w_{2,y})^2}}, \quad (3.19a)$$

$$z_{2,y} = \frac{\sigma (\nabla_y u + w_{2,y})}{\sqrt{(\nabla_x u + w_{2,x})^2 + (\nabla_y u + w_{2,y})^2}}, \quad (3.19b)$$

where the shrink factor σ is

$$\sigma = \max\left(\sqrt{(\nabla_x u + w_{2,x})^2 + (\nabla_y u + w_{2,y})^2} - \frac{\alpha_u}{\rho_2}, 0\right). \quad (3.19c)$$

For the sake of completeness, the equivalent equations for the anisotropic TV prior (3.5) are directly from (3.18)

$$z_{2,x} = \text{sign}(\nabla_x u + w_{2,x}) \max\left(|\nabla_x u + w_{2,x}| - \frac{\alpha}{\rho}, 0\right), \quad (3.20a)$$

$$z_{2,y} = \text{sign}(\nabla_y u + w_{2,y}) \max\left(|\nabla_y u + w_{2,y}| - \frac{\alpha}{\rho}, 0\right). \quad (3.20b)$$

We can see the grid bias exhibited by the anisotropic formulation. It is apparent in (3.19), or perhaps better yet in (3.17), that the gradient shrinkage induced by the isotropic TV never changes the orientation of the gradient and in particular that partial

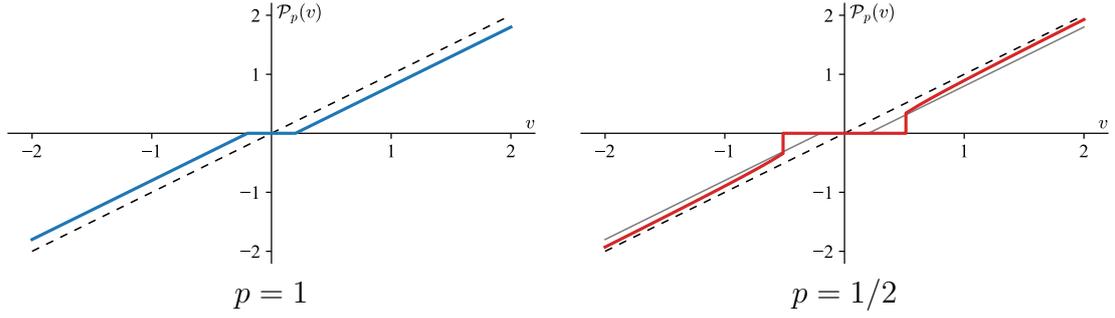


Figure 3.2: Solution of the proximal mapping of ℓ^p norm (3.16) for $p = 1$ (left) and $p = 1/2$ (right) with $\alpha = 1$ and $\rho = 5$.

derivatives vanish only if the whole gradient vanishes. With the anisotropic formulation, on the other hand, either of the partial derivatives can vanish and as a result, oblique gradient can be transformed to perfectly horizontal or vertical gradient.

Returning to the more general case, for $p < 1$ the problem (3.16) has a closed form solution only for a few selected values of p , which lead to sufficiently low-order polynomial equation (namely $p = 0, 1/2, 2/3$), and in general case it must be solved numerically; see e.g. Chen *et al.* [26] for thorough analysis. Since the proximal mapping $\mathcal{P}(v)$ depends only on the parameters p, α_u , and ρ_2 , which are typically kept constant during iterations, the mapping can be precalculated beforehand using e.g. Newton's method initialized by the $p = 1$ solution and the result stored in the form of a lookup table. In our experiments we used the value $p = 1/2$, which is one of the lucky cases for which a closed form solution to (3.16) exists [26, eq. (13)]

$$\mathcal{P}_{1/2}(v; \alpha, \rho) = \begin{cases} 0 & \text{if } |v| \leq \frac{3}{2} \left(\frac{\alpha}{\rho} \right)^{2/3}, \\ \frac{2}{3} v \left(1 + \cos \left(\frac{2}{3} \cos^{-1} \left(-\frac{3^{3/2} \alpha}{4 \rho} |v|^{-3/2} \right) \right) \right) & \text{if } |v| \geq \frac{3}{2} \left(\frac{\alpha}{\rho} \right)^{2/3}. \end{cases} \quad (3.21)$$

Notice that due to non-convexity of the ℓ^p norm, the solution of (3.16) is not continuous and ambiguous (multi-valued) at the decision boundary, as can be seen in Fig. 3.2-right. For the generalized isotropic TV image prior $Q(u) = \alpha_u \|\nabla u\|_{2,p}^p$ we get that for each pixel the iteration step (3.14b) consists of the same update equations (3.19ab) as in the $p = 1$ case but with a different shrink factor σ given by

$$\sigma = \mathcal{P}_{1/2} \left(\sqrt{(\nabla_x u + w_{2,x})^2 + (\nabla_y u + w_{2,y})^2}; \alpha_u, \rho_2 \right). \quad (3.22)$$

Finally, the update equation for u (3.14c), after differentiating with respect to u , leads to the linear system for u

$$\left(\rho_1 H^T H + \rho_2 \nabla^T \nabla \right) u = \rho_1 H^T (z_1 - w_1) + \rho_2 \nabla^T (z_2 - w_2). \quad (3.23)$$

The operator H represents circular convolution and the corresponding matrix is block-circulant, the derivative operators are defined as circular forward differences and therefore ∇ is also block-circulant. Consequently, the whole matrix of the left-hand side

is block-circulant, which means that it can be diagonalized by 2D Fourier transform. The solution u can thus be computed in a single step only at the cost of several Fourier transforms.

This concludes our detailed description of minimization of the primary loss $L(u, h)$ (3.11) with respect to u . In the experiments we use parameter values $\alpha_u = 3$, $\rho_1 = 2^6$, $\rho_2 = 2^5$, and $p = 1/2$.

Minimization with respect to h

Minimization of $L(u, h)$ (3.11) with respect to h is carried out in a similar manner. We solve the problem

$$\min_h \left(\frac{\gamma}{2} \|MUh - g\|_2^2 + \alpha_h \|h\|_1 \right) \quad \text{s.t. } h \geq 0, \quad (3.24)$$

where U is the operator performing circular convolution with u from the previous u - h alternation, i.e. $h * u \equiv Uh$. In order to split the compound matrix MU in the data term to facilitate its inversion and to isolate the non-smooth regularizer/constraint from the data term we again utilize ADMM variable splitting and introduce substitutions $z_1 = Uh$ and $z_2 = h$. ADMM procedure (3.34) then dictates the following iteration steps for $k = 1, \dots, N$

$$z_1^k = \underset{z_1}{\operatorname{argmin}} \left(\frac{\gamma}{2} \|Mz_1 - g\|_2^2 + \frac{\rho_1}{2} \|Uh^{k-1} - z_1 + w_1^{k-1}\|_2^2 \right), \quad (3.25a)$$

$$z_2^k = \underset{z_2}{\operatorname{argmin}} \left(\alpha_h \|z_2\|_1 + \frac{\rho_2}{2} \|h^{k-1} - z_2 + w_2^{k-1}\|_2^2 \right) \quad \text{s.t. } z_2 \geq 0, \quad (3.25b)$$

$$h^k = \underset{h}{\operatorname{argmin}} \left(\frac{\rho_1}{2} \|Uh - z_1^k + w_1^{k-1}\|_2^2 + \frac{\rho_2}{2} \|h - z_2^k + w_2^{k-1}\|_2^2 \right), \quad (3.25c)$$

$$w_1^k = w_1^{k-1} + Uh^k - z_1^k, \quad (3.25d)$$

$$w_2^k = w_2^{k-1} + h^k - z_2^k, \quad (3.25e)$$

assuming the initial state $h^0, w_{1,2}^0$ is given. We again reversed the order of h and z updates to be able to initialize the procedure with h^0 .

The problem (3.25a), upon differentiating with respect to z_1 , leads to the linear system for z_1 (iteration indices omitted)

$$\left(\gamma M^T M + \rho_1 I \right) z_1 = \gamma M^T g + \rho_1 (Uh + w_1). \quad (3.26)$$

Notice, though, that the substitutions $z_1 = Hu$ in the min- u step and $z_1 = Uh$ in the min- h step are identical, as are their update equations (3.14a)+(3.14d) and (3.25a)+(3.25d), respectively. Since we do not perform either of the two coordinate descents $\min_u L$ and $\min_h L$ to full convergence but instead perform one pass of the corresponding ADMM procedure and then switch to the other coordinate (u or h), we share the variable z_1 as well as the dual w_1 and perform only one update per full u - h alternation.

The update equation for z_2 (3.25b) is exactly the ℓ^1 proximal mapping studied above on page 34, in particular it is a scalar problem of the kind (3.16) for each pixel with $v = h + w_2$. We know that for $p = 1$ the solution is the soft-thresholding (3.18).

The constraint $h \geq 0$ in (3.24) is a particular instance of the convex constraint problem (3.35) discussed on page 45 in App. 3.A with $\mathcal{C} = [0, \infty]$. We see that it can be solved using the substitution z_2 by projecting $h + w_2$ to the corresponding feasible set, i.e. set negative values to zero. Putting the two results together (in fact, what we do is a proximal mapping of the function $x \rightarrow \{\|x\|_1$ if $x \geq 0$, $+\infty$ otherwise}), the update step (3.25b) for each element of z_2 is simply

$$z_2 = \max\left(h + w_2 - \frac{\alpha_h}{\rho_2}, 0\right). \quad (3.27)$$

Lastly, the update of h (3.25c), upon differentiating with respect to h , leads to the linear system for h

$$\left(\rho_1 U^T U + \rho_2 I\right) h = \rho_1 U^T (z_1 - w_1) + \rho_2 (z_2 - w_2). \quad (3.28)$$

Thanks to the variable splitting, the matrix on the left-hand side is block-circulant and can be inverted directly using Fourier transform. In the experiments we use parameter values $\alpha_h = 1$ and $\rho_2 = 2^{13}$.

Algorithm summary

As input, the algorithm gets the blurred image g and (upper bound of) the expected size of the blur PSF h . First, in the *blind stage* the blur h is estimated by the just described alternation minimization of $L(u, h)$ (3.11). After that, in the *non-blind stage* the final image u is calculated as $\min_u L(u, h)$ with h fixed and slightly adjusted parameters – namely the image prior exponent p .

Multiscale optimization

To avoid getting trapped in a local minimum and speed up the processing, we perform the blind stage in multiscale pyramidal fashion. The input image g is downsampled such that the corresponding blur PSF at this scale is small (e.g. 3×3 pixels), we estimate the blur at this scale and then upsample it (as well as the input g and current estimate u) and use as the initialization of the estimation at the next scale. This procedure is repeated until the target blur size is reached. At each scale the cropping and masking matrix M is adjusted to match the correct image sizes and to exclude from the data term pixels where g is potentially saturated and clipped. This is done by simply setting the whole row of M which corresponds to pixel where g has maximum intensity to zero. It may happen that such pixel is in fact perfectly valid, but the probability of that is small and even if we remove a small number of pixels by mistake, the estimation result is not harmed.

Model precision adjustment

Another measure we take to avoid the favored no-blur solution $h = \delta$ is adjusting the model precision γ during iterations. It has been suggested at this point to artificially sharpen the current estimate u so that $h = \delta$ no longer fits the data term and the optimization is forced to leave this point (see “Intermediate image” on page 19 in

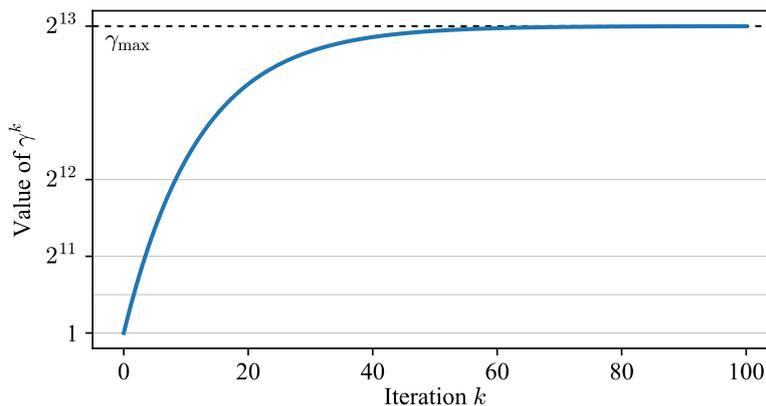


Figure 3.3: Adjustment of the data-term weight γ^k during iterations, see eq. (3.29). Displayed evolution corresponds to parameters $\gamma_{\min} = 1$, $\gamma_{\max} = 2^{13}$, $T = 12$.

Sec. 2.1). We achieve the same result by underestimating the model precision γ , which gives strong weight to the regularizers and the sparsity-promoting image prior causes emergence of strong edges in the current image estimate u . By gradually increasing γ to its expected true value the pressure on regularization is decreased and both blur and the image estimates become progressively more accurate. This technique serves the same purpose as estimating h from an artificially enhanced image and fits the Bayesian paradigm better, because it does not deviate from the MAP estimation procedure. We update the model precision γ according to the formula

$$\gamma^k = (\gamma_{\min} - \gamma_{\max}) e^{-k/T} + \gamma_{\max}, \quad (3.29)$$

where γ_{\min} is the initial (underestimated) value and γ_{\max} is the asymptotical (true) final value, T controls the rate of growth; see Fig. 3.3 for illustration. In the experiments we used parameter values $\gamma_{\min} = 1$, $\gamma_{\max} = 2^{13}$, $T = 12$, and $p = 1/2$ in the blind stage. For the final non-blind stage we fix $\gamma = 2^{13}$ and use $p = 1$ in the image prior. The assumed intensity range of g is $[0, 1]$. The whole image deblurring process is summarized in Alg. 3.1, where we omit iteration indices for brevity and it is understood that all variables are updated based on their values from the previous iteration, as is explicitly specified by the referenced update equations.

3.3 Experimental evaluation

We tested our algorithm on the dataset provided by Levin *et al.* [96] consisting of four grayscale images and eight PSFs of true motion blur, resulting in 32 test images in total; see Fig. 3.4. The original images were printed on a paper and then re-photographed with a hand-held camera to get authentic motion blur. The blur PSFs were estimated as the blur of dots placed in the corners of the image region; for more details on the acquisition process refer to [96].

In this experiment we compare our method to the method of Xu and Jia [191], which is still one of the best performing single-channel blind deblurring methods and is based on identification and restoration of salient edges in u as a means to avoid the no-blur solution, and the method of Fergus *et al.* [47], based on the marginalized h

Algorithm 3.1 Summary of the deblurring algorithm presented in Chapter 3

input: blurred image g , expected blur PSF size s_M

set $g_0 := g$

define scale levels as PSF sizes $s_1 = 3, s_2 = 5, s_3 = 9, \dots, s_M$

set $u := g, h := \delta, w_1 := 0, w_{2,u} := 0$, and $w_{2,h} := 0$, all in scale corresponding to the initial PSF size s_1

// blind stage

for all $s = s_1, \dots, s_M$ **do**

 construct g as g_0 downsampled to correspond to PSF size s

 construct M as cropping matrix from $\text{size}(u)$ to $\text{size}(g)$ and with zeros at rows corresponding to pixels where $g == 1$

for $k = 1, \dots, 100$ or until convergence **do**

 update $z_1, z_{2,u}, u, w_1$, and $w_{2,u}$ by eqs. (3.14)

 update $z_{2,h}, h$, and $w_{2,h}$ by eqs. (3.25)

 update γ according to (3.29)

end for

 unless $s == s_M$, upsample $u, h, w_1, w_{2,u}$, and $w_{2,h}$ to the next scale

end for

// non-blind stage

fix $\gamma := \gamma_{\max}$, and the load non-blind parameter set

for $k = 1, \dots, 100$ or until convergence **do**

 update $z_1, z_{2,u}, u, w_1$, and $w_{2,u}$ by eqs. (3.14)

end for

output: u and h as the final image and blur estimates

estimation using variational Bayesian inference. Please note that we use the results of [47] as presented in [96]. In our comparison we focus on the accuracy of blur estimation, which we measure by the mean squared error (MSE) of the estimated blur PSF to the ground truth (evaluated invariant to translation); see Sec. 5.1 for review of standard image and blur similarity measures.

Results of blur estimation for all images in the test dataset are in Fig. 3.5, which shows errors of the estimated blur PSFs (lower is better). For easier assessment the results are visually grouped by PSFs (8 groups) and then further by the sharp image (4 subgroups). We see that in majority of cases (24 of 32) our method is superior, though in some cases it apparently failed. Examples of blur PSFs estimated by all of the evaluated methods can be seen in Fig. 3.6, our method is arguably the most accurate.

In our next experiment we focused on how intensity clipping in images with saturated pixels affects blur estimation. In this case we worked with synthetically blurred images to be able to quantitatively measure the relation between the amount of non-Gaussian error and accuracy of blur estimation. We randomly selected some pixels in the sharp image and artificially increased their intensities to simulate overexposure, then we blurred the image and clipped all intensities above maximum allowed value (i.e. the same way it happens in normal photography). The final image served is the input to our method for blind blur estimation. To decrease fluctuations in the PSF estimation accuracy (depending on image used and the particular choice of saturated



Figure 3.4: The blind deblurring benchmark dataset by Levin *et al.* [96], containing 4 sharp images blurred by 8 different motion blurs. **Top:** All sharp images in the dataset. **Middle:** PSFs of all motion blurs in the dataset. **Bottom:** Example of blurred input images in the dataset, 4 of 32 images in total.

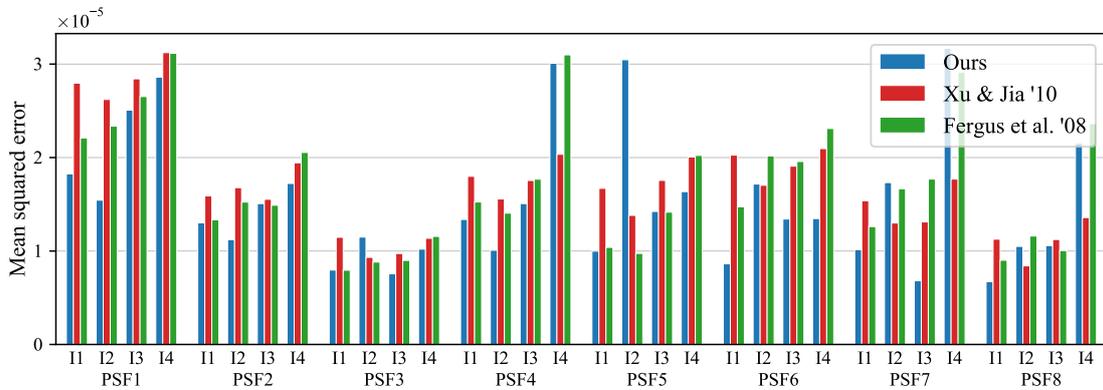


Figure 3.5: Mean squared error (lower is better) of estimated blur PSFs for all test images in the dataset by [96]. The results are visually grouped by PSFs (8 groups) and then further by the corresponding sharp image (4 subgroups).

pixels), we used 4 images and 10 different realizations of the random distribution of saturated pixels for each image, resulting in 40 test cases for each degree of image degradation for which we averaged the results of blur estimation. On top of that, we noticed that success of blur estimation in this kind of experiment highly depends on the geometry of the spatial distribution of the saturated pixels. We investigated two extremal cases – in the first case the overexposed pixels were small dots spread more or less evenly in the entire image, in the second case the overexposed pixels formed one large connected region; see examples of corresponding input images in Fig. 3.7.

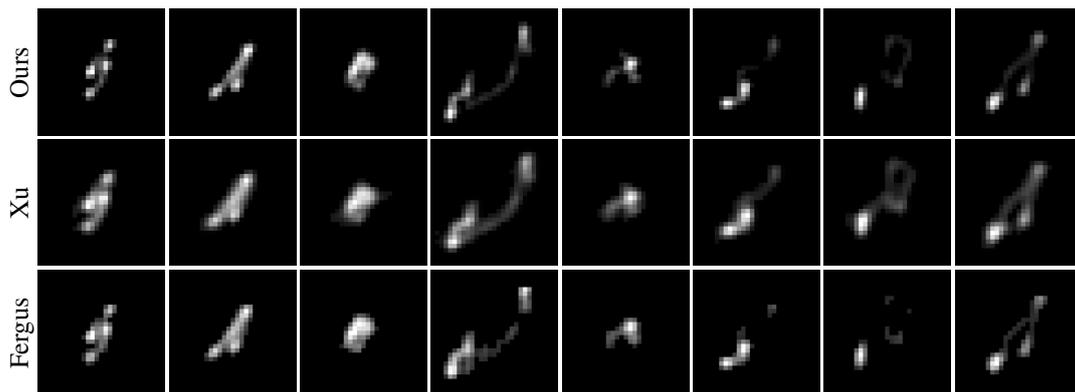


Figure 3.6: Results of blur estimation for the image 1 from the test dataset, compare with ground-truth PSFs in Fig. 3.4. Top: Our presented method. Middle: Method of Xu and Jia [191]. Bottom: Method of Fergus *et al.* [47].

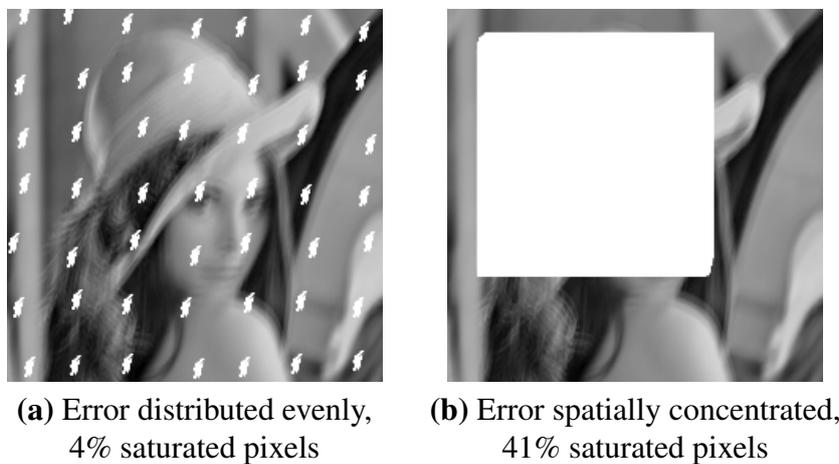


Figure 3.7: Example input to our experiment with simulated overexposure illustrating the two different geometric configurations of the distribution of saturated pixels. This percentage of affected pixels is approximately the maximum for which the blur estimation by our method was still successful.

The results of this experiment are in Fig. 3.8 which shows mean squared error of the estimated blur PSFs as a function of percentage of image pixels for which the convolution blur model is violated (blue curve). As a reference (red curve), we used our method to do the exact same blur estimation from the same inputs but with the handling of saturated pixels disabled (matrix M was used only to deal with boundary artifacts). We see that blur estimation without saturation handling fails virtually as soon as the non-Gaussian model violation appears (note that in this experiment values of MSE above $2 \cdot 10^{-3}$ indicate complete fail), while our extended method is much more robust. It is interesting, yet quite understandable, that small corruptions spread in the entire image have far more damaging effect on blur estimation than single large corrupted region (see the respective x -axis limits in Fig. 3.8a and 3.8b).

Figure 3.7 shows input images with roughly the maximum percentage of saturated pixels, for which our blur estimation method was still successful. The images are damaged to the similar degree in terms of achievable blur estimation accuracy, yet quanti-

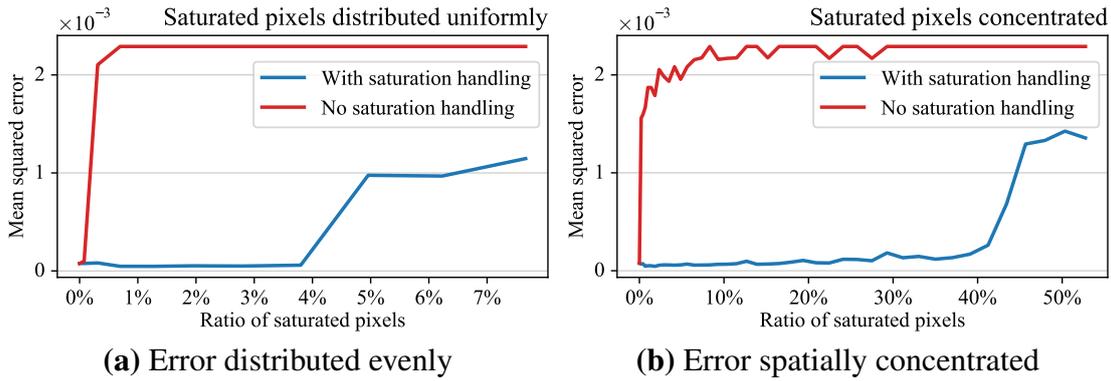


Figure 3.8: Mean squared error (lower is better) of estimated blur PSFs in presence of image saturation – evaluation of efficacy of the presented method for image saturation handling. Value of MSE above approx. $2 \cdot 10^{-3}$ indicates complete fail. Two geometric configurations of the distribution of saturated pixels were tested. **(a)** The saturated pixels were spread throughout the image, as in Fig. 3.7a. **(b)** The saturated pixels were concentrated in one large region, as in Fig. 3.7b.

tatively the right image contains an order of magnitude more pixels which violate the blur model than the left image. This is understandable, because the input in Fig. 3.7b, in which all corrupted pixels are concentrated in one place, is basically equivalent to a smaller sub-image of Fig. 3.7b without any model violation. By contrast, if the percentage of saturated pixels in the left image Fig. 3.7a increases even more, the largest available sub-image without any saturated pixels becomes too small for the particular blur size. This explains why the robustness of the presented method to the percentage of clipped pixels depends on the geometry of the spatial distribution of these pixels and why in Fig. 3.8 we see such dramatic difference in the points of failure in the two extremal geometric configurations investigated in the experiment (approx. 4% and 40%, respectively).

Our last demonstration of the efficacy of the presented method is in Fig. 3.9, which depicts deblurring result of a genuinely motion-blurred photo captured by hand-held camera, the true motion blur is unknown. Visual comparison of the blurred input and our result shows significant improvement in quality.

3.4 Conclusion

We presented a blind image deblurring method based on pure Bayesian MAP estimation of image and blur simultaneously without any “bells and whistles” that would fall outside the MAP framework. To avoid getting trapped in the no-blur solution our method uses heavy-tailed distribution of image gradient magnitudes as image prior, we gradually adjust the data term weight (model precision) during the optimization to reach the correct value asymptotically from below, and we estimate the image and blur in multiscale fashion. These three ingredients are sufficient for the MAP blind deblurring to work, as we demonstrated experimentally. The non-convexity of the problem is still present, however, and we see in the experiments that in some test cases our method failed.



Figure 3.9: Blind deblurring of image containing true motion blur with unknown ground truth. Estimated blur PSF superimposed in bottom-right (out of scale).

To increase the blur model fidelity, our method correctly and efficiently handles the convolution boundary problem using masking in the data-fidelity term. The same technique is used to deal with non-Gaussian intensity clipping caused by overexposure and saturation. We demonstrated experimentally that this addition makes blur estimation significantly more robust to overexposure. Thanks to suitable mathematical formulation, this extension of the method was not at the expense of computational efficiency. One drawback of the presented approach is that it is designed specifically to deal with nonlinear intensity clipping. The method must know a priori which pixels violate the blur model in order to remove them from the blur estimation. Boundary effects or intensity clipping are easy to anticipate or infer but it is not obvious how the method should be generalized to automatically detect and handle other types of non-Gaussian observation error. More general approach will be presented in the next chapter.

The mathematical solution and all the algorithmic steps were presented thoroughly and in full detail, which allowed us to establish a solid foundation for the rest of the thesis. For the numerical optimization we use the ADMM method, reviewed in the following appendix for reader’s convenience.

3.A Alternating direction method of multipliers

In this appendix we will briefly review the ADMM optimization method as it is used in this work. For more general and detailed information on ADMM we encourage the reader to see e.g. the introductory text by Boyd *et al.* [13] and references therein.

Consider the optimization problem

$$\min_x f(x) + g(Ax), \quad (3.30)$$

where $x \in \mathbb{R}^n$ is the primal independent variable, both $f: \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ and $g: \mathbb{R}^m \rightarrow \mathbb{R} \cup \{+\infty\}$ are closed, proper, convex functions, and $A \in \mathbb{R}^{m \times n}$ is a linear

operator with full column rank. ADMM is convenient when minimizing of f and g separately is preferable to simultaneous minimization or when minimizing $g(\cdot)$ is preferable to minimizing $g(A \cdot)$. To this end we introduce substitution $z = Ax$ and solve the constrained problem

$$\min_{x,z} f(x) + g(z), \quad \text{s.t. } Ax - z = 0, \quad (3.31)$$

that is clearly equivalent to (3.30). This technique is sometimes called *variable splitting*. We form the *augmented Lagrangian* of the problem (3.31), which is a regular Lagrangian plus a quadratic penalty term corresponding to the prescribed constraint,

$$L(x, z, y) = f(x) + g(z) + y^T(Ax - z) + \frac{\rho}{2} \|Ax - z\|_2^2, \quad (3.32)$$

where $y \in \mathbb{R}^m$ is the problem dual variable (Lagrange multiplier) and $\rho > 0$ is a user-defined parameter. By “completing the square”, L can be rewritten into the so-called *scaled* form, which is shorter and sometimes more convenient

$$L(x, z, w) = f(x) + g(z) + \frac{\rho}{2} \|Ax - z + w\|_2^2 + \text{const}(w). \quad (3.33)$$

The new scaled dual variable is simply $w = y/\rho$.

From the initial state (z^0, w^0) , ADMM consists of the following iterations for $k = 1, \dots, N$ (see [13, eqs. (3.5-7)])

$$x^k = \underset{x}{\operatorname{argmin}} L(x, z^{k-1}, w^{k-1}) = \underset{x}{\operatorname{argmin}} \left(f(x) + \frac{\rho}{2} \|Ax - z^{k-1} + w^{k-1}\|_2^2 \right), \quad (3.34a)$$

$$z^k = \underset{z}{\operatorname{argmin}} L(x^k, z, w^{k-1}) = \underset{z}{\operatorname{argmin}} \left(g(z) + \frac{\rho}{2} \|Ax^k - z + w^{k-1}\|_2^2 \right), \quad (3.34b)$$

$$w^k = w^{k-1} + Ax^k - z^k. \quad (3.34c)$$

The algorithm therefore consists of f minimization step (3.34a), g minimization step (3.34b), and the dual variable update (3.34c). The variable splitting achieved two purposes, the function f and g are minimized separately and g is minimized directly rather than composed with A , as can be seen in (3.34a) and (3.34b). Both of these effects are exploited extensively throughout the thesis. Though there exist more general formulations of the method, we will not need them, except for the trivial extension of applying variable splitting multiple times.

Convex constraint

One particular scenario in which utilization of the above procedure is convenient is worth mentioning. Note that neither f nor g are not required to be finite-valued. This can be utilized when solving a problem with the independent variable constrained to a convex set,

$$\min_x f(x) \quad \text{s.t. } x \in \mathcal{C}, \mathcal{C} \subset \mathbb{R}^n \text{ convex.} \quad (3.35)$$

This problem is equivalent to

$$\min_x f(x) + \mathcal{I}_{\mathcal{C}}(x), \quad (3.36)$$

where $\mathcal{I}_{\mathcal{C}}$ is the indicator function of the feasible set \mathcal{C} defined as

$$\mathcal{I}_{\mathcal{C}}(x) = \begin{cases} 0 & \text{if } x \in \mathcal{C}, \\ +\infty & \text{otherwise.} \end{cases} \quad (3.37)$$

Now we have unconstrained problem of the kind (3.30) where we can substitute $z = x$ to separate f from \mathcal{I} . In the corresponding iteration procedure, (3.34a) amounts to minimizing f unconstrained while (3.34b) is simply a projection of $x^k + w^{k-1}$ onto \mathcal{C} . Both of these steps are often easier than the original constrained problem.

Convergence

There are many theoretical results regarding convergence of the algorithm and its variants as well as its equivalence to special cases of other known optimization algorithms. Though very interesting, these results are not directly necessary for understanding of the material covered in this thesis. We will therefore only restate (without proof) the relevant convergence result by Eckstein and Bertsekas [45] to conclude this brief review of ADMM.

The algorithm converges to the optimal solution of (3.31), provided the solution exists and the iterations are carried out within acceptable error. These criteria are formalized by the following two assumptions.

1. Let there exist $(x^*, z^* = Ax^*, y^*)$ which is a saddle point of the unaugmented Lagrangian $L_0(x, z, y) = f(x) + g(z) + y^T(Ax - z)$.

Explicitly, let (x^*, z^*, y^*) satisfy that for all $x \in \mathbb{R}^n$ and $z \in \mathbb{R}^m$

$$L_0(x^*, z, y^*) \leq L_0(x, z, y^*) \quad \text{and} \quad L_0(x, z^*, y^*) \leq L_0(x, z, y^*). \quad (3.38)$$

It then follows that x^* is a solution to (3.30) and y^* is a solution to the dual of (3.31).

2. The problems (3.34a) and (3.34b) in the algorithm iterative procedure are solved within summable error.

Explicitly, let μ_k and ν_k denote the error bounds

$$\left\| x^k - \underset{x}{\operatorname{argmin}} \left(f(x) + \frac{\rho}{2} \|Ax - z^{k-1} + w^{k-1}\|_2^2 \right) \right\| \leq \mu_k, \quad (3.39)$$

$$\left\| z^k - \underset{z}{\operatorname{argmin}} \left(g(z) + \frac{\rho}{2} \|Ax^k - z + w^{k-1}\|_2^2 \right) \right\| \leq \nu_k. \quad (3.40)$$

Then let these bounds satisfy

$$\sum_k \mu_k < \infty \quad \text{and} \quad \sum_k \nu_k < \infty. \quad (3.41)$$

Under the assumptions (3.38), (3.41), and the assumptions on f , g , and A stated at the beginning of the section, the iterates of eqs. (3.34) satisfy

$$\{x^k\} \rightarrow x^*, \quad \{z^k\} \rightarrow z^*, \quad \text{and} \quad \{w^k\} \rightarrow w^*, \quad (3.42)$$

where $w^* = y^*/\rho$ and (x^*, z^*, y^*) satisfies assumption 1. If, on the other hand, either $\{z^k\}$ or $\{w^k\}$ are unbounded, then the problem (3.31) does not have a solution. The proof can be found in [45, Theorem 8].

Chapter 4

Variational Bayesian deblurring of images with arbitrary model violations

In the preceding chapter we covered maximum a posteriori estimation approach to blind image deblurring in the presence sensor saturation as a particular example of a non-Gaussian observation error. In this chapter we substantially generalize the target scenario as well as our approach. We will present a blind image deblurring method based on variational Bayesian inference with a unified formulation which elegantly uses the same class of prior for each of the primary unknown variables – image, blur, and observation error (noise). Using sufficiently flexible noise prior allows us to automatically deal with general violations of the image acquisition model without having to tailor the method for each particular type of model error, as in the previously presented approach to image overexposure.

We proceed from the same convolution blur model as in the previous chapter,

$$g = h * u + n, \quad (3.1)$$

and adopt the standard probabilistic formulation in which the unknown variables u , h , and n are considered as discrete random fields with corresponding distributions $p(u)$, $p(h)$, and $p(n)$. As we already discussed in the introduction to the previous Chapter 3, the advisable approach to blur estimation is maximizing $p(h|g)$, which is the joint posterior marginalized over all latent images (see eq. (3.2)), but this is typically intractable. In the variational Bayesian inference, we approximate the posterior $p(u, h|g)$ by a factorized distribution $p(u, h|g) \approx q(u)q(h)$ by minimizing its Kullback-Leibler divergence to the original distribution. Due to the factorization assumption, the marginalization effectively appears cost-free, because the distribution $q(h)$ provides an approximation of the sought marginal distribution $p(h|g)$. Estimating the sharp image is then the standard non-blind problem of maximizing $p(u|g, h)$.

As the prior distributions $p(u)$, $p(h)$, and $p(n)$ we use non-identical Gaussian distribution with automatically estimated precision, which is also called the Automatic Relevance Determination model (ARD), introduced by MacKay [106] and Neal [116]. Such prior is sufficiently rich but remains computationally tractable. Using priors from the ARD family for image deblurring is not novel per se and was suggested previously e.g. by Chantas *et al.* [24] as the image prior in the non-blind setting, or by Zhang *et al.* [203] for multichannel deconvolution, by Dong *et al.* [44] as both image and blur prior

for blind deblurring, and even as the noise prior by Christmas and Everson [32] for signal recovery, Köhler *et al.* [80] for multichannel super-resolution, and Tzikas *et al.* [173] for blind image deblurring. The approach presented in this chapter contains several novelties. We use a particular approximation of the blur covariance matrix which facilitates the optimization, we again use the noise model to deal with the boundary problem of the convolution formulation, and lastly we investigate a slightly modified formulation with explicit modeling of the global Gaussian and local non-Gaussian component of the noise and automatic estimation of the degree of non-Gaussianness of the noise distribution, which would otherwise have to be specified by user as a parameter. Our presentation of the method is accompanied by a thorough discussion and analysis of the individual prior distributions and their role in blind image deblurring as well as detailed derivation of the variational approximation and description of all algorithmic steps of the method.

We will first define and discuss our model of the prior distributions of each of the primary unknowns and validate our choice by several targeted experiments. General introduction to automatic relevance determination priors is included as a prelude for reader's convenience. Then in a little detour from image deblurring we recall the general principles of variational Bayes method (VB), which will provide foundations for our blind image deblurring problem formulated as inference on the VB approximation of the posterior. Corresponding solution is derived and presented in maximum detail in the next section. Lastly, we will evaluate the presented method and demonstrate its performance on both synthetic images with simulated arbitrary acquisition model violation combined with Gaussian noise and real images with unknown motion and out-of-focus blur. We conclude the chapter with a discussion of the presented method.

4.1 Automatic relevance determination priors

The ARD model was originally proposed for neural networks by MacKay [106] and Neal [116] to automatically determine weights of neuron connections based on the training data. Quoting from [116], “... *each input variable has associated with it a hyperparameter that controls the magnitudes of the weights on connections out of that input unit. These hyperparameters are given some prior distribution, and conditional on the values of these hyperparameters, the weights out of each input have independent Gaussian prior distributions with standard deviation given by the corresponding hyperparameter. If the hyperparameter associated with an input specifies a small standard deviation for weights out of that input, these weights will likely all be small, and the input will have little effect on the output; if the hyperparameter specifies a large standard deviation, the effect of the input will likely be significant. The posterior distributions of these hyperparameters will reflect which of these situations is more probable, in light of the training data.*” The following general introduction explains how this concept can be used in image deblurring.

We say that a discrete random variable x has ARD prior if its prior distribution is an independent non-identical zero-mean Gaussian with pixel-wise precisions (inverse

of variances) λ_i governed by independent and identical hyperprior $p(\lambda)$,

$$p(x|\lambda) = \prod_i \mathcal{N}(x_i|0, \lambda_i^{-1}) = \prod_i \left(\frac{\lambda_i}{2\pi}\right)^{\frac{1}{2}} \exp\left(\frac{\lambda_i}{2} x_i^2\right). \quad (4.1)$$

We will assume that the hyperprior $p(\lambda)$ is the so-called *conjugate prior*, which means such prior that the posterior distribution $p(\lambda|x)$ will have the same functional form as the prior $p(\lambda)$. This choice greatly facilitates subsequent calculations. In the case of precision of a Gaussian distribution, the likelihood in (4.1) has the form $\lambda^a \exp(b\lambda)$. This form is preserved when multiplied by itself, therefore the corresponding conjugate prior has the form of a Gamma distribution

$$p(\lambda) = \prod_i \mathcal{G}(\lambda_i|a, b), \quad (4.2)$$

where \mathcal{G} denotes Gamma distribution whose density, mean, and variance are respectively given by

$$\mathcal{G}(t|a, b) = \frac{1}{\Gamma(a)} b^a t^{a-1} \exp(-bt), \quad \mathbb{E}[t] = \frac{a}{b}, \quad \text{and} \quad \text{var}[t] = \frac{a}{b^2}, \quad (4.3)$$

where Γ is the gamma function. Let us note that the term ‘‘hyperprior’’, as used in this work, is not in any way different from the term ‘‘prior’’, we use to merely to follow convention and vaguely distinguish the primary unknowns (image, blur, noise) from the model parameters and latent variables.

Student’s t -distribution

ARD priors sometimes appear in literature under the name Student’s t -priors for the following reason. By marginalizing $p(x, \lambda)$ over the precisions λ we get

$$\begin{aligned} p(x) &= \int p(x|\lambda)p(\lambda)d\lambda = \prod_i \int_0^\infty \left(\frac{\lambda_i}{2\pi}\right)^{\frac{1}{2}} \exp\left(-\frac{\lambda_i}{2} x_i^2\right) \frac{b^a \lambda_i^{a-1} \exp(-b\lambda_i)}{\Gamma(a)} d\lambda_i \\ &= \prod_i \frac{b^a}{\Gamma(a)} \frac{1}{\sqrt{2\pi}} \int_0^\infty \lambda_i^{a-\frac{1}{2}} \exp\left(-\lambda_i \left(b + \frac{x_i^2}{2}\right)\right) d\lambda_i \quad (\text{see [59, eq. 3.381-4]}) \\ &= \prod_i \frac{\Gamma(a + \frac{1}{2}) b^a}{\Gamma(a) \sqrt{2\pi}} \left(b + \frac{x_i^2}{2}\right)^{-a-\frac{1}{2}} \\ &= \prod_i \mathcal{S}(x_i|0, \eta, \nu), \quad \text{with } \eta = \frac{a}{b}, \quad \nu = 2a, \end{aligned} \quad (4.4)$$

where $\mathcal{S}(\cdot|0, \eta, \nu)$ denotes the zero-mean Student’s t -distribution with precision η and degrees of freedom ν defined as

$$\mathcal{S}(t|0, \eta, \nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})} \left(\frac{\eta}{\pi\nu}\right)^{\frac{1}{2}} \left(1 + \frac{\eta t^2}{\nu}\right)^{-\frac{\nu+1}{2}}. \quad (4.5)$$

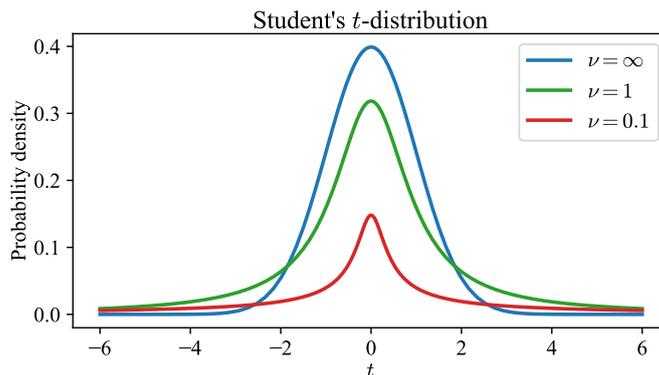


Figure 4.1: Probability density of Student’s t -distribution (4.5) for $\eta = 1$ and different values of the degrees of freedom ν .

The unconditioned prior of x is therefore independent and identical Student’s t -distribution, the original precisions λ_i are hidden (latent). The integral (4.4) is essentially an infinite sum of zero-mean Gaussians with different (Gamma-distributed) precisions and for this reason Student’s t -prior (and therefore ARD prior) can be viewed as a particular case of Gaussian scale mixture model.

Student’s t -distribution is generally more heavy-tailed than Gaussian, which means that it is more prone to producing values that fall far from its mean, but also that it is more robust (less sensitive to outliers) in regression, which may be a desirable property of a prior in inference and is the reason behind our interest in ARD. The heavy-tailedness of t -distribution depends on the parameter ν , degrees of freedom, as seen in eq. (4.5) and Fig. 4.1. For the marginalized ARD prior (4.4) the number of degrees of freedom is $\nu = 2a$, it is therefore the parameter a of the hyperprior $p(\lambda)$ what defines the robustness of our prior model. Looking at (4.3) we can identify the limit cases of the ARD model. Maximum robustness of ARD is achieved for $a \rightarrow 0$. If at the same time $a/b \rightarrow \bar{\lambda}$, then the precisions λ_i at each pixel have mean value $\bar{\lambda}$ but infinite variance, meaning maximum possible adaptation to the data; $p(\lambda)$ with this choice is called *uninformative*, since the posterior depends only on terms arising from the data and not from the prior. For $a \rightarrow \infty$ with $a/b \rightarrow \bar{\lambda}$, the distribution of pixel-wise precisions $p(\lambda_i)$ goes to delta distribution at $\bar{\lambda}$ and $p(x_i)$ goes to simple Gaussian $\mathcal{N}(x_i|0, \bar{\lambda})$, meaning zero adaptation to data. This case corresponds exactly to the ℓ^2 term appearing commonly for example as the data term $\frac{\bar{\lambda}}{2} \|h * u - g\|_2^2$ (see e.g. eq. (3.4) in the previous chapter). The advisable choice of the hyperprior parameters a and b when robustness is called-for is then setting them as small as possible while maintaining numerical stability, or making one more step in the inference hierarchy and assign these variable their respective hyperpriors $p(a)$ and $p(b)$ to let them be estimated from the data. Estimation of a and b via MAP was proposed by Tzikas *et al.* [173].

4.2 Model definition

Having defined the ARD prior model and established its basic properties, let us specify our model of the three primary unknowns in the deblurring problem – image, blur, and observation error.

Image prior

In the deblurring method presented in the previous chapter we defined the image prior in terms of the distribution of image derivatives as the well-known total variation (3.5). In this chapter we will again use image derivatives but generalization to other linear image features is obviously possible and straightforward. Using the ARD prior model (4.1) and (4.2) we define the image prior and the corresponding hyperprior of the precisions respectively as

$$p(u|\lambda) = \prod_i \mathcal{N}((\nabla u)_i | 0, \lambda_i^{-1}) \propto \prod_i \lambda_i^{1/2} \exp\left(-\frac{\lambda_i}{2}(\nabla u)_i^2\right), \quad (4.6a)$$

$$p(\lambda) = \prod_i \mathcal{G}(\lambda_i | a_\lambda, b_\lambda). \quad (4.6b)$$

where ∇ is again the concatenation of the partial derivative operators ∇_x and ∇_y as $\nabla = [\nabla_x^T, \nabla_y^T]^T$. Gaussian distribution of image derivatives is not a very good image prior per se, as it lacks any sparsity-promoting properties and as a result tends to smooth edges by over-penalizing large gradients, but we have seen that ARD is in general much more heavy-tailed than Gaussian due to the automatically determined pixel-wise precisions.

The generalized TV (3.8) that we used previously in Chapter 3 belongs to the family of commonly used ℓ^p ($p \leq 1$) priors and in comparison has the form

$$p_\ell(u|\lambda) \propto \lambda^{1/2} \prod_i \exp\left(-\frac{\lambda}{2}|\nabla u|_i^p\right). \quad (4.7)$$

It was demonstrated by Levin *et al.* [96] that ℓ^p priors counter-intuitively favor blurred images over sharp ones and therefore are not effective in avoiding the no-blur solution. The reason for this is that sparsity of image derivatives decreases with blur but the variance of image derivatives decreases as well. For natural images the latter effect is stronger and therefore the total prior probability increases with blur. It has been suggested by Krishnan *et al.* [88] to compensate this by normalizing the sparsity measure and use ℓ^1/ℓ^2 , but such measure cannot be used in probabilistic inference, because it is actually more of an explicit regularizer rather than a true prior, as it does not correspond to any probability distribution.

We subjected the ARD prior to a similar experiment to test how effective it is in avoiding image blur. First, though, we need to get rid of the dependence of $p(u|\lambda)$ on λ . One way of doing that is to marginalize over λ as in (4.4), which for $a_\lambda \rightarrow 0$ yields

$$p(u) \propto \prod_i ((\nabla u)_i^2 + 2b_\lambda)^{-\frac{1}{2}} = \prod_i \exp\left(\frac{1}{2} \log((\nabla u)_i^2 + 2b_\lambda)\right). \quad (4.8)$$

The marginalized ARD image prior is therefore the non-convex log prior of the form $\log(|\nabla u|^2 + \text{const})$, which has been used previously in image deblurring for its sparsity-promoting properties (approximately on par with ℓ^p , $p > 0$) while remaining smooth (e.g. [132]). But we do not use the image prior in its marginalized form, we instead infer the optimal precisions λ_i from the data and marginalization is essentially inference without data, relying solely on the hyperprior. Assuming the inference is carried out

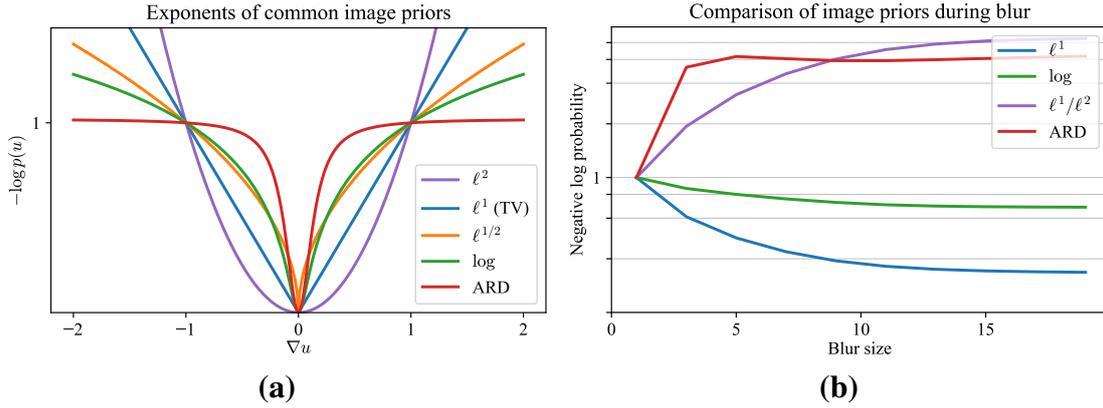


Figure 4.2: Image priors commonly used in image deblurring. **(a)** Exponents of image priors as a function of image derivative (normalized to 1 at $|\nabla u| = 1$). Displayed are ℓ^p priors (4.7), the log prior (4.8), and ARD (4.6). **(b)** Total probability (negative log, lower is better) of image prior as a function of degree of image blur, normalized to 1 for images without blur.

the same way as in the presented deblurring method, that is as the expectation of the approximate posterior, the unconditioned ARD image prior (for $a_\lambda \rightarrow 0$) has the form

$$p(u) \propto \prod_i \exp\left(-\frac{1}{2} \frac{(\nabla u)_i^2}{(\nabla u)_i^2 + 2b_\lambda}\right). \quad (4.9)$$

We see that it is a smooth, non-convex prior that very aggressively favors sparsity (asymptotically more so than any ℓ^p , $p > 0$ or the log prior) and in the limit $b_\lambda \rightarrow 0$ it goes to ℓ^0 , which is the ultimately sparse prior with improper distribution

$$-\log p(x_i) = \begin{cases} 0 & \text{if } x_i = 0, \\ 1 & \text{otherwise.} \end{cases} \quad (4.10)$$

Comparison of the discussed image priors in terms of their exponents can be seen in Fig. 4.2, which shows ℓ^p priors (4.7), the log prior (4.8), and ARD (4.6). The flatter the exponent is for large magnitudes of the image derivative, the more sparsity-promoting it is. See the striking difference between Gaussian with identical precision (violet) and ARD (red). All curves were normalized to attain value 1 for $|\nabla u| = 1$. Note that the discussed ℓ^1/ℓ^2 cannot be visualized this way.

How the image prior responds to blur can be seen in Fig. 4.2b, which depicts the total value (negative log) of the image prior as a function of the amount of blur present in the image (normalized to 1 for sharp images). We have evaluated each prior for many natural images (photos of animals, cars, indoor and outdoor scenes, humans and human faces, ...) with different amount of motion blur, from perfectly sharp to severely blurred, and averaged the results for each degree of blur. For a good image prior (in the context of image deblurring), the calculated value should be minimal for the sharp image and increase with blur, which would mean that the prior favors sharp images over blurred ones. We see in Fig. 4.2b that the ℓ^1 prior as well as the log prior fail in this test and actually decrease with blur. For the purpose of this experiment the ARD prior was calculated as though the inference of the precisions λ were optimal

with respect to the sharp image. This “ideal” ARD correctly increases with blur (red curve in Fig. 4.2b), which empirically justifies its suitability as an image prior for image deblurring, as it aids in avoiding the no-blur solution. Since the estimates of the precisions λ will not probably be as perfect in practice, this experiment demonstrates the upper bound of the ARD prior’s ability to avoid blur. The normalized measure ℓ^1/ℓ^2 passes this test even better but this measure cannot be integrated to our probabilistic framework.

Blur prior

To keep our model consistent, we use the ARD prior also for the blur kernel,

$$p(h|\beta) = \prod_i \mathcal{N}(h_i|0, \beta_i^{-1}) \propto \prod_i \beta_i^{1/2} \exp\left(-\frac{\beta_i}{2} h_i^2\right), \quad (4.11a)$$

$$p(\beta) = \prod_i \mathcal{G}(\beta_i|a_\beta, b_\beta). \quad (4.11b)$$

We have presented good reasons for using ARD as the image prior and will do so also for the noise prior, but using ARD as blur prior is perhaps hardest to justify. Blur prior is arguably less important than the image prior, because blur PSF is typically much smaller than the image and therefore the blur is well determined by the likelihood alone. Nevertheless, informative blur prior can be beneficial in avoiding local minima, increasing numerical stability, or speeding up the optimization. Common blur priors include $\|h\|_1$ or $\|h\|_2^2$ norms or convex constraints $h \geq 0$ or possibly even $\|h\|_1 = 1$. These two options can be combined, as in our method presented in the previous chapter where we used ℓ^1 norm and the constraint $h \geq 0$. We know from previous discussion that ARD is in general sparsity-promoting prior and in this respect will be more or less equivalent to using ℓ^1 . Including the non-negativity constraint, however, comes at the expense of making the VB inference substantially more complicated, as we have to forfeit using Gaussians and the resulting equations for optimal posterior approximations will no longer have a closed form. We instead opt for relative simplicity and keep using the Gaussian prior without explicitly enforcing non-negativity; later when presenting the optimization algorithm we will give some arguments while non-negativity of h is retained nevertheless. As a compromise, we tried to keep using the Gaussian priors in the VB inference and enforce non-negativity outside of the VB framework by using ADMM in the same fashion as described in “Convex constraint” on page 45. This hybrid approach, however, brings no significant improvement to plain VB and lacks its solid theoretical backing, we therefore do not use it in the subsequent presentation.

Observation error prior

At this point it will not come as a surprise to say that we use ARD model as the prior of the observation error (“noise”) $n = g - h * u$. Majority of deblurring methods make the assumption that the noise is i.i.d. zero-mean Gaussian,

$$p_G(n|\alpha) = \prod_i \mathcal{N}(n_i|0, \alpha^{-1}). \quad (4.12)$$

We have seen several demonstrations that such prior is unable to cope with local acquisition model violations, which then impede correct blur estimation or cause restoration artifacts. The assumption of Gaussian noise is not wrong, though, wrong is the assumption that the acquisition error is only due to noise. It is more realistic to assume that this error is composed of both random noise, which is close to Gaussian, and non-Gaussian perturbations like infidelity of the convolution blur model, local model violations (saturation, dead pixels, blooming), digital quantization and others. Example of actual distribution of the observation error in real blurred photograph can be seen in Fig. 4.4 (blue histogram).

To retain this separation of Gaussian and non-Gaussian component we slightly generalized the conventional ARD model (4.1) and for the observation error n use the prior

$$p(n|\alpha, \gamma) = \prod_i \mathcal{N}(n_i|0, (\alpha \gamma_i)^{-1}) \propto \prod_i (\alpha \gamma_i)^{1/2} \exp\left(-\frac{\alpha \gamma_i}{2} n_i^2\right), \quad (4.13a)$$

$$p(\alpha) = \mathcal{G}(\alpha|a_\alpha, b_\alpha), \quad (4.13b)$$

$$p(\gamma|\nu) = \prod_i \mathcal{G}(\gamma_i|\nu, \nu), \quad (4.13c)$$

$$p(\nu) = \mathcal{G}(\nu|a_\nu, b_\nu). \quad (4.13d)$$

The scalar $\alpha \in \mathbb{R}^+$ is the precision of the Gaussian noise component, modulated locally by γ_i , which models the non-Gaussian component. We call this formulation *ARD model with common precision factor and automatically estimated degrees of freedom*. In classical MAP estimation it would correspond to the data-fidelity term $\frac{\alpha}{2} \sum_i \gamma_i (h * u - g)_i^2$.

According to (4.4), calculating the expectation of $p(n|\alpha, \gamma)$ with respect to γ yields zero-mean Student's t -distribution with precision α and 2ν degrees of freedom. Notice that we took the inference one step further and instead of prescribing a hard value for the number of degrees of freedom 2ν , as in the case of image and blur prior with a_λ and a_β , respectively, we only assigned a hyperprior $p(\nu)$ and let ν be inferred from the data. We know from the discussion about Student's t -distribution on page 50 that Gaussian is the limit of t -distribution for $\nu \rightarrow \infty$. In this particular case we see from (4.3) that, independently of ν , the mean of the pixel-wise precisions is $\bar{\gamma}_i = \nu/\nu = 1$. For $\nu \rightarrow \infty$, $p(\gamma|\nu)$ goes to delta distribution at $\gamma_i = 1$ and $p(n|\alpha)$ goes exactly to $\mathcal{N}(n|0, \alpha)$. The conventional i.i.d. Gaussian noise model (4.12) is therefore a limit case of our model for zero presence of non-Gaussian error component. By contrast, the maximum capacity for local non-Gaussian error is achieved for $\nu \rightarrow 0$, then $\text{var}[\gamma_i] = 1/\nu \rightarrow \infty$ and the asymptotical decay of $p(n|\alpha)$ is slowest. The automatically estimated number of degrees of freedom 2ν can therefore be viewed as an inverse measure of non-Gaussian error present in the input.

For comparison purposes, we also define an alternative noise model using the conventional ARD without the common Gaussian precision factor α and with the number

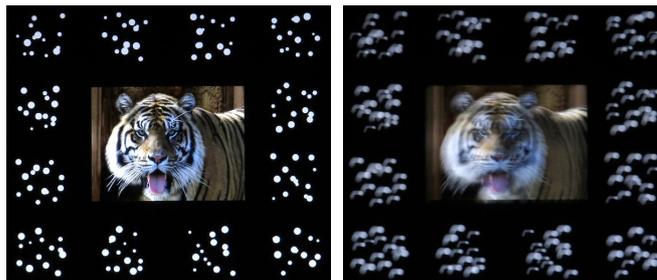


Figure 4.3: Example image pair used to determine the error distribution of real motion blur. The right image was acquired with intentional camera motion and the blur PSF was determined using the patterns surrounding both of the photographed images.

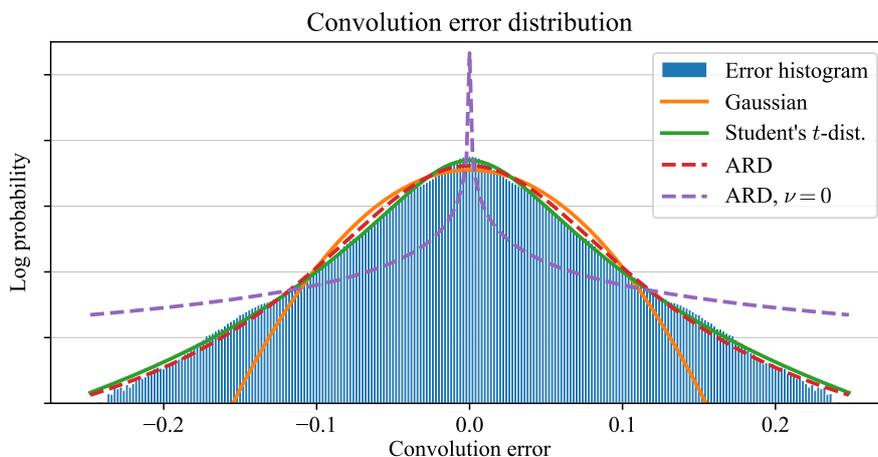


Figure 4.4: Distribution of the convolution model error for real motion blur (blue bar histogram, log scale) and maximum likelihood fits of several discussed distributions. Commonly used Gaussian (orange) cannot correctly capture the tails. ARD (red) with free choice of the a and b parameters as well as the t -distribution (green), corresponding to marginalized ARD, are perfect fits. ARD with degrees of freedom fixed to zero (violet) is too heavy-tailed.

of degrees of freedom specified as a hard-valued hyperparameter,

$$p(n|\gamma) = \prod_i \mathcal{N}(n_i|0, \gamma_i^{-1}) \propto \prod_i \gamma_i^{1/2} \exp\left(-\frac{\gamma_i}{2} n_i^2\right), \quad (4.14a)$$

$$p(\gamma) = \prod_i \mathcal{G}(\gamma_i|a_\gamma, b_\gamma). \quad (4.14b)$$

Marginalization over γ yields that $p(n)$ is a Student's t -distribution with $2a_\gamma$ degrees of freedom. We will prescribe this value as a parameter (aiming at the limit $a_\gamma \rightarrow 0$ for maximum robustness) but MAP estimation of this hyperparameter is possible and was proposed in Tzikas *et al.* [173], where this error model was also used.

Similarly to the discussion about the image prior model, we verified experimentally that the proposed error model (4.13) is suitable for image deblurring. In the first experiment we measured how well this model corresponds to actual error distribution of the convolution acquisition model (3.1). We conjecture that under real image acquisition conditions, the observation error n is never purely normally distributed. Different

factors inherently present in the acquisition process, such as lens imperfections, camera sensor discretization and quantization, or small blur spatial-variance contribute to the error of the convolution model on top of the random measurement noise and the resulting distribution is more heavy-tailed. It was reported by Tzikas *et al.* [173] that small error in the blur estimation also produces heavy-tailed error distribution in the image estimation. To verify our conjecture, we acquired several pairs of sharp–blurred images (u, g) with intentional camera motion during exposure. Except for this, we carefully avoided any other kinds of acquisition error, like pixel saturation or space-invariance of the blur, and worked strictly with raw data from the camera. For each of these pairs we estimated the blur PSF h following the procedure suggested by Tezaur *et al.* [168], which uses patterns printed around the image designed to make the blur identification stable; see example image pair in Fig. 4.3. For this data, we measured the error n of the convolution model (3.1) and plotted its histogram (log thereof) in Fig. 4.4. We performed the same measurement on the dataset by Levin *et al.* [96] with known blur PSFs and obtained the same heavy-tailed distribution of the convolution error.

We can see that the error distribution is far from normal and the maximum likelihood fit of the Gaussian noise model (4.12) (orange) provides very poor approximation. The Student’s t -distribution $\mathcal{S}(0|\eta, \nu)$ (green) and also directly the ARD model (4.13), on the other hand, approximate the true error distribution very accurately, which justifies the choice of the ARD model as the noise prior. The precisions γ_i of ARD were calculated similar way as in the presented deblurring algorithm, that is using eq. (4.32) (to be derived) but assuming that u and h are fixed values, not random variables. We also tried fitting the conventional ARD model (4.14) with degrees of freedom $\nu = 2a_\gamma$ fixed to zero and the resulting MLE fit (violet) far too heavy-tailed for images where the convolution model largely holds, which suggests that having the number of degrees estimated can be beneficial if the amount of non-Gaussian error cannot be gauged in advance.

In the next experiment we investigated how the non-normality of the observation error affects blur estimation in the blind setting. In particular, we tested whether the error prior attains its maximum for the correct blur PSF even in the presence of non-Gaussian observation error. The setup of our experiment was as follows. We took a sharp image u and set a certain percentage of randomly selected pixels to high values to simulate overexposure, which represents a non-Gaussian error after intensity clipping. We then blurred the saturated image with the “true” blur with PSF h^* (Gaussian), added some Gaussian noise and clipped the image intensities to obtain the final observed image g . We then measured the likelihood of several blur candidates h (Gaussians with different σ) by evaluating the negative log of the error prior. An error prior model suitable for blind image deblurring should attain its minimum for the true blur h^* .

Results of this experiment are in Fig. 4.5. The left plot corresponds to the standard Gaussian error model (4.12) and the right plot corresponds to the presented ARD error model (4.13). Individual line colors represent different percentage of pixels intentionally corrupted by non-Gaussian error. The y -axis shows the negative log likelihood of the tested blurs h as a function of their size (x -axis). The minimum of each line is marked by a small dot. The size of the true blur h^* is indicated by a vertical line at $x = 2.2$. For the Gaussian model (left plot), the minimum is reached for the cor-

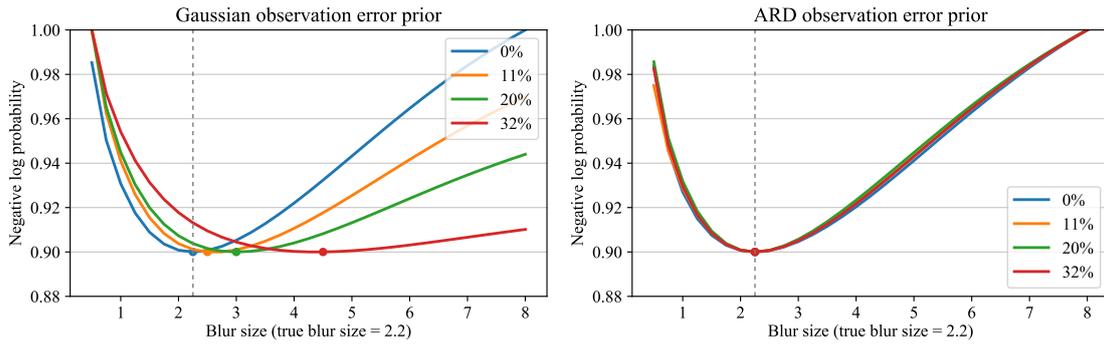


Figure 4.5: Relative likelihood value (negative log, low value means high probability) for commonly used Gaussian (left) and ARD (right) noise priors as a function of potential image blur size; the correct blur size is indicated by the vertical line at 2.2. Different curves (colors) correspond to different percentages of non-Gaussian error (sensor saturation) in the input, the curves’ minima are accentuated by dots. The Gaussian prior favors larger and larger blurs as the non-Gaussian error increases, the ARD prior remains virtually unaffected and correctly identifies the true blur size.

rect blur size only if the noise is purely Gaussian. As the presence of non-Gaussian error increases, the minimum shifts further and further away from the true value, effectively hindering the correct blur estimation. The ARD model (right plot), on the other hand, remains virtually unaffected by the non-Gaussian error. We can conclude that for blind image deblurring, ARD is much superior error prior model to standard Gaussian, especially when non-Gaussian observation error can be expected.

We will now briefly recall the general framework of VB inference and then proceed with detailed derivation of its application to our particular problem of blind image deblurring with the defined acquisition and prior models.

4.3 Variational Bayesian inference

Appearing also under the names variational Bayes, approximate inference or ensemble learning, variational Bayesian inference is a family of techniques for approximating intractable distributions arising in Bayesian inference. It can be viewed as a counterpart of sampling methods (such as Markov chain Monte Carlo), where inexact inference is done on exact distributions, whereas VB allows exact inference on approximations of these distributions. Contrary to other probabilistic techniques, VB seeks the solution in the form of a full distribution (e.g. approximation of the posterior) instead of making approximate point estimates on this distribution (e.g. mode) as in MAP.

Suppose we have a Bayesian model where X denotes the set of all observed variables and Z denotes the set of all unknown unobserved variables (latent variables or model parameters) and our model is defined by the joint distribution $p(X, Z)$ and the prior $p(Z)$. Common problem of interest is inference on the posterior $p(Z|X)$, but this distribution is usually intractable, as is in the case of our image deblurring problem. Variational approach is to find a tractable distribution $q(Z)$ that in some sense optimally approximates $p(Z|X)$ and facilitates inference. The minimized measure of

dissimilarity between $q(Z)$ and $p(Z|X)$ used in VB is the *Kullback–Leibler divergence* defined as

$$\text{KL}(q\|p) = \int q(Z) \log \left(\frac{q(Z)}{p(Z|X)} \right) dZ. \quad (4.15)$$

KL divergence satisfies $\text{KL}(q\|p) \geq 0$ and $\text{KL}(q\|p) = 0$ if and only if $q = p$ a.e., as can be seen by

$$\begin{aligned} \text{KL}(q\|p) &= \int q \log \frac{q}{p} = - \int q \log \frac{p}{q} \geq - \int q \left(\frac{p}{q} - 1 \right) \\ &= \int p(Z|X) dZ - \int q(Z) dZ = 1 - 1 = 0, \end{aligned} \quad (4.16)$$

where we used the fact that $\log(x) \leq x - 1$ and equality holds only if $x = 1$. It is not a true metric, though, as it is not symmetric and does not satisfy triangle inequality. KL divergence often emerges in other fields, e.g. in information theory, often with the role of p and q reversed. KL divergence is closely related to cross-entropy (it is sometimes even called relative entropy) and $\text{KL}(q\|p)$ can be interpreted as the information lost if q is used instead of p .

In the variational approximation of the posterior KL divergence arises quite naturally, as we show below. The posterior can be expressed as

$$p(Z|X) = \frac{p(X, Z)}{p(X)}, \quad \text{where } p(X) = \int p(X, Z) dZ. \quad (4.17)$$

The joint probability defines the model of our problem, it is fixed and usually known explicitly, whereas the marginal likelihood $p(X)$, also called *model evidence*, is in most cases analytically intractable and the reason why the posterior itself is intractable. Approximating the posterior and approximating the model evidence are therefore equivalent problems. By taking the logarithm of (4.17) and doing a small algebraic exercise

$$\begin{aligned} \log p(X) &= \log p(X) \int q(Z) dZ = \int q(Z) \log p(X) dZ \\ &= \int q(Z) \left(\log p(X, Z) - \log p(Z|X) \right) dZ \\ &= \int q(Z) \left(\log q(Z) - \log p(Z|X) + \log p(X, Z) - \log q(Z) \right) dZ \end{aligned} \quad (4.18)$$

we see that the model evidence can be expressed as

$$\log p(X) = \text{KL}(q\|p) + L(q), \quad (4.19)$$

where

$$L(q) = \int q(Z) \log \left(\frac{p(X, Z)}{q(Z)} \right) dZ. \quad (4.20)$$

The model evidence $p(X)$ is conditioned on the model structure, but independent of its state (parameters or latent variables), the left-hand side of (4.19) is therefore constant with respect to Z and especially $q(Z)$. Since KL divergence is always non-negative, we have $L(q) \leq \log p(X)$. The functional $L(q)$ is aptly named *evidence*

lower bound (conventionally abbreviated ELBO) and depending on our choice of $q(Z)$ it approaches $\log p(X)$ from below, the gap (measure of nonoptimality of $p(X)$ approximation) is given precisely by the KL divergence. For the optimal choice $q^*(Z)$ such that $L(q^*) = \log p(X)$ we also have $q^*(Z) = p(Z|X)$, which explicitly connects the optimal solution of the posterior approximation to the solution of model evidence approximation by maximizing ELBO. It is then only natural to adopt the same optimality criterion also for our primary problem of variational approximation of the posterior and choose KL divergence between the candidate distribution and the posterior as the optimized loss.

Globally optimal choice of $q(Z)$ is obviously the posterior itself but practical variational inference is based on restricting the possible choices of $q(Z)$ such that the subsequent inference (e.g. calculation of expected value) is computationally tractable. Apart from this criterion the family of admissible $q(Z)$'s should be as rich as possible, there is no advantage to being conservative – to quote from Bishop [12], “*There is no overfitting.*”

We will focus on a particular way of restricting the family of feasible approximations, which is based on the assumption that the approximation $q(Z)$ is (partially) factorizable. This idea originated in statistical physics in the form of the mean-field theory and is by far the most popular choice of variational approximation of the posterior in Bayesian learning, often correspondingly called *mean-field approximation*.

Formally, let the set of unknowns Z be partitioned into mutually disjoint groups $Z = Z_1 \cup Z_2 \cup \dots \cup Z_M$. We then restrict the family of feasible distributions by imposing that $q(Z)$ factorizes over this partition,

$$q(Z) = \prod_i q(Z_i). \quad (4.21)$$

No further assumption about the functional form of the factors $q(Z_i)$ is made. The partitioning is typically naturally dictated by the problem, for example in our case of image deblurring it is natural to consider all unknowns related directly to the image (pixel intensities) as one group and all unknowns directly related to the blur PSF as another group. Further fragmentation (e.g. treating each pixel as a standalone group) leads to more restrictive assumption and therefore poorer approximation.

We seek such factors $q(Z_i)$ that maximize ELBO (4.20). Substituting the factorized form (4.21) into (4.20) and isolating a particular factor $q(Z_j)$ we get

$$\begin{aligned} L(q) &= \int \prod_i q(Z_i) \left(\log p(X, Z) - \sum_i \log q(Z_i) \right) dZ \\ &= \int q(Z_j) \left(\int \log p(X, Z) \prod_{i \neq j} q(Z_i) dZ_i \right) dZ_j - \int q(Z_j) \log q(Z_j) dZ_j \\ &\quad - \sum_{i \neq j} \int q(Z_i) \log q(Z_i) dZ_i. \end{aligned} \quad (4.22)$$

Of the resulting three terms, the last is constant with respect to $q(Z_j)$ and the first two can be recognized to have the form of a negative KL divergence $\text{KL}(q(Z_j) \| p^*(X, Z_j))$

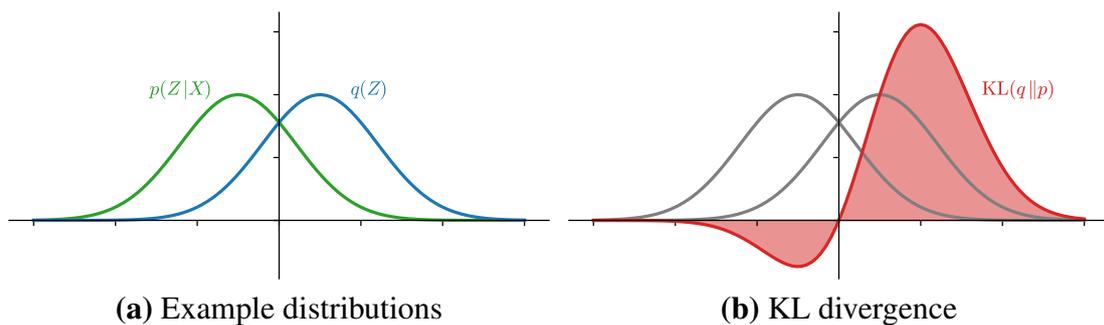


Figure 4.6: Illustration of the asymmetry of KL divergence. Example distributions p and q (shifted Gaussians) are in (a). In (b) is the corresponding integrated area of KL divergence. The penalty for $q > 0$ where $p \approx 0$ is much higher than the reward for $q > 0$ where $p > 0$.

where the role of $\log p^*(X, Z_j)$ is played by the expression **in the red** (up to a constant independent of $q(Z_j)$ due to the necessary normalization of $p^*(X, Z_j)$). In particular, the contribution of these terms is non-positive. To maximize $L(q)$ in terms of $q(Z_j)$ we therefore have to minimize $\text{KL}(q(Z_j) \| p^*(X, Z_j))$ and we know that KL divergence is minimized when its two arguments are equal, which gives us the optimality condition for $q(Z_j)$,

$$\log q(Z_j) = \int \log p(X, Z) \prod_{i \neq j} q(Z_i) dZ_i + \text{const.} \quad (4.23)$$

The expression on the right-hand side has the form of an expectation over all unknowns Z_i except Z_j , denoted $\mathbb{E}_{i \neq j}[\cdot]$. The optimality condition is then conventionally written in terms of expectation of the joint probability as

$$\log q(Z_j) = \mathbb{E}_{i \neq j} \left[\log p(X, Z) \right] + \text{const.} \quad (4.24)$$

The corresponding constant is given by the normalization of $q(Z_j)$ and is best determined ex post, if necessary.

Unfortunately, the equations (4.24) for $j = 1, \dots, M$ do not provide explicit formulas for the optimal approximating factors, since expression for each factor $q(Z_j)$ depends on all other factors $q(Z_i)$, $i \neq j$. They present a system of integral equations that must be solved, which usually is not possible. We must resort to an iterative solution via alternating coordinate ascent in the space of $q(Z_j)$. From a suitable initialization we cycle through $j = 1, \dots, M$ repeatedly and update each factor $q(Z_j)$ by evaluating the right-hand side of (4.24) using the currently available estimates of $q(Z_i)$ until a convergence criterion is met (see e.g. Tran [171] and references therein for proof of convergence to a local optimum).

Due to its asymmetry, KL divergence gives higher penalty when $q(Z)$ is large and $p(Z|X)$ is small than vice versa, this is well illustrated in Fig. 4.6. The approximation therefore tends to “underestimate” the posterior in the sense that areas in Z where $p(Z|X)$ is relatively large can be ignored by $q(Z)$ if $q(Z)$ is so restricted and would otherwise have to attain significantly nonzero values where $p \approx 0$. In a particular case when the posterior is multimodal and is approximated by a unimodal distribution, the approximation tends to concentrate around one of the modes (depending on

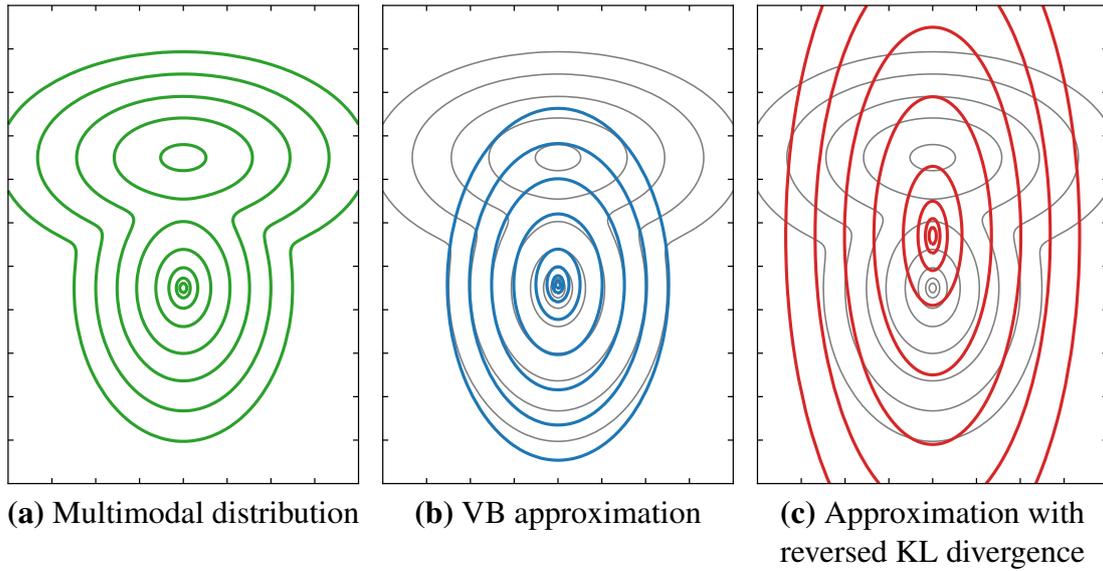


Figure 4.7: Illustration of local behavior of VB approximation. **(a)** Levelset contours of an example multimodal distribution p . **(b)** Locally optimal approximating distribution q when the family of feasible approximations is restricted to Gaussians. **(c)** Globally optimal solution of the same approximation, if the KL divergence is reversed and minimized as $\text{KL}(p\|q)$. Depicted isolines are mutually corresponding between plots (enclosing the same probability mass).

initialization of the above-mentioned procedure) and ignore the others, see Fig. 4.7 for illustration. This can be considered a good characteristic when the subsequent inference seeks e.g. the mode of $q(Z)$, which will faithfully correspond to one of the modes of $P(Z|X)$ and will therefore give a meaningful solution (e.g. a probable sharp image). For estimation of other moments (e.g. variance), the approximation may be grossly inadequate. If the situation were reversed (which can be achieved by reversing the KL divergence), the approximation would attempt to cover all areas where the posterior is large, in our particular example absorb all the modes of the posterior into a single mode (see Fig. 4.7c). The location of mode of $q(Z)$ would then correspond to a sort of average of the locations of modes of the posterior (e.g. average of several different sharp images), which is hardly a good solution.

This discussion concludes our short overview of variational Bayes method, we will now present its application to our problem of blind image deblurring.

4.4 Posterior approximation

Following the above outlined procedure, we approximate the posterior by a suitably factorizable distribution. In our problem the set of observed data consists of the blurred image g and the unknowns Z are partitioned as follows, $Z = \{u, h, \alpha, \gamma, \lambda, \beta, \nu\}$. We then approximate the posterior as $p(Z|g) \approx q(Z) = q(u)q(h)q(\alpha)q(\gamma)q(\lambda)q(\beta)q(\nu)$. Our ultimate goal regarding blur estimation is to find the mode of $q(h)$, which, due to factorization, effectively corresponds to the marginalization of $q(Z)$ over all other unknowns. The fully expanded joint probability of our model including all latent vari-

ables can be expressed in terms of the likelihood and priors as

$$p(X, Z) = p(X|Z)p(Z) = p(g|u, h, \alpha, \gamma)p(u|\lambda)p(\lambda)p(h|\beta)p(\beta)p(\alpha)p(\gamma|\nu)p(\nu), \quad (4.25)$$

where the likelihood is given by the acquisition model (3.1) and the observation error prior (4.13)

$$p(g|u, h, \alpha, \gamma) \propto \prod_i (\alpha \gamma_i)^{\frac{1}{2}} \exp\left(-\frac{\alpha \gamma_i}{2}(h * u - g)_i^2\right) \quad (4.26)$$

and all the prior distributions have been defined previously. To derive explicit iterative optimization procedure we now have to evaluate the update equation (4.24) for each factor $q(Z_j)$. It is helpful to have the likelihood (4.26) expressed in terms of its logarithm, since it appears repeatedly in the derivations.

$$\log p(g|u, h, \alpha, \gamma) = \frac{N}{2} \log \alpha + \frac{1}{2} \sum_i \log \gamma_i - \frac{\alpha}{2} \sum_i \gamma_i (h * u - g)_i^2, \quad (4.27)$$

where N denotes number of pixels in the image. To keep the derivations compact, we omit all terms that integrate to constants due to lack of dependence on $q(Z_j)$ as well as any other constants or factors related to normalization (as we have already done in the last equation). In accordance with (4.24), all expectations in the subsequent derivations are calculated with respect to all terms $q(Z_i)$ except $q(Z_j)$ whose update equation is being derived. And similarly to the previous chapter, we will denote by H and U the circulant (more precisely BCCB) matrices corresponding to (circular) convolution with h and u , respectively, such that $h * u \equiv Hu \equiv Uh$.

Despite the risk of losing the reader's interest halfway, we present the variational approximation and corresponding update equations of the resulting optimization in full detail, including derivation up to explicit equations that do not contain any "hidden" non-trivial terms. In other literature, where similar model appeared, usually only a selected part of the resulting equations were stated due to space constraints and unfortunately sometimes even with errors, which are then hard to spot and fix if the derivation is missing. We also frequently provide a commentary on what the less obvious arising terms represent and how they can be calculated in practice, which we believe should be of value to anyone willing to implement a similar model.

Likelihood terms $q(\gamma)$, $q(\alpha)$, and $q(\nu)$

From (4.13) we have

$$\log p(\gamma|\nu) = (\nu - 1) \sum_i \log \gamma_i - \nu \sum_i \gamma_i. \quad (4.28)$$

Substituting this and (4.27) to eq. (4.24) for $q(\gamma)$ we get

$$\begin{aligned} \log q(\gamma) &= \mathbb{E} [\log p(g|u, h, \alpha, \gamma)] + \mathbb{E} [\log p(\gamma|\nu)] \\ &= \frac{1}{2} \sum_i \log \gamma_i - \sum_i \gamma_i \mathbb{E} \left[\frac{\alpha}{2} (h * u - g)_i^2 \right] + \mathbb{E} [\nu - 1] \sum_i \log \gamma_i - \mathbb{E} [\nu] \sum_i \gamma_i \\ &= \left(\bar{\nu} - \frac{1}{2} \right) \sum_i \log \gamma_i - \sum_i \left(\frac{\bar{\alpha}}{2} \mathbb{E} [(h * u - g)_i^2] + \bar{\nu} \right) \gamma_i. \end{aligned} \quad (4.29)$$

By comparison with the log of general expression for a product of non-identical Gamma distributions (4.3)

$$\log \left(\prod_i \mathcal{G}(t_i | a_i, b_i) \right) = \sum_i (a_i - 1) \log t_i - \sum_i b_i t_i \quad (4.30)$$

we see that $q(\gamma_i)$ is a Gamma distribution with

$$a = \bar{\nu} + \frac{1}{2} \quad \text{and} \quad b_i = \frac{\bar{\alpha}}{2} \mathbb{E}_{u,h} [(h * u - g)_i^2] + \bar{\nu}. \quad (4.31)$$

For its expected value this implies (by (4.3))

$$\bar{\gamma}_i = \frac{2\bar{\nu} + 1}{\bar{\alpha} \mathbb{E}_{u,h} [(h * u - g)_i^2] + 2\bar{\nu}}. \quad (4.32)$$

What remains is to specify how to calculate the expectation in the denominator. Let us denote by H_i the i -th row of the convolution matrix H , such that $H_i u = (h * u)_i$. From the definition of 2D convolution it follows that H_i is nothing but a fixed permutation (and zero-padding due to different sizes of u and h) P_i of the elements of h that depends on the particular image pixel location i , i.e. $H_i^T = P_i h$. More intuitively, $P_i h$ is simply the PSF h flipped and placed (using circular wrapping) at pixel position i of u -sized zero “image” and reshaped into a column vector.

To expand the expectation we will use the elementary relation that for any two independent random vectors x and y the (co)variance of their dot product can be expressed as

$$\text{cov}(x^T y) = \bar{x}^T \text{cov}(y) \bar{x} + \bar{y}^T \text{cov}(x) \bar{y} + \text{tr}(\text{cov}(x) \text{cov}(y)). \quad (4.33)$$

We then have

$$\begin{aligned} \mathbb{E}_{u,h} [(H_i u - g_i)^2] &= (\bar{H}_i \bar{u} - g_i)^2 + \text{cov}(H_i u) \\ &= (\bar{h} * \bar{u} - g)_i^2 + \bar{H}_i \text{cov}(u) \bar{H}_i^T + \bar{u}^T \text{cov}(H_i^T) \bar{u} + \text{tr}(\text{cov}(H_i^T) \text{cov}(u)) \\ &= (\bar{h} * \bar{u} - g)_i^2 + \bar{h}^T P_i^T \text{cov}(u) P_i \bar{h} + \bar{u}^T P_i \text{cov}(h) P_i^T \bar{u} + \text{tr}(P_i \text{cov}(h) P_i^T \text{cov}(u)). \end{aligned} \quad (4.34)$$

The covariance terms look complicated but they are nothing but a simultaneous re-ordering of rows and columns of the covariance matrix in exactly the same fashion as pixels of the convolution kernel are distributed during the sliding operation of 2D convolution. In practice it is not necessary to explicitly construct any permutation matrices, with suitable representation of the covariance matrices these calculations can be done fairly efficiently using convolution-like operations. We will suggest a general approach as part of the discussion that follows the derivation of the $q(u)$ factor. In the special case that $\text{cov}(u)$ and $\text{cov}(h)$ are diagonal, we can explicitly rewrite the expectation as follows. Let $h_C = \text{diag}(\text{cov}(h))$ and $u_C = \text{diag}(\text{cov}(u))$ denote the diagonals of the covariance matrices. We shall treat these the same way we treat the blur vector h and the image vector u when used in 2D convolution. Then the expectation in the denominator of (4.32) can be written for all i as a vector

$$\mathbb{E}_{u,h} [(h * u - g)^2] = (\bar{h} * \bar{u} - g)^2 + \bar{h}^2 * u_C + h_C * \bar{u}^2 + h_C * u_C, \quad (4.35)$$

where all powers of a vector are calculated elementwise.

When using the simplified noise prior (4.14) without automatically estimated degrees of freedom ν , the expected value of γ is given similarly by

$$\bar{\gamma}_i = \frac{2a_\gamma + 1}{\mathbb{E}_{u,h} [(h * u - g)_i^2] + 2b_\gamma}. \quad (4.36)$$

Recall that a_γ and b_γ are user-specified parameters.

To derive $q(\alpha)$ we proceed analogously. From (4.13) we have

$$\log p(\alpha) = (a_\alpha - 1) \log \alpha - b_\alpha \alpha. \quad (4.37)$$

Substituting to (4.24) for $q(\alpha)$ we get

$$\begin{aligned} \log q(\alpha) &= \mathbb{E} [\log p(g|u, h, \alpha, \gamma)] + \mathbb{E} [\log p(\alpha)] \\ &= \frac{N}{2} \log \alpha - \frac{\alpha}{2} \sum_i \mathbb{E} [\gamma_i (h * u - g)_i^2] + (a_\alpha - 1) \log \alpha - b_\alpha \alpha \\ &= \left(\frac{N}{2} + a_\alpha - 1 \right) \log \alpha - \left(\frac{1}{2} \sum_i \bar{\gamma}_i \mathbb{E} [(h * u - g)_i^2] + b_\alpha \right) \alpha. \end{aligned} \quad (4.38)$$

By comparison with (4.30) (recall that α is a scalar so the sums contain only one term) we see that $q(\alpha)$ is again Gamma distribution with expected value

$$\bar{\alpha} = \frac{N + 2a_\alpha}{\sum_i \bar{\gamma}_i \mathbb{E}_{u,h} [(h * u - g)_i^2] + 2b_\alpha}. \quad (4.39)$$

The expectation in the denominator is calculated in the same fashion as in (4.34).

Deriving the update equation for $\bar{\nu}$ is slightly different because ν is a parameter of Gamma distribution, not Gaussian as in the previous cases. Including all ν -related terms and omitting others (hence the difference from (4.28)), we have from (4.13)

$$\log p(\gamma|\nu) = N \log \left(\frac{\nu^\nu}{\Gamma(\nu)} \right) + \nu \sum_i \log \gamma_i - \nu \sum_i \gamma_i, \quad (4.40)$$

$$\log p(\nu) = (a_\nu - 1) \log \nu - b_\nu \nu. \quad (4.41)$$

To get rid of the awkward gamma function in the logarithm, we approximate gamma function by the Stirling's formula as

$$\Gamma(z) \approx \sqrt{\frac{2\pi}{z}} \left(\frac{z}{e} \right)^z, \quad (4.42)$$

then

$$\log \left(\frac{\nu^\nu}{\Gamma(\nu)} \right) \approx \nu + \frac{1}{2} \log \nu. \quad (4.43)$$

Substituting these to (4.24) for $q(\nu)$ we get

$$\begin{aligned}
 \log q(\nu) &= \mathbb{E} [\log p(\gamma|\nu)] + \mathbb{E} [\log p(\nu)] \\
 &= N\nu + \frac{N}{2} \log \nu + \nu \mathbb{E} \left[\sum_i \log \gamma_i \right] - \nu \mathbb{E} \left[\sum_i \gamma_i \right] + (a_\nu - 1) \log \nu - b_\nu \nu \\
 &= \left(\frac{N}{2} + a_\nu - 1 \right) \log \nu - \left(\sum_i \bar{\gamma}_i - \sum_i \mathbb{E} [\log \gamma_i] - N + b_\nu \right) \nu,
 \end{aligned} \tag{4.44}$$

which we recognize again as Gamma distribution with expected value

$$\bar{\nu} = \frac{N + 2a_\nu}{2 \sum_i (\bar{\gamma}_i - 1 - \mathbb{E}_\gamma [\log \gamma_i]) + 2b_\nu}. \tag{4.45}$$

General expression for $\mathbb{E}[\log x]$ for Gamma-distributed random variable x is known as (Bishop [12, eq. (B.30)])

$$\mathbb{E}[\log x] = \psi(a) - \log b, \quad x \sim \mathcal{G}(x|a, b), \tag{4.46}$$

where ψ is the digamma function defined as the derivative of $\log \Gamma(\cdot)$. This expression can be derived by substituting $x = \exp(y)$ and calculating the expectation of y using differentiating by a under the integral. In our case this yields (substituting from (4.31) and denoting $\bar{\nu}_{\text{old}}$ the current estimate of $\bar{\nu}$, i.e. before (4.45) is evaluated)

$$\mathbb{E}[\log \gamma_i] = \log \bar{\gamma}_i + \psi \left(\bar{\nu}_{\text{old}} + \frac{1}{2} \right) - \log \left(\bar{\nu}_{\text{old}} + \frac{1}{2} \right). \tag{4.47}$$

Image terms $q(u)$ and $q(\lambda)$

To derive the expression for $q(u)$, it is practical to express the logarithm of the likelihood and of the image prior using vector notation rather than pixel-wise, as we have done so far. From (4.26) and (4.6) we have

$$\log p(g|u, h, \alpha, \gamma) = -\frac{\alpha}{2} (Hu - g)^T \Gamma (Hu - g), \tag{4.48}$$

$$\log p(u|\lambda) = -\frac{1}{2} u^T \nabla^T \Lambda \nabla u, \tag{4.49}$$

where $\Gamma = \text{diag}(\gamma)$ and $\Lambda = \text{diag}(\lambda)$ are the diagonal precision matrices of the likelihood and the image prior, respectively. Substituting these to (4.24) for $q(u)$ we get

$$\begin{aligned}
 \log q(u) &= \mathbb{E} [\log p(g|u, h, \alpha, \gamma)] + \mathbb{E} [\log p(u|\lambda)] \\
 &= -\frac{1}{2} \mathbb{E} \left[\alpha (Hu - g)^T \Gamma (Hu - g) + u^T \nabla^T \Lambda \nabla u \right].
 \end{aligned} \tag{4.50}$$

The right-hand side is quadratic with respect to u , which suggests that $q(u)$ is a Gaussian. We can determine the mean by taking the first derivative and set it equal to zero and the covariance as the negative inverse of the second derivative. We get respectively

$$\left(\bar{\alpha} \mathbb{E}_h [H^T \bar{\Gamma} H] + \nabla^T \bar{\Lambda} \nabla \right) \bar{u} = \bar{\alpha} \bar{H}^T \bar{\Gamma} g, \tag{4.51}$$

$$\text{cov}(u) = \left(\bar{\alpha} \mathbb{E}_h [H^T \bar{\Gamma} H] + \nabla^T \bar{\Lambda} \nabla \right)^{-1}. \tag{4.52}$$

Both equations require solving large linear systems with non-circulant matrix. Using the elementary relation for the expectation of the dot product of random vectors $\mathbb{E}[x^T y] = \bar{x}^T \bar{y} + \text{tr}(\text{cov}(x, y))$, the expectation term can be expanded as

$$\mathbb{E}_h [H^T \bar{\Gamma} H] = \bar{H}^T \bar{\Gamma} \bar{H} + C^h, \quad (4.53)$$

with

$$C_{ij}^h = \text{tr}(\text{cov}(H_{\cdot,i}, \bar{\Gamma} H_{\cdot,j})) = \text{tr}(\bar{\Gamma} P_i^* \text{cov}(h) P_j^{*T}), \quad (4.54)$$

where $H_{\cdot,i}$ denotes i -th column of H . It follows that $H_{\cdot,i}$ is again just a permutation of h as $H_{\cdot,i} = P_i^* h$ but the permutation matrices P_i^* are different from P_i in the previous case in the sense that there is no flipping of the PSF before its distribution. $P_i^* h$ is constructed by placing the PSF h (as is, unflipped) at pixel position i of u -sized zero image and reshaping it into a column vector. We can write $P_i = P_i^* F$ where F is the flipping operator (square matrix with 1s on its anti-diagonal and zeros elsewhere).

To get better insight into the last expression, imagine that $\bar{\Gamma}$ is identity, then C^h becomes a convolution matrix created by summing (the non-circulant matrix) $\text{cov}(h)$ along its different diagonals depending on the spatial difference of i -th and j -th pixel (the summation must be done using circular wrap-around, so e.g. sum of the first sub-diagonal includes also the top right matrix element etc.). Another important special case is when $\text{cov}(h)$ is diagonal – then C^h is also diagonal, because whenever $i \neq j$ then the diagonal of $\text{cov}(h)$ is distributed to non-diagonal elements and vanishes, as a result the whole product has zeros on the diagonal. In this case the diagonal of C^h can be easily calculated as a circular *correlation* of the diagonal of $\text{cov}(h)$ and the diagonal of $\bar{\Gamma}$, for the expectation in (4.51) and (4.52) we get

$$\mathbb{E}_h [H^T \bar{\Gamma} H] = \bar{H}^T \bar{\Gamma} \bar{H} + \text{diag}(h_C \star \bar{\gamma}), \quad (4.55)$$

where again $h_C = \text{diag}(\text{cov}(h))$ and \star denotes 2D correlation (h_C is the sliding kernel). The vectors h_C and $\bar{\gamma}$ are treated the same way as h and u when in 2D convolution.

The last two simplified examples suggest a general approach of working with expressions involving the covariance matrices (which can be used to calculate also the similar terms in (4.34)). It turns out that it is useful to represent $\text{cov}(h)$ (and analogously $\text{cov}(u)$) as a generalized indexed-by-vector matrix which for each spatial difference of i -th and j -th pixel in the blur PSF, denoted $\mathbf{i} - \mathbf{j}$ (recall that this is a 2D tuple), contains all elements of $\text{cov}(h)$ corresponding to pixels with this spatial difference and in the same topological layout as these pixels are in the PSF of h . These “planes” of $\text{cov}(h)$ then correspond to the diagonals that we awkwardly described in the previous example, the plane corresponding to $i = j$ is precisely the main diagonal of $\text{cov}(h)$ reshaped as h -sized convolution kernel etc. All calculations involving terms of the kind $P_i \text{cov}(h) P_j^T$ for all i, j can be efficiently calculated by convolution-like operations (or simple summations) on the planes corresponding to $\mathbf{i} - \mathbf{j}$ and loop over the difference $\mathbf{i} - \mathbf{j}$. This becomes especially beneficial when $\text{cov}(h)$ or $\text{cov}(u)$ are approximated by band matrices (e.g. diagonal), then the outer loop over $\mathbf{i} - \mathbf{j}$ is short and bulk of the work is done efficiently by the inner convolutions. Since this is more or less an implementation detail and probably of little interest to most readers, we will leave this discussion in the form of this suggestion but will not pursue it any further by detailed formalization.

Deriving the expression for $q(\lambda)$ is analogous to $q(\gamma)$. From (4.6) we have

$$\log p(u|\lambda) = \frac{1}{2} \sum_i \log \lambda_i - \frac{1}{2} \sum_i \lambda_i (\nabla u)_i^2, \quad (4.56)$$

$$\log p(\lambda) = (a_\lambda - 1) \sum_i \log \lambda_i - b_\lambda \sum_i \lambda_i. \quad (4.57)$$

Substituting these to (4.24) for $q(\lambda)$ we get

$$\begin{aligned} \log q(\lambda) &= \mathbb{E} [\log p(u|\lambda)] + \mathbb{E} [\log p(\lambda)] \\ &= \frac{1}{2} \sum_i \log \lambda_i - \frac{1}{2} \sum_i \lambda_i \mathbb{E} [(\nabla u)_i^2] + (a_\lambda - 1) \sum_i \log \lambda_i - b_\lambda \sum_i \lambda_i \\ &= \left(a_\lambda - \frac{1}{2} \right) \sum_i \log \lambda_i - \sum_i \left(\frac{1}{2} \mathbb{E} [(\nabla u)_i^2] + b_\lambda \right) \lambda_i. \end{aligned} \quad (4.58)$$

We recognize $q(\lambda)$ as Gamma distribution with expected value

$$\bar{\lambda}_i = \frac{2a_\lambda + 1}{\mathbb{E}_u [(\nabla u)_i^2] + 2b_\lambda}. \quad (4.59)$$

The expectation in the denominator can be expanded as

$$\mathbb{E}_u [(\nabla u)_i^2] = (\nabla \bar{u})_i^2 + \nabla_i \text{cov}(u) \nabla_i^T, \quad (4.60)$$

where ∇_i is i -th row of the derivative operator. If $\text{cov}(u)$ is diagonal with u_C on the diagonal, the second term can be again expressed concisely using 2D convolution and for the expectation as a vector of all i 's we get

$$\mathbb{E}_u [(\nabla u)^2] = (\nabla \bar{u})^2 + d^2 * u_C, \quad (4.61)$$

where d is the convolution kernel by which the derivative is defined, e.g. $d = [1, -1]$ for simple horizontal forward differences etc., and all powers are elementwise.

Blur terms $q(h)$ and $q(\beta)$

Derivation of the blur-related terms mimics analogous derivation of the image terms. From (4.26) and (4.11) we have

$$\log p(g|u, h, \alpha, \gamma) = -\frac{\alpha}{2} (Uh - g)^T \Gamma (Uh - g), \quad (4.62)$$

$$\log p(h|\beta) = -\frac{1}{2} h^T B h, \quad (4.63)$$

where $B = \text{diag}(\beta)$ is the diagonal precision matrix of the blur prior. Substituting these to (4.24) for $q(h)$ we get

$$\begin{aligned} \log q(h) &= \mathbb{E} [\log p(g|u, h, \alpha, \gamma)] + \mathbb{E} [\log p(h|\beta)] \\ &= -\frac{1}{2} \mathbb{E} \left[\alpha (Uh - g)^T \Gamma (Uh - g) + h^T B h \right]. \end{aligned} \quad (4.64)$$

As in the image case, the right-hand side is quadratic with respect to h , which tells us that $q(h)$ is a Gaussian. By setting the first derivative equal to zero and calculating the second derivative we get the equations for mean and covariance, respectively, as

$$\left(\bar{\alpha} \mathbb{E}_u [U^T \bar{\Gamma} U] + \bar{B}\right) \bar{h} = \bar{\alpha} \bar{U}^T \bar{\Gamma} g, \quad (4.65)$$

$$\text{cov}(h) = \left(\bar{\alpha} \mathbb{E}_u [U^T \bar{\Gamma} U] + \bar{B}\right)^{-1}. \quad (4.66)$$

The expectation term can be expanded in the similar fashion as in (4.53) and (4.54). We get

$$\mathbb{E}_u [U^T \bar{\Gamma} U] = \bar{U}^T \bar{\Gamma} \bar{U} + C^u, \quad (4.67)$$

$$C_{ij}^u = \text{tr}(\text{cov}(U_{\cdot,i}, \bar{\Gamma} U_{\cdot,j})) = \text{tr}(\bar{\Gamma} P'_i \text{cov}(u) P'_j{}^T), \quad (4.68)$$

where P'_i is yet another set of permutation matrices. $P'_i u$ is constructed by placing the origin of the whole image u at i -th pixel (using circular wrapping and without flipping) and reshaping into a column vector; we have $U_{\cdot,i} = P'_i u$. The same discussion can be made as in the case of (4.54) regarding suitable representation of $\text{cov}(u)$ that facilitates calculations. In particular, if $\text{cov}(u)$ is diagonal, then C^u is also diagonal and we can write the expectation as

$$\mathbb{E}_u [U^T \bar{\Gamma} U] = \bar{U}^T \bar{\Gamma} \bar{U} + \text{diag}(u_C \star \bar{\gamma}), \quad (4.69)$$

where $u_C = \text{diag}(\text{cov}(u))$ and \star is again 2D circular correlation (u_C is the sliding window and $\bar{\gamma}$ is fixed; be careful that correlation is not commutative).

To derive $q(\beta)$ we have from (4.11)

$$\log p(h|\beta) = \frac{1}{2} \sum_i \log \beta_i - \frac{1}{2} \sum_i \beta_i h_i^2, \quad (4.70)$$

$$\log p(\beta) = (a_\beta - 1) \sum_i \log \beta_i - b_\beta \sum_i \beta_i. \quad (4.71)$$

Substituting these to (4.24) for $q(\beta)$ we get

$$\begin{aligned} \log q(\beta) &= \mathbb{E}[\log p(h|\beta)] + \mathbb{E}[\log p(\beta)] \\ &= \frac{1}{2} \sum_i \log \beta_i - \frac{1}{2} \sum_i \beta_i \mathbb{E}[h_i^2] + (a_\beta - 1) \sum_i \log \beta_i - b_\beta \sum_i \beta_i \\ &= \left(a_\beta - \frac{1}{2}\right) \sum_i \log \beta_i - \sum_i \left(\frac{1}{2} \mathbb{E}[h_i^2] + b_\beta\right) \beta_i. \end{aligned} \quad (4.72)$$

We recognize $q(\beta)$ as Gamma distribution with expected value

$$\bar{\beta}_i = \frac{2a_\beta + 1}{\mathbb{E}[h_i^2] + 2b_\beta} = \frac{2a_\beta + 1}{\bar{h}_i^2 + \text{cov}(h)_{ii} + 2b_\beta}. \quad (4.73)$$

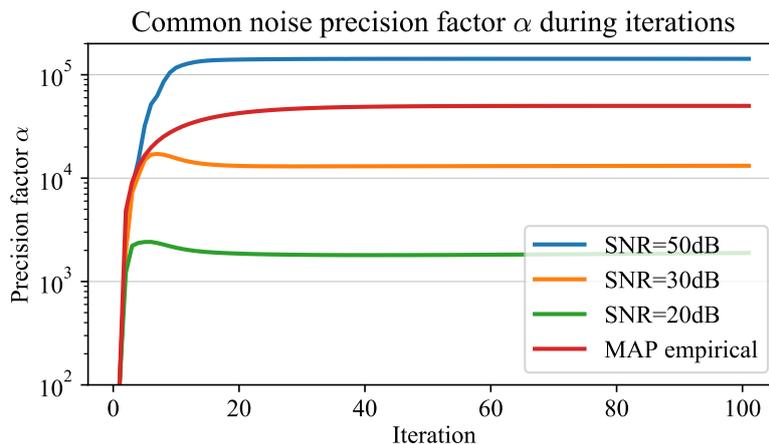


Figure 4.8: Evolution of the estimated noise precision factor $\bar{\alpha}$ during iterations. Different curves depict its typical development for different input image SNRs from 50dB to 20dB. For comparison, we included the hard-coded adjustment (3.29) of the data term weight during iterations used previously in Chapter 3.

4.5 Algorithm summary

Before proceeding with description of the overall blind deblurring algorithm, let us give a short commentary on the derived update equations and compare them to their counterparts if equivalent model were used in blind deblurring formulated as MAP estimation, instead of variational posterior approximation.

Pixel-wise precision γ can be compared to weights of individual pixels in the data-fidelity term in the classical MAP formulation, as we already discussed. From the update equation (4.32) we see that such weight of each pixel is inversely proportional (up to relaxation by $\bar{\nu}/\bar{\alpha}$) to the expected quadratic residual at that pixel. For strongly Gaussian observation noise (when $\nu \rightarrow \infty$) we have $\gamma \rightarrow 1$ and the weight of the likelihood term as a whole is given by α . For finite ν the expression for γ has the form of relaxed ℓ^0 reweighting as used in iteratively reweighted least squares methods to achieve greater robustness, but in variational Bayes it arises naturally due to the prior model. We will see in the experimental section how γ nicely identifies the violations of the blur model (Fig. 4.15). To evaluate the expected value of the quadratic residual we need the covariance matrices of both u and h , which is a typical “inconvenience” of equations arising from VB, since the covariances are expensive to calculate.

Common noise precision α has a very similar update equation (4.39), except that the estimated noise level is averaged over the whole image. Notice also that pixels suspected of local non-Gaussian error do not contribute to the overall $\bar{\alpha}$ estimate due to their low γ_i . In MAP formulation, α determines the weight of the whole data-fidelity term relative to the weight of the regularizers (priors). We have seen in the previous chapter based on MAP estimation that suitable adjustment of the ratio between the data-fidelity weight and regularizer weight during the optimization procedure is used as a means of avoiding the trivial no-blur solution (see eq. (3.29) and Fig. 3.29 and note that therein the data-term weight is denoted γ). The major difference is that in our MAP

approach such weight and its adjustment were hard-valued and specified by the user, whereas in VB it is a random variable with automatically inferred expectation. We did an experiment to see how $\bar{\alpha}$ evolves during optimization and whether it corresponds to actual noise level in the image. We ran our blind deblurring algorithm on synthetically blurred images with deferent levels of added Gaussian noise and measured $\bar{\alpha}$ during iterations; the results are in Fig. 4.8 and for comparison purposes we also plotted values corresponding to the manual adjustment in MAP according to (3.29). We can see that the evolution of $\bar{\alpha}$ during VB optimization very much resembles what was proposed in the MAP approach on purely empirical basis. During the early iterations the precision sharply increases from a very low initial value and then flattens at value corresponding to the true noise level. The indisputable advantage of the VB is that it gives us the optimal rate of change and automatic asymptotical value.

Degrees of freedom ν determine the degree of non-Gaussianness of the observation noise (see the discussion on Student’s t -distribution on page 50). In the MAP formulation it has no counterpart and arguably none is necessary. In fact, we include it in our VB model by choice for experimental purposes, since the number of degrees of freedom could have easily been hard-valued in the form of a hyperparameter as in the more conventional formulation (4.14). In the update equation for $\bar{\nu}$ (4.45) we see that the degree of non-Gaussianness is measured by how much the pixel-wise precisions γ differ from 1 (which is their expected value dictated by the hyperprior (4.13c)). If the observation noise is strictly Gaussian, then $\gamma \equiv 1$ and $\bar{\nu} \rightarrow \infty$. The more γ approaches zero (indicating poor model fidelity), the more $\bar{\nu}$ approaches zero as well.

Image u is determined by its mean and covariance. Without the expectation $\mathbb{E}_h[\cdot]$, the update equation for \bar{u} (4.51) would be fairly standard u update based on the current estimate of the blur h commonly encountered in image deblurring (compare e.g. with (3.23) in which the right-hand side would read $\rho_1 H^T g$ if it were not for the ADMM splittings). We know from (4.53) that the expectation term contains also the covariance of h , which is more accurate characterization of the current estimate of h than simple mean. Covariance of u has no counterpart in MAP and we have seen that in VB it is omnipresent in update equations of other unknowns and is arguably one of the primary reasons why VB is superior to MAP. Unfortunately, the covariance matrix is very large (for image with N pixels, covariance represents more than $N^2/2$ unknowns) and is therefore virtually always somehow approximated in practice.

Image gradient precision λ is updated according to (4.59), which again resembles reweighting that effectively achieves relaxed ℓ^0 regularization of $|\nabla u|$, a very sparsity-promoting prior. The VB addition is the covariance term, which again provides better characterization of the current image estimate. In this case the effect of the covariance term can be relatively easily visualized. In the final iteration of the optimization procedure we calculated $\bar{\lambda}$ using the correct update (4.59) and then using the same update equation but without the covariance term in (4.60), which is comparable to MAP estimation, and visualized the estimated $\bar{\lambda}$ in Fig. 4.9 (with suitable but equal scaling for visualization purposes). It is apparent that with the covariance term the precisions λ have higher variance, since the estimated image edges are more pronounced, which

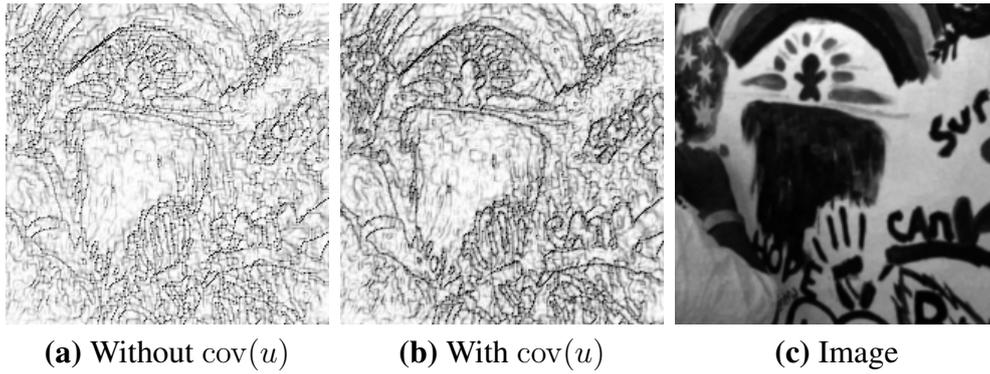


Figure 4.9: Visualization of the image prior precision λ and effect of the covariance of u on its estimation. **(a)** Estimation of $\bar{\lambda}$ by (4.59) using directly \bar{u} instead of properly evaluating the expectation. **(b)** Estimation with proper evaluation of the expectation in (4.59) using both \bar{u} and $\text{cov}(u)$.

means that the corresponding image prior is effectively more heavy-tailed when image covariance is correctly included.

Blur h is updated according to (4.65), which is again a very standard update equation, except for the expectation operator. The situation is analogous to the image update – blur is estimated based on the current image estimate and including image covariance provides more accurate description of the current image, which improves the blur estimation.

Blur precision β is approximately inversely proportional to the squared PSF pixel value, which again effectively results in a very sparse blur prior roughly on par with relaxed ℓ^0 regularization. If you recall, the blur prior does not contain any means of enforcing non-negativity of the blur values, but experiments show that this does not present any problem and seemingly no such constraint is needed. At this point we can comment on why. First, if the input image g is truly a result of convolution of non-negative image u and non-negative blur h , then the likelihood (data-fidelity term) alone has non-negative local minimum which attracts the optimization if the initial blur estimate is also non-negative. And second, the more any positive h_i approaches zero during the optimization, the more the corresponding precision β_i increases (theoretically up to $1/(2b_\beta)$) and slows down any further change of h_i . In the linear system (4.65), more and more weight is on the diagonal term \bar{B} in the left-hand side, which then outweighs the data term and traps h_i at zero.

Optimization algorithm

We know from previous discussion that general VB inference consists of repeatedly looping over all unknown variable groups and update each according to its just derived update equation using the current estimates of all other variables. All the update equations are relatively easy to calculate, except for the updates of $\text{cov}(u)$ and $\text{cov}(h)$ in eq. (4.52) and (4.66), which require inversion of large non-circulant matrices. To facilitate the calculations even for large images we resort to the following approximation.

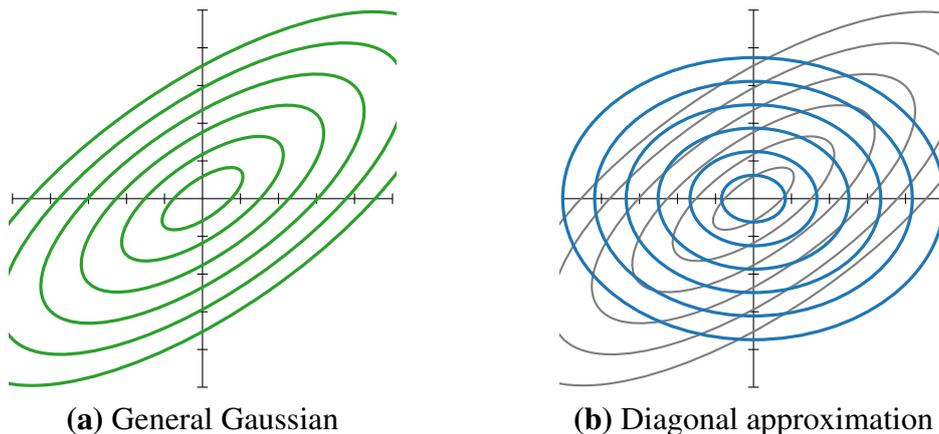


Figure 4.10: Diagonal approximation of Gaussian precision matrix. **(a)** Levelset contours of a general 2D Gaussian. **(b)** Contours of the Gaussian in (a) with the precision matrix approximated by its diagonal. Depicted contours are mutually corresponding between plots (enclosing equal probability mass).

The equations (4.52) and (4.66) explicitly define the *precision* matrices of the respective distributions. We approximate each of these precision matrices by a diagonal matrix by dropping all its non-diagonal elements and then easily calculate the covariance matrix by inverting the diagonal elements only. Diagonal approximation of a precision matrix of a general Gaussian unfortunately results in an axis-aligned Gaussian that is necessarily “tighter” than the original distribution, see Fig. 4.10 for illustration. Such distribution therefore presents a less conservative estimate of the posterior, as it is more concentrated around the mode and therefore closer to the MAP estimate of the mode only. We argue that this approximation still makes more sense and performs better than other possibilities that appeared in literature, such as approximating the covariance by a convolution matrix (which was used e.g. by Tzikas *et al.* [173] for $\text{cov}(h)$). All calculations involving the covariance matrices are then greatly simplified since the covariance matrices become diagonal; we have already covered this special case in the discussion of all update equations where the covariance matrices appear.

As in our MAP solution in the previous Chapter 3, we perform the blur estimation in a multiscale fashion by first estimating the blur from a downsampled input g and then proceeding with estimation on larger scale initialized by the estimate from the previous scale. This technique has been described in full detail in the previous chapter (see “Multiscale optimization” on page 38 and the outer loop of the blind section of Alg. 3.1). When moving from one scale to the next we upsample the blur PSF \bar{h} and the precisions γ and λ . Passing other variables (e.g. covariances or other precisions) between scales proved superfluous or detrimental. We assume that upper bound of the PSF size (which defines the number of scales) is provided by the user.

You may recall that in the previous chapter we focused on proper treatment of the convolution operation and avoided directly using circular convolution because it is unrealistic and performs poorly for real blurs. In this chapter we used nothing but circular convolutions for their efficiency but due to the pixel-wise data-fidelity precisions γ we can easily prevent any damage caused by the unrealistic periodic assumption. We fix $\gamma_i = 0$ for all pixels i close enough to the image boundary to be affected by the circular

wrap-around caused by convolution. This effectively removes any terms arising from the acquisition model from the update equations for pixels in the boundary region and using circular convolutions is safe. Also, the pixels with γ_i fixed are excluded from the summations in the update equations for $\bar{\alpha}$ (4.39) and $\bar{\nu}$ (4.45) and N is adjusted accordingly.

Algorithm 4.1 Summary of the deblurring algorithm presented in Chapter 4

input: blurred image g , blur size upper bound

initialize $\bar{u} := g$, $u_C \equiv 0$, $\bar{h} := \delta$, $h_C \equiv 0$, $\bar{\gamma} \equiv 1$ (except near boundary, where $\bar{\gamma}_i \equiv 0$ fixed), $\bar{\nu} \equiv 1$, $\bar{\lambda} \equiv 1$, $\bar{\beta} \equiv 0$

repeat

$\bar{u} :=$ solution of (4.51) using eq. (4.55)

$u_C := \text{diag} \left(\bar{\alpha} \bar{H}^T \bar{\Gamma} \bar{H} + \bar{\alpha} \text{diag}(h_C \star \bar{\gamma}) + \nabla^T \bar{\Lambda} \nabla \right)^{-1}$ \triangleright see eq. (4.52) and (4.55)

$\bar{h} :=$ solution of (4.65) using eq. (4.69)

$h_C := \text{diag} \left(\bar{\alpha} \bar{U}^T \bar{\Gamma} \bar{U} + \bar{\alpha} \text{diag}(u_C \star \bar{\gamma}) + \bar{B} \right)^{-1}$ \triangleright see eq. (4.66) and (4.69)

for all pixels i do

$\bar{\gamma}_i :=$ value of (4.32) using eq. (4.35)

$\bar{\lambda}_i :=$ value of (4.59) using eq. (4.61)

$\bar{\beta}_i :=$ value of (4.73) where $\text{cov}(h)_{ii} = (h_C)_i$

end for

$\bar{\alpha} :=$ value of (4.39) using eq. (4.35) \triangleright reuse (4.35) computed in $\bar{\gamma}_i$ update

$\bar{\nu} :=$ value of (4.45) using eq. (4.47) \triangleright see comment regarding $\bar{\nu}_{\text{old}}$ in (4.47)

until convergence

output: \bar{u} and \bar{h} as the final image and blur estimates

The blur estimation process for one scale is summarized in Alg. 4.1, all other scales are identical except that the initialization of certain variables is done by upscaling as described above. The listed algorithm already includes the described diagonal approximation of the precision matrices, as a result the diagonals of $\text{cov}(u)$ and $\text{cov}(h)$ are represented by vectors u_C and h_C , respectively. The only significantly time consuming steps are the linear systems for \bar{u} and \bar{h} update, which we solve iteratively using conjugate gradient method.

The only user-specified parameters are the hyperparameters a_* and b_* of the Gamma distributions used as hyperpriors, namely a_α , a_ν , a_λ , a_β and their b_* companions. We set all a_* to zero to achieve the most uninformative hyperpriors. Looking at the update equations (4.39) and (4.45) for $\bar{\alpha}$ and $\bar{\nu}$ we see that b_α and b_ν have negligible effect when added to large summations over all image pixels, we safely set them to zero. By contrast, b_λ and b_β are important because the updates (4.59) and (4.73) are calculated for each pixel and the b_* parameters provide necessary regularization. By search in the parameter space we determined that setting $b_\lambda = b_\beta = 10^{-6}$ is optimal (generally, any low value approximately in the range $(10^{-9}, 10^{-6})$ works equally well). In the experimental section we evaluated also the more conventional noise prior (4.14), where the number of degrees of freedom is fixed and user-specified. In this case we used $a_\gamma = 0$ and $b_\gamma = 10^{-4}$ which corresponds to fixing the degrees of freedom to zero.

Table 4.1: List of methods included in the experimental evaluation.

Abbreviation	Description
Ours- $\alpha\gamma$	Our method, ARD noise model (4.13) with common precision factor α and estimated degrees of freedom.
Ours- γ	Our method, conventional ARD noise model (4.14) with fixed degrees of freedom.
Tzikas09	Method of Tzikas <i>et al.</i> [173], our implementation.
Pan16	Method of Pan <i>et al.</i> [125], authors' implementation.
Xu10	Method of Xu and Jia [191], authors' implementation.
Zhong13	Method of Zhong <i>et al.</i> [211], authors' implementation.
Babacan12	Method of Babacan <i>et al.</i> [7], our implementation.

4.6 Experimental evaluation

We evaluated the presented method with the primary intention to test its ability to blindly identify regions violating the assumed convolution model in input images and exclude those from blur estimation without affecting its accuracy. For the quantitative evaluation we used synthetically blurred images with known ground-truth blur and compared the performance of the presented method with other blind deblurring methods targeting the same scenario or based on similar principle. The methods and their abbreviations are listed in Tab. 4.1. Our presented method is included in two flavors that differ in the observation error model. Ours- $\alpha\gamma$ is based on the noise model defined in (4.13) that contains the common factor α , which is the precision of the Gaussian noise component, and automatically estimated degrees of freedom ν as the amount of non-Gaussian noise. Ours- γ is based on the simpler noise model defined in (4.14), which is the conventional ARD model without any separation of Gaussian and non-Gaussian noise and with the degrees of freedom fixed to zero (maximally non-Gaussian). We included both of these models in the comparison to investigate if the more complex model is superior in practice and in which scenarios. The method Tzikas09 [173] is principally closest to ours, in particular the simpler Ours- γ , but differs in several aspects such as covariance matrix approximation and handling of convolution boundary effect. The method Pan16 [125] resembles objective-wise our method but is not based on variational approximation and instead uses a different approach that is similar to the method presented in the preceding Chapter 3. The method Zhong13 [211] specifically addresses strong Gaussian noise in the input by applying a series of directional filters to suppress the noise while retaining blur information in the orthogonal direction. The method Babacan12 [7] is based on variational Bayes and adaptively estimates the overall input noise level. The method Xu10 [191] already appeared in the experimental section of Chapter 3, it is a very well performing deblurring method, though without any extra handling of local model violations. All methods tagged with “our implementation” in Tab. 4.1 were run with the same multiscale pyramid scheme and parameter settings (when possible) for meaningful comparison.

Our first synthetic experiment is conceptually very similar to that in Chapter 3, where we simulated overexposure and intensity clipping. We used three sharp images (see Fig. 4.11) and blurred them by eight motion blur PSFs from the test dataset by



Figure 4.11: Original images and blur PSFs used in the synthetic experiments.



Figure 4.12: Example of input images in the first synthetic experiment. **(a)** Pixels with non-Gaussian error spatially distributed (7% erroneous pixels). **(b)** Erroneous pixels spatially concentrated (2.2% pixels). **(c)** Combination of non-Gaussian error (5%) and Gaussian noise (SNR=25dB); the ground-truth PSF is in the top-right corner. Results of blur estimation for these inputs are in Fig. 4.14 and the corresponding estimated precision γ is in Fig. 4.15.

Levin *et al.* [96]; see Sec. 3.3 and Fig. 3.4. This presents 24 input images in total; see example inputs in Fig. 4.12. After blurring, we intentionally set random parts of the blurred images to 0 or 1, which could simulate overexposed or underexposed regions but in general represents arbitrary non-Gaussian error (affected pixel i in image g was altered according to $g(i) = \text{round}(1 - g(i))$ to achieve an error of at least .5), and finally added mild Gaussian noise. An important difference from the similar experiment in the preceding chapter is that this time the method does not attempt to identify the altered regions by explicitly anticipating any particular kind of model violation. Because we already know from Chapter 3 that geometry of the spatial distribution of the altered pixels affects the result of blur estimation, we again tested two different extremal configurations. In one case we altered an increasing number of small 3×3 pixel squares randomly scattered in the image (Fig. 4.12a), simulating e.g. several

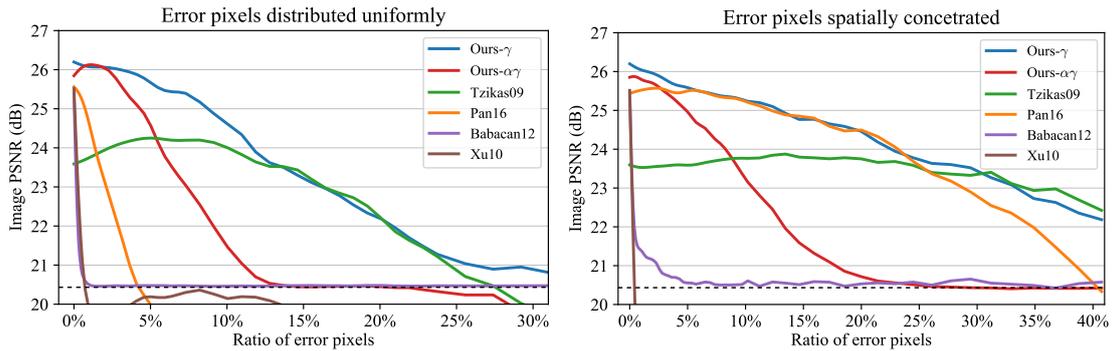


Figure 4.13: Restored image PSNR (higher value is better) as a function of degree of model violation. **Left:** The erroneous pixels are spread throughout the image, as in Fig. 4.12a. **Right:** The erroneous pixels are concentrated in one large region, as in Fig. 4.12b. Dashed horizontal line shows blurred image PSNR for reference.

small saturated regions in the scene, and in the other case we altered one large disk of pixels (Fig. 4.12b), simulating e.g. an object occluding an otherwise uniformly blurred scene. Since we are primarily interested in efficacy and accuracy of the blur estimation, we used the tested methods only for the blur estimation and then in each case used the same non-blind deblurring method (the method presented in Chapter 3) for the final image restoration to make the results of blur estimation directly comparable and uninfluenced by the non-blind image restoration procedure. The final deblurring was performed on the blurred image without the intentional non-Gaussian degradation (otherwise the input noise would cause most of the restoration error). We repeated this process for several realizations of the random placement of altered pixels and averaged the image estimation accuracy as a function of percentage of damaged pixels. The image restoration quality was measured in terms of PSNR, see Sec. 5.1 for review of standard image quality measures, such as PSNR or SNR. The results of the experiment are summarized in Fig. 4.13.

In the ideal case, if the model-violating regions are identified exactly, the restored image accuracy should remain constant as long as the rest of the image constitutes enough data for accurate blur estimation. After this point, the accuracy can be expected to drop rapidly. In this regard, Ours- γ exhibits the best performance – it retains high restoration PSNR even if a significant portion of the input is corrupted. A good characteristic is that the outlier rejection works in both of the tested geometric configurations of the altered pixels in the input. The method Ours- $\alpha\gamma$ also shows certain robustness to input error but unfortunately its accuracy drops quite quickly. Performance of Pan16 apparently depends on the geometry of the spatial distribution of the model-violating pixels. When the erroneous pixels are spatially concentrated (as in Fig. 4.12b), Pan16 performs almost on par with Ours- γ , but when the erroneous pixels are scattered in the whole area of the input (as in Fig. 4.12a), the performance of Pan16 drops below that of other methods. Tzikas09 shows similar robustness as Ours- γ but performs overall worse. Babacan12 and Xu10, with no explicit handling of local model violations other than being generally robust, fail as soon as the non-Gaussian noise becomes non-negligible. We did not test Zhong13 in this experiment but it can be seen in the results of the next experiment that this method cannot deal with non-Gaussian

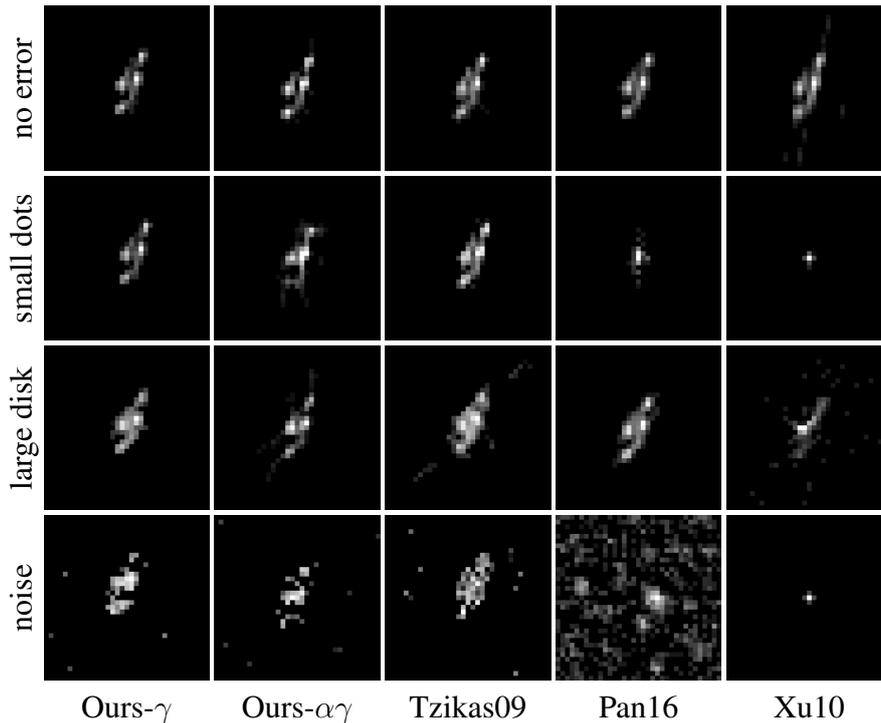


Figure 4.14: Results of blur estimation for the example inputs in Fig. 4.12, the ground-truth blur PSF is in Fig. 4.12c top-right. **Row 1:** No intentionally added input error. **Row 2:** Input image in Fig. 4.12a, erroneous pixels spatially distributed. **Row 3:** Input image in Fig. 4.12b, erroneous pixels spatially concentrated. **Row 4:** Input image in Fig. 4.12c, mixture of non-Gaussian and Gaussian noise.

noise at all.

A qualitative demonstration of the blur estimation is in Fig. 4.14, which depicts results for the particular inputs in Fig. 4.12. The top row corresponds to unaltered blurred input without any intentional perturbation, second through fourth row correspond to the three input images in Fig. 4.12, respectively. All methods perform well on the error-free input (top row), but as the noise increases the performance of all methods deteriorates to varying degree. Pan16 fails when the erroneous pixels are scattered. The method Xu10 does not actively seek outliers and as a result fails when the non-Gaussian error is non-negligible. The presented methods (Ours- γ and Ours- $\alpha\gamma$), and also Tzikas09 produce results comparable to the error-free case, unless the overall input error is too severe (bottom row).

This good performance is due to the automatic identification of the local model violations via γ as shown in Fig. 4.15 for the same input images. The dot pattern on the left image is recognized almost flawlessly. The large disk in the middle image has zeros mainly around its edges. This is because the homogeneous inside of the disk can be explained by any PSF, including the current estimate, and the method therefore considers this area as in accordance with the convolution model. The black outer frame are enforced zeros as a means to avoid convolution boundary artifacts.

The difference in performance between Ours- γ and Ours- $\alpha\gamma$ is worth attention. Results in Fig. 4.13 indicate that the apparent elegance of separating the Gaussian and non-Gaussian noise and automatic estimation of the ν parameter by Ours- $\alpha\gamma$ are

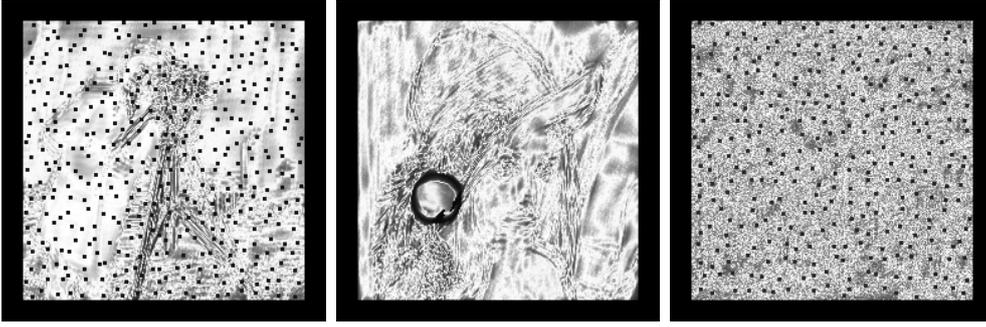


Figure 4.15: Results of γ estimation (precision of the non-Gaussian reconstruction error) for the example inputs in Fig. 4.12. Dark pixels indicate low model fidelity, light pixels indicate high model fidelity. The black frame are enforced zeros to avoid convolution boundary artifacts.

not worth the increased complexity, which requires that more unknowns need to be estimated from potentially unreliable data. In this experiment the observation noise is caused almost exclusively by non-Gaussian error and closer inspection reveals that the method Ours- $\alpha\gamma$ overestimates the number of degrees of freedom $\bar{\nu}$ and struggles to approach the limit $\bar{\nu} \rightarrow 0$, which would be appropriate for this kind of input. One possible explanation is that the Stirling’s approximation used in (4.43) to derive the optimal $q(\nu)$ factor becomes more and more inaccurate as ν approaches zero and as a result the derived update equations are not sufficiently accurate in this range. Another plausible explanation is that this behavior is a consequence of the non-uniqueness of blind deblurring. In the early stages of the optimization, when blur estimation is still very inaccurate, the generally large data-fidelity error results in low value of the global precision α instead of being distributed locally to pixel-wise precisions γ , which then remain uninformatively close to their prior-induced mean value 1 and as a result ν is continually overestimated. When the number of degrees of freedom is fixed to zero (as in Ours- γ), the local precisions γ are forced to adapt which for the kind of input used in this experiment may be sufficient to push the blur estimation off to a good start.

In the second synthetic experiment, we investigated robustness to Gaussian noise and the scenario of mixed Gaussian and non-Gaussian input noise. We used the same kind of input as in the previous experiment but with substantially increased amount of Gaussian noise (see example in Fig. 4.12c). Results of the blur estimation accuracy as a function of the Gaussian noise level for four different percentages of pixels affected by the non-Gaussian error (thresholding) are in Fig. 4.16 (plots from left to right and top to bottom correspond to increasing percentage of non-Gaussian error). Unlike in the previous experiment, the result is evaluated directly as the error of the estimated blur PSF to the ground truth. For low levels of non-Gaussian noise and reasonable input SNR, Ours- $\alpha\gamma$ method performs best and outperforms Ours- γ . This can be explained by the fixed degrees of freedom in Ours- γ to zero, which is too extreme for the input image containing only Gaussian noise or as little as 5% of saturated pixels, as in the top row of Fig. 4.16. By manually increasing the a_γ and b_γ parameters the performance of Ours- γ can be improved but only at the expense of performance on inputs with increased percentage of perturbed pixels, as in the bottom row of Fig. 4.16, where Ours- γ outperforms Ours- $\alpha\gamma$ because in this case the low number of degrees of free-

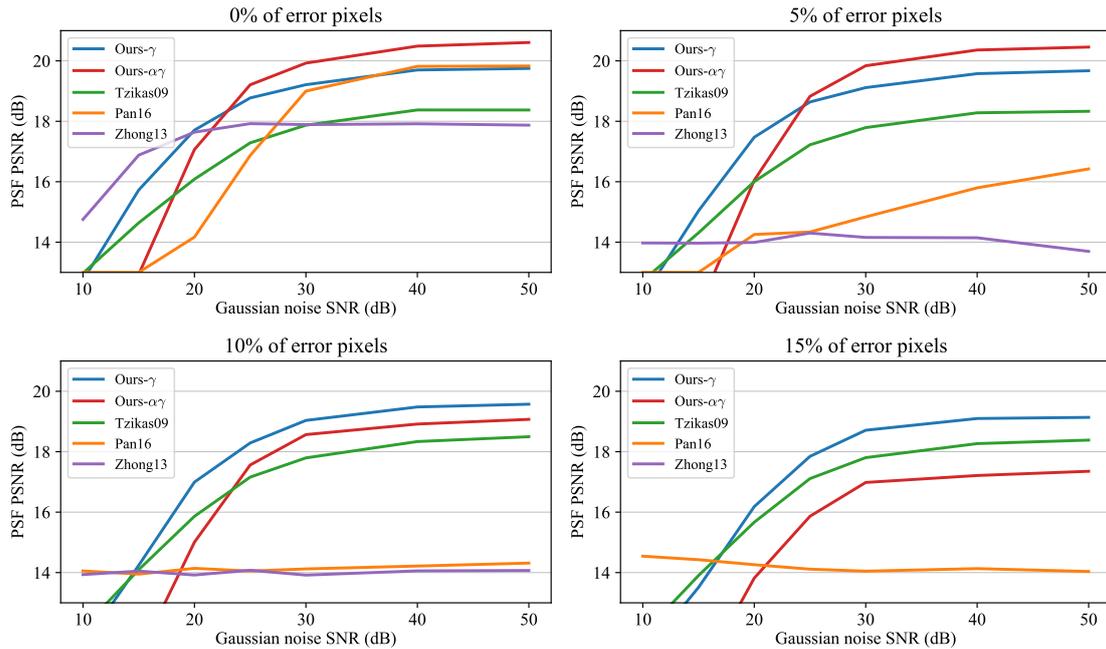


Figure 4.16: Estimated blur PSNR (high value is better) as a function of Gaussian noise in input images (SNR on x -axis) for four different levels of non-Gaussian noise (0% through 15% of pixels with random perturbation). Result is measured as error of the estimated blur PSF to the ground-truth PSF.

dom is appropriate and unfortunately \bar{v} again struggles to reach sufficiently low value. Tzikas09 is similarly stable as Ours- γ but with performance trailing by 1dB in average. Pan16 is slightly less robust to Gaussian noise even with zero non-Gaussian component and, as in the first experiment, when non-Gaussian error increases its performance drops quickly. Zhong13 shows stable (though somewhat subpar) performance for zero non-Gaussian noise and even outperforms other methods for exceedingly noisy images. However, it fails with any presence of non-Gaussian noise.

Besides synthetic experiments, we also tested the presented method on a number of images captured in real world conditions with intentional blur caused either by camera motion or incorrect focus. This presents a different kind of challenge than the previous two experiments, because in the synthetic tests the convolution model was violated either completely or not at all (except for the Gaussian noise). If the outliers are identified correctly, they can be excluded from the blur estimation without much affecting its accuracy. In real imaging conditions the convolution model approximates reality with varying degree of fidelity so the distinction between inliers and outliers is less clear and the blind deblurring method has a harder job of determining the relevance of each pixel.

We demonstrate the results of blind blur and image estimation in Fig. 4.17 and 4.18 for camera motion and out-of-focus blur, respectively. We ran all blind methods on each image to estimate the blur PSFs, but to be able to account for noise, saturations, or other potential model violations without compromising the image restoration result, we always used our method as the non-blind deblurring method for the final image estimation. Explicitly, given the blur h estimated by each of the tested methods

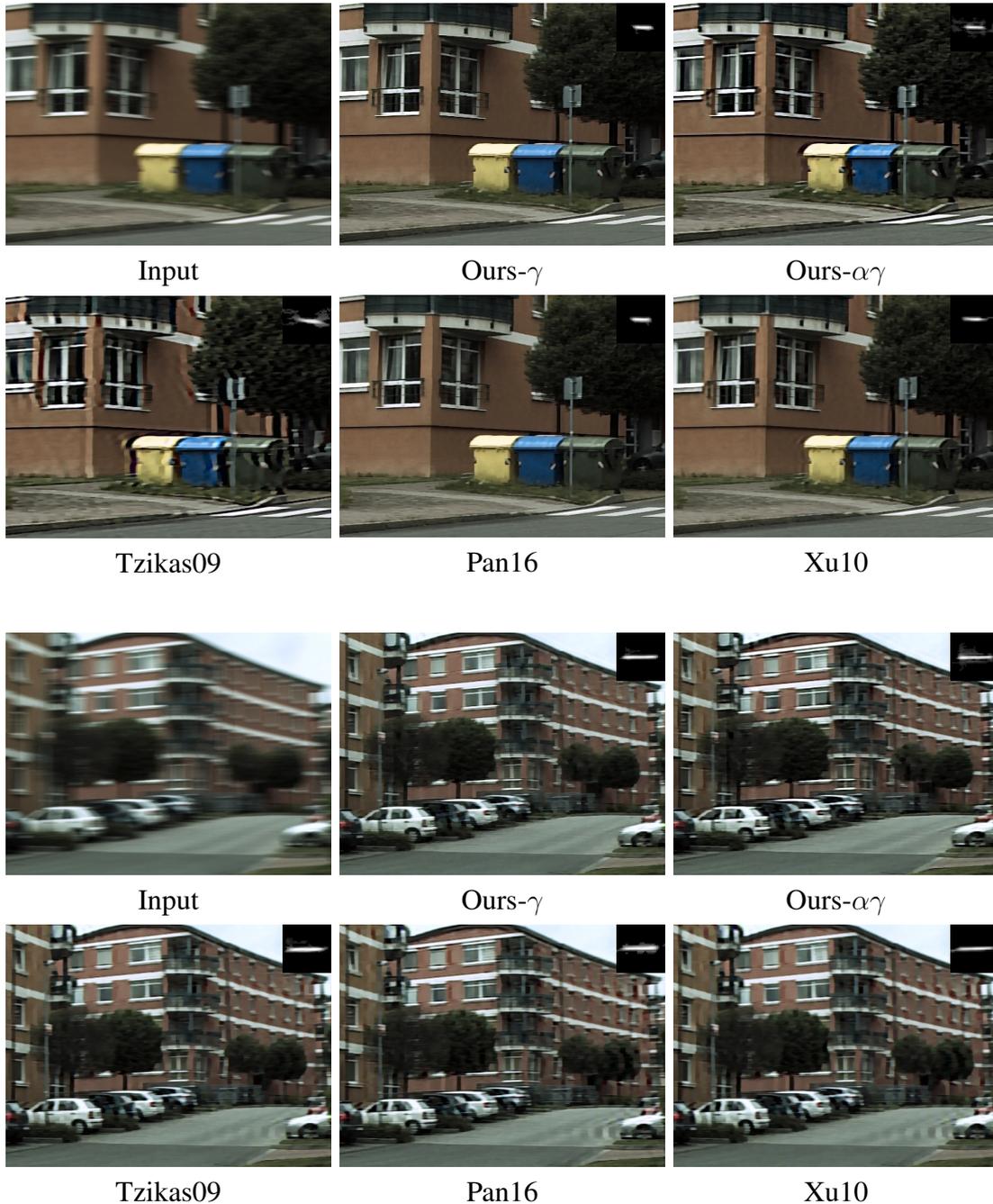


Figure 4.17: Deblurring of images with real motion blur. Estimated blur PSFs are superimposed in the top-right corner.

we ran Ours- $\alpha\gamma$ (at the final pyramid scale) with all blur-related variables (h , $\text{cov}(h)$, and β) fixed to the provided input while other variables such as α , γ , and all image-related variables were inferred from the data to account for noise, saturations, or other model violations. For this experiment we present only qualitative comparison between the presented method and several others. We can see that especially Ours- γ method performs well in both motion blur and out-of-focus scenario – the ringing is less prominent while the level of detail and the quality of the PSF estimation is better than in the results of other methods.



Figure 4.18: Deblurring of images with real out-of-focus blur. Estimated blur PSFs are superimposed in the top-right corner.

4.7 Conclusion

We have presented a blind image deblurring algorithm based on fully Bayesian formulation with automatic relevance determination priors on all the primary unknowns – image, blur, and observation error. The solution is based on the variational approximation of the posterior distribution in a factorizable form, which is theoretically a better estimator of the unknown blur than direct maximization of the joint posterior. We took great care to document explicitly and in full detail all the necessary steps required to derive the optimal posterior approximation and provided numerous comments and implementation suggestions. For reader’s convenience we also included a brief general introduction to variational Bayes and automatic relevance determination priors.

We have provided theoretical discussion and empirical evidence in the form of targeted experiments supporting that ARD is a suitable choice of prior for image, blur, and observation error and has several advantages compared to other priors commonly used in image deblurring. We showed that ARD as image prior correctly favors sharp images over blurred ones. We showed that ARD better approximates the true distribution of the observation noise than commonly used Gaussian. Most importantly, we showed that when ARD is used in the likelihood, the deblurring algorithm is able to automatically detect outliers of arbitrary kind and remove them from the estimation, relieving us of the necessity to explicitly consider all different kinds of model violations. Experiments demonstrated that the presented method compares favorably to other blind deblurring methods, especially in the presence of non-Gaussian input error.

In terms of computational complexity the resulting iterative algorithm is equivalent to common MAP deblurring algorithms that use conjugate gradient method (CG) to solve the resulting linear systems but this applies only when the suggested diagonal approximation of the precision matrices of the image and blur distributions is used. Otherwise, the computational burden of the VB approach increases substantially. In contrast, the MAP algorithm presented in Chapter 3 uses more efficient numerical scheme and the linear system are solved non-iteratively using Fourier transform. One CG iteration of our VB method is approximately equivalent to the full solution of the corresponding linear system in the MAP method from Chapter 3 (in terms of number of required Fourier transforms). We usually use approximately 20-30 CG iterations in each outer iteration of the VB method, the presented method is therefore an order of magnitude slower than the MAP method from Chapter 3.

We presented the method in two variants. The more complex one contains common precision factor corresponding to the Gaussian input noise and the parameter degrees of freedom, which determines heavy-tailedness of the noise distribution, is automatically estimated from the data. The simpler variant estimates precisions of the noise only pixel-wise and degrees of freedom are fixed as user-defined parameter. Synthetic experiments show that the more complex model is beneficial only in the scenario of mixed Gaussian and mild non-Gaussian input noise. Otherwise the simpler model is superior, which is true also for real blurred images. Separating the precision of the observation noise into two factors is apparently superfluous and unnecessarily results in more parameters that need to be estimated in an already under-determined problem, although we could use it to nicely justify the empirically optimal data-term weight adjustment in MAP based blind deblurring methods. Automatically determining the number of degrees of freedom is principally a good idea, because prescribing the correct value manually may be problematic, but requires better approximation of gamma function near zero, where the Stirling's formula used in the update equation for the degrees of freedom is inaccurate, and possibly better initialization of the automatic estimation. Otherwise the algorithm can get stuck with low value of the global noise precision and poor blur estimate instead of distributing the model infidelity to local pixel-wise precisions, which facilitates blur estimation from the good regions of the input.

Chapter 5

Blur estimation accuracy assessment

During presentation of two different approaches to blind image deblurring in the preceding two chapters we have seen that it is common to model the image degradation using convolution with an unknown blur kernel and that estimation of this blur is the first step and prerequisite of subsequent restoration of the input image itself. As a final topic related to general image deblurring we will talk about how accuracy of blur estimation can be meaningfully assessed by similarity of the estimated blur and the known ground truth using a measure which reflects how inaccuracy in the blur estimation affects the subsequent image restoration, so that performance of different blur estimation methods is absolutely comparable.

Recently, the deep-rooted paradigm of separating the restoration process into blur estimation and non-blind deblurring has been complemented by efforts to estimate the sharp image directly, without explicitly considering the blur process. This does not mean that explicit blur modeling is wrong and replaced for good, in many image acquisition processes convolution or spatially-variant convolution blur model is still the optimal approach and having a convenient accuracy measure of different blur candidates is valuable. Similar situation arises when inherent blur of some imaging device can be measured exactly but must be approximated by a simpler blur kernel (e.g. for computational reasons), then again different candidates must be compared.

In the rest of the chapter, we will first recall and illustrate the most common metrics of image similarity frequently appearing in literature. After that, we will explain our motivation for proposing a new way of measuring blur similarity and derive the proposed measure, which will be experimentally validated in the next section. As always, we conclude the chapter with a discussion.

5.1 Common image similarity metrics

Surely the most common measure of similarity between the ground-truth image u and its estimate \hat{u} is the mean squared error (MSE) defined as

$$\text{MSE}(u, \hat{u}) = \frac{1}{M} \sum_i (u_i - \hat{u}_i)^2, \quad (5.1)$$

where M is the total number of pixels in the image. It has been argued (e.g. [184]) that MSE does not faithfully correspond to human perception of image similarity, but

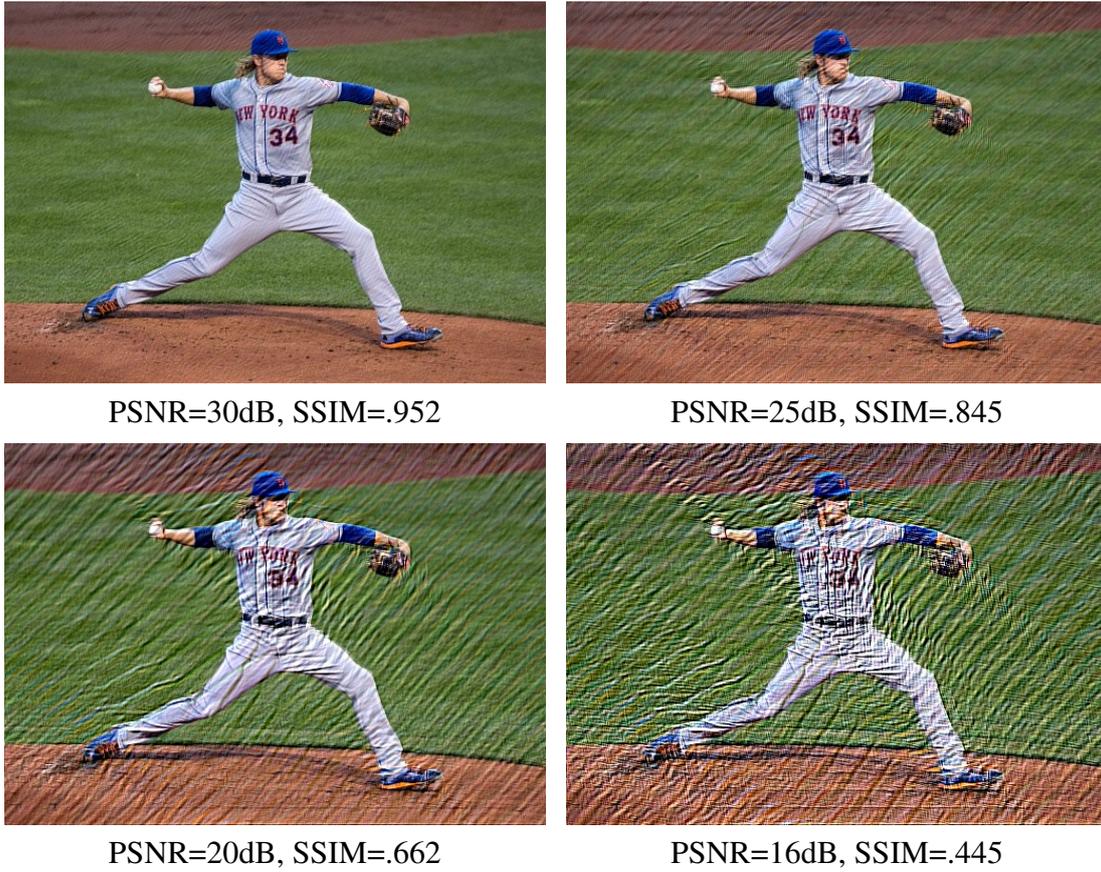


Figure 5.1: Illustration of the PSNR and SSIM measure of quality of image deblurring; the true sharp image is in Fig. 1.2a. The visible artifacts are caused by intentional inaccuracy of the blur. Input blurred image is noiseless version of the image in Fig. 5.2.

due to its numerical convenience it is widely being used nonetheless. Several metrics derived directly from MSE exist, such as root mean squared error (RMSE), which is simply the square root of MSE, or the peak signal-to-noise ratio (PSNR) in logarithmic scale defined as

$$\text{PSNR}(u, \hat{u}) = 10 \log_{10} \left(\frac{M}{\sum_i (u_i - \hat{u}_i)^2} \right), \quad (5.2)$$

assuming the $[0, 1]$ image intensity range. Since PSNR is essentially only an inverse of MSE, it inherits all its properties. Specifically in the context of image restoration the quality of the image estimate is sometimes reported as an improvement to the input g in terms of PSNR. Such metric is called ISNR and is defined simply as

$$\text{ISNR}(u, \hat{u}) = \text{PSNR}(u, \hat{u}) - \text{PSNR}(u, g) = 10 \log_{10} \left(\frac{\sum_i (u_i - g_i)^2}{\sum_i (u_i - \hat{u}_i)^2} \right), \quad (5.3)$$

where g is the (blurred or noisy) input image to the restoration process. For comparison purposes, all the discussed error measures are equivalent.

An active field of research exists to improve how image similarity measurement correlates with human perception of image similarity and hundreds of solutions have been proposed. One of the most widely known and adopted measures is the structural similarity index (SSIM) by Wang *et al.* [184], which outputs values between 0 and 1,

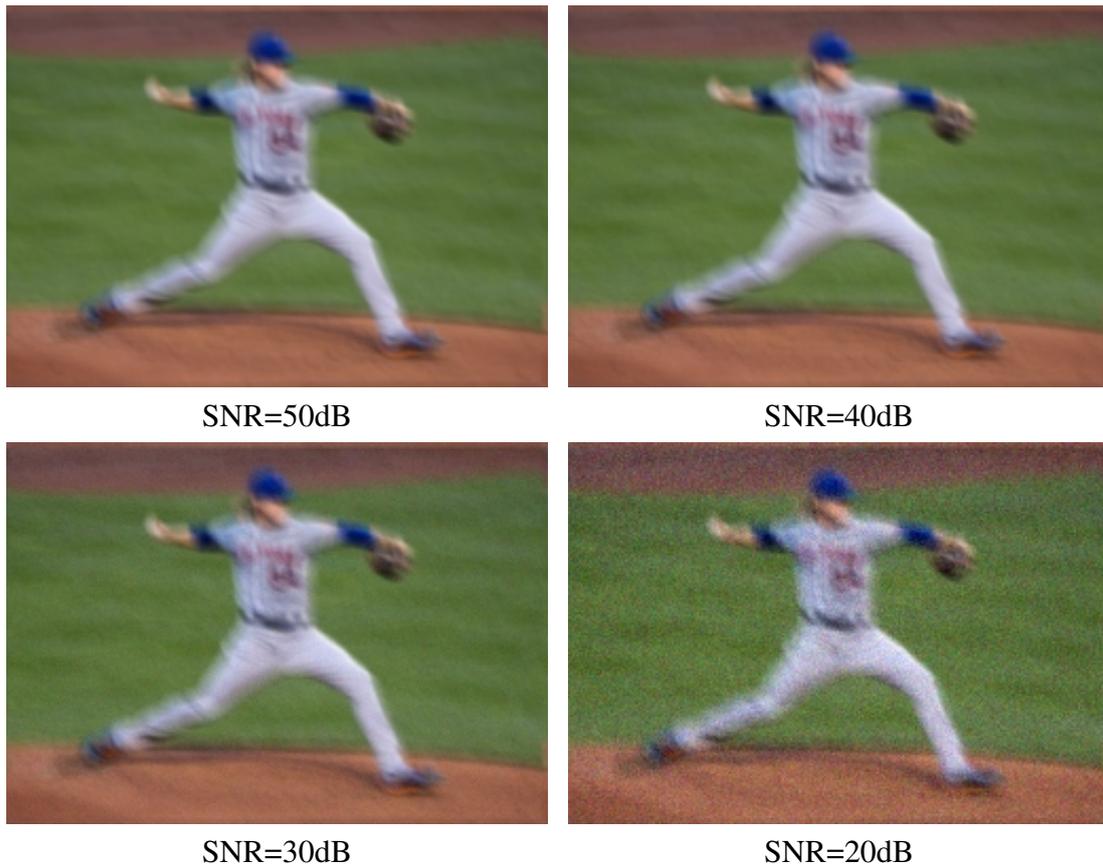


Figure 5.2: Illustration of the SNR measure of noise amount in the input to image deblurring. Noise with SNR above 30dB is hardly visible.

the latter attainable only for identical images (i.e. higher values indicate higher perceived similarity). In our limited experience gained during the work on image deblurring we observed that SSIM actually correlates with PSNR much more than originally claimed by its authors. Optimizing e.g. parameter settings of a deblurring method to achieve maximum SSIM or PSNR yields the same results; similar findings were reported by other authors [66].

Figure 5.1 illustrates the behavior of PSNR and SSIM when used as a measure of image deblurring success. A blurred noiseless input was restored using increasingly inaccurate blur and the PSNR decreases correspondingly. All the visible deblurring artifacts are due to inaccuracy in the blur, not noise. The true sharp image is in Fig. 1.2a on page 5. It is apparent that due to the inverse relation of PSNR to MSE it is ridiculously sensitive in high values (difference between 30dB and 35dB result is hardly visible) but rather insensitive in low values (visual difference between 15dB and 20dB is huge). SSIM is arguably more linear.

Often it is necessary to specify the amount of error (most typically noise) of the input to the restoration process rather than its output. If the ideal blur model dictates that $g = h * u$ but due to noise we instead observe $\hat{g} = h * u + n$ where $n \sim \mathcal{N}(0, \sigma^2)$ is zero-mean Gaussian noise, the noise content of \hat{g} is most commonly expressed using

the signal-to-noise ratio (SNR) defined in the context of image processing as

$$\text{SNR}(\hat{g}) = 10 \log_{10} \left(\frac{\sum_i g_i^2}{M\sigma^2} \right). \quad (5.4)$$

Some authors prefer using simply the value of σ as the specification of noise amount, so we can frequently encounter expressions such as “the input noise is $\sigma = 5/255$,” where the implicit assumption is that image intensity range is $[0, 1]$. Such specification is usually sufficient for the intended purpose but is not strictly equivalent to using SNR because SNR depends also on the image content (nearly white image has higher SNR than nearly black image with the same noise σ).

Figure 5.2 provides visual comparison of different input SNR values of the blurred input image. Noise is hardly visible until SNR reaches about 30dB, but it has noticeable detrimental effect on the deblurring result for SNR values as high as 50dB (which is roughly the limit of current high-end consumer cameras). This illustration concludes our little detour to the most commonly used image quality metrics and we now return to the rationale for and derivation of the proposed measure of blur estimation accuracy.

5.2 Motivation and goals

Image blur is commonly modeled as a convolution of the unknown sharp image u with an unknown blur h ,

$$g = h * u + n. \quad (3.1)$$

Most blind image deblurring methods can be separated into two stages: blur estimation, in which the blur h is computed, and non-blind image restoration, in which the sharp image u is estimated using the blur estimate from the previous step. Non-blind deblurring is generally better posed problem and success of the blind image restoration therefore largely depends on accuracy of the blur estimation.

As with any problem in applied science, it is necessary to quantitatively compare results of different methods or different versions of the same method for a given task. Arguably the most common measure of image deblurring success is the mean squared error (MSE) of the restored image with respect to the ground truth. Some other image similarity measures are derived from MSE and therefore equivalent, such as PSNR that we use in the previous and several other chapters.

The drawback of this approach is that it does not provide enough insight into how successful the key blur estimation alone actually was, because we evaluate the overall result only after the image restoration. Furthermore, such result is highly dependent on the used non-blind deblurring method. Some methods may be more sensitive to blur error than others, some explicitly deal with phenomena like the convolution boundary problem or blur model violation, discussed at length in previous chapters, while other do not and restoration error caused by these effects may be comparable with or exceed the error caused by inaccuracy of the blur. All non-blind methods require setting usually several parameters. As a result, we can get completely different orderings of the compared blur estimates just by using different non-blind methods or one method with different parameters, meaning that such comparison is not transferable between different author teams, where test conditions differ.

This suggests that preferable approach is evaluating directly the estimated blur, rather than the final restored image. One possible way is to measure the MSE, or equivalently cross correlation or PSNR, of the estimated blur point spread function (PSF) with respect to the known ground-truth PSF, as we used e.g. in Chapter 3. The problem with PSF-MSE is that it measures the similarity of the blur PSFs, it does not measure the similarity of the corresponding restored images and the two may not correlate. Two blurs with the same PSF-MSE to the common ground truth can produce different results (in terms of image MSE) when used for non-blind deblurring (as is demonstrated in the experimental section). While blur estimation is a necessary intermediate step, the ultimate goal is the sharp restored image. Therefore, blur estimation error should be measured as it would propagate to the image restoration.

To our knowledge, there are no well-established standard methods for evaluation of blur estimation without explicit non-blind deblurring. In Levin *et al.* [96] authors used the ratio between MSE of the image restored with the estimated kernel and MSE of the image restored with the ground-truth kernel as the blur error measure. This approach was adopted in many later papers but as a measure of accuracy of a blur estimate to the ground truth it is equivalent to using directly the MSE of the restored image and therefore inherits all the disadvantages discussed above.

We propose a method which can accurately calculate the MSE introduced by the blur inaccuracy to the image restoration just by one formula evaluated directly on the estimated blur, without the need to perform the actual image restoration. Such measure can be used for absolute comparison of several blur estimates with common ground truth and due to its simplicity it does not require careful specification of the test conditions.

5.3 Blur accuracy measure

We propose to measure the error ρ of a blur estimate \hat{h} with respect to the true blur h as

$$\rho[h, \hat{h}] = \mathbb{E} [\text{MSE}(u_h, u_{\hat{h}})], \quad (5.5)$$

where u_h and $u_{\hat{h}}$ are sharp images obtained by using the correct blur h and its estimate \hat{h} , respectively, and \mathbb{E} is expectation over images and noise, which are treated as random variables. The proposed metric is defined as the expected error of the image restoration caused by inaccuracy of the blur estimation. For correctly estimated blur the error is zero and increases with deviation of the estimate from the ground truth but the increase reflects how the blur error affects the image restoration. In the following text we provide a formula for evaluating the expression in (5.5) without having to actually compute u_h or $u_{\hat{h}}$, which of course we want to avoid.

As an analogue to the image improvement measure ISNR (5.3) we can define PSF-ISNR as a measure of improvement due to blur identification as

$$\text{PSF-ISNR}[h, \hat{h}] = 10 \log_{10} \left(\frac{\rho[h, \delta]}{\rho[h, \hat{h}]} \right), \quad (5.6)$$

where δ is the Dirac function (convolution identity). We now proceed with deriving the expression for (5.5).

Let us carry out the analysis in the Fourier domain, which is equivalent since ρ is quadratic and ℓ^2 norm is preserved by Fourier transform. Contrary to previous chapters, we use uppercase letters to denote the Fourier transform of the particular variable, e.g. $H = \text{FT}[h]$.

The acquisition model (3.1) expressed in Fourier domain is

$$G = HU + N, \quad (5.7)$$

where HU is elementwise multiplication due to the fact that convolution operator is diagonalized by the Fourier transform. If the correct blur H is known, the optimal (in the sense of statistically mean squared error) estimate U_H of U obtainable by linear space-invariant filtering of G is calculated using the Wiener filter W

$$U_H = WG, \quad W = \frac{H^*}{|H|^2 + R}, \quad R = \frac{S_N}{S_U}, \quad (5.8)$$

where

$$S_U = \mathbb{E}[U^*U] \quad \text{and} \quad S_N = \mathbb{E}[N^*N] \quad (5.9)$$

is the power spectrum of the image and noise, respectively, and R is called noise-to-signal ratio. We denote by A^* the complex conjugate of A (conjugate transpose of the corresponding operator or vector). If the image is restored using the estimated blur \hat{H} we get

$$U_{\hat{H}} = \hat{W}G = \frac{\hat{H}^*}{|\hat{H}|^2 + R}G. \quad (5.10)$$

The measure ρ in eq. (5.5) is defined as the expected squared difference of the restored images U_H and $U_{\hat{H}}$,

$$\rho[h, \hat{h}] = \mathbb{E} \left[\frac{1}{M^2} \sum_{\omega} |U_H - U_{\hat{H}}|^2 \right] = \frac{1}{M^2} \sum_{\omega} \mathbb{E} [|U_H - U_{\hat{H}}|^2], \quad (5.11)$$

where ω is the frequency index of U etc., which we omit for brevity. Substituting from (5.7), (5.8), and (5.10) we get for the difference of the restored images

$$\begin{aligned} U_H - U_{\hat{H}} &= (W - \hat{W})G = \left(\frac{H^*}{|H|^2 + R} - \frac{\hat{H}^*}{|\hat{H}|^2 + R} \right) (HU + N) \\ &= \frac{\left(H^*(|\hat{H}|^2 + R) - \hat{H}^*(|H|^2 + R) \right) (HU + N)}{(|H|^2 + R)(|\hat{H}|^2 + R)}. \end{aligned} \quad (5.12)$$

The expectation in eq. (5.11) acts non-trivially only on the last term in the numerator of (5.12), $HU + N$. In the squared magnitude we have

$$\begin{aligned} \mathbb{E} [|HU + N|^2] &= |H|^2 \mathbb{E} [|U|^2] + H \mathbb{E} [UN^*] + H^* \mathbb{E} [U^*N] + \mathbb{E} [|N|^2] \\ &= |H|^2 S_U + S_N. \end{aligned} \quad (5.13)$$

The second and third term vanish because U and N are independent and n has zero mean (then so does $N(\omega)$ for each ω).

By substituting (5.12) and (5.13) into (5.11) and using the relation $R = S_N/S_U$ in the numerator we finally get

$$\begin{aligned}\rho[h, \hat{h}] &= \frac{1}{M^2} \sum_{\omega} \frac{\left| H^*(|\hat{H}|^2 + R) - \hat{H}^*(|H|^2 + R) \right|^2 (|H|^2 S_U + S_N)}{(|H|^2 + R)^2 (|\hat{H}|^2 + R)^2} \\ &= \frac{1}{M^2} \sum_{\omega} \frac{S_U \left| H^*(|\hat{H}|^2 + R) - \hat{H}^*(|H|^2 + R) \right|^2}{(|H|^2 + R)(|\hat{H}|^2 + R)^2}.\end{aligned}\tag{5.14}$$

Remarks

Equation (5.14) is the final proposed blur error measure. It is a single formula which can be readily calculated as it does not require any iterative deblurring algorithm. Besides the evaluated blur and corresponding ground truth it depends also on the sharp image power spectrum S_U and the input noise level R , which is the only free parameter.

The presence of S_U is understandable because different images respond differently to change in the (de)blurring PSF. For example, an image with horizontal translation symmetry will be unaffected by horizontal blur, therefore if h and \hat{h} for such image differ only by a horizontal perturbation b (e.g. $\hat{h} = b * h$), then $\rho[h, \hat{h}] = 0$, which corresponds to the fact that the reconstructed images are identical. For calculation, we use the autocorrelation of the true image u as S_U , which presents no practical problem because for quantitative testing both true h and true u must be known. For results independent of particular image one can use the average formula $S_N(\omega) \propto |\omega|^{-2}$ approximately valid for natural images [16].

The dependence on $R = S_N/S_U$ also warrants discussion. Assuming that the input image u was corrupted with i.i.d. Gaussian noise n , $n \sim \mathcal{N}(0, \sigma^2)$, then $S_N \equiv M\sigma^2$. For a fixed test image u , R is then completely determined by a single parameter σ as $R = M\sigma^2/S_U$. In practice, the value of σ can be chosen almost arbitrarily within reasonable limits (e.g. $\sigma = 1/255$ for image with $[0, 1]$ intensity range to simulate approx. SNR=40dB), since relative comparison of different blur candidates with common ground truth does not depend on σ . As a simplification with minor loss in fidelity, one can even use $R \equiv \text{const}$. In Wiener filtering the variable R serves not only as an estimate of the input noise, but also as a general regularizer compensating the error in the blur estimation. If R is set too low for near-noiseless input image, even small errors in the estimated blur result in visible artifacts in the restored image, while with R set higher, the inverse filter is closer to identity and therefore less dependent on the blur estimate; the image restoration will be more forgiving to potential inaccuracy of the blur estimation. The same situation occurs with the error measure ρ . If R is set too low, any deviation from the ground truth will result in high error values while high R will make the error measure less sensitive. Consequently, any published test results will be fully reproducible only if they include the value of R used for evaluation.

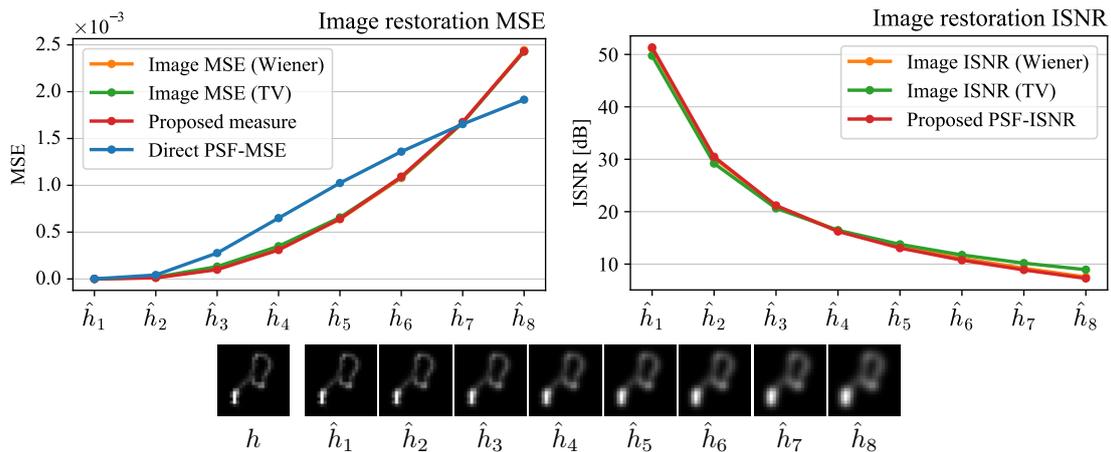


Figure 5.3: Value of the proposed blur accuracy measure compared with true mean squared error or ISNR of the restored images. Evaluated blur PSFs \hat{h}_i (differing by their support) are displayed in the bottom, the correct PSF h is on the left. Each blur candidate was used to restore the image (blurred by h) and MSE of the result image is compared with the value of the proposed measure ρ calculated with $\sigma = 2^{-6}$. Two non-blind methods were used for the image restoration, Wiener filtering (orange) and total variation method (green). **Left:** Mean squared error of the restored images (orange and green), the value of the proposed measure (red), and direct MSE between the blur PSFs (blue). **Right:** The PSF-ISNR measure (5.6) corresponding to the same experiment.

5.4 Experimental validation

In this section we present experimental verification that the proposed blur accuracy measure ρ corresponds to the true mean squared error calculated after image restoration and provide comparison with direct PSF-MSE of the blur PSFs. In these synthetic experiments we investigated three kinds of inaccuracy of blur estimation: incorrect support estimation (PSFs are too “thick”), incorrect length of motion blur, and incorrect diameter of out-of-focus blur; corresponding results are in Fig. 5.3 and Fig. 5.4, respectively. In each case we constructed several blur PSFs \hat{h}_i (displayed below each corresponding plot in Figs. 5.3 and 5.4) with various degrees of the aforementioned error with respect to the true blur h and for each of those we calculated

- our proposed error measure $\rho[h, \hat{h}]$ (red),
- true $\text{MSE}[u_h, u_{\hat{h}}]$ of the images restored using Wiener filtering (orange),
- true $\text{MSE}[u_h, u_{\hat{h}}]$ of the images restored using the total variation method presented in Chapter 3, specifically the problem (3.12) for $p = 1$ (green), and
- direct $\text{PSF-MSE}[h, \hat{h}]$ of the blur PSFs (blue).

All experiments were computed with the value $\sigma = 2^{-6}$ and using the well-known cameraman image.

It is well apparent that the proposed accuracy measure ρ very accurately captures the true image error introduced into the restoration by inaccuracy of the blur estimation. Although our derivation is based on Wiener filter, we can see that TV-based deblurring

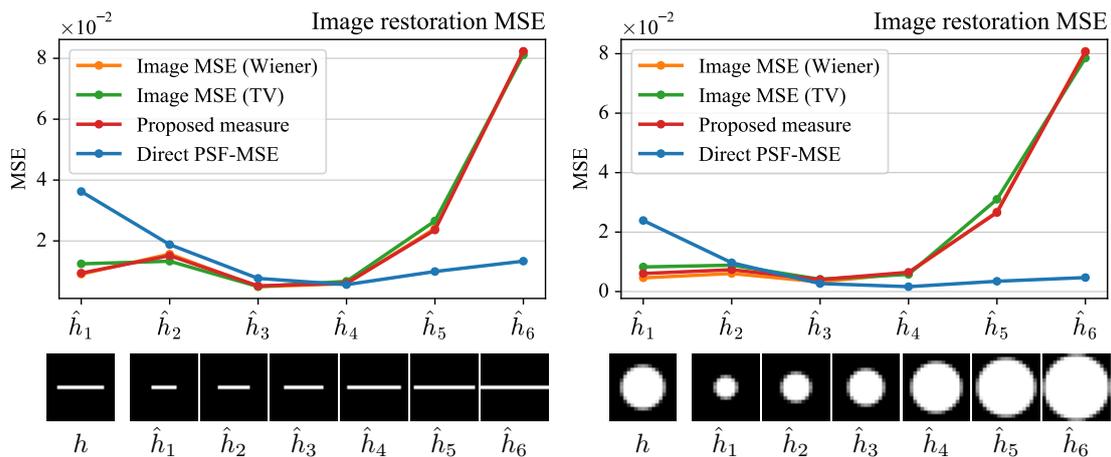


Figure 5.4: Value of the proposed blur accuracy measure compared with true mean squared error of the restored images; same experimental setup as in Fig. 5.3. **Left:** Incorrect motion blur length. **Right:** Incorrect out-of-focus blur diameter.

behaves very similarly when it comes to image mean squared error caused by blur inaccuracy. The proposed measure can therefore be used to gauge image restoration error even for more sophisticated non-blind deblurring methods. Direct PSF-MSE calculated between the estimated blur PSF and the ground truth, on the other hand, provides poor insight into how much the particular blur estimate affects the image restoration. This can be seen especially in Fig. 5.4 (blue line).

Results of the same experiment with ρ evaluated using different value of the noise parameter σ are available in Kotera *et al.* [87]. They show that while the absolute values of the accuracy measure depend on σ , the relative comparison of the blur candidates and fidelity to true image restoration error remain the same.

For completeness we also show the value of the PSF-ISNR measure, derived from the proposed measure ρ by (5.3); see Fig. 5.3 right, which corresponds to the same experiment as in Fig. 5.3 left. As in the case of image MSE, this alternative accuracy metric also well corresponds to actual image restoration ISNR. For results of PSF-ISNR corresponding to experiments in Fig. 5.4 we refer the reader to [87].

5.5 Conclusion

We explained the necessity and advantages of evaluating accuracy of blur estimation separately from image restoration and discussed and demonstrated that commonly used direct measures such as PSF-MSE or cross-correlation are inadequate. As a solution we proposed and presented full derivation of full-reference measure of blur accuracy that faithfully corresponds to the mean squared error of image restoration caused by inaccuracy of the evaluated blur.

The proposed measure depends on one free parameter σ which effectively determines how steeply the value of the proposed measure increases the further the evaluated blur is from the ground truth. Relative comparison of different blur candidates with common ground truth is independent of σ , but for absolute comparison and reproducibility, its value should be specified with any published test results. We advocate

using a fixed value corresponding to reasonable noise level, for example $\sigma = 1/255$ for images with $[0, 1]$ intensity range.

As a counterpart of image ISNR, a well-established measure of image restoration performance, we proposed PSF-ISNR as a direct product of the primary presented blur accuracy measure and therefore also evaluated directly on the estimated blur.

We verified experimentally that the proposed error measure accurately captures the error introduced into image restoration by inaccuracy of blur estimation. Although the presented derivation is based on Wiener filter, experiments show that even more sophisticated non-blind methods respond similarly to blur inaccuracy and therefore the proposed measure provides realistic results of image restoration error obtainable by other more advanced non-blind methods.

Part II

Fast moving objects

Chapter 6

Deblurring of fast moving objects

Blind image deblurring encompasses wide range of tasks, from restoration of a single blurred image to deblurring of video sequences, and can be based on varying complexity of the blur model, from simple convolution via constrained spatially-variant convolution (e.g. limited to 3D camera motion) to fully spatially-variant blur. But the underlying assumption of most deblurring methods based on forward modeling of the acquisition process is that for each blurred image (or video sequence frame) there exists a latent sharp image and the two are functionally related by the considered blur model – most typically that the blurred image is some linear function of the sharp image. For example, in the previous part of this work we addressed the problem of single image blind deblurring and worked with the convolution blur model. Our assumption was that under perfect imaging conditions, for any given blurred image g there exists a convolution kernel h and a sharp image u such that $g = h * u$. If we extend the blur model or work with video sequences, it will not change the basic notion that each blurred image is a function of same static sharp image, which we ultimately aim to recover. This paradigm works well in the most common real scenarios, for example for small motion blurs where the camera motion is negligible compared to the scene geometry, or blur due to incorrect camera focus or media turbulence and many others. However, not all physical processes that cause blur have this form.

The primary counterexample is when the blur is caused by change of the photographed scene during exposure. In this case the acquired image is an integration of the sharp snapshots of the scene, each corresponding to an infinitesimal part of the exposure time, but it is not in general possible to isolate any of these snapshots and declare it as the sharp image corresponding to the blurred acquired image, because there is no functional relation (that we could use in the forward model of the acquisition) between any one of the snapshots and the blurred acquired image simply because none of the snapshots alone contains everything that is visible in the blurred image. This phenomenon can be small and negligible – in fact it is usually implicitly assumed, that the rate of potential changes in the scene is negligible compared to the camera shutter speed. Strictly speaking, blur due to camera motion in front of a 3D scene also in general falls into this category, because slightly different scene is visible at each time during the camera motion, but in majority of cases this effect is so small that it can be ignored or taken care of by the technique for increasing robustness to model violation presented in the previous part of the thesis.



Figure 6.1: Examples of objects blurred due to their motion during camera exposure: (left-to-right, top-to-bottom) car, motorcycle, model airplane, skier, volleyball, ping-pong ball.

In this part we will consider scenarios where change of the photographed scene is the dominant cause of image blur. In particular, we will focus on images of objects in fast motion (with respect to the camera shutter speed) while the rest of the scene is static and sharp; see Fig. 6.1 for examples of such images. Of course, the two discussed causes of blur can be combined – an image can be blurred due to motion of the camera and on top of that display a moving object that is further blurred due to a motion of its own. But just like in the previous part, where we addressed uniform blur of the whole image alone, in this part we will focus on motion-blurred objects separately to be able to better analyze the properties of the problem. In published literature, this scenario is sometimes called *dynamic scene deblurring*, see Sec. 2.3 for review of existing works.

Deblurring of fast moving objects (FMO deblurring) shares some similarities with traditional blind image deblurring, but there are many aspects in which the problem is quite different. One of the differences was already mentioned – in image deblurring the blurred image is a transformation of some sharp image, whereas in FMO deblurring the input is a blend of the blurred image of the foreground object and the sharp, partially occluded, image of the background. Another very important difference is that the sought sharp image of the object cannot be represented by a standard rectangular image, as it is in traditional deblurring, because the object can have arbitrary shape that is unknown and must be estimated from the blurred input along with the appearance.

Let us emphasize that due to the blending of the object and the background images, FMO deblurring cannot be regarded as a special case a spatially-variant blur. Similarly, it is not possible to segment the blurred object and process it with a standard blind deblurring method separately from the background, because due to gradual transition between the object and the background, the contour of the object is not well defined. In theory it would be possible to determine the alpha matte (non-binary opacity mask) of the blurred object image using one of the available alpha-matting algorithms such as Levin *et al.* [94] or Xu *et al.* [194], separate the object from the background and then proceed with blind deblurring. However, alpha matting is strongly ill-posed and the results are notoriously unreliable, unless meticulous manual pre-segmentation is

provided by the user. In its current state of the art it is not conceivable to use this approach in the blind setting.

The notion of a *fast moving object* was first introduced by Rozumnyi *et al.* [150] in the context of visual object tracking. The image formation model was presented and the non-blind problem with known object shape solved. This was later extended in Kotera *et al.* [82] where the fully blind problem was considered and therein called “deblatting” to emphasize the fact that two inverse problems are being solved: motion *deblurring* and image *matting*. It is certainly true that the problem exceeds the boundaries of ordinary image deblurring and possibly deserves its own name. We try to be consistent with the already published works on this topic and will keep using the more customary term *deblurring* in this chapter, where appearance restoration is the primary objective. In the next chapter focused on object tracking, where the primary interest is in object localization and motion trajectory estimation, we will use the term *deblatting* under which the technique was first published. The distinction is subtle and for all practical purposes the terms “FMO deblurring” and “deblatting”, as used in this thesis, can be regarded as synonyms.

In its full generality the FMO deblurring problem is indeed quite complex. We have to consider an object with unknown appearance and 3D shape that can undergo arbitrary motion (including 3D rotation or translation towards or away from the camera) and potentially change its shape as well as appearance during motion. Solving the deblurring problem then amounts to estimating the full 3D shape and appearance of the object, both as a function of time, from a single blurry input. Such problem is beyond our current capabilities and we instead consider a scenario that is substantially simplified, yet still general enough to encompass some useful real-world cases and interesting enough to reveal the characteristics of the general problem. In the theoretical analysis we will limit our focus on 2D objects undergoing arbitrary 2D motion and rotation. This is general enough to be practically applicable to some 3D scenarios as well, in particular 3D objects moving in a plane perpendicular to the camera optical axis and with imperceptible or in-plane rotation. We also make the assumption that the background image (image of the scene without the object of interest) is known. This may seem restrictive at first, but in practical scenarios obtaining such image is usually easy – for example in case of video sequences we can use one of the neighboring frames or a temporal median of several neighboring frames. It also does not simplify the problem as much as one might think – even segmenting the blurred object in the known background is still ill-posed, with infinite number of possible solutions.

Our presentation of the topic is divided into two chapters. In this chapter we focus on the task of recovering the object shape, unblurred appearance, and motion from a single blurred input. We define the image acquisition model, formulate the FMO deblurring as a constrained optimization problem and present a solution, which we then evaluate on real images of motion-blurred objects. We accompany the solution with extensive discussion on the theoretical properties of the FMO deblurring problem, state necessary conditions for solution uniqueness and identify the similarities and differences between FMO deblurring and standard blind image deblurring. In the next chapter we show how image deblurring can be used in visual tracking of fast moving objects.

6.1 Problem formulation

We consider the scenario of an unknown 2D object photographed at arbitrary motion in front of a known background. The object is fully defined by a pair (f, m) , where f denotes the object appearance (a 2D image) and m denotes the projected object shape (a 2D binary mask); e.g. if f is an image of a ball then m is a disk mask. Let us assume just for the purpose of initial discussion that the object motion is pure translation without any rotation, so that its blur can be expressed using convolutions – it will simplify the comparison to standard image deblurring. The mathematical formulation is invariant to the exact type of blur and we will describe an extension to rotational motion later. Under this assumption, the acquired image g of the scene can be written as

$$g = h * f + (1 - h * m) \cdot b + n, \quad (6.1)$$

where h is the motion blur (the point spread function is simply projection of the object trajectory onto the image plane), b is the background image and n denotes observation noise. We refer to the inverse problem of estimating h , f , and m given the input g and b as the *FMO deblurring problem*. See Fig. 6.2 for graphical illustration of the input (top row) and unknown variables (middle row) and their respective domains (sizes). Note that the unknown appearance f is limited only to the foreground object and does not contain any parts of the background, similarly for the shape mask m . The bottom row of Fig. 6.2 illustrates the components of the image formation model in eq. (6.1) – the blurred foreground object and partially occluded background.

Let us briefly explore the differences between our FMO image formation model (6.1) and the common image deblurring formation model (3.1) used in the previous chapters,

$$g = h * u + n. \quad (3.1)$$

The first term in eq. (6.1) represents the blurred foreground object (see Fig. 6.2 bottom-left) and roughly corresponds to the term $h * u$ in (3.1), though there are several important differences. In image deblurring the blur h is typically small compared to the image u and encodes only relative local transformation of the sharp image. In FMO deblurring the situation is reversed. The image of the foreground object can be quite small, depending on the size of the object, while the blur h must in fact have the same size of the acquired image g because it also encodes absolute position of the foreground object in the whole scene, as can be seen in Fig. 6.2-middle. The second term in (6.1) has no counterpart in traditional deblurring and corresponds to the sharp background with a “hole” where the background is occluded by the blurred object, which is determined by the motion of the object silhouette m (see Fig. 6.2 bottom-right). Note that even if m is strictly binary (i.e. the object is fully opaque without any semi-transparent parts), the transition between the images of the object and the background is gradual due to the blur. Because of the reversed roles of the blur and object image, we do not have to care about the convolution boundary problem as in Chapters 3 and 4 and if the blur model permits (e.g. the object does not rotate), we can freely use circular convolutions (provided that the object is sufficiently far from the image boundary).

We are going to solve the FMO deblurring the traditional way by reformulating it as an optimization problem in the same fashion we tackled image deblurring in Chapter 3. To mitigate its ill-posedness, we need to add some regularizers or feasibility

Input to FMO deblurring:

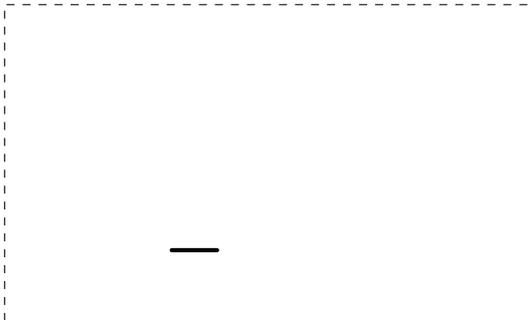


Input image g



Background image b

Output of FMO deblurring:



Blur PSF h (black and white inverted)

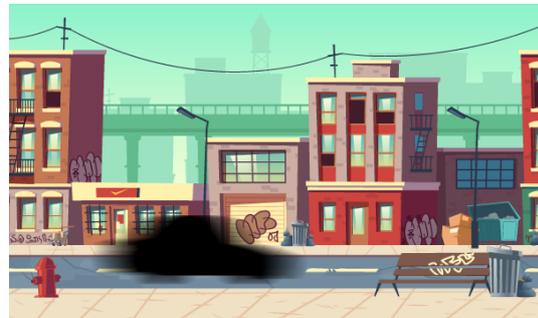


Foreground image f and shape mask m

Image formation model:



The term $h * f$ in eq. (6.1)



The term $(1 - h * m) \cdot b$ in eq. (6.1)

Figure 6.2: Illustration of the acquisition model of the FMO deblurring problem. **Top:** Input to FMO deblurring – image with blurred foreground object and separate background image. **Middle:** Output of FMO deblurring – motion blur (displayed as simple PSF without rotation for simplicity), sharp foreground object, and shape mask. **Bottom:** Components of the image formation model (6.1) – blurred foreground and partially occluded background. All images are in proper relative scale. Notice that the blur has the same size as the input image and effectively encodes position of the object in the input. The foreground object and shape mask are smaller than the input. Source images [49].

constraints. For image regularization we will use the isotropic total variation $\|\nabla f\|_{2,1}$ (see eq. (3.6) and comments therein)

$$\|\nabla f\|_{2,1} = \sum_i \sqrt{(\nabla_x f)_i^2 + (\nabla_y f)_i^2}, \quad (6.2)$$

where as previously $\nabla = [\nabla_x^T, \nabla_y^T]^T$ is the concatenation of partial derivative operators ∇_x and ∇_y . The blur h is large but nonzero only along the object motion trace, which suggests sparsity-promoting regularization – we will use ℓ^1 as previously in Chapter 3. All the unknowns h , f , and m are naturally non-negative and f and m are further bounded from above – f is bounded by the value which represents the maximum image intensity (assumed 1) and m is bounded by value signifying maximum opacity, which is also 1. There is also a deeper relation between f and m that we will specify momentarily. Putting it all together, the FMO deblurring formulated as a minimization problem becomes

$$\begin{aligned} \min_{h,f,m} \frac{1}{2} \|Hf - b \cdot (Hm) - (g - b)\|_2^2 + \alpha_f \|\nabla f\|_{2,1} + \alpha_h \|h\|_1 \\ \text{s.t. } h \geq 0, (f, m) \in S. \end{aligned} \quad (6.3)$$

The prescribed constraints are to be interpreted elementwise. For the sake of generality, we have already expressed blur as a general linear operation H defined by a set of parameters h related to the object motion (e.g. for translational motion, h is the point spread function given by the motion trajectory and H is the corresponding convolution operator).

The set S in (6.3) represents feasible values of f and m simultaneously, since the two are related. We initially defined m as a binary variable with f “visible” only where $m = 1$. This would imply

$$S' = \{(f, m) : (m = 1 \text{ and } f \in [0, 1]) \text{ or } (m = 0 \text{ and } f = 0)\} \quad (6.4)$$

shown in Fig. 6.3a. Instead, we propose a different definition, depicted in Fig. 6.3b,

$$S = \{(f, m) : m \in [0, 1], 0 \leq f \leq m\}. \quad (6.5)$$

This can be viewed as a real-valued relaxation of m as well as convex relaxation of S . The problem with binary m can be formulated as a mixed-integer or binary quadratic program, but this is known to be NP-hard and in practice approximated by spectral or semidefinite relaxation; Olsson *et al.* [120], Wang *et al.* [180]. We tested different large-scale binary quadratic programs and besides being extremely slow in convergence they show no improvement in results compared to the proposed relaxation.

We can also provide a point of view from which the relaxation (6.5) is in fact the correct formulation and with a further advantage that it allows considering semi-transparent objects (e.g. tinted glass). Let $\hat{f} \in [0, 1]$ be the latent image of the object and $m \in [0, 1]$ its real-valued opacity (alpha channel). Then the perceived object image is $\hat{f} \cdot m$ and modified eq. (6.1) reads

$$g = h * (\hat{f} \cdot m) + (1 - h * m) \cdot b + n. \quad (6.1')$$

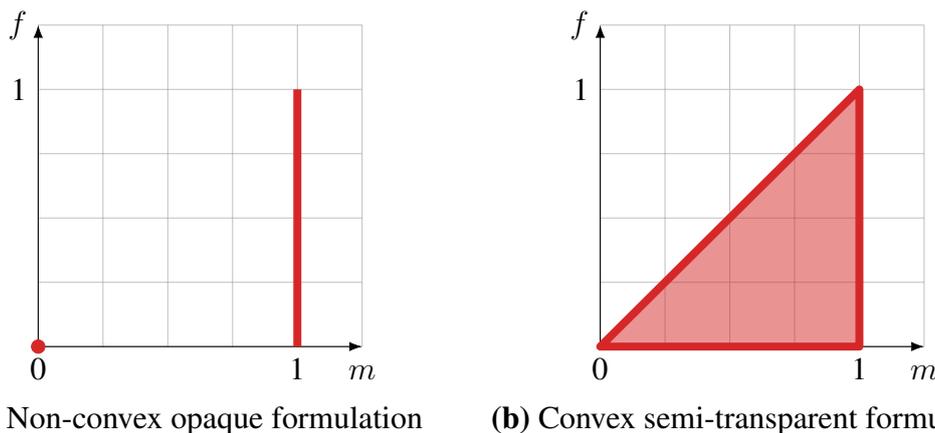


Figure 6.3: The set S of feasible values of pixel image intensity f and mask value m for the case of grayscale image. **(a)** Opaque formulation (6.4) with binary m and $f \in [0, 1]$ only if $m = 1$ resulting in non-convex set S (a point and a line segment). **(b)** Semi-transparent formulation (6.5) (real-valued relaxation) resulting in convex S . In the case of color (RGB) image the corresponding sets are (a) an isolated point and a cube, and (b) an oblique polyhedral pyramid in \mathbb{R}^4 .

This is in fact the correct formation model even if we do not consider semi-transparency. Under this formulation, \hat{f} and m are independent with separate feasibility constraints $\hat{f} \in [0, 1]$ and $m \in [0, 1]$ (or $m \in \{0, 1\}$ in the binary formulation). If instead we make the substitution $f := \hat{f} \cdot m$, we arrive at the originally stated acquisition model (6.1) and since now trivially $f \leq m$, the feasibility of the two (now related) variables is exactly given by (6.5). In practice, allowing semi-transparency is advantageous even for fully opaque objects, because fractional values of m can account for small shadows and mixed-pixels near object boundary (pixels which due to finite discretization extend both inside and outside the object). We will therefore continue the presentation with the concept that f is (in general) an RGB image and m is a single-channel real-valued opacity, though still referred to as “mask”.

6.2 Problem solution

Similarly to Chapter 3, we minimize (6.3) in a coordinate-descent manner by alternating between minimization with respect to (f, m) (fm -step) and minimization with respect to h (h -step), while keeping the other variable constant (f and m are optimized simultaneously). For each subproblem we use the alternating direction method of multipliers (ADMM, see e.g. Boyd *et al.* [13]) reviewed in Appendix 3.A to Chapter 3. As previously, we do not solve each coordinate sub-problem (h or (f, m)) to convergence, instead we perform one pass of the ADMM update equations of the particular sub-problem and switch to the other coordinate; this results in faster convergence of the whole problem (6.3). Just like in the previous chapter, we will express blur as a matrix-vector multiplication. Let F , M , and H denote the matrices performing blur conditioned on f , m , and h , respectively, such that $Fh \equiv Hf$ is the blurred object and $Mh \equiv Hm$ is the blurred object mask. In case the motion is pure translation, we have

$h * f \equiv Hf$ etc. Also, let $B = \text{diag}(b)$ denote the diagonal matrix which performs elementwise multiplication with b , such that $b \cdot v \equiv Bv$. In all the subsequent equations related to the optimization algorithm we will omit iteration indices and it is understood that all updates are performed in the same fashion as explicitly specified in Chapter 3; i.e. in each iteration, all variables are updated based on their values from the previous iteration.

Minimization with respect to (f, m)

We minimize the primary problem (6.3) with respect to f and m simultaneously. For given and fixed blur h we solve the problem

$$\min_{f,m} \frac{1}{2} \|Hf - BHm - (g - b)\|_2^2 + \alpha_f \|\nabla f\|_{2,1} \quad \text{s.t. } (f, m) \in S. \quad (6.6)$$

Let $\bar{f} = [f^T, m^T]^T$ denote the concatenation of the two unknowns into a single variable. The constraint in (6.6) is a particular instance of the convex constraint problem discussed on page 45 in App. 3.A. As suggested, we utilize ADMM variable splitting and introduce substitution $z_1 = \bar{f}$, upon which the constraint will be imposed. Similarly to Chapter 3, to facilitate minimization of the non-smooth total variation we introduce another substitution $z_2 = \nabla f$. According to the ADMM optimization procedure outlined in eqs. (3.34), minimization of (6.6) consists of the iteration steps

$$z_1 := \underset{z_1}{\text{argmin}} \left(\frac{\rho_1}{2} \|\bar{f} - z_1 + w_1\|_2^2 \right) \quad \text{s.t. } z_1 \in S, \quad (6.7)$$

$$z_2 := \underset{z_2}{\text{argmin}} \left(\alpha_f \|z_2\|_{2,1} + \frac{\rho_2}{2} \|\nabla f - z_2 + w_2\|_2^2 \right), \quad (6.8)$$

$$\begin{aligned} \bar{f} := \underset{f,m}{\text{argmin}} & \left(\frac{1}{2} \left\| \begin{bmatrix} H & -BH \end{bmatrix} \bar{f} - (g - b) \right\|_2^2 \right. \\ & \left. + \frac{\rho_1}{2} \|\bar{f} - z_1 + w_1\|_2^2 + \frac{\rho_2}{2} \|\nabla f - z_2 + w_2\|_2^2 \right), \end{aligned} \quad (6.9)$$

$$w_1 := w_1 + \bar{f} - z_1, \quad (6.10)$$

$$w_2 := w_2 + \nabla f - z_2. \quad (6.11)$$

The new variables w_1 and w_2 are Lagrange multipliers corresponding to the constraints caused by the variable substitutions and $\rho_{1,2}$ are user-defined parameters; see App. 3.A for details.

Solution of the problem (6.7) is (elementwise) projection of the minimizer of the quadratic term onto the convex feasibility set,

$$z_1 = \text{proj}_S(\bar{f} + w_1). \quad (6.12)$$

For grayscale image $f \in [0, 1]$ the set S is depicted in Fig. 6.3b, but more common is the situation that f is a vector-valued RGB image, then $S \subset \mathbb{R}^4$ (three dimensions for f and one for m) and has a shape of a bounded hyper-pyramid (hyper-cube “base” for $m = 1$, sides formed by simplexes, and one tip point for $m = 0$). The projection

(6.12) can be in principle carried out directly, but we employ an equally fast and clearer method. The set S can be defined as the intersection of convex sets S_0, \dots, S_3

$$S_0 = \{m \geq 0, m \leq 1, f^{(i)} \geq 0 \forall i\}, \quad (6.13a)$$

$$S_k = \{m - f^{(k)} \geq 0\}, \quad (6.13b)$$

where $f^{(k)}$ denotes k -th color channel of f . Each of the sets S_* is fast to project to, S_0 is axis-aligned and $S_k, k > 0$ are half-spaces. We therefore evaluate the projection to S by alternating projection onto the sets S_k using the algorithm by Boyle and Dykstra [14] (best see Tibshirani [169]), which converges in just a few iterations.

Problem (6.8) is related to the minimization of the isotropic TV regularization, which we covered at length in “ ℓ^p proximal mapping” on page 34 of Sec. 3.2; in particular see eqs. (3.19). The solution is given by

$$z_{2,x} = \frac{\sigma(\nabla_x f + w_{2,x})}{\sqrt{(\nabla_x f + w_{2,x})^2 + (\nabla_y f + w_{2,y})^2}}, \quad (6.14a)$$

$$z_{2,y} = \frac{\sigma(\nabla_y f + w_{2,y})}{\sqrt{(\nabla_x f + w_{2,x})^2 + (\nabla_y f + w_{2,y})^2}}, \quad (6.14b)$$

where

$$\sigma = \max\left(\sqrt{(\nabla_x f + w_{2,x})^2 + (\nabla_y f + w_{2,y})^2} - \frac{\alpha_f}{\rho_2}, 0\right). \quad (6.14c)$$

Recall that the derivative operator is a concatenation of horizontal and vertical partial derivatives, therefore z_2 has twice as many pixels as the object image f , which we denoted by $z_{2,x}$ and $z_{2,y}$ and analogously for w_2 .

Lastly, by differentiating (6.9) with respect to \bar{f} we get the optimality condition in the form of a linear system for $\bar{f} = [f^T, m^T]^T$,

$$\begin{aligned} & \begin{bmatrix} H^T H + \rho_1 I + \rho_2 \nabla^T \nabla & -H^T B H \\ -H^T B H & H^T B^2 H + \rho_1 I \end{bmatrix} \begin{bmatrix} f \\ m \end{bmatrix} \\ & = \begin{bmatrix} H^T \\ -H^T B \end{bmatrix} (g - b) + \rho_1 (z_1 - w_1) + \begin{bmatrix} \rho_2 \nabla^T \\ 0 \end{bmatrix} (z_2 - w_2). \end{aligned} \quad (6.15)$$

Recall that B is diagonal. Even if the object motion is pure translation, the left-hand side is a mixture of convolution and diagonal matrices, which prevents solving the system efficiently using the Fourier transform. Instead we use the conjugate gradient method.

Minimization with respect to h

For the given image f and mask m we solve the problem

$$\min_h \frac{1}{2} \|(F - BM)h - (g - b)\|_2^2 + \alpha_h \|h\|_1 \quad \text{s.t. } h \geq 0. \quad (6.16)$$

To isolate the positivity constraint and the non-smooth ℓ^1 regularizer from the data term we again use variable splitting and introduce substitution $z = h$. Following the recipe in (3.34), the optimization of (6.16) consists of the iteration steps

$$z := \underset{z}{\operatorname{argmin}} \left(\alpha_h \|z\|_1 + \frac{\rho}{2} \|h - z + w\|_2^2 \right) \quad \text{s.t. } z \geq 0, \quad (6.17)$$

$$h := \underset{h}{\operatorname{argmin}} \left(\frac{1}{2} \|(F - BM)h - (g - b)\|_2^2 + \frac{\rho}{2} \|h - z + w\|_2^2 \right), \quad (6.18)$$

$$w := w + h - z. \quad (6.19)$$

The new variable w is again related to ADMM (App. 3.A).

The problem in eq. (6.17) is a combination of ℓ^1 and convex constraint proximal mapping and is exactly equivalent to the problem (3.25b) we already solved in Sec. 3.2; see the solution in eq. (3.27). For our current problem we get

$$z = \max \left(h + w - \frac{\alpha_h}{\rho}, 0 \right). \quad (6.20)$$

By differentiating (6.18) with respect to h , we get the optimality condition as a linear system for h

$$\left((F - BM)^T (F - BM) + \rho I \right) h = (F - BM)^T (g - b) + \rho(z - w). \quad (6.21)$$

The left-hand side again mixes convolution and diagonal matrices and cannot be solved efficiently in the Fourier domain. As in the fm -step, we solve the system (6.21) using the conjugate gradient method.

Algorithm summary

The FMO deblurring problem is less susceptible to getting trapped in the no-blur solution encountered previously in standard image deblurring, as we will discuss in the following section. For this reason it is not necessary to employ the multiscale minimization approach as we did in Chapter 3 and 4 and we can solve (6.3) directly at the original scale of the input.

The overview of the optimization can be seen in Alg. 6.1. All steps are straightforward, except for the linear systems (6.15) and (6.21), which are solved using the conjugate gradient method. We initialize the algorithm with “uninformative” $h \equiv \text{const}$, unless the location or approximate trajectory of the object are known. If it is more natural to initialize the algorithm with the approximate estimate of the appearance and shape (f, m) (as is the case in the next chapter), the order of the two minimization blocks in Alg. 6.1 can be reversed. For the regularization parameters we use values $\alpha_f = 10^{-3}$, $\alpha_h = 1$.

Extension to 2D rotation

The convolution blur model is extensible to 2D in-plane rotation without introducing any substantial conceptual changes. Pure motion blur is traditionally modeled as a

Algorithm 6.1 Summary of the deblurring algorithm presented in Chapter 6

input: blurred image g , background b

initialize $h \equiv 1/N$ where $N = \text{size}(h)$, $f \equiv 1$, $m \equiv 1$, $w_* \equiv 0$

repeat

 // minimization of (f, m)

 update z_1 by (6.12) using the Dykstra's method [169] and sets (6.13)

 update z_2 by (6.14)

 update f and m by solving (6.15)

 update w_1 and w_2 by (6.10) and (6.11)

 // minimization of h

 update z by (6.20)

 update h by solving (6.21)

 update w by (6.19)

until convergence

output: f , m , and h as the final foreground image, mask, and blur estimates

weighted superposition of finite number of discretized displacements,

$$Hf = \sum_i h_i T_i f, \quad (6.22)$$

where T_i is a translation operator that moves f into position i and h_i is the weight. We are used to thinking of uniform motion blur in terms of convolution, where the PSF is exactly the sum $\sum_i h_i T_i$ if T_i is viewed as a convolution kernel (shifted delta pulse). The indices i correspond to particular pixels in the blur PSF and h_i are intensities of these pixels. Rotation can be discretized in the same fashion and we can write

$$Hf = \sum_{ij} h_{ij} T_i R_j f, \quad (6.23)$$

where R_j is the rotation operator into the j -th angle. In other words, motion and rotation blur of f is equivalent to pure motion blur of pre-rotated images of f . From the mathematical perspective, the problem remains the same, the only differences are its practical numerical properties (rotation increases the dimensionality of the unknowns) and in the implementation, in part because H is no longer a circulant matrix. Similar extension to 3D rotation, on the other hand, increases the problem complexity significantly, because on top of further increase of dimensionality it requires modeling of f and m in 3D and considering their projection onto the image plane in each of the discretized poses, which is no longer trivial.

In the following theoretical discussion we will consider only translational motion purely for the sake of clarity of the presented statements and arguments, all conclusions can be easily generalized to rotational motion.

6.3 Uniqueness of solutions

There is no general proof of solution uniqueness in blind image deblurring due to the image-blur coupling in the convolution, which makes the problem non-convex. The

following theoretical section states some necessary conditions for solution uniqueness in the FMO deblurring problem and discusses similarities and differences of the FMO deblurring and standard blind deblurring (BD).

We consider the discrete, noiseless FMO image formation model

$$g = h * f + (1 - h * m) \cdot b \quad (6.24)$$

and assume that every variable has a prescribed bounded domain denoted by dom and nonzero support denoted by supp , e.g. $\text{supp } f \subseteq \text{dom } f$. An image f is called *compactly supported* within its domain if increasing its domain does not increase its support (i.e. the support can be overestimated). For example, in standard image deblurring the true blur has a finite size and prescribing a larger domain has no effect, the blur support remains the same. The domain of variables is determined either by the size of input data (as in the case of the acquired image g), or prescribed by the user (as in the case of the object appearance f , which is not known in advance).

For the purposes of the discussion we make the following natural assumptions about the input (both in terms of user choice and physical reality) to the FMO deblurring problem:

- (a1) All variables satisfy $0 \leq g, b, h, f, m \leq 1$.
- (a2) h preserves energy, i.e. $\sum_i h_i = 1$.
- (a3) h , f , and m are irreducible, i.e. cannot be non-trivially factorized as $h = h' * h''$ etc.
- (a4) g , b , and h have the same bounded rectangular domain and f and m also have the same bounded rectangular domain which is smaller than the previous one, i.e. $\text{dom } f \subset \text{dom } g$. Further, the motion of the object satisfies $\text{supp}(h * m_1) \subset \text{dom } g$ where m_1 is such that $\text{dom } m_1 = \text{dom } m$ and $m_1 \equiv 1$.

A tuple (f, m, h) is called a solution of the FMO deblurring if for the given b and g it satisfies (6.24) and assumptions (a1, a2, a4). Energy preservation is just a technical requirement to avoid trivial equivalent solutions of the form $(\alpha f, \alpha m, (1/\alpha)h)$. If h , or f and m are not irreducible then there is an inherent ambiguity in the solution, as it is impossible to determine which of the factors is the sought blur and which can be included into f and m as $f' = h' * f$ etc. This in fact imposes further constraints on the factorization, e.g. non-negativity of the factors. The last condition in (a4) precludes motion that is too close to the boundary of the input, in particular motion that originates or terminates outside the observed scene, as this is not even properly modeled by (6.24).

No-blur solutions

Let us start the discussion considering only grayscale images and then generalize to color images. First notice that if the background is zero (black), FMO deblurring almost reduces to standard blind deblurring, since for the image formation model we have from (6.24)

$$g = h * f, \quad (6.25)$$

which is the image formation of BD (see eq. (3.1) in Chapter 3 and note that therein the unknown sharp image is denoted u). The difference remains in the respective domains of the sought image and blur, which has consequences for the possible trivial solutions.

In standard BD, f is the unknown sharp image with domain similar to the input g and its support is not compact, since g is typically a cropped subset of some larger (potentially infinite) observed scene. The blur h describes the local averaging of f which has only finite extent, its domain is therefore much smaller than the domain of f and it has compact support.

In FMO deblurring the situation is almost reversed. Image f is the moving object, which covers only small part of the observed scene as well as the input. It has bounded domain $\text{dom } f \subset \text{dom } g$ and is compactly supported; the same holds for m . The blur h encodes the object motion as well as its position within the input frame, its domain must therefore be of the same size as the input, $\text{dom } h = \text{dom } g$, and since the motion of the object is spatially bounded by the assumption (a4), h is also compactly supported. Refer to Fig. 6.2 for illustration of domains of the unknowns in FMO deblurring problem.

Blind image deblurring is prone to spurious local solutions, most notable of which being the already encountered no-blur solution, where the estimated sharp image is the input blurred image and the estimated blur is delta function (convolution identity), which perfectly satisfies (6.25).

In FMO deblurring this corresponds to the solution

$$(f', m', h') = (h * f, h * m, \delta), \quad (6.26)$$

as can be verified by substituting to (6.24). However, this solution is possible only if $\text{supp}(h * f) \subseteq \text{dom } f$, that is if the domain of f is overestimated enough to accommodate increase in size due to blur. This type of no-blur solution can therefore be prevented by tight setting of $\text{dom } f$ as a minimum rectangle that includes $\text{supp } f$, but this can be difficult in practice as the true size of f is not known. We can conclude that the shorter the true blur and the more uncertain the true size of the object, the higher the probability that this type of no-blur solution occurs.

Even if the domain of f is not overestimated so much that the full no-blur solution (6.26) is possible, other spurious solutions can still appear. They are of the form $(f', m', h') = (s * f, s * m, s^\dagger * h)$ for some blur kernel s with sufficiently small support such that $\text{supp}(s * f) \subseteq \text{dom } f$ and an approximate inverse s^\dagger such that $\text{supp } s^\dagger * h \subseteq \text{dom } h$. It is known that if an exact inverse of a compactly supported s exists, i.e. $s^{-1} * s = \delta$, then s^{-1} has infinite support. With the compactly supported approximation s^\dagger , (f', m', h') is not an exact solution to (6.24) but it could still present a local minimum of (6.3) in which the optimization gets trapped in practice. Fortunately, this problem is partially mitigated by the prescribed constraints in (6.3). If $s \geq 0$ then s^{-1} (and to certain degree also s^\dagger) is bound to have negative values or vice versa. This implies that one of f' , m' , or h' would contain negative values, which is prohibited by the positivity constraints.

We have seen that the roles of f and h are somewhat reversed between standard BD and FMO deblurring, which is the reason why another type of no-blur solution exists in FMO deblurring that has no counterpart in BD. If the object image f were estimated

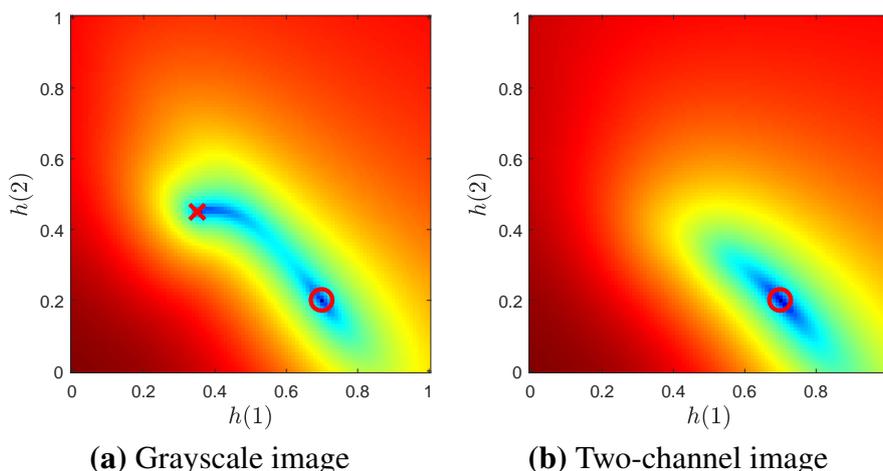


Figure 6.4: Toy experiment with two-pixel image and four-pixel blur, plot of the loss function $L(h)$ marginalized over h_3, h_4 , and image f . **(a)** Grayscale image f . **(b)** Color (two-channel) image f . Red circle indicates the correct blur, red cross indicates no-blur solution present only in the grayscale case.

trivially as a delta function, we can derive by substituting to (6.24) that

$$(f', m', h') = (\delta, \delta, (h * f - b(h * m))/(1 - b)) \quad (6.27)$$

is also a perfect solution. In the case of zero background, this simplifies to $(\delta, \delta, h * f)$, which is easier to interpret. The object image is estimated as a single nonzero pixel (which surely fits inside $\text{dom } f$) and the whole input image is explained by the estimated blur h' . As in the previous case, in practice this potential problem is reduced by the used regularizers and prescribed constraints. Sparsity-promoting regularization $\|h\|_1$ penalizes the very non-sparse solution h' while at the same time the bound constraints $f, m \leq 1$ prevent the tendency to decrease intensities in h' (as a way of decreasing $\|h\|_1$) and compensate with increased intensities in f' and m' . Unlike the previously discussed kind of spurious solutions, this type of no-blur solution is actually rarely encountered in practice, mainly because it is very unlikely to occur for general color images, as we explain in the following experiment.

To illustrate the phenomenon of no-blur solutions, we conducted an experiment similar to that in Perrone and Favaro [131] but modified for FMO deblurring. We consider a simplified FMO deblurring problem (6.3) in which we assume that the mask m is known and fixed, and define the loss function of h as the marginal minimization of the data term over f

$$L(h) = \min_f \|Hf - BHm - (g - b)\|_2^2 \quad \text{s.t. (a1) and (a2)}. \quad (6.28)$$

The minimization with respect to f is carried out analogously to the presented minimization of the (f, m) subproblem. To be able to visualize the loss function (6.28) we assume a toy 1D problem with $h = [h_1, h_2, h_3, h_4]$ of length four and $f = [f_1, f_2]$ of length two. Assumption (a2) reduces dimensionality of the problem by one since $h_4 = 1 - \sum_i^3 h_i$ so $L(h)$ is really a function of only three variables. Final reduction to a 2D function is done by marginalization and plotting $\min_{h_3} L(h)$. Fig. 6.4a shows the

plot of $L(h)$ for the FMO deblurring problem with $h = [.7, .2, .1, 0]$, $f = [.5, .5]$, and zero background $b = [0, 0, 0, 0]$. The correct solution is marked by circle at location $[h_1, h_2] = [.7, .2]$. Another equally strong minimum is at $[h'_1, h'_2] = [.35, .45]$, which corresponds to the no-blur solution $h' = h * f = [.35, .45, .15, .05]$.

The situation is quite different for color images. In the case of zero background, the acquisition model (6.25) becomes

$$g^{(i)} = h * f^{(i)}, \quad i \in \{R, G, B\}, \quad (6.29)$$

where $g^{(i)}$ is the i -th color channel of the observation and likewise for f . Reversing again the roles of f and h , this is equivalent to a multichannel blind deblurring problem (MBD), where a single “image” h is blurred by different “blurs” $f^{(i)}$ corresponding to the color channels of the object latent image. A well-known advantage of MBD is the absence of no-blur solution and better conditionality of the inverse problem. The no-blur solution of the second kind (6.27) is therefore much less likely for color images, because single grayscale blur h would have to explain three potentially different colors channels of the image f . This is fundamental difference from BD, where additional color channels bring no advantage because each color channel constitutes a separate, unrelated problem. Unfortunately, the situation remains unchanged for the no-blur solution of the first kind (6.26).

We repeated the same toy 1D experiment in color and calculated $L(h)$ in eq. (6.28) for f having two color channels, $f^{(1)} = [.5, .5]$ and $f^{(2)} = [.8, .2]$; see Fig. 6.4b. The plot shows a unique minimum at the correct location, the no-blur solution disappeared. We can conclude that for color images and if the domains of f and m are sufficiently tight, no-blur solutions present less of a problem for FMO deblurring than for standard BD.

As a final remark, in the case of color images the assumption of irreducibility (a3) can be weakened. It suffices to have $f^{(i)}$ mutually *weakly coprime*, i.e. having no common factor except delta function. Color channels of natural images usually are weakly coprime, though we can find special cases when they are not. For example the color channels of an object with arbitrary shape m and uniform color $[r, g, b] \in \mathbb{R}^3$ have m as the common factor. In case of zero background the no-blur solution of the second kind is then $(f', m', h') = ([r\delta, b\delta, g\delta], \delta, h * m)$.

Image–mask ambiguity

Among other things, FMO deblurring differs from standard image blind deblurring by the presence of the object shape mask m , which is estimated together with the latent object appearance f . This causes an ambiguity in the fm -step that has no counterpart in standard image deblurring. In this part we will assume that the blur h is known and analyze properties of the (f, m) subproblem alone (a sort of “non-blind FMO deblurring”), but obviously any identified problems propagate to the fully blind case.

For simplicity, let us consider grayscale images with square domains, but the same arguments can be made also for arbitrary color images. We can rewrite the acquisition model (6.24) using the combined variable $\bar{f} = [f^T, m^T]^T$ as

$$g - b = A\bar{f} = \begin{bmatrix} H & -BH \end{bmatrix} \bar{f}. \quad (6.30)$$

Without regularization, the fm -step has a unique solution if the matrix $A = [H \quad -BH]$ is full column rank, since the system matrix of the inverse problem is $A^T A$. Denoting S_h the number of pixels in the domain of h (as well as b, g) and S_f the number of pixels in the domain of f (as well as m), the matrix A is of size $S_h \times 2S_f$, i.e. it has more rows than columns since $S_h \gg S_f$. There are two notable situations when it is column-rank deficient: H has fewer than $2S_f$ non-zero rows or the background b is uniform.

Assuming continuous linear motion blur, the number of non-zero rows in H depends on the blur length L and it can be derived that it is equal to $(L - 1)\sqrt{S_f} + S_f$. Recall that $\sqrt{S_f}$ is the length of the side of the square domain of f , so in our model situation it is the length of the object. This implies the necessary condition for the minimal blur length

$$L_{\min} = \sqrt{S_f} + 1. \quad (6.31)$$

If $L < L_{\min}$ (i.e. blur is shorter than or equal to the size of the object), then the system matrix $A^T A$ is singular and estimation of f and m is ambiguous. For color images, the condition is weaker. The size of matrix A is $3S_h \times 4S_f$ (assuming three-channel image) and the corresponding minimal blur length is

$$L_{\min}^{\text{RGB}} = \frac{\sqrt{S_f}}{3} + 1 \approx \frac{L_{\min}}{3}. \quad (6.32)$$

Blur longer than one-third of the object size is sufficient to distinguish f and m .

For general motion blur such simple condition cannot be written, as the matrix rank depends not only on the motion length but also on its structure. For example, a discontinuous “stroboscopic” blur that places individual snapshots of the object far apart has better conditionality than continuous motion blur with the same number of nonzero entries. Qualitatively speaking, the f -part and m -part of the combined unknown \bar{f} are uniquely determined if the blur is sufficiently long or complex that it provides enough snapshots of the object against different parts of the background. In particular, if the true blur is delta function (the object is sharp on top of the background), it is impossible to determine the shape of the object, because for every pixel i in the observed image g (that is affected by the blur) we have a scalar equation of the type

$$g_i - b_i = f_i - m_i \cdot b_i, \quad (6.33)$$

where both f_i and m_i are unknown, so the number of solutions is in general infinite and 1D measure of the solution space is limited only by the bound constraints in assumption (a1). This characteristic of FMO deblurring is in stark contrast to standard image deblurring, where no-blur non-blind deblurring works flawlessly.

We conducted a synthetic experiment to verify these conclusions. We blurred an image of fixed size by a motion blur of increasing length and then performed (f, m) estimation based solely on the data term of the (f, m) subproblem (6.6) (in case of rank-deficient system matrix we chose the minimum-norm solution, i.e. used pseudoinverse). Results of the image restoration error (averaged over 100 different images and backgrounds) as a function of blur length are in Fig. 6.5. The size of the object image f is 60×60 px. In the single-channel grayscale case (solid gray) the error decreases slowly as the blur length increases but does not drop to zero until the minimum

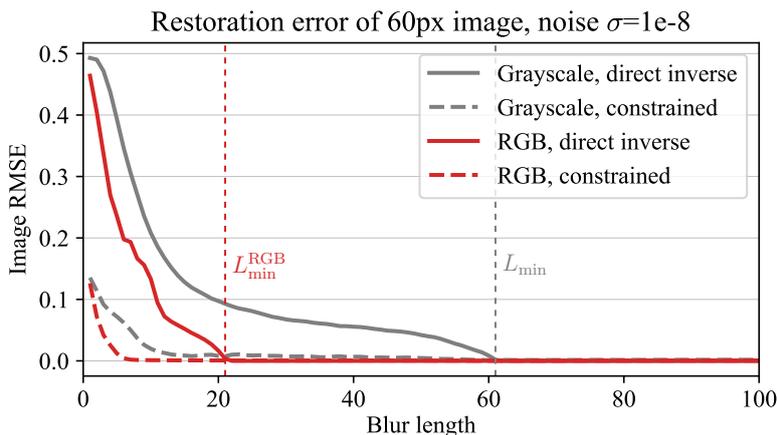


Figure 6.5: Non-blind image restoration error as a function of blur length. Unless the blur length exceeds the minimum value L_{\min} (for grayscale images, solid gray) or L_{\min}^{RGB} (for color images, solid red) that depends on the object size, the foreground appearance f and shape m cannot be determined unambiguously. The solid lines correspond to direct unconstrained inversion of the data term in (6.6). If the proposed constraint (6.5) is used, the restoration is successful for much shorter blurs (dashed lines).

blur length condition for grayscale images is met and the system matrix becomes full rank. In the RGB case (solid red) the error vanishes roughly at $60/3 + 1 = 21\text{px}$ as predicted.

To emphasize the important role of the proposed constraint relating f and m defined by (6.5), we repeated the same experiment but with the bound constraint included, i.e. we solved the full (f, m) subproblem (6.6) with $\alpha_f = 0$ (otherwise zero restoration error is unattainable due to the TV regularization). The results are depicted by the dashed lines in Fig. 6.5. We see that the estimation is successful for much shorter blurs (in the RGB case approx. 5px for 60px object), the ambiguity has been substantially reduced. If we repeat the same experiment with the shape mask m known and fixed (which is equivalent to standard non-blind deblurring), the image estimation error is practically zero for any blur length (increasing only slightly due to noise). The ambiguity in the (f, m) subproblem of FMO deblurring for small blurs is caused by the coupling of f and m in the image formation model, a phenomenon that does not exist in standard deblurring.

Another ambiguous situation arises when the background is uniform. Then the matrix A rewrites to $A = [H \quad -\gamma H]$, where the “scalar” γ is the background color, and it is readily seen that the matrix is singular with rank at most S_f . In this case it is also impossible to unambiguously determine the object latent image f and mask m , regardless of blur length. In the previous discussion we said that the object motion must provide enough snapshots of the object against different parts of the background. If the background is uniform, clearly different placements of the object in the input image do not provide any information. Although this is related to the fm -step alone, the problem naturally propagates to the fully blind case. Fig. 6.6 shows a synthetic example of how uniform background negatively affects blind estimation of all the unknowns – blur, image, and mask.

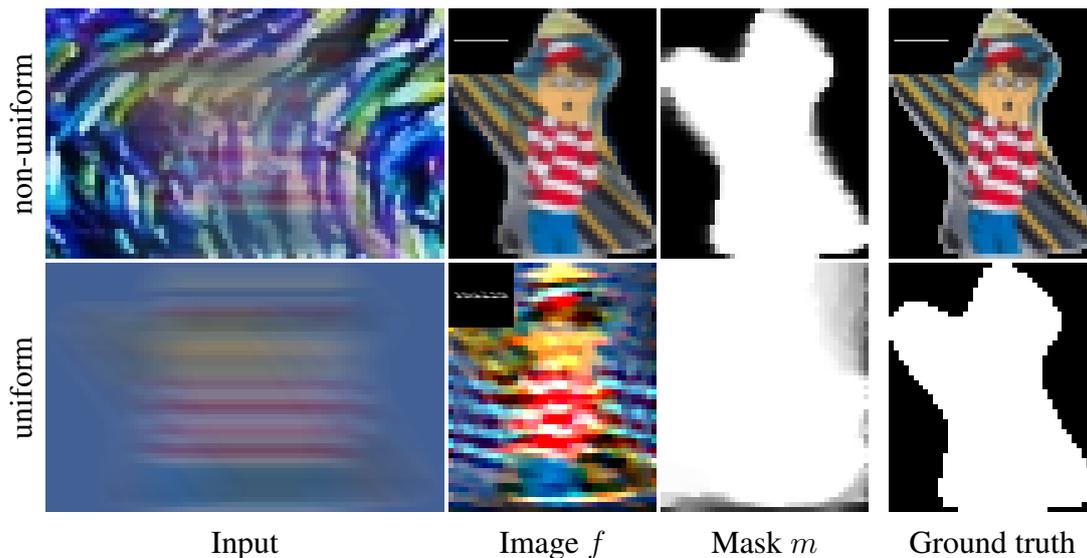


Figure 6.6: Effect of background content on the result of blind FMO deblurring. **Top:** Natural image in the background and corresponding estimation of motion-blurred image and mask; compare with ground truth in the right column. **Bottom:** The same foreground image and blur on uniform background (background from the top row averaged) – the results is significantly worse due to ambiguity between f and m .

We discussed two canonical cases of the system matrix rank deficiency – due to H and due to B . Since the two are combined in A , any mixture of the two cases can happen in practice, for example when only part of the object motion is over nearly uniform background or when the background contains repetitive structures in the direction of the motion. Our intention was to provide a deeper insight into the less intuitive property of FMO deblurring, which is that even a seemingly easy non-blind problem – small blur and simple background – is in fact ill-posed because the shape of the object is unknown. In practice this problem is partially alleviated by the proposed regularization.

6.4 Experimental evaluation

To evaluate the presented method and compare with other deblurring methods we created a dataset containing images of motion-blurred objects in front of a sharp background. The images were obtained by sliding a picture on a table while taking a photograph from above. Five physically different pictures ranging in size from 150×100 px to 240×170 px were each photographed 15 times while undergoing different motion ranging in length from about 10px to 130px; the dataset thus consists of 75 different input images with real motion blur in total. We also photographed the cards at rest to get the ground-truth appearance (see the second column in Fig. 6.8) and the error between the reconstructed picture and the ground truth was measured in terms of PSNR (see Sec. 5.1 for review of standard image quality measures).

As the benchmark for quantitative comparison of our results we chose the robust blind image deblurring method by Pan *et al.* [125] and the deep learning-based method DeblurGAN by Kupyn *et al.* [90]. Neither of the two methods explicitly addresses

the FMO deblurring problem we are investigating, we therefore adjusted how each of the methods is used to mimic steps one would take to reconstruct motion-blurred foreground using that method.

The BD method Pan16 [125] already appeared in comparison in the experimental section of Chapter 4. It is a standard image deblurring method in the sense that it assumes an input image blurred uniformly in its whole area. In the FMO setting this assumption is violated by the sharp background partially visible through parts the blurred foreground object, so for Pan16 we cropped the full-size input image to contain only the blurred foreground object and as little of the background as possible, to get closer to the scenario of standard image deblurring. This is not always possible because for blur of a given length the input must have a certain minimum size, in these cases we had to compromise and the input to Pan16 necessarily contains some parts of the background. We chose Pan16 in part because it includes extra handling of outliers, therefore it has a good chance of being successful even if small parts of the input violate the convolution blur assumption. To further mitigate the influence of background, we used Pan16 only to estimate the motion blur and then applied the non-blind deblurring part of Kotera and Šroubek [85], which is essentially the presented (f, m) subproblem with the blur and mask fixed (at this point we supplied the ground-truth mask, because Pan16 naturally cannot estimate the object shape) to estimate the full-size foreground image. We therefore effectively evaluate only blind motion blur estimation of Pan16 in the FMO setting.

DeblurGAN by Kupyn *et al.* [90] uses a convolutional neural network that directly outputs a natural-looking sharp image without intermediate blur estimation. The network is trained for various spatially variant motion blurs but does not explicitly assume any particular blur model. It is applicable and advertised also for the dynamic scene scenario, i.e. object motion deblurring, but we have noticed that if the background occupies a large area of the input or if the sharp background is visible through the motion-blurred object, the performance of DeblurGAN drops significantly, as the network tends to output the same image it received as input. As in the case of Pan16 we therefore cropped the input image for DeblurGAN to limit the amount of apparent background. For all three methods the restoration accuracy was measured only in the central part of the foreground image, so that the results of DeblurGAN are not affected by different input size.

Quantitative results of foreground image deblurring in terms of restoration PSNR are in Fig. 6.7. For easier comparison we averaged the results over the five different picture cards and also grouped the test inputs by blur length into 5 categories (with 3 examples of each foreground image in each category) and averaged those as well. Complete results for each image are available in the supplementary of Kotera *et al.* [81]. As a reference the chart in Fig. 6.7 contains also direct PSNR of the blurred input. We can see that for small blurs all methods succeed in improving the input to a certain degree, the presented method being superior. As the blur length increases, the performance of Pan16 drops rapidly – it fails to estimate the blur correctly and as a result the restored images contain increasing amount of deblurring artifacts, rendering them worse than the blurred input itself. The performance of DeblurGAN does not deteriorate as much because it does not perform explicit algebraic inversion and therefore escapes the typical pitfall of classical methods, but apparently it is not trained to

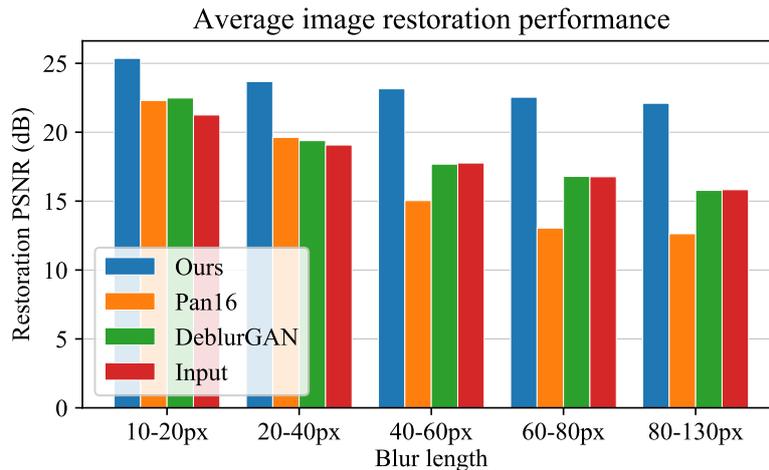


Figure 6.7: Results of blind object restoration (in PSNR, higher is better) as a function of blur length. The results are averaged over 5 different foreground images and 3 blur realizations. The foreground images range in size from 150×100 px to 240×170 px. Direct PSNR of the blurred input (red) shown as a reference. See Fig. 6.8 for examples of input images in the test dataset and corresponding deblurring results.

perform on such large blurs and cannot cope with foreground-background blending. From certain blur length it starts to return the blurred input as output. The presented method is visibly superior and although its performance also gets progressively worse with increasing blur length, the result is still much improved over the input.

Fig. 6.8 contains several examples of results for different input images and blur lengths from the evaluation dataset. It well illustrates the behavior of each method and demonstrates the effectiveness of the presented method compared to the other two approaches. This experiment shows that the task of FMO deblurring is not solvable by general existing deblurring methods and that a specialized method is necessary.

Our method estimates not only the object translational motion but also its rotation. Fig. 6.9 demonstrates that in practice this is a necessity. Results where rotation is included in the model (two rightmost columns) are significantly better than if the motion is approximated by pure translation even if the total image rotation in the input is relatively small.

Temporal super-resolution

An exciting application of FMO deblurring and perhaps the most natural way of presenting the obtained results is the so-called *temporal super-resolution* of the input image. This refers to interpreting the input image with the motion-blurred object as a frame of a video sequence and using deblurring to re-synthesize this sequence at higher framerate. We show examples of temporal super-resolution of the images from the test dataset in Fig. 6.10. In the top row are the input images and estimated blur PSFs (including color-coded rotation), below are the results of temporal super-resolution. The motion of the foreground object is encoded in its motion blur, but to be able to exploit this information and re-synthesize the object motion it is necessary to establish some temporal structure in the estimated blur. To this end we fit the blur PSF with a

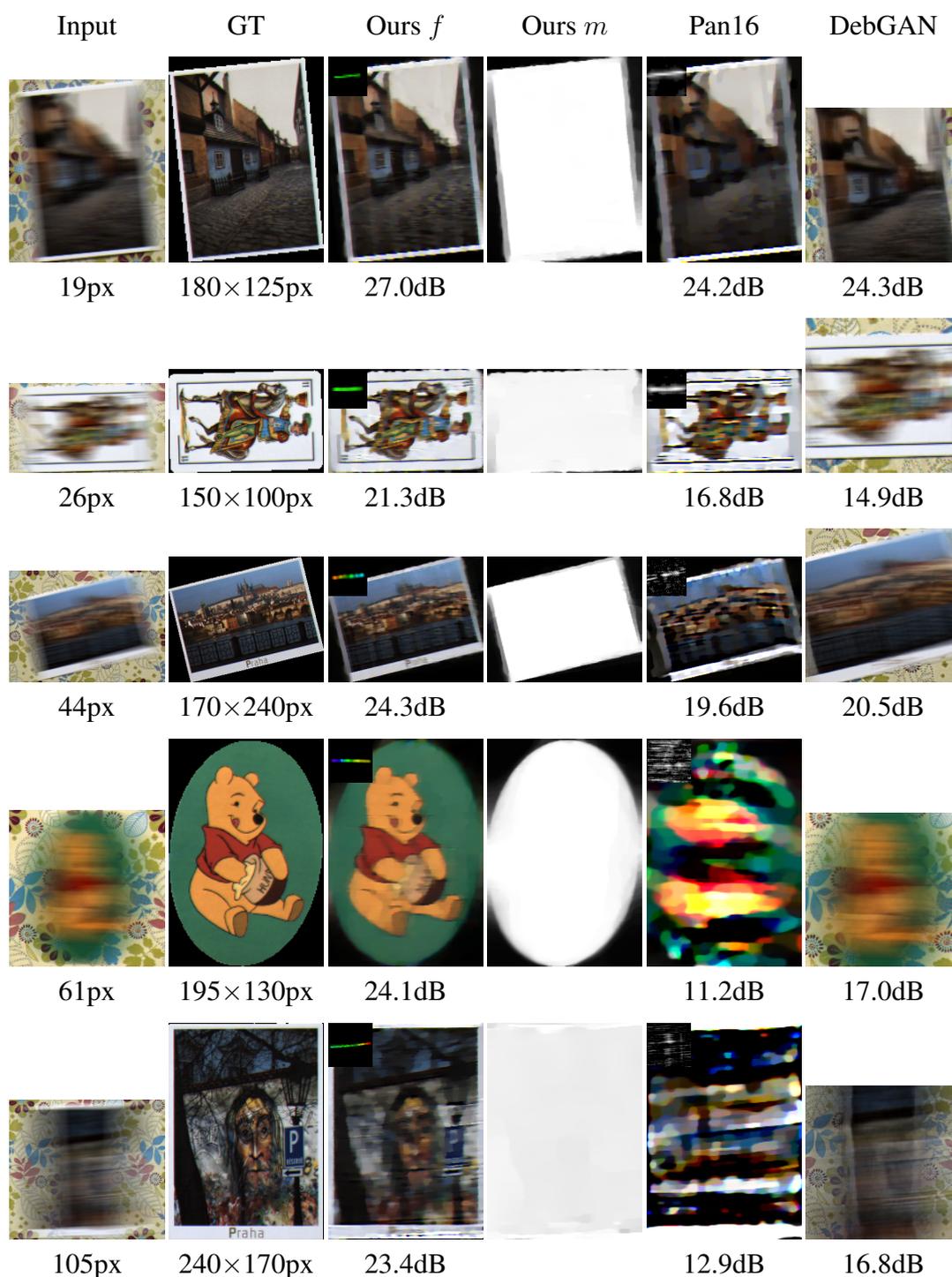


Figure 6.8: Examples of input images from the test dataset and deblurring results of the evaluated methods with corresponding restoration PSNR below (higher is better). The two leftmost columns show the input image with the motion blur length below (increasing from top to bottom) and the ground-truth foreground image with its size below. Results of Pan16 and ours include estimated blur in top left. The color in the blur PSF of the presented method indicates estimated rotation.

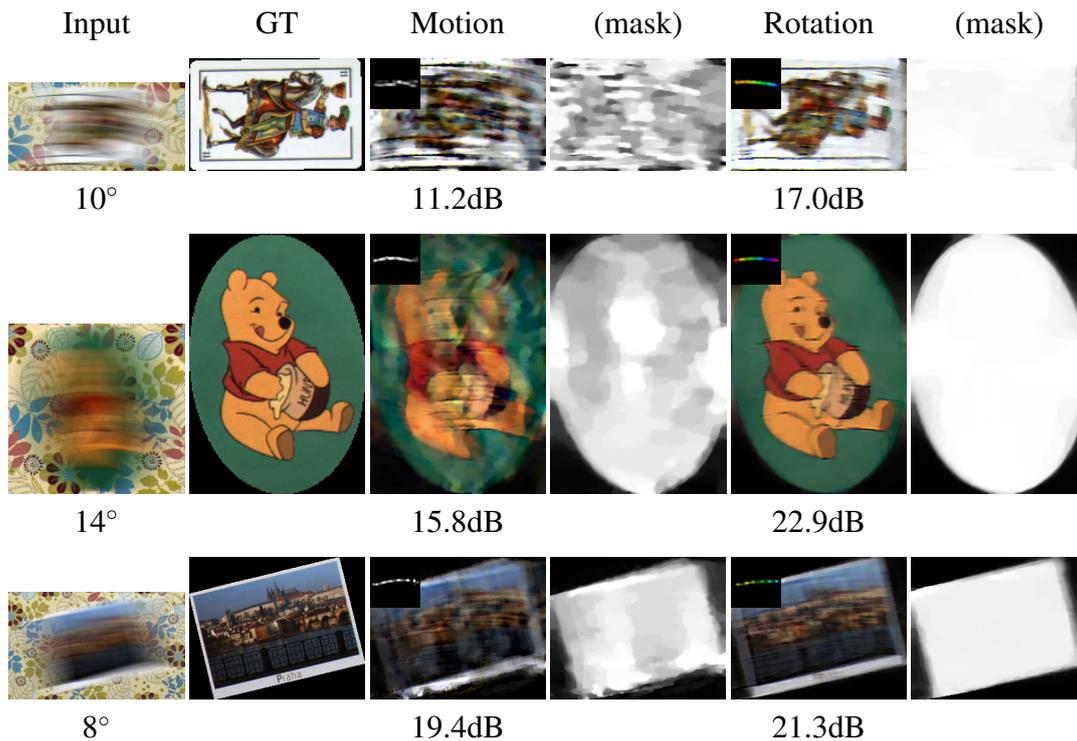


Figure 6.9: Comparison of blur model with translational motion only and the full model with both translation and rotation. From left to right: input with the amount of rotation present indicated below, ground-truth image, result of the pure-translation model with PSNR and corresponding mask, result of the full model and corresponding mask. The blur caused by rotation of the object is small compared to the translation blur, yet the performance is significantly worse when rotation is neglected. Color in the blur PSF indicates estimated rotation angle.

parametric polynomial, which corresponds to time-parametrized motion trajectory of the object. This approximate approach cannot capture for example arbitrary change of speed of the object, but it is nevertheless sufficient for many scenarios because significant change of speed during the short exposure time is rare. In Fig. 6.10 we show the motion discretized into four “video frames”.

Qualitative results on real-world images

Examples of results obtained with the presented method in fully blind, uncontrolled real-world setting without known ground truth are in Fig. 6.11. The left column contains long exposure pictures of cars driving by and the two right columns contain restored images of the cars and corresponding masks. The images suffer from deblurring artifacts but nevertheless bring undeniable value over the inputs, for example when attempting to recognize the make and model of the cars. Another similar example is in Fig. 6.12 containing images of badminton shuttlecock and volleyball from the dataset by Kotera *et al.* [82] that contains videos of small fast moving objects captured with standard and high-speed cameras. The objects undergo 3D rotation, which is not included in our model, yet the restoration and motion estimation are successful if the

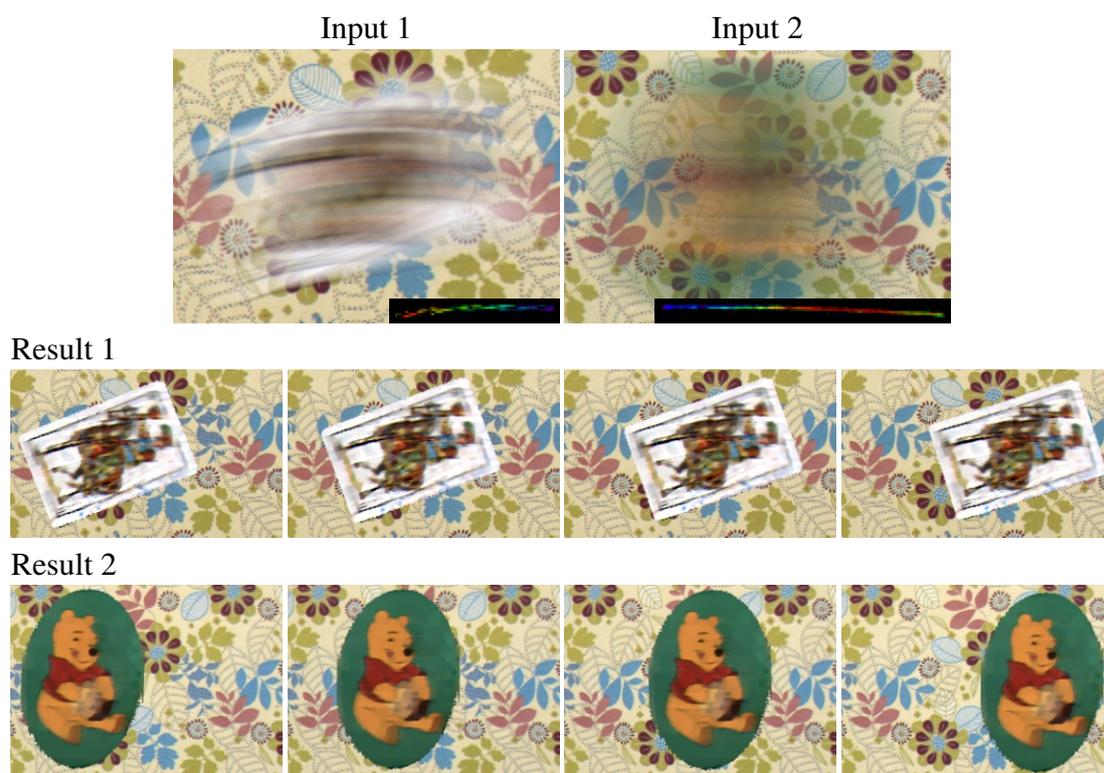


Figure 6.10: Temporal super-resolution of images in the test dataset. Top row: Different input images with estimated blur PSFs (including color-coded rotation). Second and third row: Result of deblurring of the particular input displayed as if the motion of the object were captured by a high-speed video camera (only 4 frames shown due to size).

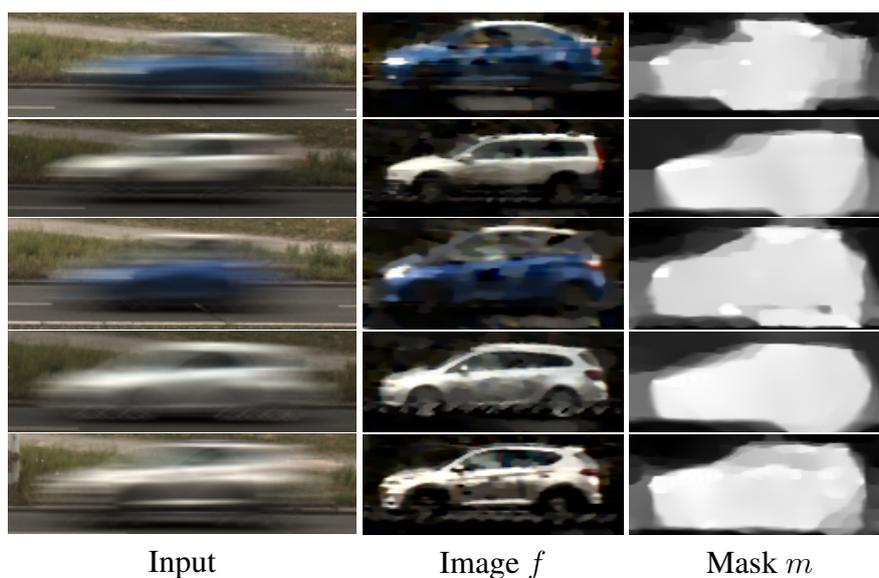


Figure 6.11: Motion deblurring results in real-world setting. From left to right: input image, estimated appearance f , and shape mask m .

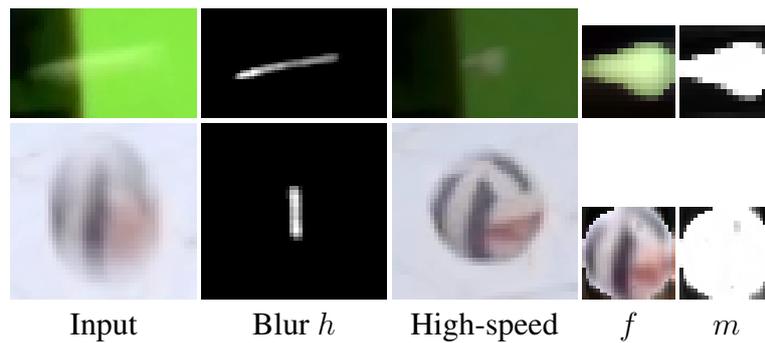


Figure 6.12: Further deblurring examples of real-world objects. From left to right: the input image, estimated blur h , high-speed camera frame corresponding to the input, estimated appearance f , and shape mask m .

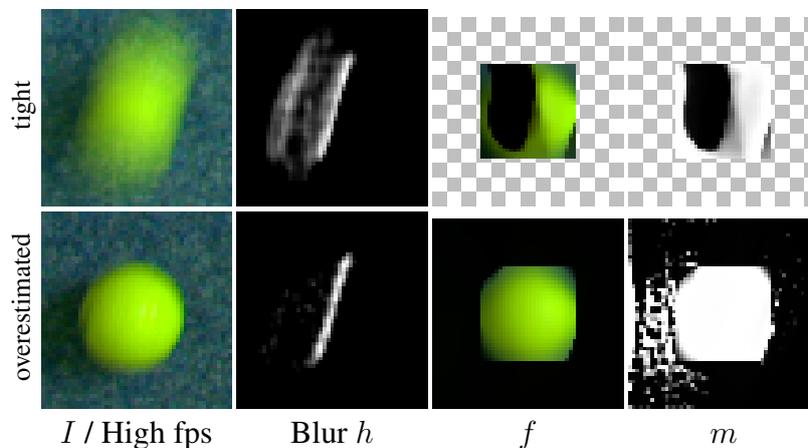


Figure 6.13: Object shadow and blur estimation. Input image with a ball casting a shadow is in top-left, corresponding high-speed camera frame in bottom-left. **Top:** The size of f is set too small and the shadow causes artifacts in blur estimation. **Bottom:** Size of f is overestimated so that the mask m can compensate for the shadow, the blur is then estimated correctly.

angular velocity is small compared to the camera shutter speed.

Interesting phenomenon relevant to the topic of the next chapter occurs in Fig. 6.13, where the object in the top-left image is rolling on the ground and casts a shadow. If the prescribed size of the estimated appearance f is too small, the shadow presents a violation of the blur model and the estimated blur contains artifacts (see Fig. 6.13-top). By increasing the size of f and m the shadow can effectively become a part of the object and the blur is estimated correctly (Fig. 6.13-bottom).

As a final demonstration of the efficacy of the presented method we offer the results of our tentative venture into the 3D world in Fig. 6.14, where the goal was restoration of a blurred volleyball undergoing 3D rotation. The deblurring method had to be suitably modified to be able to handle 3D rotation and in certain aspects simplified – in this case we assumed that the object shape is known to be a ball and the radius can be determined by some morphological operations and does not have to be estimated as part of the inverse problem. We do not present all the technical details of the necessary modifications since most of them have implementational character. The simplified task

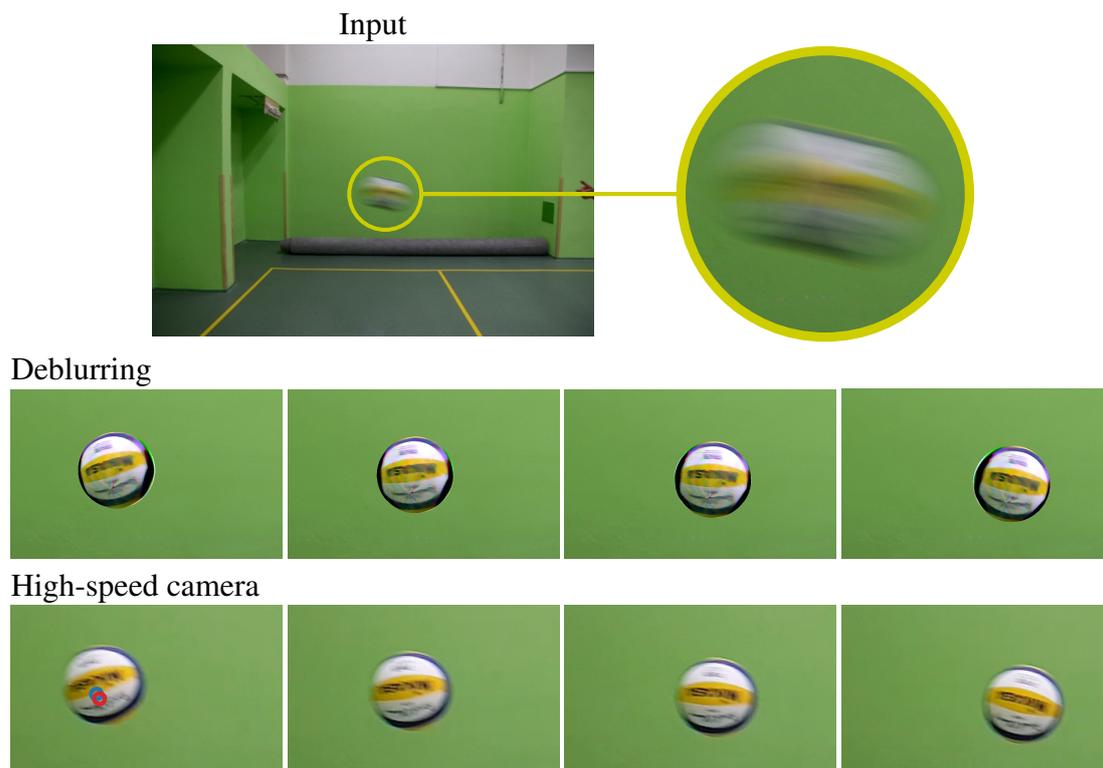


Figure 6.14: Temporal super-resolution of video frame with object blurred by motion and 3D rotation. **Top:** Input frame (and closeup) from a sequence captured with a regular video camera (25fps). **Middle:** Result of deblurring, the input frame re-synthesized to simulate video camera with $10\times$ higher frame-rate. **Bottom:** Real footage from a high-speed camera with 250fps. In the bottom left the circles indicate position of the 3D axis of rotation (with respect to the center of the ball) estimated from the high-speed camera (blue) and the blurred input (red).

is conceptually similar to the presented 2D problem because rendering of a sphere is easy regardless of its pose, so we did not have to deal with the problem of rendering an arbitrary 3D shape that at the same time has to be estimated. Similarly to Fig. 6.10 we present the results in the form of a temporal super-resolution of the input frame. In the top of Fig. 6.14 we see the input blurred image (including closeup on the blurred ball) and below the result of the temporal super-resolution – a sequence of 4 frames simulating that the input were obtained by a video camera with $10\times$ higher shutter speed (we only show 4 of the 10 resulting frames so that the images are not too small). In the bottom row we see an actual real footage of a high-speed video camera with $10\times$ higher framerate for comparison (necessarily shot from a slightly different vantage point). The restoration results are very good, though this task is admittedly slightly easier than the previously presented fully blind deblurring due to the known and simple shape of the object. Thanks to the 3D modelling of the motion we are able to determine the axis of rotation of the ball – purely from the blurred input frame in the top of Fig. 6.14 – which we indicated by a red circle in the bottom left high-speed image. We do not know the ground-truth rotation, but we estimated the axis from the high-speed camera images and the result is very close (indicated by blue circle in the same image).

6.5 Conclusion

We presented a new problem called *FMO deblurring* (deblurring of fast moving objects) that consists of estimating shape, appearance, and motion of a foreground object from an input image in which the object is motion-blurred in front of a known and still background. We showed that although the problem is similar to standard blind image deblurring, it differs in several important aspects and consequently cannot be solved by existing blind deblurring methods.

The principal difference from blind image deblurring lies in the necessity of estimating the object shape and the shape-appearance coupling in the acquisition model. FMO deblurring can be viewed as a combination of image deblurring and image matting, which is why it is sometimes called *deblatting*, and inherits some undesirable properties related to solution ambiguity from both of these inverse problems. We stated several necessary conditions for solution uniqueness and defined some theoretical limitations of FMO deblurring that were demonstrated on synthetic experiments.

We derived and presented an effective method for solving the FMO deblurring problem with object motion limited to arbitrary 2D translation and rotation. The solution is based on real-valued relaxation of the binary shape representation, for which we provided good theoretical reasons, and reformulation of the problem as a functional minimization with suitable constraint relating the object shape and appearance.

Performance of the presented method was experimentally validated on a new dataset of real blurred images and favorably compared to other blind image deblurring methods. We also presented results of successful deblurring of several real-world 3D objects, provided that the motion and rotation of the objects are confined to 2D or otherwise suitably constrained.

The biggest practical shortcoming of the presented approach is its limitation to 2D objects and motions. Unfortunately, extension to 3D is not straightforward precisely due to the necessity to estimate the object shape. The real-valued representation of the object shape that greatly facilitates shape estimation in 2D cannot be readily transferred to 3D. The mathematical formulation and solution must be suitably revised, which is the subject of our ongoing work.

Chapter 7

Tracking by deblatting

In the second chapter devoted to fast moving objects (FMO) we will explore how image deblurring can be utilized in the task of visual object tracking, in particular determining object location and pose with sub-frame frequency. This chapter will be less theoretical and more practically oriented than the previous one. We will focus mainly on balls and ball-like objects in flight and make adjustments to the previously presented solution to FMO deblurring to make it more suitable for the task of object tracking.

The goal of visual tracking is to continually determine the location of a predefined object in each frame of a video sequence, in which the object is at uninterrupted motion (to distinguish from e.g. re-identification task). A typical example might be tracking of the ball in a video recording of a soccer match. The premise of such task is that the motion of the object of interest can be regarded as a quasistatic process – the object is at motion throughout the whole sequence but in each frame it is static enough that we can pinpoint its exact location in the frame. In other words, the motion of the object must be slow enough that the trajectory can be adequately represented by sampling the object position at the framerate of the video sequence. It was shown by Rozumnyi *et al.* [150] that standard benchmark databases for visual object tracking are mostly limited to slow objects and performance of non-specialized trackers plummets when faced with the task of tracking an object with significant intra-frame motion. Note that even seemingly fast objects can actually be slow from the perspective of the tracker, if they are followed by the camera. This contributes to the illusion that we are tracking e.g. a very fast car or a ski jumper when in reality its position with respect to the image coordinates hardly changes from frame to frame.

Our interest in fast moving objects stems precisely from the fact that motion and consequently blur are integral parts of their perceived appearance. Their defining characteristic is that they travel non-negligible distance during the exposure time of a single video frame. Their location in the frame is therefore not even well defined, they have to be localized in terms of their intra-frame trajectory. Since motion blur is directly related to the object motion, the pivotal idea of *tracking by deblatting* (TbD) is to use a suitably modified algorithm for FMO deblurring presented in the previous chapter to determine the motion blur of the object image in the video frame and interpret the blur as a continuous object intra-frame motion. Recall that the term deblatting is an alternative name for FMO deblurring and is derived from the terms “deblurring” and “matting”. We respect the terminology under which this technique appears in literature

for tracking-related tasks and use the term *deblatting* in this chapter, where the primary interest is in determining the object trajectory, and appearance estimation is used as a means to this goal.

Object tracking is a very complex task and most algorithms, even those based on one key principle, are a conglomerate of many processing or decision steps to achieve maximum performance on the test datasets. Tracking by deblatting is no exception. Since the primary topic of this thesis is blind image deblurring in its various forms and we want to focus on the central principles and problems of TbD from the image deblurring standpoint, we will skim over the details of the tracking pipeline that are not directly related to FMO deblurring, and merely provide references to particular sources, similarly for applications of TbD where deblurring does not play any essential role.

The rest of the chapter is organized as follows. First we provide a general high-level overview of the tracking by deblatting method. Then we present in full detail the modifications to the FMO deblurring algorithm from the preceding Chapter 6 as it is used in TbD and show results of object tracking in pseudo-2D setting. In the final sections we extend the deblatting to partially address 3D objects and environments and present experimental results. The chapter concludes with a discussion.

7.1 General overview

At its core, tracking by deblatting works as follows:

1. For each new video frame g containing the object of interest obtain the background image b (image identical to g but without the tracked object) and approximate location of the tracked object. If available, provide also approximate object appearance model (f, m) , where f is the object appearance and m is the object shape mask.
2. Process the preselected region of interest of the frame g with the deblatting algorithm to determine the updated object appearance f , shape m , and mainly the motion blur h .
3. Convert the blur to a time-parametrized motion trajectory by fitting a piecewise polynomial curve c to the nonzero values in the PSF of the blur h . The curve c represents the intra-frame trajectory of the tracked object and is the result of the tracking for frame g .
4. Use the newly estimated properties of the object and its trajectory to prepare for tracking in the next frame (e.g. update appearance model and extrapolate motion trajectory) and go to 1.

Our interest lies mainly in step 2, which will be covered in detail shortly, but let us now briefly go over the most important ingredients of the remaining steps.

Background image We know from Chapter 6 that FMO deblurring requires not only the image containing the blurred foreground object but also separately the image of the

background. In TbD, background is estimated as a temporal median of several previous frames, exploiting the fact that due to its motion the tracked object disappears under the median but the rest of the scene remains unaltered. This works only if the video camera is stationary, which is the case in the sequences in the test datasets. Otherwise, camera stabilization has to be used as part of preprocessing.

Domain restriction Only preselected region of the frame g is processed by the deblurring algorithm for two reasons. First, the iterative algorithm is rather slow to converge for large images, restricting the size of the input results in considerable speedup of the whole tracking pipeline. And second, the deblurring algorithm is based on the image formation model (6.1), which assumes that a single foreground object is at motion in a static scene. This assumption is naturally violated in many real-world video sequences containing many objects at motion or other kind of scene dynamics. By limiting the processing to only a small part of the frame where the object is predicted to appear, the negative impact of other objects at motion on accuracy of the blur estimation is diminished.

Trajectory fit When introducing temporal super-resolution of fast moving objects on page 114 of the previous chapter we already discussed that although motion blur encodes the motion trajectory of the object, it is entirely unstructured. Motion blur is merely a set of duration-weighted positions where the object appeared at any moment during the exposure, whereas motion trajectory is a mapping from the exposure time to the position. The relationship is analogous to that of a raster and a vector image. To establish the time order and convert the motion blur to a motion trajectory, we fit a parametric piecewise-polynomial curve to the non-zero values of the blur PSF and the curve then represents the time-parametrized motion trajectory. The fitting algorithm is described in detail in Kotera *et al.* [82].

This solution is not perfect, because the curve fit is based only on spatial approximation of the blur PSF by the curve, but does not take into account speed of the object, hidden in intensities of the blur PSF. For example, constant-speed linear motion and accelerated linear motion would result in the same inferred trajectory. Better solution is a subject of ongoing work. The choice of piecewise polynomial curves is given by the fact that TbD is primarily designed for tracking of balls and ball-like objects in sport videos and their trajectories are most often smooth arcs when in free flight or piecewise smooth when in contact with the players or surrounding environment, but piecewise polynomials are sufficiently general to describe nearly any naturally occurring intra-frame motion.

Motion prediction, failure recovery etc. Using the assumption of free-flight motion, the intra-frame trajectory is extrapolated to the next video frame in which an appropriate region of interest is selected and the tracking loop restarts. The estimated object appearance is used to update the maintained appearance model, which in turn is used to initialize or regularize appearance estimation in the next frame.

In the listed sequence of the main tracking steps, a lot of things can go wrong for various reasons. Deblurring can fail due to poor background estimation, trajectory

fitting may fail to identify the salient part of the blur PSF corresponding to the object motion, motion prediction can fail due to interference of the object with the scene – these are just a few examples. The presented barebone pipeline is therefore augmented by additional decision steps that perform failure detection and take appropriate action, for example search in an increased domain, redetection of the object using other means etc. We refer the reader to Kotera *et al.* [82] for more details and diagrams of the full tracking procedure.

7.2 Deblatting for 2D tracking

Let us now present the modifications to the general FMO deblurring problem defined in Chapter 6, which is mathematically formulated as the minimization problem (6.3). The primary scenario for which tracking by deblatting is designed is tracking of small objects, mostly balls, in sport videos (tennis, ping-pong, volleyball, etc.), because this is where fast moving objects appear most naturally. The presented solution to the deblatting problem is limited to motions restricted to a particular 2D plane, but balls in sports typically fly in all three directions and rotate freely. Although this is an obvious contradiction, we claim and demonstrate experimentally that despite its formal limitation, the presented solution is well usable for the purposes of motion blur estimation even for 3D objects, provided that during single frame exposure the perceived object motion can be approximated as two-dimensional; i.e. the 3D effects are small. In particular, we assume that during exposure time the inner 3D rotation of the object is imperceptible (e.g. nearly homogeneous ball) or negligible (e.g. dart or shuttlecock) and the same holds for the non-lateral component of the object motion (towards or away from the camera). Note that we are not restricting the motion of the object throughout the whole sequence but only within a single frame.

Following this assumption, we drop the support for 2D rotations and work with the convolution blur model (6.1). This is motivated primarily by reduction of computational burden, because the dimensionality of rotational blur that must be estimated is typically orders of magnitude higher than the dimension of pure translation, and modeling 2D rotation would not help if the potential rotation is almost surely 3D. On top of that, we did the following two modifications to the FMO deblurring problem as formulated in (6.3):

- Visual object tracking must be initialized by some identification of the object that is to be tracked. Most typically this is provided by user in terms of a bounding box of the object in the first frame of the video sequence. The deblatting must therefore be augmented by some notion of the initial object appearance extracted from the initialization f_0 . It is assumed that in the video frame used for initialization the tracked object is at rest.
- We noticed that finding globally optimal choice of the parameter α_h in (6.3), which determines the weight of the ℓ^1 regularization of the blur, can be problematic for sequences or datasets with varying contrast between the tracked object and the background. We therefore replaced the ℓ^1 regularization with the energy-conservation constraint $\sum_i h_i = 1$, which is universal for all sequences and works equally well.

The reformulated deblatting problem is (newly added terms highlighted)

$$\begin{aligned} \min_{h,f,m} \frac{1}{2} \|h * f - b \cdot (h * m) - (g - b)\|_2^2 + \frac{\lambda}{2} \|f - m \cdot f_0\|_2^2 + \alpha_f \|\nabla f\|_{2,1} \\ \text{s.t. } h \geq 0, \sum_i h_i = 1, (f, m) \in S. \end{aligned} \quad (7.1)$$

The derivative operator is again denoted by ∇ and recall from (6.5) that $S = \{(f, m) : m \in [0, 1], 0 \leq f \leq m\}$ is the feasibility set relating f and m ; see Fig. 6.3. The second, λ -weighted, term is a regularization of the estimated appearance f in the form of “template matching” to f_0 , which is the content of the user-provided initializing bounding box of the object. Since f_0 contains the object only loosely delimited but not precisely segmented from the background, it must be multiplied by the estimated shape mask m , which effectively segments the object from the background, before comparison with current appearance estimate f .

Solving (7.1) is done in exactly the same fashion as solving the original FMO deblurring problem (6.3) as described in Chapter 6, in particular Alg. 6.1 remains unchanged except for initialization (to be specified). We will therefore describe only differences arising from the introduced modifications.

Minimization with respect to (f, m)

The only modification to the procedure in Sec. 6.2 is related to the template-matching term with f_0 in (7.1). By inspection of the minimization steps (6.7)-(6.11) we see that only the step in eq. (6.9) needs to be revised as (new terms highlighted)

$$\begin{aligned} \bar{f} := \operatorname{argmin}_{f,m} \left(\frac{1}{2} \|[H \quad -BH] \bar{f} - (g - b)\|_2^2 + \lambda \|[I \quad -F_0] \bar{f}\|_2^2 \right. \\ \left. + \frac{\rho_1}{2} \|\bar{f} - z_1 + w_1\|_2^2 + \frac{\rho_2}{2} \|\nabla f - z_2 + w_2\|_2^2 \right), \end{aligned} \quad (7.2)$$

where $F_0 = \operatorname{diag}(f_0)$ performs elementwise multiplication with f_0 and recall that H is the convolution matrix performing convolution with h and $\bar{f} = [f^T, m^T]^T$ is the concatenation of the two unknowns. By differentiating (7.2) with respect to \bar{f} we see that the corresponding previous solution (6.15) needs to be updated as the linear system

$$\begin{aligned} \begin{bmatrix} H^T H + (\lambda + \rho_1)I + \rho_2 \nabla^T \nabla & -H^T B H - \lambda F_0 \\ -H^T B H - \lambda F_0 & H^T B^2 H + \lambda F_0^2 + \rho_1 I \end{bmatrix} \begin{bmatrix} f \\ m \end{bmatrix} \\ = \begin{bmatrix} H^T \\ -H^T B \end{bmatrix} (g - b) + \rho_1 (z_1 - w_1) + \begin{bmatrix} \rho_2 \nabla^T \\ 0 \end{bmatrix} (z_2 - w_2). \end{aligned} \quad (7.3)$$

Minimization with respect to h

Modification to the corresponding procedure in Sec. 6.2 is related to the removal of the ℓ^1 term and the new energy-conservation constraint. Only the update eq. (6.17) of the

auxiliary variable z needs to be revised as

$$z := \operatorname{argmin}_z \left(\frac{\rho}{2} \|h - z + w\|_2^2 \right) \quad \text{s.t. } z \geq 0, \sum_i z_i = 1. \quad (7.4)$$

Solution of this problem is projection of $h + w$ onto the *probability simplex* $z \geq 0$, $\sum_z z_i = 1$, for which we use the very fast algorithm [183].

Results and applications

To test the tracking capabilities of the TbD we collected or created two datasets of sequences containing fast moving objects. The first dataset is called simply *FMO dataset* and was published by Rozumnyi *et al.* [150], the second is called *TbD dataset* and was published by Kotera *et al.* [82]. The datasets contain frames with significant intra-frame motion as well as frames where the object is slow or at rest. Each frame is annotated with ground-truth object trajectory and size (manually or semi-automatically with manual correction). The TbD dataset was obtained using high-speed camera with nearly zero gap between exposure of consecutive frames (sometimes called “360° shutter”) and then averaged to simulate acquisition at standard framerate (for example, the high-speed video was shot at 300fps and then each non-overlapping 10 consecutive frames were averaged to get video corresponding to standard 30fps). The advantage of this procedure is that we also get a good estimate of the object appearance with little blur densely sampled throughout the sequence.

Graphical examples of the tracking on selected sequences from the two datasets are in Fig. 7.1. Each picture represents a selected frame from one sequence and each arrow corresponds to one frame of the sequence and indicates estimated trajectory of the object in that frame. The accuracy of trajectory estimation in each frame is measured using *trajectory intersection-over-union* (TIOU), which is a generalization of the traditional position accuracy measure IoU to trajectories and is defined as

$$\text{TIOU}(c, c^* | m^*) = \int \text{IoU}(m_{c(t)}^*, m_{c^*(t)}^*) dt, \quad (7.5)$$

where c is the estimated trajectory, c^* and m^* are ground-truth trajectory and object shape mask, respectively, and m_x denotes mask m placed at position x ; see Fig. 7.2 for illustration. TIOU can be regarded as the standard IoU averaged over the intra-frame trajectory. In Fig. 7.1 the TIOU values for each frame are indicated using the color of the corresponding arrow; the colormap is in the bottom row of the figure. For more detailed quantitative results including comparison with other tracking methods please see Kotera *et al.* [82] and its supplementary material.

As a next result we show the temporal super-resolution of two sequences from the TbD dataset, which simulates acquisition of the sequences at higher framerate (see page 114 of the previous chapter). The results are in Fig. 7.3. In the left column we see the original sequence (selected frames aggregated) and in the right column the result of temporal super-resolution. Naturally, only few of the simulated high-speed frames can be displayed (we show one per each frame displayed in the left). In the bottom right corner of the volleyball sequence we show a closeup on the ball in one of the sequence input frames (yellow outline in the left column) and result of deblurring for



Figure 7.1: Intra-frame trajectory estimation using TbD for selected sequences from the FMO dataset [150] and TbD dataset [82] (one sequence per picture, all evaluated frames are aggregated). Each arrow corresponds to one frame of the sequence and indicates estimated trajectory of the object in that frame, the color encodes TIoU accuracy to the ground truth. The TIoU colormap is in the bottom row (greener is better).

that frame (blue outline in the right column) in comparison with an actual frame from a high-speed camera (green). Given the quality of the input, in which the ball is hardly even visible, the restoration is quite good.

To stay on topic, we showed only those results of TbD that are directly related to image deblurring, but additional details and applications are available in the provided reference Kotera *et al.* [82] (including supplementary material), where the method is favorably compared to other tracking methods, such as the original FMO detector by Rozumnyi *et al.* [150] or a well-performing non-specialized tracker CSR-DCF by Lukežić *et al.* [103]. The presented version of TbD is formulated as causal sequential processing of each frame individually, but in Rozumnyi *et al.* [147] we use dynamic programming and the assumption of piecewise smooth motion throughout the whole sequence and show that by global processing of all the intra-frame trajectories esti-

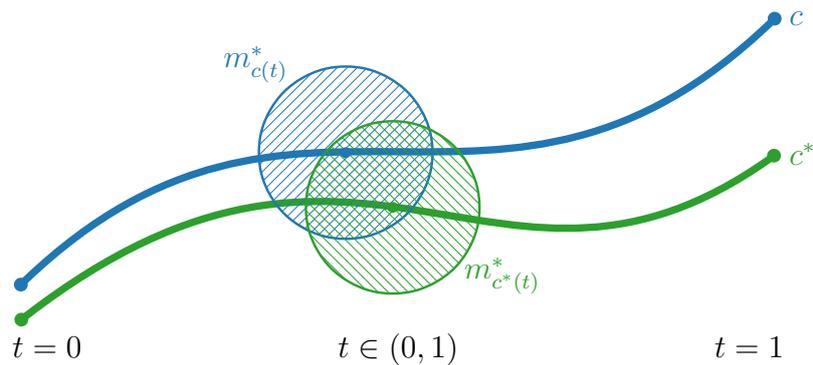


Figure 7.2: Illustration of the definition of the TIoU accuracy measure (7.5). The two compared curves (trajectories) c and c^* are t -parametrized curves $[0, 1] \rightarrow \mathbb{R}^2$ and for each t during the integration we evaluate the intersection over union (IoU) of the true shape m^* of the object placed at points corresponding to t of the two curves.

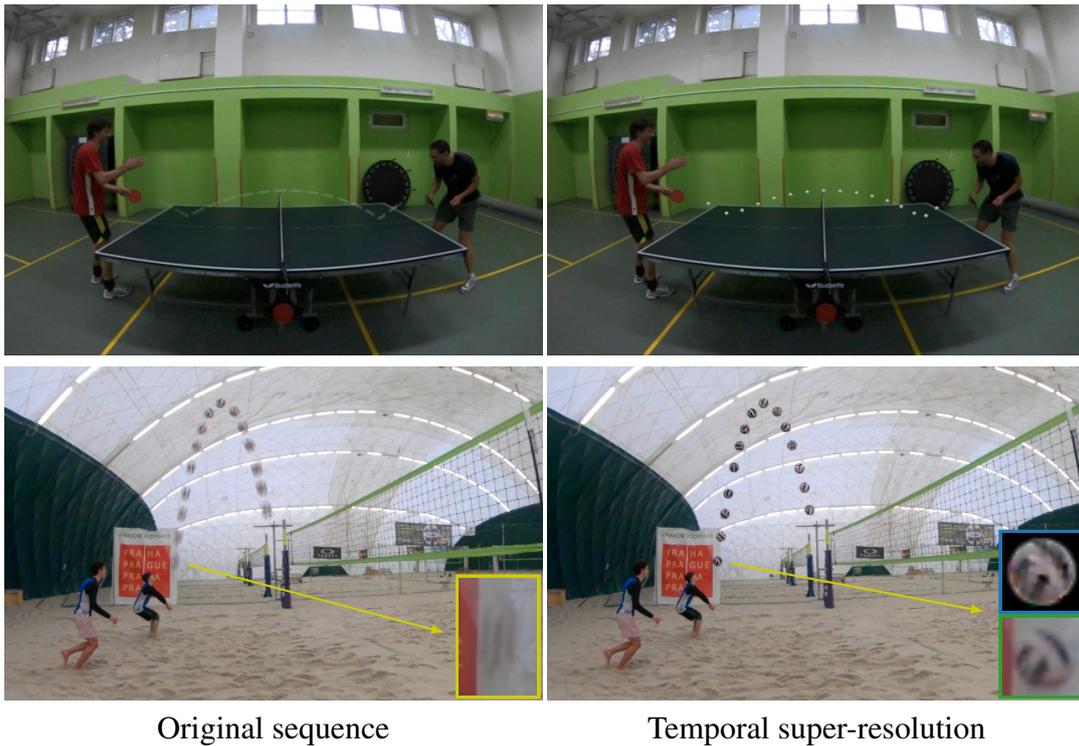


Figure 7.3: Temporal super-resolution (simulation of high-speed camera acquisition; see page 114) of two sequences from the TbD dataset. Left: Original sequence (only selected frames displayed). Right: Temporal super-resolution (selected frames). In the bottom-right corners of the pictures in the bottom row is a closeup on one of the input frames (yellow outline) and results of deblurring (blue) next to image from a true high-speed camera (green).

mated from individual frames their accuracy can be further improved. In the same work TbD is shown to be capable of estimating the speed of the object and even its size, when it is in free-flight curved by gravity.

7.3 Deblatting for 3D tracking

The original formulation of the deblatting problem (6.3) as well as the modified version intended for tracking (7.1) are based on the assumption that in each separate frame the perceived appearance f of the object is constant during exposure time. This implicitly limits intra-frame motion of the object to 2D plane perpendicular to camera axis, otherwise any 3D rotation or motion of the object generally changes its perceived appearance or size. The only exception was experiment in Fig. 6.14, in which the object was fully modeled in 3D. We would like to have the best of both worlds – allow 3D motion in a single frame without having to explicitly model the object in 3D, which is computationally prohibitive in tracking-related applications.

It turns out that it suffices to weaken the assumption of constant appearance and instead assume that the gradual transformation of the perceived object appearance during exposure can be approximated by a sequence of stepwise changes. In other words, the appearance as a function of time is assumed piecewise constant, with the points of discontinuity predefined and fixed. In its current formulation, TbD returns the average object appearance f and shape mask m for each frame. To get more densely sampled estimates we propose to partition the exposure interval T into subintervals T_i , $T = \bigcup T_i$, and estimate (f_i, m_i) corresponding to the time interval T_i with the approximating assumption that within this time interval, appearance and shape are constant.

We utilize the estimated sub-frame appearance and shape in two ways. First, from the change of the object size during exposure we estimate its relative distance to the camera (e.g. we complement the 2D trajectory estimated by TbD with the depth dimension). And second, from the change of the perceived appearance during exposure we estimate the 3D rotation (angular velocity) of the object. In its full generality this is quite nontrivial task, we therefore make a simplification and assume that the tracked object is a ball. Note that this approach is quite different from the 3D experiment in Fig. 6.14, in which the size of the ball was assumed known and the continuous 3D motion fully included in the blur model. In the present case we assume that the original TbD is sufficient to determine the 2D projection of the object trajectory and we then use the proposed extension to get more densely sampled sub-frame appearance snapshots and estimate the parameters of the 3D motion essentially by post-processing of these snapshots. We will now state the mathematical formulation of the problem and present its solution, show how depth and 3D rotation are calculated from the estimated appearance snapshots, and finally provide some experimental results.

Piecewise-constant appearance deblatting

The proposed modifications affect only the (f, m) -subproblem of the deblatting procedure, because it is assumed that the blur h (and consequently the intra-frame motion trajectory c) are provided by the above-presented 2D flavor of TbD and therefore known and fixed. As a natural extension, after estimating the sub-frame appearance f_i

and shape m_i one could re-run the h -step of the outer deblatting loop and refine the estimation of the motion blur as well.

The exposure time interval $T = [0, 1]$ is polychotomously partitioned into N sub-intervals $T_i, i = 1, \dots, N$ (we use isochronous partition $T_i = (\frac{i-1}{N}, \frac{i}{N})$), and the intra-frame motion trajectory c is correspondingly partitioned into sub-frame trajectories $\{c_i\}, c_i : T_i \rightarrow \mathbb{R}^2$. For each interval T_i we construct a motion blur h_i corresponding to trajectory c_i (i.e. the curve c_i is simply rendered as a PSF). With the assumption of constant appearance f_i and shape m_i within each time interval T_i the acquisition model (6.1) of the video frame g is modified accordingly as

$$g = \sum_i h_i * f_i + \left(1 - \sum_i h_i * m_i\right) \cdot b. \quad (7.6)$$

To facilitate achieving the goals set above we made two more modifications to the original (f, m) -subproblem (6.6). First, we enforce similarity of the appearance f_i to some prescribed appearance estimate f_i^0 (clarified later), and second, since the method is tailored for tracking of balls, we enforce rotational symmetry of the estimated shape masks m_i .

The optimization problem (6.6) updated for joint estimation of (f_i, m_i) for all trajectory segments c_i is

$$\begin{aligned} \min_{\{f_i, m_i\}} \frac{1}{2} \left\| \sum_i H_i f_i - B \sum_i H_i m_i - (g - b) \right\|_2^2 &+ \alpha_f \sum_i \|\nabla f_i\|_{2,1} + \frac{\lambda}{2} \sum_i \|f_i - f_i^0\|_2^2 \\ &+ \frac{\lambda_R}{2} \sum_i \|Rm_i - m_i\|_2^2 + \lambda_f \sum_i \|f_{i+1} - f_i\|_1 + \lambda_m \sum_i \|m_{i+1} - m_i\|_1 \\ &\text{s.t. } (f_i, m_i) \in S \quad \forall i, \end{aligned} \quad (7.7)$$

where R is linear operator performing rotational averaging (in practice it is sufficient to average over small, finite set of mutually coprime angles). The last two ℓ^1 terms are a form of *temporal total-variation* regularization, they postulate that the change of appearance and shape mask between two neighboring time intervals is small.

The problem (7.7) is minimized in the same fashion as the original (f, m) -subproblem (6.6) in Sec. 6.2, but due to the added temporal-TV terms we will use more auxiliary variables for the corresponding ADMM variable splitting (see App. 3.A for more information on ADMM). Let $f = [f_1^T, \dots, f_N^T]^T, m = [m_1^T, \dots, m_N^T]^T$ and again $\bar{f} = [f^T, m^T]^T$ denote the concatenation of all the primary unknowns. On top of the existing substitutions $z^{(1)} = \bar{f}$ and $z^{(2)} = \text{diag}(\nabla)f$ defined previously when solving (6.6) we introduce new substitutions $z^{(f)} = Df$ and $z^{(m)} = Dm$ where the block matrix D acts on the temporal components of f and m and performs simple forward differences, e.g.

$$Df = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ & & & \ddots & \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}. \quad (7.8)$$

For the sake of simplicity we will slightly abuse notation and use the subscript $(\cdot)_i$ to extract components related to i -th temporal segment T_i , for example $\bar{f}_i = [f_i^T, m_i^T]^T$ or $z_i^{(f)} = f_{i+1} - f_i$; similarly for other variables.

Analogously to solving (6.6), the minimization procedure of (7.7) consists of the iteration steps (iteration indices omitted)

$$z_i^{(1)} := \operatorname{argmin}_{z_i^{(1)}} \left(\frac{\rho_1}{2} \left\| \bar{f}_i - z_i^{(1)} + w_i^{(1)} \right\|_2^2 \right) \quad \text{s.t. } z_i^{(1)} \in S, \quad (7.9)$$

$$z_i^{(2)} := \operatorname{argmin}_{z_i^{(2)}} \left(\alpha_f \|z_i^{(2)}\|_{2,1} + \frac{\rho_2}{2} \left\| \nabla f_i - z_i^{(2)} + w_i^{(2)} \right\|_2^2 \right), \quad (7.10)$$

$$z_i^{(f)} := \operatorname{argmin}_{z_i^{(f)}} \left(\lambda_f \|z_i^{(f)}\|_1 + \frac{\rho_f}{2} \left\| f_{i+1} - f_i - z_i^{(f)} + w_i^{(f)} \right\|_2^2 \right), \quad (7.11)$$

$$z_i^{(m)} := \operatorname{argmin}_{z_i^{(m)}} \left(\lambda_m \|z_i^{(m)}\|_1 + \frac{\rho_m}{2} \left\| m_{i+1} - m_i - z_i^{(m)} + w_i^{(m)} \right\|_2^2 \right), \quad (7.12)$$

$$\begin{aligned} \bar{f} := \operatorname{argmin}_{f,m} & \left(\frac{1}{2} \left\| [H_1 \cdots H_N \quad -BH_1 \cdots -BH_N] \bar{f} - (g - b) \right\|_2^2 \right. \\ & + \frac{\lambda}{2} \|f - f^0\|_2^2 + \frac{\lambda_R}{2} \left\| \operatorname{diag}(R)m - m \right\|_2^2 \\ & + \frac{\rho_1}{2} \left\| \bar{f} - z^{(1)} + w^{(1)} \right\|_2^2 + \frac{\rho_2}{2} \left\| \operatorname{diag}(\nabla)f - z^{(2)} + w^{(2)} \right\|_2^2 \\ & \left. + \frac{\rho_f}{2} \left\| Df - z^{(f)} + w^{(f)} \right\|_2^2 + \frac{\rho_m}{2} \left\| Dm - z^{(m)} + w^{(m)} \right\|_2^2 \right), \end{aligned} \quad (7.13)$$

$$w^{(1)} := w^{(1)} + \bar{f} - z^{(1)}, \quad (7.14)$$

$$w_i^{(2)} := w_i^{(2)} + \nabla f_i - z_i^{(2)}, \quad (7.15)$$

$$w_i^{(f)} := w_i^{(f)} + f_{i+1} - f_i - z_i^{(f)}, \quad (7.16)$$

$$w_i^{(m)} := w_i^{(m)} + m_{i+1} - m_i - z_i^{(m)}. \quad (7.17)$$

The update steps (7.9) and (7.10) are solved separately for each i -th component and therefore have identical solutions to those derived when solving (6.6), namely (6.12) and (6.14), respectively. Similarly, (7.11) and (7.12) are solved per i -th component separately, $i = 1, \dots, N - 1$, and each case is an instance of scalar ℓ^1 proximal mapping in each pixel (see page 34) with solution in the form of soft-thresholding (3.18). In particular we get for (7.11) (analogously for (7.12); all operations are evaluated elementwise)

$$z_i^{(f)} = \operatorname{sign} \left(f_{i+1} - f_i + w_i^{(f)} \right) \max \left(\left| f_{i+1} - f_i + w_i^{(f)} \right| - \frac{\lambda_f}{\rho_f}, 0 \right). \quad (7.18)$$

The iteration step (7.13) is also structurally very similar to the previously derived solution (6.15) but due to multiple ‘‘channels’’ of \bar{f} and added regularization terms, the resulting system is substantially bulkier. Let us establish more succinct, though slightly abused, notation and define $H = [H_1, \dots, H_N]$ and $BH = [BH_1, \dots, BH_N]$ (analogously B^2H). By differentiating with respect to \bar{f} we get the optimality condition in

the form of a linear system for \bar{f} that can be written in block form as

$$\begin{aligned}
 & \begin{bmatrix} H^T H + (\rho_1 + \lambda)I + \rho_2 \nabla^T \nabla + \rho_f D^T D & -H^T B H \\ -H^T B H & H^T B^2 H + \rho_1 I + \rho_m D^T D + \lambda R \mathbf{R}_0^T \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} f \\ m \end{bmatrix} \\
 = & \begin{bmatrix} H^T \\ -H^T B \end{bmatrix} (g - b) + \rho_1 (z^{(1)} - w^{(1)}) + \begin{bmatrix} \rho_2 \nabla^T (z^{(2)} - w^{(2)}) + \rho_f D^T (z^{(f)} - w^{(f)}) + \lambda f^0 \\ \rho_m D^T (z^{(f)} - w^{(f)}) \end{bmatrix}, \tag{7.19}
 \end{aligned}$$

where $R_0 = R - I$ (difference of rotational averaging and identity) and we typeset in bold operators that denote diagonal blocks of appropriate size composed of the particular operator, e.g. $\nabla = \text{diag}(\nabla)$.

From given input, the piecewise-constant appearance deblatting is solved by repeatedly looping over (7.9)-(7.17) using the derived solutions for each update step. Unfortunately, the data term of the problem (first term in (7.7)) is generally under-determined. Unless the blur is very long (minimum required length increases with the size of the object and number of segments N), the system matrix $A^T A$ with $A = [H \quad -BH]$ is singular. The root cause is that pixels where the images of the object belonging to different trajectory segments intersect each other (this necessarily happens even for non-intersecting trajectories due to finite size of the object) contribute to the data-term only once per two (or more) intersecting images, which results in fewer equations than unknowns. This phenomenon was already discussed in ‘‘Image–mask ambiguity’’ on page 109 in the previous chapter. We mitigate this problem by the added regularizers and by adjusting the image template f^0 in a hierarchical manner. For desired number of segments $N = 2^K$ we calculate piecewise deblatting (7.7) for $k = 1, \dots, K$ and at each level prescribe f^0 as the corresponding f from the previous level. The process is detailed in Alg. 7.1.

Algorithm 7.1 Hierarchical piecewise-constant appearance deblatting

input: blurred image g , background b , appearance f , shape m , and trajectory c estimated by TbD, required number of trajectory segments 2^K

for $k=1, \dots, K$ **do**

 set $N = 2^k$

 split curve c into N isochronous curve segments c_i

 construct blurs h_i corresponding to c_i

 set image template $f^0 = [f_1^T, f_1^T, f_2^T, f_2^T, \dots, f_{N/2}^T, f_{N/2}^T]^T$

 find $\{f_i\}$ and $\{m_i\}$ by solving (7.13) by looping over (7.9)-(7.17)

end for

output: final appearance and shape estimates $\{f_i\}, \{m_i\}, i = 1, \dots, 2^K$

Depth and 3D rotation estimation

TbD provides the 2D projection of the object trajectory, it remains to contribute the relative distance from the camera. The distance is inversely proportional to the size of

the object, and since we work with the assumption that the tracked object is a ball, the relative distance can be calculated as an inverse of its diameter. Due to the regularizer enforcing rotational symmetry of the shape mask, the estimated mask m is a well defined disk. It is nevertheless more robust to infer the area of the disk than estimating its diameter directly and the area is proportional to the sum of the values in the mask. We therefore calculate the relative depth position d of the object from the estimated mask m as

$$d \propto \sqrt{\frac{\pi}{\|m\|_1}}. \quad (7.20)$$

Estimating the 3D rotation is more difficult. We represent 3D rotation in terms of the angular velocity of the object. Angular velocity $\omega \in \mathbb{R}^3$ is a vector whose direction defines the rotation axis orientation and whose magnitude is equal to the rotation angle per time unit. We calculate ω by analyzing how the perceived appearance of the object transforms between frames or between the sub-frame snapshots estimated by piecewise deblatting.

Let R_ω be an operator transforming a perceived 2D image of a ball by performing 3D rotation given by ω of a virtual 3D representation of the ball and reprojecting to 2D. More explicitly, if $f_2 = R_\omega f_1$, then f_2 is calculated by mapping the 2D image f_1 to a virtual 3D sphere using camera projection, rotating the sphere by ω , and projecting back onto the 2D image plane. We define the error of such transformation between two images as

$$E(f_1, f_2 | \omega) = \|R_\omega f_1 - f_2\|_1. \quad (7.21)$$

Since different parts of the ball are visible before and after the rotation, the sum in eq. (7.21) is evaluated only in some pre-defined image region visible in both images after arbitrary considered rotation (from a bounded set), so that errors corresponding to different rotations are mutually comparable.

Having recovered the object appearance f_1 and f_2 at two different video sequence times t_1 and t_2 , we can find the average angular velocity ω between t_1 and t_2 as the minimizer of the transformation error $E(f_1, f_2 | (t_2 - t_1)\omega)$. However, velocity estimation from just two restored images at close times is prone to errors, especially if either of the images is affected by restoration artifacts. We therefore postulate that angular velocity is locally constant in small time interval of the motion (which is physically nearly correct even in the long term if the ball is in free flight) and estimate angular velocity more robustly in a sliding-window manner from several restored images belonging to the corresponding time window.

Let f_1, \dots, f_n be a set of estimated ball appearances corresponding to video sequence times t_1, \dots, t_n ; a short time window of the whole sequence. We estimate the average angular velocity ω at this time window as follows. Let ω_{ij} be the minimizer of the transformation error from f_i to f_j and S_{ij} the inverse of the attained error (“score”):

$$\omega_{ij} = \operatorname{argmin}_\omega E(f_i, f_j | (t_j - t_i)\omega), \quad (7.22)$$

$$S_{ij} = \frac{1}{E(f_i, f_j | (t_j - t_i)\omega_{ij}) + \varepsilon}. \quad (7.23)$$

In other words, ω_{ij} is the vote of the corresponding image pair (f_i, f_j) for the true ω and S_{ij} is the confidence of such vote. We minimize (7.22) by searching the discretized space of all possible angular velocities.

Simple weighted average of ω_{ij} is a very non-robust estimate sensitive to outliers, e.g. situations when the appearance restoration fails. Instead we proceed with an approach similar to RANSAC. Based on the discretization step used in the minimization of (7.22), an inlier threshold ρ is defined as the maximum acceptable error in determining ω . We treat $\{\omega_{ij}\}$ as hypotheses for the final estimate ω and for each hypothesis calculate its consensus set C_{ij} as the set of all image pairs whose candidate for the angular velocity is ρ -neighborhood of ω_{ij} ,

$$C_{ij} = \{(k, l) : \|\omega_{kl} - \omega_{ij}\| \leq \rho\}. \quad (7.24)$$

The winning candidate ω_{pq} is the one with the best total score of its consensus set,

$$(p, q) = \underset{(i,j)}{\operatorname{argmax}} \sum_{(k,l) \in C_{ij}} S_{kl}. \quad (7.25)$$

The final estimate of angular velocity is then the weighted average of the votes in the consensus set of the winning candidate

$$\omega = \frac{\sum_{kl} S_{kl} \omega_{kl}}{\sum_{kl} S_{kl}}, \quad (k, l) \in C_{pq}. \quad (7.26)$$

Using piecewise deblatting enables us to have several higher quality (sharper) appearance estimates per frame instead of just one estimate of averaged appearance. As a result, both angular velocity and depth estimation are more accurate and densely sampled, because we can use smaller time windows and still have enough samples for robust estimation.

Results and applications

As in the case of 2D tracking by deblatting we created a dataset of sequences with fast moving objects, but unlike in the previous case, the new dataset contains objects with pronounced 3D motion and change of appearance during exposure. The dataset is called *TbD-3D dataset* and was published by Rozumnyi *et al.* [148], it contains nine sequences featuring three different balls with complex texture. The sequences were obtained in the same way as the TbD dataset using high-speed camera with 360° shutter and averaged to simulate acquisition at standard framerate. This way we could actually create sequences corresponding to 30fps, 60fps, and 120fps and test the method in differently demanding conditions, as motion blur decreases with framerate.

Using the high-speed footage with little to no blur, the dataset sequences are annotated with ground-truth 3D trajectory and angular velocity. The 2D trajectory is obtained the same way as for the TbD dataset – by automatic tracking and manual correction of the ball position in all frames of the high-speed sequence. The depth information and 3D angular velocity are calculated using the method presented above, but using the image from the high-speed sequence without any deblurring. We realize that using even a subset of the evaluated method to obtain the ground truth sounds suspicious, so we tested this approach in a setup with known 3D geometry from which the actual ground truth can be inferred by other means. A ball was rolled without slipping

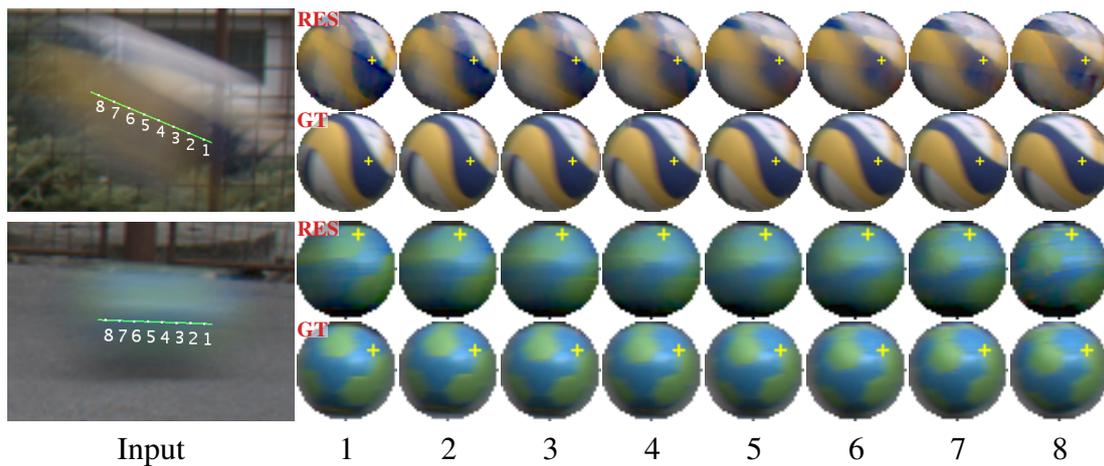


Figure 7.4: Two examples of sub-frame appearance estimation of the balls in the TbD-3D dataset [148]. **Left:** 30fps input images with overlaid 2D projections of the recovered trajectories in green. White points indicate midpoints of the sub-frame trajectories, for which the restoration on the right was calculated. **Right, top row:** Result of sub-frame appearance estimation (marked “RES”). **Right, bottom row:** High-speed camera frame (ground truth, marked “GT”). Estimated position of the rotation axis is marked by a yellow plus.

on a constrained path on the ground and filmed by the high-speed camera. We calculated its 3D position and rotation two ways – using the above-presented method and from the known geometry of the setup (the rotation is given by the travelled distance and circumference of the ball). The average angle between the estimated rotation axis using the proposed method and the ground-truth axis known from the scene geometry was approximately 4 degrees, which is less than half of the resolution of the angular velocity space discretization. The error between the estimated and ground-truth rotation velocity was 0.037 degrees, which corresponds to approx. 1.2% relative error in average. Getting more accurate ground truth of 3D rotation for objects in free flight without expensive dedicated hardware is highly nontrivial. An alternative approach is to use a 3D reconstruction method such as COLMAP by Schönberger and Frahm [152]. We tried several reconstruction and structure-from-motion methods and none of them were able to deal with small objects containing few features, even when applied to the images from the high-speed camera with negligible motion blur.

We show two examples of piecewise deblatting in Fig. 7.4. In the left part you can see the single low-speed input frame with indicated intra-frame ball trajectory estimated by TbD. The trajectory is split into eight sub-frame trajectories with their midpoints marked by dots and numbers 1 through 8. In the right part of Fig. 7.4 (top row) we show the result of sub-frame appearance restoration for these eight sub-trajectories (marked “RES”) and in each case the corresponding image from the high-speed camera is shown below (marked “GT”). The restoration suffers from deblurring artifacts caused primarily by the low rank of the system matrix due to huge overlap of the object images corresponding to adjacent sub-trajectories, as we discussed above. The added regularizers partially help and we are able to use the sub-frame snapshots for estimation of the ball 3D rotation – we marked the rotation axis direction (with

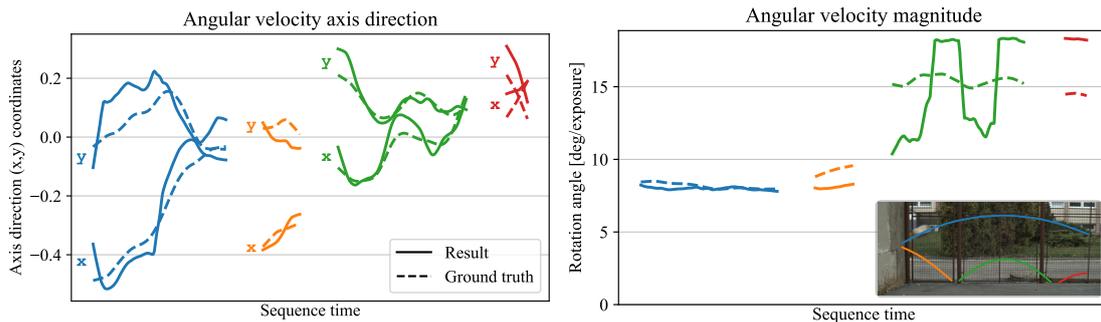


Figure 7.5: Axis and speed of rotation in different parts of the 30fps sequence out2 from the TbD-3D dataset [148]. **Left:** Estimated rotation axis direction (x, y) coordinates (unit vector); solid line is our result, dashed line is ground truth from the high-speed sequence. **Right:** Estimated speed of rotation in degrees per exposure; solid line is our result, dashed is ground truth. Different colors correspond to different parts of the sequence between ball bounces, see pictorial legend on the lower right.

respect to the ball center) by a yellow cross and especially in the top example (blue and yellow volleyball) the result is very accurate.

More detailed result of the rotation estimation for a whole sequence is in Fig. 7.5. In the left we show the (x, y) coordinates of the unit vector of the rotation axis, identified by a small x or y at the beginning of each run of values. Solid line represents our result and dashed line is ground truth from the high-speed sequence. Different colors correspond to different parts of the sequence, as shown in the legend of the right plot. The results are somewhat noisy but still relatively well correspond to the ground truth. In the right plot in Fig. 7.5 is the estimated rotation speed. Surprisingly, estimating the speed of rotation is not any easier than estimation the axis direction, the results of the green segment are very noisy and would apparently benefit from even more robust approach and post-filtering, but it can be seen, for example, that after bouncing from the ground the ball rotation accelerates.

Overall result of the 3D tracking throughout the whole sequence for three selected sequences from the TbD-3D dataset is shown in Fig. 7.6. In the left column is our result, in the right column is ground-truth 3D position. The distance of the ball from the camera is indicated by the thickness of the blue curve and it corresponds to the ground truth quite well. More detailed results of the presented TbD-3D method of both 3D tracking and rotation estimation are available in Rozumnyi *et al.* [148] (including supplementary material), we do not repeat them here in full as they are only marginally connected to image deblurring, which is the primary topic of the thesis.

7.4 Conclusion

In this chapter we focused on tracking of fast moving objects, a phenomenon widely overlooked by traditional trackers and benchmark tracking datasets. Our approach is based on two principle ideas. One, a fast moving object should be localized in terms of its intra-frame trajectory in each video frame, rather than its position, and two, the trajectory is inferable from its motion blur. We presented two different methods, *tracking by deblatting* (TbD) and its extension *tracking by deblatting-3D* (TbD-3D), designed

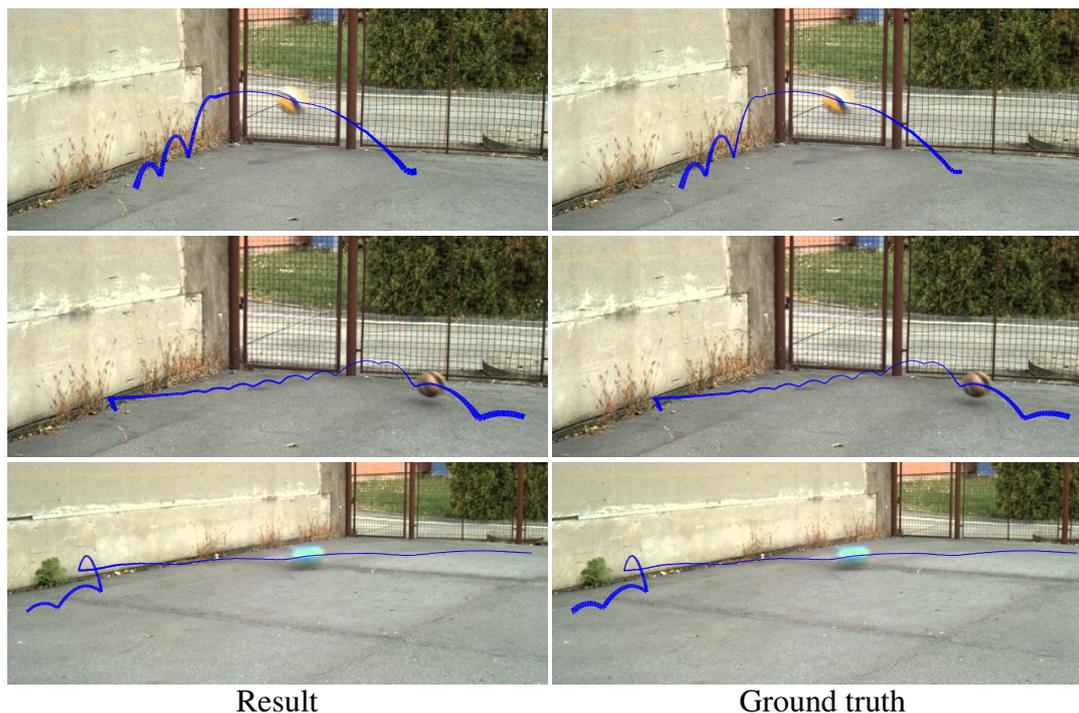


Figure 7.6: Graphical demonstration of the 3D tracking on three 30fps sequences from the TbD-3D dataset [148]. Line thickness indicates distance from the camera (depth). Left: Our result. Right: Ground-truth object position.

specifically for tracking of fast moving objects and based on the central *deblatting* operation, which is a joint process of motion deblurring and separation of the blurred foreground object from the background (matting). During the work on the topic we have published three datasets of video sequences with fast moving objects in different kinds of motion, annotated with ground-truth position and in the case of TbD-3D dataset also rotation information.

The original TbD method is based on a 2D approximation of the object and its motion. It assumes that in each separate frame of the video sequence, the perceived appearance and shape of the object are constant, which implicitly constrains the intra-frame motion of the object to a 2D plane and precludes 3D rotation. Though limited on paper, it was shown that the method performs well on real 3D sequences. Although the method consists of several processing blocks, the key step is the deblatting of pre-selected region of the input frame, which consists primarily of estimating the motion blur of the tracked object. The motion blur is then further processed and converted to time-parametrized intra-frame trajectory.

We have shown how the previously presented FMO deblurring problem can be modified to better suit the purposes of visual object tracking by addition of a template-matching term and adjustment of the regularization terms; corresponding algorithmic solution was presented in full detail with emphasis on the differences to the general FMO deblurring. Accuracy of the tracking procedure is measured using a proposed TIoU metric, which extends the standard intersection-over-union, and efficacy of the method was graphically demonstrated on several sequences from two different datasets. As an interesting application for which deblatting is essential we showed a

result of temporal super-resolution of a video sequence with standard framerate. For more detailed results and quantitative comparison with other trackers we encourage the reader to see Kotera *et al.* [82]. The basic tracking method can be extended by global processing of all detections in the sequence and its performance thus further improved – we refer the reader to Rozumnyi *et al.* [147] and [149] for details and results of other applications, such as speed or size estimation.

The limitations of TbD are partially lifted by the presented TbD-3D method, which extends TbD to 3D object motion and rotation, although it is limited to spherical objects. In particular, we modified and extended the non-blind deblatting step, which estimates appearance and shape for the given trajectory, and relaxed the initial assumption of globally constant appearance (per frame) to only piecewise-constant appearance. Unfortunately, the fully blind deblatting and trajectory fitting, which are necessary prerequisites of TbD-3D, are still calculated using the 2D TbD, so TbD-3D does not help in cases when the requirements of TbD are not met to such degree that TbD fails entirely. More general solution is naturally the topic of our ongoing work.

Despite this limitation, we have demonstrated that TbD-3D is a valuable extension of TbD and is able to provide more detailed information about the object motion in full six degrees of freedom. We have again presented how the appearance and shape estimation step of the original FMO deblurring is modified to work with sub-frame appearance and shape representation and derived the algorithmic solution in full detail. Since the resulting problem is under-determined, as follows from our previous discussion on the general FMO deblurring, we introduced several suitable regularizers and proposed a hierarchical optimization approach, so that each level can be appropriately regularized. The recovered sub-frame appearances are used to determine the relative distance of the object to the camera and its 3D angular velocity, for which we presented a robust estimation procedure based on minimization of the reprojection error. We showed results of both sub-frame appearance estimation and rotation estimation as well as graphical results of global tracking throughout the whole sequence. More detailed results are available in Rozumnyi *et al.* [148]. It can be seen that rotation estimation from blurred sequences is quite difficult and closely depends on success of the appearance recovery. The primary issue that needs to be addressed and improved upon in sub-frame appearance estimation is the discussed rank deficiency of the system matrix. The model of piecewise-constant appearance is convenient and presents relatively straightforward extension of the established FMO deblurring problem, but unfortunately its numerical properties are less than desirable.

Overall, the most notable weakness of tracking by deblatting is its over-reliance on the image formation model and quick tendency to fail if the model assumptions are not satisfied. In practice this occurs mainly in two cases: when other disruptive motion is present in the image region selected for deblatting by the TbD pipeline and when the object appearance changes significantly during single frame. As a result, in sport videos the tracked ball is often temporarily lost and needs to be redetected after having been in close contact with one of the players. Similarly, visual effects like shadows or illumination changes can be problematic. Despite these shortcomings, we believe that fast moving objects are an interesting research topic both for the image deblurring and visual object tracking community and that our initial and ongoing work will find its applications as well as ignite wider interest in the topic.

Chapter 8

Thesis conclusion

The thesis addressed some less frequently studied problems related to blind image deblurring in demanding conditions, in particular outlier handling in standard image deblurring and fast moving objects in dynamic scene deblurring.

In the first part we focused on standard blind image deblurring based on convolution blur model and presented methods of increasing its robustness to phenomena violating the linear acquisition model, such as intensity clipping caused by sensor saturation in overexposed pixels, occlusion in 3D scenes, local motion in the scene, dead pixels, and others. In Chapter 3 we formulated a deblurring method based on MAP blur estimation that identifies and removes overexposed pixels from the blur estimation. Drawback of this approach is that the erroneous pixels must be known a priori or identified by some external means, which is not trivial for other kinds of model violation. In Chapter 4 we generalized our approach and presented a “VB” method based on Bayesian formulation with spatially-variant noise prior and variational approximation of the posterior distribution of the unknown blur and sharp image, in which the model-violating pixels are identified automatically as part of the deblurring process. The VB approach is superior in all aspects except processing time, because it does not require any prior knowledge of the location of the erroneous regions or the physical cause of the model violation, and all primary parameters of the method are estimated automatically. All these aspects are hard-coded or user-specified in the MAP approach. The MAP method, on the other hand, uses very efficient numerical scheme in which the time consuming steps of the iterative procedure – the linear systems corresponding to blur inversion – are solved non-iteratively using Fourier transform. This is not possible in the VB approach, where conjugate gradient method is used to solve the resulting linear systems and the method is considerably slower. More detailed description of the properties of the methods is available in the conclusion section of the respective chapters.

In Chapter 5 we presented a method for assessment of blur estimation accuracy of blind deblurring methods. The derived formula evaluates error of the estimated blur PSF to the ground-truth PSF, it is simple to use and faithfully corresponds to the error of subsequent image restoration. The proposed method therefore provides better alternative to direct commonly used blur similarity measures, such as mean squared error of the blur PSFs, or performing blur estimation assessment after non-blind image restoration.

In the second part of the thesis we addressed the scenario of motion-blurred objects in an otherwise sharp image, a particular example of dynamic scene deblurring. In Chapter 6 we formulated the deblurring problem of recovering the object shape, appearance, and motion from the blurred input and studied its properties, solvability, and relationship to standard image deblurring. For a simplified case of the deblurring problem, in which the object motion is restricted to 2D plane (including rotation), we proposed a solution. In Chapter 7 we applied this method to the problem of visual tracking of fast moving objects in video sequences. Several suitable modifications were made to achieve better tracking performance as well as enable restoration of sharp sub-frame snapshots of the tracked object even if it slightly violates the assumption of 2D motion. Latest results indicate that pursuing this approach further should allow extending the method for fully 3D objects and motions. More detailed description of the properties of the presented methods is available in the conclusion section of the respective chapters.

All the presented deblurring methods have several common properties. They are based on Bayesian formulation in which the likelihood term is formulated using exact forward model of the assumed degradation process and the priors are relatively simple distributions that do not very faithfully approximate the distributions of natural images or motion blurs, but on the other hand allow efficient implementation. We demonstrated experimentally that the presented methods work well as they are but it is conceivable that their performance could be improved by some of the techniques reviewed in Chapter 2, such as learned image priors. Such extension of the proposed methods is part of our future work on the subject.

We hope that the thesis provided an interesting and informative read and will serve as a reference material for other works addressing similar topics.

Bibliography

- [1] Igor Aizenberg, Dmitriy Paliy, Claudio Moraga, and Jaakko Astola, “Blur identification using neural network for image restoration,” in *Computational Intelligence, Theory and Applications*, Bernd Reusch, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 441–455.
- [2] Raied Aljadaany, Dipan K. Pal, and Marios Savvides, “Douglas-Rachford networks: Learning both the image prior and data fidelity terms for blind image deconvolution,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 10 227–10 236.
- [3] Mariana S. C. Almeida and Mário Figueiredo, “Deconvolving images with unknown boundaries using the alternating direction method of multipliers,” *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3074–3086, Aug. 2013.
- [4] Mariana S. C. Almeida and Mário A. T. Figueiredo, “Blind image deblurring with unknown boundaries using the alternating direction method of multipliers,” in *2013 IEEE International Conference on Image Processing*, Sep. 2013, pp. 586–590.
- [5] Harry C. Andrews and Bobby Ray Hunt, “Digital image restoration,” 1977.
- [6] Saeed Anwar, Cong Phuoc Huynh, and Fatih Porikli, “Image deblurring with a class-specific prior,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2112–2130, Sep. 2019.
- [7] S. Derin Babacan, Rafael Molina, Minh N. Do, and Aggelos K. Katsaggelos, “Bayesian blind deconvolution with general sparse image priors,” in *Computer Vision – ECCV 2012*, Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 341–355.
- [8] S. Derin Babacan, Rafael Molina, and Aggelos K. Katsaggelos, “Parameter estimation in TV image restoration using variational distribution approximation,” *IEEE Transactions on Image Processing*, vol. 17, no. 3, pp. 326–339, Mar. 2008.
- [9] S. Derin Babacan, Rafael Molina, and Aggelos K. Katsaggelos, “Variational Bayesian blind deconvolution using a total variation prior,” *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 12–26, Jan. 2009.

-
- [10] Hicham Badri and Hussein Yahia, “Handling noise in image deconvolution with local/non-local priors,” in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct. 2014, pp. 2644–2648.
- [11] Yuval Bahat, Netalee Efrat, and Michal Irani, “Non-uniform blind deblurring by reblurring,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 3306–3314.
- [12] Christopher M. Bishop, *Pattern Recognition and Machine Learning*. Berlin, Heidelberg: Springer-Verlag, Jan. 2006.
- [13] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, Jan. 2011.
- [14] James P. Boyle and Richard L. Dykstra, “A method for finding projections onto the intersection of convex sets in Hilbert spaces,” in *Advances in Order Restricted Statistical Inference*, Richard Dykstra, Tim Robertson, and Farroll T. Wright, Eds., New York, NY: Springer New York, 1986, pp. 28–47.
- [15] Alon Brifman, Yaniv Romano, and Michael Elad, “Turning a denoiser into a super-resolver using plug and play priors,” in *2016 IEEE International Conference on Image Processing (ICIP)*, Sep. 2016, pp. 1404–1408.
- [16] Geoffrey J. Burton and Ian R. Moorhead, “Color and spatial structure in natural scenes,” *Applied Optics*, vol. 26, no. 1, pp. 157–170, Jan. 1987.
- [17] Jian-Feng Cai, Hui Ji, Chaoqiang Liu, and Zuowei Shen, “Blind motion deblurring from a single image using sparse approximation,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 104–111.
- [18] Jian-Feng Cai, Hui Ji, Chaoqiang Liu, and Zuowei Shen, “Framelet-based blind motion deblurring from a single image,” *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 562–572, Feb. 2012.
- [19] Patrizio Campisi and Karen Egiazarian, *Blind image deconvolution: Theory and applications*. CRC press, 2017.
- [20] Xiaochun Cao, Wenqi Ren, Wangmeng Zuo, Xiaojie Guo, and Hassan Foroosh, “Scene text deblurring using text-specific multiscale dictionaries,” *IEEE Transactions on Image Processing*, vol. 24, no. 4, pp. 1302–1314, Apr. 2015.
- [21] Ayan Chakrabarti, “A neural approach to blind motion deblurring,” in *Computer Vision – ECCV 2016*, Bastian Leibe, Jiří Matas, Nicu Sebe, and Max Welling, Eds., Cham: Springer International Publishing, 2016, pp. 221–235.
- [22] Stanley H. Chan, Xiran Wang, and Omar A. Elgendy, “Plug-and-play ADMM for image restoration: Fixed-point convergence and applications,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 84–98, Mar. 2017.
- [23] Tony F. Chan and Chiu-Kwong Wong, “Total variation blind deconvolution,” *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, Mar. 1998.

-
- [24] Giannis Chantas, Nikolaos Galatsanos, Aristidis Likas, and Michael Saunders, "Variational Bayesian image restoration based on a product of t -distributions image prior," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1795–1805, Oct. 2008.
- [25] Giannis Chantas, Nikolaos P. Galatsanos, Rafael Molina, and Aggelos K. Katsaggelos, "Variational Bayesian image restoration with a product of spatially weighted total variation image priors," *IEEE Transactions on Image Processing*, vol. 19, no. 2, pp. 351–362, Feb. 2010.
- [26] Feishe Chen, Lixin Shen, and Bruce W. Suter, "Computing the proximity operator of the ℓ^p norm with $0 < p < 1$," *IET Signal Processing*, vol. 10, no. 5, pp. 557–565, 2016.
- [27] Liang Chen, Faming Fang, Tingting Wang, and Guixu Zhang, "Blind image deblurring with local maximum gradient prior," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 1742–1750.
- [28] Zezhou Cheng, Matheus Gadelha, Subhransu Maji, and Daniel Sheldon, "A Bayesian perspective on the deep image prior," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 5438–5446.
- [29] Sunghyun Cho and Seungyong Lee, "Fast motion deblurring," *ACM Transactions on Graphics*, vol. 28, no. 5, pp. 1–8, Dec. 2009.
- [30] Sunghyun Cho and Seungyong Lee, "Convergence analysis of MAP based blur kernel estimation," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 4818–4826.
- [31] Sunghyun Cho, Jue Wang, and Seungyong Lee, "Handling outliers in non-blind image deconvolution," in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 495–502.
- [32] Jacqueline Christmas and Richard Everson, "Robust autoregression: Student-t innovations using variational Bayes," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 48–57, Jan. 2011.
- [33] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [34] Shengyang Dai and Ying Wu, "Estimating space-variant motion blur without deblurring," in *2008 15th IEEE International Conference on Image Processing*, Oct. 2008, pp. 661–664.
- [35] Shengyang Dai and Ying Wu, "Motion from blur," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2008, pp. 1–8.
- [36] Shengyang Dai, Ming Yang, Ying Wu, and Aggelos K. Katsaggelos, "Tracking motion-blurred targets in video," in *2006 International Conference on Image Processing*, Oct. 2006, pp. 2389–2392.

- [37] Aram Danielyan, Vladimir Katkovnik, and Karen Egiazarian, “BM3D frames and variational image deblurring,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1715–1728, Apr. 2012.
- [38] Jianwei Ding, Yongzhen Huang, Wei Liu, and Kaiqi Huang, “Severely blurred object tracking by learning deep image representations,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 2, pp. 319–331, Feb. 2016.
- [39] Chaoqun Dong, Sergio Barbero, and Javier Portilla, “Nonconvex Bayesian restoration of blurred foreground images,” in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 754–758.
- [40] Jiangxin Dong, Jinshan Pan, Zhixun Su, and Ming-Hsuan Yang, “Blind image deblurring with outlier handling,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2497–2505.
- [41] Jiangxin Dong, Jinshan Pan, Deqing Sun, Zhixun Su, and Ming Hsuan Yang, “Learning data terms for non-blind deblurring,” English, in *Computer Vision – ECCV 2018*, Vittorio Ferrari, Cristian Sminchisescu, Yair Weiss, and Martial Hebert, Eds., ser. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 15th European Conference on Computer Vision, ECCV 2018 ; Conference date: 08-09-2018 Through 14-09-2018, Germany: Springer Verlag, Jan. 2018, pp. 777–792.
- [42] Weisheng Dong, Peiyao Wang, Wotao Yin, Guangming Shi, Fangfang Wu, and Xiaotong Lu, “Denoising prior driven deep neural network for image restoration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 10, pp. 2305–2318, Oct. 2019.
- [43] Weisheng Dong, Lei Zhang, Guangming Shi, and Xin Li, “Nonlocally centralized sparse representation for image restoration,” *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1620–1630, Apr. 2013.
- [44] Wende Dong, Huajun Feng, Zhihai Xu, and Qi Li, “Blind image deconvolution using the Fields of Experts prior,” *Optics Communications*, vol. 285, no. 24, pp. 5051–5061, 2012.
- [45] Jonathan Eckstein and Dimitri P. Bertsekas, “On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators,” *Mathematical Programming*, vol. 55, no. 1, pp. 293–318, Apr. 1992.
- [46] Michael Elad and Michal Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [47] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman, “Removing camera shake from a single photograph,” *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 787–794, Jul. 2006.
- [48] Mário A. T. Figueiredo and Robert D. Nowak, “An EM algorithm for wavelet-based image restoration,” *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 906–916, Aug. 2003.

-
- [49] Freepik.com. (2020). “Image resources from freepik.com,” [Online]. Available: <https://www.freepik.com/> (visited on 04/01/2020).
- [50] Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia, “Dynamic scene deblurring with parameter selective sharing and nested skip connections,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 3843–3851.
- [51] Jochen Gast, Anita Sellent, and Stefan Roth, “Parametric object motion from blur,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 1846–1854.
- [52] Daniel Glasner, Shai Bagon, and Michal Irani, “Super-resolution from a single image,” in *2009 IEEE 12th International Conference on Computer Vision*, Sep. 2009, pp. 349–356.
- [53] Amit Goldstein and Raanan Fattal, “Blur-kernel estimation from spectral irregularities,” in *Computer Vision – ECCV 2012*, Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 622–635.
- [54] Dong Gong, Mingkui Tan, Yanning Zhang, Anton van den Hengel, and Qinfeng Shi, “Blind image deconvolution by automatic gradient activation,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 1827–1836.
- [55] Dong Gong, Mingkui Tan, Yanning Zhang, Anton van den Hengel, and Qinfeng Shi, “Self-paced kernel estimation for robust blind image deblurring,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 1670–1679.
- [56] Dong Gong, Jie Yang, Lingqiao Liu, *et al.*, “From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 3806–3815.
- [57] Dong Gong, Zhen Zhang, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, and Yanning Zhang, “Learning deep gradient descent optimization for image deconvolution,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2020.
- [58] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, *et al.*, “Generative adversarial nets,” in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014, pp. 2672–2680.
- [59] Izrail Solomonovich Gradshteyn and Iosif Moiseevich Ryzhik, *Table of integrals, series, and products*, Seventh, Alan Jeffrey and Daniel Zwillinger, Eds. Elsevier/Academic Press, Amsterdam, 2007, pp. xlviii+1171, Translated from the Russian, Translation edited and with a preface by Alan Jeffrey and Daniel Zwillinger.

- [60] Ankit Gupta, Neel Joshi, C. Lawrence Zitnick, Michael Cohen, and Brian Curless, “Single image deblurring using motion density functions,” in *Computer Vision – ECCV 2010*, Kostas Daniilidis, Petros Maragos, and Nikos Paragios, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 171–184.
- [61] Per Christian Hansen, James G. Nagy, and Dianne P. O’Leary, *Deblurring Images: Matrices, Spectra, and Filtering*. USA: Society for Industrial and Applied Mathematics, 2006.
- [62] James L Harris, “Image evaluation and restoration,” *Journal of the Optical Society of America*, vol. 56, no. 5, pp. 569–574, 1966.
- [63] Kaiming He, Jian Sun, and Xiaoou Tang, “Single image haze removal using dark channel prior,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.
- [64] Natalia Hidalgo-Gavira, Javier Mateos, Miguel Vega, Rafael Molina, and Aggelos K. Katsaggelos, “Variational Bayesian blind color deconvolution of histopathological images,” *IEEE Transactions on Image Processing*, vol. 29, pp. 2026–2036, 2020.
- [65] Michael Hirsch, Christian J. Schuler, Stefan Harmeling, and Bernhard Schölkopf, “Fast removal of non-uniform camera shake,” in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 463–470.
- [66] Alain Horé and Djemel Ziou, “Image quality metrics: PSNR vs. SSIM,” in *2010 20th International Conference on Pattern Recognition*, Aug. 2010, pp. 2366–2369.
- [67] Michal Hradiš, Jan Kotera, Pavel Zemčík, and Filip Šroubek, “Convolutional neural networks for direct text deblurring,” in *Proceedings of the British Machine Vision Conference (BMVC)*, Mark W. Jones Xianghua Xie and Gary K. L. Tam, Eds., BMVA Press, Sep. 2015, pp. 6.1–6.13.
- [68] Zhe Hu, Sunghyun Cho, Jue Wang, and Ming-Hsuan Yang, “Deblurring low-light images with light streaks,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 3382–3389.
- [69] Zhe Hu and Ming-Hsuan Yang, “Fast non-uniform deblurring using constrained camera pose subspace,” in *British Machine Vision Conference, BMVC 2012, Surrey, UK, September 3-7, 2012*, Richard Bowden, John P. Collomosse, and Krystian Mikolajczyk, Eds., BMVA Press, 2012, pp. 1–11.
- [70] Zhe Hu and Ming-Hsuan Yang, “Good regions to deblur,” in *Computer Vision – ECCV 2012*, Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 59–72.
- [71] Bobby R. Hunt, “The application of constrained least squares estimation to image restoration by digital computer,” *IEEE Transactions on Computers*, vol. C-22, no. 9, pp. 805–812, 1973.
- [72] Hui Ji and Kang Wang, “Robust image deblurring with an inaccurate blur kernel,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1624–1634, Apr. 2012.

- [73] Meiguang Jin, Zhe Hu, and Paolo Favaro, “Learning to extract flawless slow motion from blurry videos,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 8104–8113.
- [74] Meiguang Jin, Givi Meishvili, and Paolo Favaro, “Learning to extract a video sequence from a single motion-blurred image,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6334–6342.
- [75] Meiguang Jin, Stefan Roth, and Paolo Favaro, “Noise-blind image deblurring,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 3834–3842.
- [76] Aggelos K. Katsaggelos, Ed., *Digital Image Restoration*. Berlin, Heidelberg: Springer-Verlag, 1991.
- [77] Tae Hyun Kim, Byeongjoo Ahn, and Kyoung Mu Lee, “Dynamic scene deblurring,” in *2013 IEEE International Conference on Computer Vision*, Dec. 2013, pp. 3160–3167.
- [78] Tae Hyun Kim and Kyoung Mu Lee, “Segmentation-free dynamic scene deblurring,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 2766–2773.
- [79] Youngjung Kim, Hyungjoo Jung, Dongbo Min, and Kwanghoon Sohn, “Deeply aggregated alternating minimization for image restoration,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 284–292.
- [80] Thomas Köhler, Xiaolin Huang, Frank Schebesch, André Aichert, Andreas Maier, and Joachim Hornegger, “Robust multiframe super-resolution employing iteratively re-weighted minimization,” *IEEE Transactions on Computational Imaging*, vol. 2, no. 1, pp. 42–58, Mar. 2016.
- [81] Jan Kotera, Jiří Matas, and Filip Šroubek, “Restoration of fast moving objects,” *IEEE Transactions on Image Processing*, vol. 29, pp. 8577–8589, 2020.
- [82] Jan Kotera, Denys Rozumnyi, Filip Šroubek, and Jiří Matas, “Intra-frame object tracking by deblatting,” in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct. 2019.
- [83] Jan Kotera, Václav Šmídl, and Filip Šroubek, “Blind deconvolution with model discrepancies,” *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2533–2544, May 2017.
- [84] Jan Kotera and Filip Šroubek, “Blind deconvolution of images with model discrepancies using maximum a posteriori estimation with heavy-tailed priors,” in *Digital Photography XI*, Nitin Sampat, Radka Tezaur, and Dietmar Wüller, Eds., International Society for Optics and Photonics, vol. 9404, SPIE, 2015, pp. 64–75.
- [85] Jan Kotera and Filip Šroubek, “Motion estimation and deblurring of fast moving objects,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct. 2018, pp. 2860–2864.

-
- [86] Jan Kotera, Filip Šroubek, and Peyman Milanfar, “Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors,” in *Computer Analysis of Images and Patterns (CAIP)*, Richard Wilson, Edwin Hancock, Adrian Bors, and William Smith, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 59–66.
- [87] Jan Kotera, Barbara Zitová, and Filip Šroubek, “PSF accuracy measure for evaluation of blur estimation algorithms,” in *2015 IEEE International Conference on Image Processing (ICIP)*, Sep. 2015, pp. 2080–2084.
- [88] Dilip Krishnan, Terence Tay, and Rob Fergus, “Blind deconvolution using a normalized sparsity measure,” in *CVPR 2011*, Jun. 2011, pp. 233–240.
- [89] D. Kundur and D. Hatzinakos, “Blind image deconvolution,” *IEEE Signal Processing Magazine*, vol. 13, no. 3, pp. 43–64, 1996.
- [90] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas, “DeblurGAN: Blind motion deblurring using conditional adversarial networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 8183–8192.
- [91] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang, “DeblurGAN-v2: Deblurring (orders-of-magnitude) faster and better,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 8877–8886.
- [92] Wei-Sheng Lai, Jian-Jiun Ding, Yen-Yu Lin, and Yung-Yu Chuang, “Blur kernel estimation using normalized color-line priors,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 64–72.
- [93] Hyukzae Lee, Chanho Jung, and Changick Kim, “Blind deblurring of text images using a text-specific hybrid dictionary,” *IEEE Transactions on Image Processing*, vol. 29, pp. 710–723, 2020.
- [94] Anat Levin, Dani Lischinski, and Yair Weiss, “A closed-form solution to natural image matting,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 228–242, Feb. 2008.
- [95] Anat Levin, Yair Weiss, Fredo Durand, and William T. Freeman, “Efficient marginal likelihood optimization in blind deconvolution,” in *CVPR 2011*, Jun. 2011, pp. 2657–2664.
- [96] Anat Levin, Yair Weiss, Fredo Durand, and William T. Freeman, “Understanding blind deconvolution algorithms,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2354–2367, Dec. 2011.
- [97] Lerenhan Li, Jinshan Pan, Wei-Sheng Lai, Changxin Gao, Nong Sang, and Ming-Hsuan Yang, “Learning a discriminative prior for blind image deblurring,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6616–6625.
- [98] A. C. Likas and N. P. Galatsanos, “A variational approach for Bayesian blind image deconvolution,” *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2222–2233, Aug. 2004.

-
- [99] Renting Liu and Jiaya Jia, “Reducing boundary artifacts in image deconvolution,” in *2008 15th IEEE International Conference on Image Processing*, Oct. 2008, pp. 505–508.
- [100] Yang Liu, Jinshan Pan, Jimmy Ren, and Zhixun Su, “Learning deep priors for image dehazing,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 2492–2500.
- [101] Yuan-Yuan Liu, Lu-Yue Ye, Wen-Ze Shao, *et al.*, “Adversarial representation learning for dynamic scene deblurring: A simple, fast and robust approach,” in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 4644–4648.
- [102] Marina Ljubenović and Mário A. T. Figueiredo, “Blind image deblurring using class-adapted image priors,” in *2017 IEEE International Conference on Image Processing (ICIP)*, Sep. 2017, pp. 490–494.
- [103] Alan Lukežič, Tomáš Vojříř, Luka Cehovin Zajc, Jiří Matas, and Matej Kristan, “Discriminative correlation filter with channel and spatial reliability,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 4847–4856.
- [104] Liyan Ma, Lionel Moisan, Jian Yu, and Tiejiong Zeng, “A dictionary learning approach for Poisson image deblurring,” *IEEE Transactions on Medical Imaging*, vol. 32, no. 7, pp. 1277–1289, Jul. 2013.
- [105] David Pearce MacAdam, “Digital image restoration by constrained deconvolution,” *Journal of the Optical Society of America*, vol. 60, no. 12, pp. 1617–1627, 1970.
- [106] David J.C. MacKay, “Bayesian non-linear modelling for the prediction competition,” in *ASHRAE Transactions, V.100, Pt.2*, ASHRAE, 1994, pp. 1053–1062.
- [107] A Maréchal and K Dietzel, “Amélioration du contraste des détails des images photographiques par filtrage des frequences spatiales,” *Optica Acta*, vol. 5, pp. 256–262, 1958.
- [108] Yosra Marnissi, Yuling Zheng, Emilie Chouzenoux, and Jean-Christophe Pesquet, “A variational Bayesian approach for image restoration—application to image deblurring with Poisson–Gaussian noise,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 4, pp. 722–737, Dec. 2017.
- [109] Antonios Matakos, Sathish Ramani, and Jeffrey A. Fessler, “Accelerated edge-preserving image restoration without boundary artifacts,” *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 2019–2029, May 2013.
- [110] Tomer Michaeli and Michal Irani, “Blind deblurring using internal patch recurrence,” in *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 783–798.
- [111] James Miskin and David J. C. MacKay, “Ensemble learning for blind image separation and deconvolution,” in *Advances in Independent Component Analysis*, Mark Girolami, Ed. London: Springer London, 2000, pp. 123–141.

-
- [112] Rafael Molina, Javier Mateos, and Aggelos K. Katsaggelos, “Blind deconvolution using a variational approach to parameter, image, and blur estimation,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3715–3727, Dec. 2006.
- [113] Ali Mosleh, J. M. Pierre Langlois, and Paul Green, “Image deconvolution ringing artifact detection and removal via PSF frequency analysis,” in *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 247–262.
- [114] Ali Mosleh, Yasser Elmi Sola, Farzad Zargari, Emmanuel Onzon, and J. M. Pierre Langlois, “Explicit ringing removal in image deblurring,” *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 580–593, Feb. 2018.
- [115] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee, “Deep multi-scale convolutional neural network for dynamic scene deblurring,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 257–265.
- [116] Radford M. Neal, *Bayesian Learning for Neural Networks*. Berlin, Heidelberg: Springer-Verlag, 1996.
- [117] T. M. Nimisha, Akash Kumar Singh, and A. N. Rajagopalan, “Blur-invariant deep learning for blind-deblurring,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 4762–4770.
- [118] T.M. Nimisha, Vijay Rengarajan, and A. N. Rajagopalan, “Semi-supervised learning of camera motion from a blurred image,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct. 2018, pp. 803–807.
- [119] Mehdi Noroozi, Paramanand Chandramouli, and Paolo Favaro, “Motion deblurring in the wild,” in *Pattern Recognition*, Volker Roth and Thomas Vetter, Eds., Cham: Springer International Publishing, 2017, pp. 65–77.
- [120] Carl Olsson, Anders P. Eriksson, and Fredrik Kahl, “Solving large scale binary quadratic problems: Spectral methods vs. semidefinite programming,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2007, pp. 1–8.
- [121] Ido Omer and Michael Werman, “Color lines: Image specific color representation,” in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2, Jun. 2004, pp. II–II.
- [122] Jinshan Pan, Jiangxin Dong, Yu-Wing Tai, Zhixun Su, and Ming-Hsuan Yang, “Learning discriminative data fitting functions for blind image deblurring,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 1077–1085.
- [123] Jinshan Pan, Zhe Hu, Zhixun Su, Hsin-Ying Lee, and Ming-Hsuan Yang, “Soft-segmentation guided object motion deblurring,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 459–468.

-
- [124] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang, “ L_0 -regularized intensity and gradient prior for deblurring text images and beyond,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 342–355, Feb. 2017.
- [125] Jinshan Pan, Zhouchen Lin, Zhixun Su, and Ming-Hsuan Yang, “Robust kernel estimation with outliers handling for image deblurring,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 2800–2808.
- [126] Jinshan Pan, Risheng Liu, Zhixun Su, and Xianfeng Gu, “Kernel estimation from salient structure for robust motion deblurring,” *Signal Processing: Image Communication*, vol. 28, no. 9, pp. 1156–1170, 2013.
- [127] Jinshan Pan and Zhixun Su, “Fast ℓ^0 -regularized kernel estimation for robust motion deblurring,” *IEEE Signal Processing Letters*, vol. 20, no. 9, pp. 841–844, Sep. 2013.
- [128] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang, “Blind image deblurring using dark channel prior,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 1628–1636.
- [129] Liyuan Pan, Richard Hartley, Miaomiao Liu, and Yuchao Dai, “Phase-only image based kernel estimation for single image blind deblurring,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 6027–6036.
- [130] Daniele Perrone and Paolo Favaro, “Total variation blind deconvolution: The devil is in the details,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 2909–2916.
- [131] Daniele Perrone and Paolo Favaro, “A clearer picture of total variation blind deconvolution,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 6, pp. 1041–1055, Jun. 2016.
- [132] Daniele Perrone and Paolo Favaro, “A logarithmic image prior for blind deconvolution,” *International Journal of Computer Vision*, vol. 117, pp. 159–172, 2016.
- [133] Daniele Perrone, Avinash Ravichandran, René Vidal, and Paolo Favaro, “Image priors for image deblurring with uncertain blur,” in *British Machine Vision Conference, BMVC 2012, Surrey, UK, September 3-7, 2012*, Richard Bowden, John P. Collomosse, and Krystian Mikolajczyk, Eds., BMVA Press, 2012, pp. 1–11.
- [134] Gabriel Peyré, Sébastien Bougleux, and Laurent Cohen, “Non-local regularization of inverse problems,” in *Computer Vision – ECCV 2008*, David Forsyth, Philip Torr, and Andrew Zisserman, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 57–68.
- [135] Javier Portilla, “Image restoration through l0 analysis-based sparse optimization in tight frames,” in *2009 16th IEEE International Conference on Image Processing (ICIP)*, Nov. 2009, pp. 3909–3912.

- [136] Javier Portilla, “Maximum likelihood extension for non-circulant deconvolution,” in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct. 2014, pp. 4276–4279.
- [137] Kuldeep Purohit and A. N. Rajagopalan, “Efficient motion deblurring with feature transformation and spatial attention,” in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 4674–4678.
- [138] Kuldeep Purohit, Anshul Shah, and A. N. Rajagopalan, “Bringing alive blurred moments,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 6823–6832.
- [139] Congbin Qiao, Rynson W. H. Lau, Bin Sheng, Benxuan Zhang, and Enhua Wu, “Temporal coherence-based deblurring using non-uniform motion optimization,” *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4991–5004, Oct. 2017.
- [140] A.N. Rajagopalan and Rama Chellappa, Eds., *Motion Deblurring: Algorithms and Systems*. Cambridge University Press, 2014.
- [141] Sainandan Ramakrishnan, Shubham Pachori, Aalok Gangopadhyay, and Shanmuganathan Raman, “Deep generative filter for motion deblurring,” in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, Oct. 2017, pp. 2993–3000.
- [142] Dongwei Ren, Wangmeng Zuo, David Zhang, Jun Xu, and Lei Zhang, “Partial deconvolution with inaccurate blur kernel,” *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 511–524, Jan. 2018.
- [143] Wenqi Ren, Xiaochun Cao, Jinshan Pan, Xiaojie Guo, Wangmeng Zuo, and Ming-Hsuan Yang, “Image deblurring via enhanced low-rank prior,” *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3426–3437, Jul. 2016.
- [144] Wenqi Ren, Jiawei Zhang, Lin Ma, *et al.*, “Deep non-blind deconvolution via generalized low-rank approximation,” in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Curran Associates, Inc., 2018, pp. 297–307.
- [145] William Hadley Richardson, “Bayesian-based iterative method of image restoration,” *Journal of the Optical Society of America*, vol. 62, no. 1, pp. 55–59, Jan. 1972.
- [146] Yaniv Romano, Michael Elad, and Peyman Milanfar, “The little engine that could: Regularization by denoising (RED),” *SIAM Journal on Imaging Sciences*, vol. 10, no. 4, pp. 1804–1844, 2017. eprint: <https://doi.org/10.1137/16M1102884>.
- [147] Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas, “Non-causal tracking by deblatting,” in *Pattern Recognition, DAGM GCPR 2019*, Gernot A. Fink, Simone Frintrop, and Xiaoyi Jiang, Eds., Cham: Springer International Publishing, 2019, pp. 122–135.

- [148] Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas, “Sub-frame appearance and 6D pose estimation of fast moving objects,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 6777–6785.
- [149] Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas, “Tracking by deblatting,” *International Journal of Computer Vision*, 2020.
- [150] Denys Rozumnyi, Jan Kotera, Filip Šroubek, Lukáš Novotný, and Jiří Matas, “The world of fast moving objects,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 4838–4846.
- [151] Pablo Ruiz, Xu Zhou, Javier Mateos, Rafael Molina, and Aggelos K. Katsaggelos, “Variational Bayesian blind image deconvolution: A review,” *Digital Signal Processing*, vol. 47, pp. 116–127, 2015, Special Issue in Honour of William J. (Bill) Fitzgerald.
- [152] Johannes L. Schönberger and Jan-Michael Frahm, “Structure-from-motion revisited,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 4104–4113.
- [153] Christian J. Schuler, Harold Christopher Burger, Stefan Harmeling, and Bernhard Schölkopf, “A machine learning approach for non-blind image deconvolution,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2013, pp. 1067–1074.
- [154] Christian J. Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf, “Learning to deblur,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1439–1451, Jul. 2016.
- [155] Juan G. Serra, Javier Mateos, Rafael Molina, and Aggelos K. Katsaggelos, “Spike and slab variational inference for blind image deconvolution,” in *2017 IEEE International Conference on Image Processing (ICIP)*, Sep. 2017, pp. 3765–3769.
- [156] Jakub Ševčík, Václav Šmídl, and Filip Šroubek, “An adaptive correlated image prior for image restoration problems,” *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 1024–1028, Jul. 2018.
- [157] Qi Shan, Jiaya Jia, and Aseem Agarwala, “High-quality motion deblurring from a single image,” in *ACM SIGGRAPH 2008 Papers*, ser. SIGGRAPH ’08, Los Angeles, California: Association for Computing Machinery, 2008.
- [158] Eero P. Simoncelli, “Bayesian denoising of visual images in the wavelet domain,” in *Bayesian Inference in Wavelet-Based Models*, Peter Müller and Brani Vidakovic, Eds., New York, NY: Springer New York, 1999, pp. 291–308.
- [159] David Slepian, “Linear least-squares filtering of distorted images,” *Journal of the Optical Society of America*, vol. 57, no. 7, pp. 918–922, 1967.
- [160] Filip Šroubek, Václav Šmídl, and Jan Kotera, “Understanding image priors in blind deconvolution,” in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct. 2014, pp. 4492–4496.

-
- [161] T. G. Stockham, T. M. Cannon, and R. B. Ingebretsen, “Blind deconvolution through digital signal processing,” *Proceedings of the IEEE*, vol. 63, no. 4, pp. 678–692, 1975.
- [162] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce, “Learning a convolutional neural network for non-uniform motion blur removal,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 769–777.
- [163] Libin Sun, Sunghyun Cho, Jue Wang, and James Hays, “Edge-based blur kernel estimation using patch priors,” in *IEEE International Conference on Computational Photography (ICCP)*, Apr. 2013, pp. 1–8.
- [164] Yu-Wing Tai, Xiaogang Chen, Sunyeong Kim, *et al.*, “Nonlinear camera response functions and image deblurring: Theoretical analysis and practice,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2498–2512, Oct. 2013.
- [165] Jianwei Ding ; Kaiqi Huang ; Tieniu Tan, “Tracking blurred object with data-driven tracker,” in *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, Sep. 2012, pp. 331–336.
- [166] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia, “Scale-recurrent network for deep image deblurring,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 8174–8182.
- [167] Afonso M. Teodoro, José M. Bioucas-Dias, and Mário A. T. Figueiredo, “Image restoration and reconstruction using targeted plug-and-play priors,” *IEEE Transactions on Computational Imaging*, vol. 5, no. 4, pp. 675–686, Dec. 2019.
- [168] Radka Tezaur, Tetsuji Kamata, Li Hong, and Stephen S. Slonaker, “A system for estimating optics blur PSFs from test chart images,” in *Digital Photography XI*, Nitin Sampat, Radka Tezaur, and Dietmar Wüller, Eds., International Society for Optics and Photonics, vol. 9404, SPIE, 2015, pp. 84–93.
- [169] Ryan J Tibshirani, “Dykstra’s algorithm, ADMM, and coordinate descent: Connections, insights, and extensions,” in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., Curran Associates, Inc., 2017, pp. 517–528.
- [170] Tom Tirer and Raja Giryes, “Image restoration by iterative denoising and backward projections,” *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1220–1234, Mar. 2019.
- [171] Viet Hung Tran, *Copula variational Bayes inference via information geometry*, 2018. arXiv: 1803.10998 [cs.IT].
- [172] Jumpei Tsujiuchi, “Correction of optical images by compensation of aberrations and by spatial frequency filtering,” in *Progress in optics*, vol. 2, Elsevier, 1963, pp. 131–180.
- [173] Dimitris G. Tzikas, Aristidis C. Likas, and Nikolaos P. Galatsanos, “Variational Bayesian sparse kernel-based blind image deconvolution with Student’s-t priors,” *IEEE Transactions on Image Processing*, vol. 18, no. 4, pp. 753–764, Apr. 2009.

-
- [174] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, “Deep image prior,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 9446–9454.
- [175] V.S. Unni and Kunal N. Chaudhury, “Non-local patch-based regularization for image restoration,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct. 2018, pp. 1108–1112.
- [176] Subeesh Vasu, Venkatesh Reddy Maligireddy, and A. N. Rajagopalan, “Non-blind deblurring: Handling kernel uncertainty with CNNs,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 3272–3281.
- [177] Singanallur V. Venkatakrisnan, Charles A. Bouman, and Brendt Wohlberg, “Plug-and-play priors for model based reconstruction,” in *2013 IEEE Global Conference on Signal and Information Processing*, Dec. 2013, pp. 945–948.
- [178] Cédric Vonesch and Michael Unser, “A fast multilevel algorithm for wavelet-regularized image restoration,” *IEEE Transactions on Image Processing*, vol. 18, no. 3, pp. 509–523, Mar. 2009.
- [179] C. Wang, Y. Yue, F. Dong, *et al.*, “Enhancing Bayesian estimators for removing camera shake,” *Computer Graphics Forum*, vol. 32, no. 6, pp. 113–125, 2013. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12074>.
- [180] Peng Wang, Chunhua Shen, and Anton van den Hengel, “A fast semidefinite approach to solving binary quadratic problems,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2013, pp. 1312–1319.
- [181] Ruxin Wang and Dacheng Tao, *Recent progress in image deblurring*, 2014. arXiv: 1409.6838 [cs.CV].
- [182] Ruxin Wang and Dacheng Tao, “Training very deep CNNs for general non-blind deconvolution,” *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2897–2910, Jun. 2018.
- [183] Weiran Wang and Miguel Á. Carreira-Perpiñán, *Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application*, 2013. arXiv: 1309.1541 [cs.LG].
- [184] Zhou Wang, Alan C. Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [185] Oliver Whyte, Josef Sivic, and Andrew Zisserman, “Deblurring shaken and partially saturated images,” *International Journal of Computer Vision*, vol. 110, pp. 185–201, 2014.
- [186] Oliver Whyte, Josef Sivic, Andrew Zisserman, and Jean Ponce, “Non-uniform deblurring for shaken images,” *International Journal of Computer Vision*, vol. 98, no. 2, pp. 168–186, 2012.
- [187] David Wipf and Haichao Zhang, “Revisiting Bayesian blind deconvolution,” *Journal of Machine Learning Research*, vol. 15, no. 111, pp. 3775–3814, 2014.

-
- [188] Yi Wu, Haibin Ling, Jingyi Yu, Feng Li, Xue Mei, and Erkang Cheng, “Blurred target tracking by blur-driven tracker,” in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 1100–1107.
- [189] Lei Xiao, Felix Heide, Wolfgang Heidrich, Bernhard Schölkopf, and Michael Hirsch, “Discriminative transfer learning for general image restoration,” *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 4091–4104, Aug. 2018.
- [190] Xiaofen Xing, Nanhai Zhang, Kailing Guo, Chunmei Qing, Jiali Deng, and Huiping Qin, “Blurred target tracking based on sparse representation of online updated templates,” in *2016 10th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Jul. 2016, pp. 1–4.
- [191] Li Xu and Jiaya Jia, “Two-phase kernel estimation for robust motion deblurring,” in *Computer Vision – ECCV 2010*, Kostas Daniilidis, Petros Maragos, and Nikos Paragios, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 157–170.
- [192] Li Xu, Jimmy SJ Ren, Ce Liu, and Jiaya Jia, “Deep convolutional neural network for image deconvolution,” in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014, pp. 1790–1798.
- [193] Li Xu, Shicheng Zheng, and Jiaya Jia, “Unnatural L0 sparse representation for natural image deblurring,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2013, pp. 1107–1114.
- [194] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang, “Deep image matting,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 311–320.
- [195] Xiangyu Xu, Jinshan Pan, Yu-Jin Zhang, and Ming-Hsuan Yang, “Motion blur kernel estimation via deep learning,” *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 194–205, Jan. 2018.
- [196] Ruomei Yan and Ling Shao, “Blind image blur estimation via deep learning,” *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1910–1921, Apr. 2016.
- [197] Yanyang Yan, Wenqi Ren, Yuanfang Guo, Rui Wang, and Xiaochun Cao, “Image deblurring via extreme channels prior,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 6978–6986.
- [198] Liuge Yang and Hui Ji, “A variational EM framework with adaptive edge selection for blind motion deblurring,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 10 159–10 168.
- [199] Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum, “Progressive inter-scale and intra-scale non-blind image deconvolution,” *ACM Transactions on Graphics*, vol. 27, no. 3, pp. 1–10, Aug. 2008.

-
- [200] Zhiyuan Zha, Xin Yuan, Bihan Wen, Jiachao Zhang, Jiantao Zhou, and Ce Zhu, “Simultaneous nonlocal self-similarity prior for image denoising,” in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 1119–1123.
- [201] Ganchi Zhang and Nick Kingsbury, “Variational Bayesian image restoration with group-sparse modeling of wavelet coefficients,” *Digital Signal Processing*, vol. 47, pp. 157–168, 2015, Special Issue in Honour of William J. (Bill) Fitzgerald.
- [202] Haichao Zhang and David Wipf, “Non-uniform camera shake removal using a spatially-adaptive sparse penalty,” in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2013, pp. 1556–1564.
- [203] Haichao Zhang, David Wipf, and Yanning Zhang, “Multi-observation blind deconvolution with an adaptive sparse prior,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1628–1643, Aug. 2014.
- [204] Haichao Zhang, Jianchao Yang, Yanning Zhang, and Thomas S. Huang, “Sparse representation based blind image deblurring,” in *2011 IEEE International Conference on Multimedia and Expo*, Jul. 2011, pp. 1–6.
- [205] Jiawei Zhang, Jinshan Pan, Wei-Sheng Lai, Rynson W. H. Lau, and Ming-Hsuan Yang, “Learning fully convolutional networks for iterative non-blind deconvolution,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 6969–6977.
- [206] Jiawei Zhang, Jinshan Pan, Jimmy Ren, *et al.*, “Dynamic scene deblurring using spatially variant recurrent neural networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 2521–2529.
- [207] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang, “Learning deep CNN denoiser prior for image restoration,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 2808–2817.
- [208] Mingli Zhang and Christian Desrosiers, “High-quality image restoration using low-rank patch regularization and global structure sparsity,” *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 868–879, Feb. 2019.
- [209] Xinxin Zhang, Ronggang Wang, Yonghong Tian, Wenmin Wang, and Wen Gao, “Image deblurring using robust sparsity priors,” in *2015 IEEE International Conference on Image Processing (ICIP)*, Sep. 2015, pp. 138–142.
- [210] Shuai Zheng, Zhenfeng Zhu, Jian Cheng, Yandong Guo, and Yao Zhao, “Edge heuristic GAN for non-uniform blind deblurring,” *IEEE Signal Processing Letters*, vol. 26, no. 10, pp. 1546–1550, Oct. 2019.
- [211] Lin Zhong, Sunghyun Cho, Dimitris Metaxas, Sylvain Paris, and Jue Wang, “Handling noise in single image deblurring using directional filters,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2013, pp. 612–619.

- [212] Yipin Zhou and Nikos Komodakis, “A MAP-estimation framework for blind deblurring using high-level edge priors,” in *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 142–157.
- [213] Maria Zontak, Inbar Mosseri, and Michal Irani, “Separating signal from noise using patch recurrence across scales,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2013, pp. 1195–1202.
- [214] Daniel Zoran and Yair Weiss, “From learning models of natural image patches to whole image restoration,” in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 479–486.
- [215] Wangmeng Zuo, Dongwei Ren, Shuhang Gu, Liang Lin, and Lei Zhang, “Discriminative learning of iteration-wise priors for blind deconvolution,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3232–3240.
- [216] Wangmeng Zuo, Dongwei Ren, David Zhang, Shuhang Gu, and Lei Zhang, “Learning iteration-wise generalized shrinkage–thresholding operators for blind deconvolution,” *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1751–1764, Apr. 2016.