UNIVERSITY OF CALIFORNIA

Los Angeles

Local, Non-local and Global Methods in Image Reconstruction

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Mathematics

by

Yifei Lou

2010

The dissertation of Yifei Lou is approved.

Stanley Osher

Luminita Vese

Stefano Soatto, Committee Co-chair

Andrea Bertozzi, Committee Co-chair

University of California, Los Angeles

2010

Dedicated to my unborn child. . .

TABLE OF CONTENTS

1 Introduction **1**

2 Literature Review **5**

2.1 Image Denoising 5

2.2 Deconvolution 7

2.3 Super-Resolution 9

2.3.1 Focus on Registration 11

2.3.2 Focus on Reconstruction 12

2.3.3 Implicit Motion Estimation 13

3 Local Constraints via Sparsity **14**

3.1 Prior Work on Sparse Denoising 15

3.2 Direct Sparse Deblurring 16

3.2.1 Continuum Formulation 19

3.2.2 Boundary Issues 22

3.2.3 From Continuum to Discrete 23

3.2.4 Weighted Averaging 24

3.3 Experiments 25

3.3.1 Binary Text Images 25

3.3.2 Blind Deconvolution of the Text Images 26

3.3.3 General Case 28

3.3.4 Trained Dictionary 30

3.4 Discussion 32
3.4.1 Coherence 32
3.4.2 Domain Overlapping 35

4 A Nonlocal Framework 37
4.1 Nonlocal Operators 39
4.2 Computing the Weights 41
4.2.1 Similarity-Invariant Weights 41
4.2.2 Preprocessing the Data by Linear Models 43
4.3 Applications 48
4.3.1 Denoising 48
4.3.2 Super-resolution 50
4.3.3 Image deconvolution 51
4.3.4 Tomographic reconstruction 54

5 A Global Approach for Multi-image Denoising 67
5.1 Preliminaries, Anterior Works 71
5.1.1 Image Matching 71
5.1.2 Noise Estimation 72
5.1.3 Image/Video Denoising Algorithms 73
5.2 The Main Tools of the Burst Denoising Chain 74
5.2.1 Registration of an Image Sequence 75
5.2.2 Reliable Dominant Homography Estimation 77
5.2.3 Video Equalization 79

5.2.4 Signal-Dependent Noise Estimation 79
5.2.5 Hybrid Denoising Scheme 80
5.3 Experiments 81
5.3.1 Accurate SIFT 81
5.3.2 Multi-image Registration 82
5.3.3 Video Equalization 82
5.3.4 Noise Estimation 83
5.3.5 Multi-image Denoising 83

6 Conclusion **92**

References **94**

LIST OF FIGURES

3.1 Diagram of the continuum formulation. 20

3.2 Left: the blurry patch with zero-padding. Right: the blur basis. The red square indicates the region for our refined dictionary. 23

3.3 The training data. The dictionary is obtained by randomly sampling raw 10×10 patches from the text images as well as the template shown on the top left. All the text images are from different categories of CNN news. 26

3.4 Text Deblurring with 20,000 dictionary elements. 27

3.5 Influence of the number of the elements in the dictionary on the deblurring performance. The average of 10 experiments for each column is reported. 28

3.6 Blind deconvolution with comparison to non-blind case. 29

3.7 Examples in the training images. 30

3.8 Grayscale image deblurring with 20,000 dictionary elements. 31

3.9 The dictionary is trained using all the 16×16 patches in the training image (top left), which has a similar structure to the test one. 33

3.10 The dictionary is either trained or generic (comprised of random samples from the training image). Our method using either dictionary improves upon traditional methods, with the generic dictionary providing further improvement over the trained one. 34

4.1 Procedure to align patches. Three patches are selected to illustrate the alignment, as shown in the first row. From top to bottom: (1) noisy patches whose size corresponds to the scale of its center; (2) rotate the patch with the angle assigned by SIFT; (3) crop the black boundary due to the rotation; (4) down-sample to a uniform size patch 7×7. . 55

4.2 Fifteen most similar patches to the target one (red square on the left) are selected (middle) and aligned via similarity (right). On the right, the pose of the patch corresponds to the scale and orientation of its center as obtained by SIFT. 56

4.3 Radon transform in $\mathbb{R}^2$. 56

4.4 Experiment with Gaussian noise: $\sigma = 20$. The flat regions in the residual of NLM show that the central part has not been denoised. . . 57

4.5 Experiment with Gaussian noise: $\sigma = 40$. The flat regions in the residual of NLM show that the serifs of the A characters have not been captured. 58

4.6 Denoising examples in T. Brox's paper [17]. From top to bottom: original image, noisy image (cropped from his paper), nonlocal means, iterated NL (his), NL similarity (ours). 59

4.7 Color image denoising with Gaussian noise with $\sigma = 40$. From left to right, top to bottom: (a) original image, (b) noisy input f, (c) nonlocal means u_1, (d) nonlocal Similarity u_2, (e) NL method noise $f - u_1$ and (f) NL similarity method noise $f - u_2$. The stripes tend to be restored better in (f) than in (e). 60

4.8 Super Resolution. From top to bottom and left to right: (a) original image, (b) low-resolution noisy image), (c) standard nonlocal means and (d) nonlocal similarity (ours). The characters are sharper in (d) than (c). 61

4.9 A 150×150 image with Gaussian blur $\sigma_b = 2$ and Gaussian noise $\sigma_n = 10$. 62

4.10 A 200×200 image with Gaussian blur $\sigma_b = 1$ and Gaussian noise $\sigma_n = 5$ cropped to 75×75 pixels. 63

4.11 A 256×256 image with box average kernel 9×9 and Gaussian noise $\sigma_n = 3$. 64

4.12 Results of reconstruction from noisy projection data with SNR=28.1db. 65

4.13 Results of reconstruction from noisy projection data with SNR=26.04db. On the last row is the enlarged central part of each reconstruction. . . 66

5.1 From left to right: one long-exposure image (time = 0.4 sec, ISO=100), one of 16 short-exposure images (time = 1/40 sec, ISO = 1600) and the average after registration. All images have been color-balanced to show the same contrast. The long exposure image is blurry due to camera motion. The middle short-exposure image is noisy, and the third one is some 4 times less noisy, being the result of averaging 16 short-exposure images. 68

5.2 Two images used to test the accurate SIFT. The right image is generated from the left one by a translation+rotation. 82

5.3 Multi-image registration. Top: three frames from an image sequence with a rotating pattern and a fixed pedestal. Bottom: the corresponding ones after registration. The dominant homography we find is on the plane of the rotating pattern, since it contains more SIFT points than the pedestal region. As a result we observe the rotating pedestal and its background after registration. The images are a courtesy of DxO Labs, Boulogne. 84

5.4 Video Equalization. Top: three frames from an image sequence with different illuminations. Bottom: after registration and equalization. . 85

5.5 Noise curve. From top to bottom: the original image, one of the simulated images by moving the image and adding Poisson noise, and the noise curve from our algorithm using 16 images. The standard deviation of the noise (Y-axis) fits to the square root of the intensity (X-axis). 86

5.6 Noise curves of the real data sets. Left: one of the images in the sequence; right: the noise curves of the three color channels. 88

5.7 Real dataset from DxO Labs, Boulogne. Due to mis-registration, the simple average fails to denoise the region of the pedestals. In the middle example, it does not remove some dust stick to the camera objective. The hybrid scheme works everywhere and gives roughly the same result with NLM and BM3D. 89

5.8 Real dataset of an indoor setting. The average fails completely with the moving mouse on the right example, while block matching succeeds since it uses the similarity patches in the template image itself. 90

5.9 A burst of images of a painting. The last two results are almost identical, which indicates that the registration has been detected correct almost everywhere. 91

LIST OF TABLES

3.1 RMS errors for different methods. (G) and (T) indicate what kind of dictionary to use for our method with (G) for generic and (T) for trained. In some case, BM3D is marginally better than our method (by about 10%), while in other cases our method fares significantly better (three times better in the Text examples and 50% better in the Texture). 32

4.1 RMS errors for the input images and different denoising methods. . . 50

4.2 The statistics of blur and noise we add to the images. 52

4.3 SNR for different methods 53

5.1 The average error in each octave for Lowe's classical SIFT and for accurate SIFT. The precision decreases for Lowe's classical SIFT, while accurate SIFT remains stable through octaves. This is essentially obtained by removing the sub-sampling step in the SIFT method. 83

5.2 RMS for different methods 87

5.3 RMSE on synthetic data with 16 images. AR and GT stand for "average after registration" and "ground-truth" in the sense of registration back by the ground-truth motion. In principle GT divides the RMSE by 4, while AR is very close but higher than GT due to misregistration and interpolation errors. In all cases video BM3D gets close to the ratio 4 limit, but still is overcome by AR for all the images. 87

ACKNOWLEDGMENTS

First and foremost I wish to thank my thesis advisors, Professor Andrea Bertozzi and Professor Stefano Soatto. I feel fortunate to work with them for the past five years. As a computer science professor, Stefano always brings up a lot of intriguing problems in the computer vision and image processing. Thanks to him, I have diverse research interests. Andrea Bertozzi, on the other hand, is always a good example to me as a mathematician. I regret not learning enough mathematics from her. Also her careful revising of all my writings will never be forgotten.

My thanks go to my other committee members, Professor Stanley Osher and Professor Luminita Vese. It was Professor Osher who gave me an introductory but substantial course on image processing, which led me into this field. Without his help, the majority of the content in Chapter 4 would never be possible. I thank Professor Luminita Vese for her excellent lectures on the numerical analysis and helpful discussions on various aspects. I also wish to give my special thanks to Professor Jean-Michel Morel in ENS Cachan, France for his remarkable guidance during my 6-month internship in Paris.

Though only my name appears on the cover of this dissertation, many of my collaborators have contributed in these works. I enjoyed the collaboration with Xiaoqun Zhang, Toni Buades and Zhongwei Tang, and I would like to acknowledge them.

I want to thank David Y. Mao, James Y. Jian, Bin Dong, Yongning Zhu, Wenhua Gao for the helpful discussions that I had with them over the years. My gratitude also goes to the people in the Vision Lab, who inspire me in various ways.

My husband, Xun Jia, provides me with endless support from every aspect. Not only does he add color to my dull routine of everyday life, he helps with my research

as well. More often than not, he is able to point out the critical issues for the obstacles that I encounter. Now that our research directions are crossing, we have collaborated a couple of projects, which is a great experience.

My deepest gratitude goes to my parents, Shuichao Lou and Xiaohong Sun. They have been a constant source of love and support that have helped me reach my goals. They, as first teachers in my life, offered me the desire to obtain the Ph.D degree. I owe them everything and I wish I could express my sincerest appreciation to my parents.

Last but not the least, my first child is kicking while I'm writing this thesis. This work is dedicated to my naughty girl.

The research presented in this dissertation was supported by NSF grant CBET-0940417, ONR grant N000140810363 and Department of Defense.

VITA

1983	Born, Shanghai, the People's Republic of China.
2005	B.S. School of Mathematical Sciences, Peking University, Beijing, China.
2007	M.S. Department of Mathematics, University of California, Los Angeles, California, USA.
2005–present	Teaching and Research Assistant, Department of Mathematics, University of California, Los Angeles, California, USA.

PUBLICATIONS AND PRESENTATIONS

X. Jia, Y. Lou, B. Dong, Z. Tian, and S. B. Jiang. 4D computed tomography reconstruction from few-projection data via temporal non-local regularization. In *Proc. of Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2010

X. Jia, Y. Lou, R. Li, W. Y. Song, and S. B. Jiang. GPU-based fast cone beam CT reconstruction from undersampled and noisy projection data via total variation. *Med. Phys.*, **37**(4), 1757-1760, April 2010.

Y. Lou, X. Zhang, S. Osher and A. Bertozzi. Image Recovery via Nonlocal Operators. *Journal of Scientific Computing*, 42(2), 185-197, February 2010.

Y. Lou, P. Favaro, S. Soatto and A. Bertozzi. Nonlocal Similarity Image Filtering. In *Proc. of International Conference on Image Analysis and Processing (ICIAP)*, 2009.

T. Buades, Y. Lou, J-M. Morel and Z. Tang. A Note on Multi-Image Denoising. In *Proc. of Local and Non-Local Approximation (LNLA) in Image Processing*, 2009.

Y. Lou, A. Bertozzi and S. Soatto. Direct Sparse Deblurring. *CAM report* 09-15, 2009.

Y. Lou, P. Favaro, A. Bertozzi and S. Soatto. Autocalibration and Uncalibrated Reconstruction of Shape from Defocus. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

Y. Lou, B. Zhang, J. Wang and M. Jiang. Blind Deconvolution for Symmetric Point-spread Functions. In *Proc. of the IEEE Conf. Eng. Med. Biol. Soc. (EMBS)*, 4, 3459-3462, 2005.

Direct Sparse Deblurring. Poster presented at the SIAM Conference on Imaging Sciences, Chicago, IL, USA, April 12, 2010.

Nonlocal Similarity Image filtering. Presented at the Conference on Image Analysis and Processing (ICIAP), Salerno, Italy, September 9, 2009.

Burst Denoising. Presented at the Workshop on Local and Non-Local Approximation (LNLA) in Image Processing, Tuusula Finland, Aug. 21, 2009

Image Recovery via Nonlocal Operators. Presented at the SIAM Conference on Imag-

ing Sciences, San Diego, CA, USA, July 8, 2008.

Autocalibration and Uncalibrated Reconstruction of Shape from Defocus. Presented at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, NM, USA, June 21, 2007.

ABSTRACT OF THE DISSERTATION

Local, Non-local and Global Methods in Image Reconstruction

by

Yifei Lou

Doctor of Philosophy in Mathematics

University of California, Los Angeles, 2010

Professor Andrea Bertozzi, Co-chair

Professor Stefano Soatto, Co-chair

Image restoration has been an active research topic in image processing and computer vision. There are vast of literature, most of which rely on the regularization, or prior information of the underlying image. In this work, we examine three types of methods ranging from local, nonlocal to global with various applications.

A classical approach for local regularization term is achieved by manipulating the derivatives. We adopt the idea in the local patch-based sparse representation to present a deblurring algorithm. The key observation is that the sparse coefficients that encode a given image with respect to an over-complete basis are the same that encode a blurred version of the image with respect to a modified basis. Following an "analysis-by-synthesis" approach, an explicit generative model is used to compute a sparse representation of the blurred image, and its coefficients are used to combine elements of the original basis to yield a restored image.

We follows the framework that generates the neighborhood filters to an variational formulation for general image reconstruction problems. Specifically, two extensions

regarding to the weight computation are investigated. One is to exploit the recurrence of structures at different locations, orientations and scales in an image. While previous methods based on "nonlocal filtering" identify corresponding patches only up to translations, we consider more general similarity transformation. The second algorithm utilizes a preprocessed data as input for the weight computation. The requirements for preprocessing are (1) fast and (2) containing sharp edges. We get superior results in the applications of image deconvolution and tomographic reconstruction.

A Global approach is explored in a particular scenario, that is, taking a burst of photographs under low light conditions with a hand-held camera. Since each image of the burst is sharp but noisy, our goal is to efficiently denoise these multiple images. The proposed algorithm is a complex chain involving accurate registration, video equalization, noise estimation and the use of state-of-the-art denoising methods. Yet, we show that this complex chain may become risk free thanks to a key feature: the noise model can be estimated accurately from the image burst.

CHAPTER 1

Introduction

Image restoration is a fundamental problem to improve image quality for high-level vision tasks. A vast variety of methods are available touching very different fields of mathematics and statistics. To the best of our knowledge, many successful algorithms are based on

- a forward image formation model, *e.g.* (1.1)
- a generic image smoothness model, *i.e.* regularization term.

In this dissertation, we examine a general model of image formation:

$$f(x) = \mathcal{K}u(x) + n(x), \qquad \forall x \in \Omega \tag{1.1}$$

where Ω is the image domain, f is the observed data, $\mathcal{K}$ is a continuous linear operator, u is the original image and n is white noise. Image reconstruction is an inverse problem, which refers to the estimation of u given the data f and operator $\mathcal{K}$. The simplest case is when $\mathcal{K}$ degenerates to be the identity operator. As a result, our goal is to remove the noise n from the data f, hence the term "denoising". When the operator $\mathcal{K}$ is formulated as a convolution with a shift-invariant kernel, the inverse problem becomes image deconvolution. This shift-invariant kernel is referred to as the point-spread function (PSF), which usually describes the response of an imaging system to a point source. A close relative to deblurring [1] is super-resolution in the sense that its

[1] Although usually deblurring refers to non-shift-invariant kernels, we use deblurring and deconvolution interchangeably in the dissertation.

PSF corresponds to block-averaging of neighboring pixels. Another example occurs in the context of tomographic reconstruction in which $\mathcal{K}$ represents an attenuated Radon transform and the data f are observed as photon counts [3].

It is standard to approach the inverse problem by the method of regularization. Since there is always no unique solution, a prior is imposed to guarantee that the reconstructed image satisfies some desirable properties, such as smooth in homogeneous regions and sharp in edges. In this dissertation, we look into variational approaches where the image u is computed by a minimization of some energy functional, which typically consists of a data fidelity term and a regularization term.

The local smoothing model goes back to Gabor [83] in 1960, where the smoothness of u is measured by the H^1 semi-norm,

$$u = \arg\min \int_\Omega (f - \mathcal{K}u)^2 + \nu \int_\Omega |\nabla u|^2 \ .$$

It is also called Wiener Filter [2]. The minimizer u is a solution of the Euler-Lagrange equation

$$\mathcal{K}^*(\mathcal{K}u - f) - \nu \Delta u = 0 \ ,$$

where Δu is the Laplacian of u and ν is a parameter. If a gradient descent is applied, it boils down to the heat equation, which performs poorly on singular parts of u, namely edges or textures, where the Laplacian of the image is large. Another popular regularization is total variation (TV). Due to its virtue of preserving edges, TV is widely used in the many applications of image processing, such as blind deconvolution [27, 70, 93], inpainting [26] and super-resolution [94]. However, these TV-based methods tend to create a taxonomy of artifacts: "blur" in the texture regions, "staircase effect" and "checkerboard effect" *etc*.

With the emergence of the compressive sensing, the concept of sparsity has gained popularity in image processing and computer vision. It is based on the assumption

that any signal can be accurately represented with a few atoms from an over-complete dictionary. Due to the over-completeness, the dimension of signals considered here can not be the same as natural images, but has to reduce to a small image patch, for example, 8×8. Surprisingly, the sparse representation of local patches yields very appealing restoration results, such as denoising [46], color image restoration (denoising, inpainting and demosaicing) [90] and super-resolution [138, 36]. Our contribution, image deconvolution via sparsity, is addressed in Chapter 3. In particular, we devise a deblurring algorithm that (a) explicitly takes into account the "sparse" natural statistics of the image as a regularizer, and (b) does not suffer from the numerical conditioning issues associated with solving an inverse diffusion PDE. We also extend the method to blind deconvolution by simply augmenting our dictionary to include several different blurring kernels.

Neighborhood filters consider to average the pixels which have higher similarity with, instead of closer to, the pixel being processed. Referred to as nonlocal means (NLM), the idea proposed by Buades *et. al.* [20] is to replace every pixel with a weighed average of its neighborhood. The weight is measured in terms of similarity between image patches centered around the center pixel. Since similar pixels can be located far from each other, leading to an essentially nonlocal filtering. On the other hand, however, due to the computational complexity, the similarity is not calculated between any two pixels on the whole image domain, but within a searching window; hence the term is "nonlocal", but not "global". Such kind of filters includes the SUSAN filter [121], bilateral filter [129] and UINTA filter [4]. Inspired by graph theory [149], Gilboa and Osher formalize a systematic and coherent framework [61] using nonlocal operators. It provides an effective mechanism for general image restoration, as elaborated on Chapter 4. Our contribution is to design two particular ways of computing the weight. In the original NLM, image patches in the weight function are invariant only up to translation, while we extend this into a more general transformation, *i.e.*,

invariant to similarity. In addition, a key step for nonlocal methods to work in a general inverse problem is to use a crude solution of the inverse problem to construct the weight. It is especially important for applications such as tomographic reconstruction when the observed data and the image lie in different spaces.

Global smoothness is usually proposed in the transform domain. For example, Tikhonov regularization [128], which is to minimize the L^2 norm of the image, can be formulated as a one-step filter via Fourier transform, in the case of image deconvolution. However, this procedure tends to amplify the high frequencies where noise is conspicuous. In addition, global image characteristics may prevail over local ones, thus creating the "ringing" artifact. In Chapter 5, we consider a different global approach. It aims at denoising multiple images, any two of which are related by homography. The main technical objection is to register globally the images of a burst (to take multiple images of the same object). Once the registration is done, denoising follows naturally by simple averaging.

The dissertation is organized as follows. Chapter 2 provides literature review in the area of denoising, image (blind) deconvolution and super-resolution. In Chapter 3, we develop a local sparsity scheme for image deconvolution. A nonlocal framework is studied in Chapter 4 with two extensions on the ways of computing the weight function. The global approach for multi-image denoising is discussed in Chapter 5. Finally our conclusions are given in Chapter 6.

CHAPTER 2

Literature Review

2.1 Image Denoising

Image "denoising" refers to a series of inference tasks whereby the effects of various nuisance factors in the image formation process are removed or mitigated. Like all inference tasks, denoising hinges on an underlying model – implicit or explicit – where nuisance factors are processes that affect the data, but whose inference is not directly of interest. The generic term "noise" then refers loosely to all unmodeled phenomena, so illumination could be treated as noise in one application, or signal in another.

A simple example, used in most classical work in image processing, is the additive model, where the scene underlying the image is the image itself, and "noise" is just a realization of an additive process independent and typically characterized in terms of simple statistics such as its local mean and standard deviation, or its norm. The measured data is just a sampled version of the image plus noise. This model corresponds to assuming that the scene is flat, Lambertian, with diffuse albedo exhibiting piecewise smooth statistics, without an explicit model of illumination. The "noise" term lumps together everything else, and the resolution is determined by the sampling rate.

A variety of methods are available for image denoising, such as wavelet-based approaches [110, 98] and statistical filters [129, 4]. A popular denoising model is

proposed by Rudin, Osher and Fatemi, so-called ROF model,

$$u = \arg\min_u \int_\Omega |\nabla u| + \frac{\lambda}{2} \int_\Omega (f-u)^2, \tag{2.1}$$

where $\lambda > 0$ is Lagrange multiplier. One way of solving this minimization problem is to evolve a PDE, whose steady state corresponds to the local minima of (2.1),

$$\begin{cases} u_t = \nabla \dfrac{\nabla u}{|\nabla u|} + \lambda(f-u) & \text{in} \quad \Omega\,, \\ \dfrac{\partial u}{\partial n} = 0 & \text{on} \quad \partial\Omega\,. \end{cases}$$

There are fast methods to minimize (2.1) such as [25, 63, 64].

The nonlocal means filter [20] recently emerged as a generalization of the Yaroslavsky filter [139], but also taps on "exemplar-based" methods in texture synthesis [45] and super-resolution [60], as well as on "procedural methods" in computer graphics [136, 29]. Its advantage is to exploit similar patches in the same image, without an explicit model of the image formation process. The approach is taken one step further in [17], where similarity is computed hierarchically and efficiently. Another accelerating method is proposed by Mahmoudi and Sapiro [89] via eliminating unrelated neighborhoods from the weighted average. There are several other methods based on the idea of nonlocal means filter [20]. For example, Kervrann and Bruckner [75] improve it by using an adaptive window size. Gilboa and Osher [61, 76] formalize a variational nonlocal framework motivated from graph theory [149]. Chatterjee and Milanfar [28] generalize nonlocal means to high-order kernel regression.

Another trend in image denoising is to use sparse representation of signals. The basic assumption is that any signal x can be sparsely represented by a linear combination of atoms in an over-complete dictionary $\mathrm{D} \in \mathbb{R}^{n \times K}$ *i.e.*, $x = \mathrm{D}\alpha$ where the coefficients α has only a few non-zero elements. The over-completeness of the dictionary refers to the fact that $K > n$. If the data y is a corrupted version of x by the additive Gaussian

noise, then the "denoised" signal x is given by $\mathrm{D}\hat{\alpha}$, where

$$\hat{\alpha} = \arg\min_{\alpha} ||\alpha||_0 \quad \text{s.t.} \quad ||\mathrm{D}\alpha - y||_2^2 \leqslant T \ ,$$

where T is dictated by standard deviation of the additive noise. In this way, denoising is achieved. There are several caveats for sparsity applied to image denoising. First, due to the over-completeness of the dictionary, the dimension of atoms in the dictionary can not be very large. Therefore, the signal considered in the sparse representation is limited to small image patches. Second, the dictionary can be either chosen from a prespecified set of functions, such as wavelets of various sorts, or designed by adapting its content to fit a given set of signal examples. In [1], authors introduce the K-SVD algorithm to learn a dictionary iteratively from training samples using orthogonal matching pursuit (OMP) [92, 131]. The follow-up work [46] claims that their denoising algorithm achieves the best results when using K-SVD to learn a dictionary from the noisy image itself. The extension to color images is discussed in [90] by adapting the OMP inner-product definition. A multiscale framework is proposed in [91] for the use of different sizes of atoms simultaneously.

2.2 Deconvolution

Deblurring refers to the task of "undoing" the effects of convolving the data with a known kernel. A common instance occurs when an image is taken with a finite-aperture system that is not well focused, so the measured image is a blurred version of the "ideal image," convolved with the point-spread function of the lens. Ideally, one would like to recover, or "restore," the image as would be captured by a well-focused lens. Unfortunately, deblurring is well-known to be an ill-posed inverse problem, so small perturbations in the data (for instance noise or quantization errors in the measured "blurred" image) lead to large errors in the reconstruction. These arti-

facts are usually kept at bay by means of regularization, following the classical work of Tikhonov [128]. Several choices of such *generic regularizers* have been proposed, mostly made for mathematical convenience, some based on empirical observations on the quality of the reconstruction. Recently, there has been some convergence towards regularizers that enforce the statistics of natural images, which are well-known to possess highly kurtotic behavior [71] due to the presence of large homogeneous regions bounded by sharp discontinuities at visibility boundaries such as occlusions and cast shadows. Some classic regularizers, such as the Total Variation, implicitly favor these kind of solutions, and remain among the most competitive deblurring algorithms to this day. Nevertheless, deblurring in this context involves solving an inverse diffusion partial differential equation (PDE). In [52], Favaro *et. al.* have approached the problem of deblurring using a "direct" method: Rather than deconvolving the measured image and the noise that goes with it, they convolve the "model image," which is noiseless by definition, with the known kernel. This yields a simple diffusion PDE whose (space-varying) stopping time encodes the value of the kernel. They do not, however, exploit the statistics of natural images in their solution.

Deconvolution is mainly solved by regularization. For example, the Wiener filter [2] uses the H^1 semi-norm of the solution, which favors smooth reconstructions. Total Variation (TV) [114], as already mentioned, favors piece-wise constant solutions, whereas many wavelet-based deconvolution method do not directly enforce a regularizer, but rather enforce regularization through complexity bounds, see *e.g.*, [107, 39, 54, 55, 57]. One exception is discussed in [40]. Segmentation-based regularization is discussed in [97]. Deconvolution is also directly extended from denoising algorithms, such as BLS-GSM [110, 66], kernel regression [122, 123] and BM3D [33, 31].

Blind deconvolution is to decompose the data f as $f = g*u$, where both g and u are

unknown. One undesired solution is the no-blur explanation: g is the identity kernel and $u = f$. Additionally, in order for blind deconvolution algorithms to work, one important assumption is that the image and PSF must be irreducible [78]: *an irreducible signal cannot be exactly expressed as the convolution of two or more component signals of the same family, on the understanding that the two-dimensional delta function is not a component signal*. However, the Gaussian function as a widely utilized PSF or approximation to the real PSFs in many applications is reducible and not covered by the methods in [79]. In recent paper [80], Levin *et. al.* explain the failure of the naive maximum-a-posteriori (MAP) approaches by demonstrating that they mostly favor the no-blur solution. It also follows from their analysis that modern natural image priors [113, 137] do not help to overcome the MAP limitation. Fortunately, satisfactory results are achieved in some special applications, such as bar code reconstruction [48] and motion deblurring [53, 117, 23].

2.3 Super-Resolution

A single-frame super-resolution is often referred to as interpolation. Recently a directional interpolation [141, 142] is computed by estimating sparse image mixture models in a wavelet frame. Its idea is to use cubic spline interpolation for the low-frequency image of wavelet decomposition, and a directional interpolation for the high-frequency ones according to local orientation. Baker and Kanade derive analytical results in [6] to show that the reconstruction constraints and smoothness prior provides less useful information for larger magnification factors. They also propose a solution, which is to introduce a recognition-based prior in the context of face or text images. Similarly, example-based approaches [60] use a nearest-neighbor search to find the best match for local patches, and replace them with corresponding high-resolution patches in the training set, thus enhancing the resolution. To make the neighbors compatible, they

use a belief-propagation algorithm linked to the Markov network. In another work, Datsenko and Elad [36] consider a weighted average by surrounding pixels (analogue to NLM). Instead of the nearest-neighbor search, Yang *et. al.* [138] propose to incorporate sparsity in the sense that each local patch can be sparsely represented as a linear combination of low-resolution image patches; and a high-resolution image is reconstructed by the corresponding high-resolution elements. Example-based video enhancement is discussed in [11], where a simple frame-by-frame approach is combined with temporal consistency between successive frames. Also to mitigate the flicker artifacts, a stasis prior is introduced to ensure the consistency in the high frequency information between two adjacent frames.

Multi-frame super-resolution usually consists of two steps: registration and image fusion. A review [13] in 1998 further categorizes super-resolution methods into two divisions: those in the frequency domain and those in the spatial domain. Knowledge of the affine motion in sub-pixel accuracy is required for frequency methods, while spatial methods highly rely on regularization and optimization techniques.

Super-resolution and motion deblurring are combined in the work [8]. First the object is tracked through the sequence, which gives a reliable sub-pixel segmentation of a moving object [7]. Then a high-resolution image is constructed by merging multiple images with motion estimation. The deblurring algorithm, which mainly deals with motion blur [74], is applied only to the region of interest. In [118, 119], super-resolution is performed simultaneously in time and in space. Authors introduce a directional space-time regularization term to enforce smoothness only in directions within the space-time volume where the derivatives are low. The algorithm can be applied directly to revolve motion blur, which is caused by temporal blurring and not by spatial blurring. The recent paper by Baboulaz and Dragotti [5] presents several methods of registration and image fusion to solve the super-resolution problem. The

registration can be performed either globally by continuous moments from samples or locally by step edge extraction. Besides, they merge the set of registered images into a single image and then apply either Wiener filter or the iterative Modified Residual Norm Steepest Descent (MRNSD) [106] to remove the blur and the noise.

2.3.1 Focus on Registration

In terms of image registration, most of the existing SR methods rely either on a computationally intensive optical flow calculation, or on a parametric global motion estimation. In [148], Zhao *et. al.* discuss the effects of multi-image alignment on super-resolution, which addresses two issues: flow consistency (flow computed from frame A to frame B should be consistent with that computed from B to A) and flow accuracy. The flow consistency can be generalized to multiple frames by computing a consistent bundle of flow fields. Global motion, on the other hand, can be estimated either in the frequency domain or by feature-based approaches. For example, Vandewalle *et. al.* [133] consider the problem of registering a set of images based on their low-frequencies, aliasing-free part. They assume a planar motion, and as a result, the rotation angle and shifts between any two images can be precisely calculated in the frequency domain. As for the feature-based approaches, the standard procedure is to detect the feature points via Harris corner detector [24, 5] or SIFT [144, 116] and then match the corresponding points by RANSAC to fit a proper transformation such as homography.

The Harris corner detector [67] finds the local maximum of the points using a large corner response function. On the other hand, SIFT [87] is short for Scale Invariant Feature Transform (SIFT). This is one of the most successful methodologies to match regions up to a similarity transformation. The main steps in computing SIFT are

- *Scale-space extrema detection:* Scale is identified by searching for extrema in

the scale-space of the image via a difference-of-Gaussian convolution.

- *Keypoint localization:* Keypoints are selected based on the stability of fitting a 3-D quadratic function (obtained via Taylor expansion in the scale-space).

- *Orientation assignment:* A rotation with respect to a canonical reference frame is computed based on local image gradients.

- *Keypoint descriptor:* A vector composed of local image gradients is built, so that it is not sensitive to similarity transformations and, to some extent, changes in illumination.

More details on how each step is implemented in practice can be found in [87].

Both the Harris corner detector and SIFT will produce a large number of outliers which are inconsistent with the desired homography. The RANSAC algorithm [56] is applied to simultaneously estimate the homography and a set of consistent matching points. RANSAC is an abbreviation for "RANdom SAmple Consensus." It is composed of two steps in an iterative fashion. First, a subset of samples points are randomly selected to fit the parameters of the homography model. In the second step, RANSAC checks which elements of the entire dataset are consistent with the model. The set of such elements is called a consensus set.

2.3.2 Focus on Reconstruction

There are a number of papers that focus on image fusion by assuming the motion between two frames is either known or easily computed. Elad and Feuer [47] formulate the super-resolution of image sequences in the context of the Kalman filter. They assume the matrices, which define the state-space system, are known. For example, the blurring kernel can be estimated by a knowledge of the camera characteristics,

and the warping between two consecutive frames is computed by a motion estimation algorithm. But due to the curse of dimensionality of the Kalman filter, they can only deal with small images, *e.g.* of size 50×50. The work [94] by Marquina and Osher limits the forward model to be spatial-invariant blurring kernel with the down-sampling operator, while no local motion is present. They solve a TV-based reconstruction with Bregman iterations.

A joint approach on demosaicing and super-resolution of color images is addressed in [49], based on their early super-resolution work [50]. The authors use the bilateral-TV regularization for the spatial luminance component, the Tikhonov regularization for the chrominance component and a penalty term for inter-color dependencies. The motion vectors are computed via a hierarchical model-based estimation [9]. The initial guess is the result of the Shift-And-Add method. In addition, the camera PSF is assumed to be a Gaussian kernel with various standard deviation for different sets of experiments.

2.3.3 Implicit Motion Estimation

Inspired by NLM, researchers nowadays turn their attention into super-resolution without motion estimation [43, 44, 111]. Similar methodologies include the steering kernel regression [124], BM3D [33] and so forth. The forward model in [35] does not assume the present of the noise, so the authors prefilter the noisy low-resolution input by V-BM3D [30]. They upsample each image progressively several times, and at each time, the initial estimate is obtained by zero-padding the spectra of the output from the previous stage, followed by filtering.

CHAPTER 3

Local Constraints via Sparsity

There are relatively fewer works addressing the use of sparse priors for deblurring, which is the goal in this chapter. Although super-resolution is a close relative to deblurring (the point-spread function corresponds to block-averaging of neighboring pixels), the latter has not been addressed directly in a sparse setting. There are a few papers on image deblurring using global sparse transforms [88, 23], while we focus on local sparsity.

A related literature stream encodes the image as a discrete array of positive numbers, approximated by linear combinations of local overcomplete bases, where the natural statistics are captured by the fact that the vector of coefficients is *sparse*, so at any location only few bases contribute to the approximation [81]. In this case, there is no need for an explicit regularizer, due to the finite dimensionality of the representation, but there is still a trade-off between fidelity of the approximation and complexity of the model. While the measured image is undoubtedly a discrete object, with quantization of both the domain and the range, the object of inference, or the "ideal image," is best represented in the continuum, with the final discretization left only for the numerical implementation of the optimization scheme. We therefore take the continuum approach, and explicitly write cost functionals that have the ideal image as an infinite-dimensional unknown.

Since we use the sparse image denoising algorithm [46] as a building block, we will briefly review it in Sect. 3.1 to make this chapter self-contained. We will then

extend it to the continuum and apply it to deblurring in Sect. 3.2 with the discussion of the details on the implementation. Finally, Sect. 3.3 contains numerical experiments. We discuss some caveats in Section 3.4.1.

3.1 Prior Work on Sparse Denoising

If we consider discrete image patches, *i.e.* positive-valued matrices of size $\sqrt{n} \times \sqrt{n}$ pixels, ordered lexicographically as column vectors $x \in \mathbb{R}^n$, then the sparsity assumption corresponds to assuming the existence of a matrix $\mathrm{D} \in \mathbb{R}^{n \times K}$, the "dictionary," such that every image patch x can be represented as a linear combination of its columns with a vector of coefficients with small L^0 norm. If we measure y, a version of x corrupted by additive Gaussian noise that is spatially white (independent and identically distributed) with standard deviation σ, then the maximum a-posteriori (MAP) estimator of the "denoised" patch x is given by $\mathrm{D}\hat{\alpha}$, where

$$\hat{\alpha} = \arg\min_{\alpha} ||\alpha||_0 \quad \text{s.t.} \quad ||\mathrm{D}\alpha - y||_2^2 \leqslant T \,, \tag{3.1}$$

where T is dictated by σ. If one wishes to encode a larger image X of size $\sqrt{N} \times \sqrt{N}$ ($N \gg n$), with a combination of columns of the low-dimensional dictionary D, a natural approach is to use a block-coordinate relaxation.

$$\begin{aligned} \hat{\mathrm{X}} \;\; = \;\; & \arg\min_{\mathrm{X},\alpha_{ij},\mathrm{D}} ||\mathrm{X} - \mathrm{Y}||_2^2 + \lambda \sum_{i,j} ||\alpha_{ij}||_0 \\ & + \mu \sum_{i,j} ||\mathrm{D}\alpha_{ij} - R_{ij}\mathrm{X}||_2^2 \,. \end{aligned} \tag{3.2}$$

The first term measures the fidelity between the measured image Y and its denoised (and unknown) version X. The second term enforces sparsity of each patch; the $n \times N$ matrix R_{ij} extracts the (i,j)th block from the image. A simple denoising algorithm [46] based on sparse coding goes as follows,

1. Initialization: Set $\mathrm{X} = \mathrm{Y}, \mathrm{D} =$ an overcomplete discrete cosine transform (DCT) dictionary.

2. Repeat until it converges:

   - *Sparse Coding:* fix X and D, compute the representation vectors α_{ij} for each patch $R_{ij}\mathrm{X}$

$$\begin{aligned} \hat{\alpha}_{ij} &= \arg\min_{\alpha} ||\alpha||_0 \\ &\quad \text{s.t. } ||\mathrm{D}\alpha - R_{ij}\mathrm{X}||_2^2 \leqslant T \ . \end{aligned} \tag{3.3}$$

   - *Dictionary Update:* fix X and $\{\alpha_{ij}\}$, compute D via K-SVD [1] one column at a time.

3. Set:

$$\mathrm{X} = \frac{\mathrm{Y} + \mu \sum_{ij} R_{ij}^T \mathrm{D}\alpha_{ij}}{Id + \mu \sum_{ij} R_{ij}^T R_{ij}} \ , \tag{3.4}$$

   which is a simple averaging of shifted patches.

One could also fix the dictionary and only perform sparse coding in the iteration. Alternatively, the dictionary can be learned from a large number of patches in natural images via K-SVD [1] so that it is tailored to the data.

3.2 Direct Sparse Deblurring

The idea of direct sparse deblurring is simple and can be illustrated in three steps.

First, we assume that the image is square-integrable and sparse in some basis defined on the entire real plane. This is a common assumption underlying most image compression algorithms, in particular those based on over-complete bases, or "dictionaries" $\{d_k\} \in L^2(\mathbb{R}^2 \to \mathbb{R}), k = 1, \cdots, K$ where K is the number of atoms in the

dictionary. A dictionary can be used to approximate the original image u to an arbitrary degree by a sparse linear combination $u \doteq \sum_{k=1}^{K} d_k \alpha_k + n \doteq \mathrm{D}\alpha + n$ where $\|\alpha\|_0$ is small[1] and so is $\|n\|_2$ [46]. For the sake of illustration, let us pretend that this representation is *exact*, that is $\|n\| = 0$, and $u = \mathrm{D}\alpha$ with $\|\alpha\|_0 \leq L$, where L is the bound of L^0 norm. We will discuss the role of n later.

Now, convolving an image with a shift-invariant kernel g yields a blurred image $f = g * u = g * \mathrm{D}\alpha$; this shows that the coefficients α that represent the *sharp image* u relative to the *clear basis* $\{d_k\}$ are the same that represent the **blurred image** f relative to the **blurred basis** $\{b_k\} \doteq \{g * d_k\}$.

But while we do not have access to the sharp image u, we *do* have access to the original basis elements $\{d_k\}$. Therefore, all we need to recover the encoding of the sharp image are the coefficients of the encoding of the blurred image relative to the blurred basis. The ensuing algorithm is as follows:

1. Take a dictionary $\{d_k\}$, either from a generic over-complete basis or learned form the image using any of a variety of sparse coding algorithms, for instance [1]. Convolve the basis with the kernel g to obtain a "blurred basis" $\{b_k\} \doteq \{g * d_k\}$.

2. Perform sparse coding of the blurred image f relative to the blurred basis $\{b_k\}$ to obtain $\hat{\alpha}$ with $\|\hat{\alpha}\|_0$ small such that $\|f - \sum_{k=1}^{K} b_k \hat{\alpha}_k\|$ is also small.

3. Reconstruct the original (deblurred) image *directly* via $\hat{u} = \sum_{k=1}^{K} d_k \hat{\alpha}_k$.

Note that this algorithm performs deblurring without solving a backward diffusion or other numerically ill-conditioned procedure. Instead, it solves the inverse problem by "direct methods", an approach sometimes referred to as "analysis by synthesis" [65]

[1] $\forall\, \epsilon\ \exists\, \bar{K} = K(\epsilon)$ such that if $M > \bar{K}$, then $\|f - \sum_{k=1}^{M} d_k \alpha_k\| < \epsilon$.

whereby an explicit generative model is used to match the statistics of the measured data, and the model itself provides both the necessary regularization and the solution to the desired inverse problem.

Now a few caveats. In practice, no image fits the model $u = \mathrm{D}\alpha$ exactly, sparsely or otherwise. Therefore, one typically looks for the optimal representation $\hat{\alpha}$, defined as the solution of the following optimization problem

$$\hat{\alpha} = \arg\min_{\alpha} \left\{ \int \|n\|_2^2 \mathrm{d}x \;\mid\; u = \mathrm{D}\alpha + n, \;\; \|\alpha\|_0 \leq L \right\},$$

where the integral is on all of $\mathbb{R}^2$. Following the prescription outlined above to extend the algorithm to blurred images, one would get $f = g * u + g * n = g * \mathrm{D}\alpha + g * n$. Therefore, the algorithm we have suggested does *not* solve the problem

$$\hat{\alpha} = \arg\min_{\alpha} \left\{ \int \|n\|_2^2 \mathrm{d}x \;\mid\; f = g * \mathrm{D}\alpha + n, \;\; \|\alpha\|_0 \leq L \right\},$$

where the constraint $f = g * u + n$ describes the image formation model. Instead, we solve the modified problem

$$\tilde{\alpha} = \arg\min_{\alpha} \left\{ \int \|g * n\|_2^2 \mathrm{d}x \;\mid\; f = g * \mathrm{D}\alpha + g * n, \;\; \|\alpha\|_0 \leq L \right\},$$

Note that, in principle, $\tilde{\alpha} \neq \hat{\alpha}$. However, the blurring kernel g is zero-mean (lest images would get brighter and dimmer as the focus changes), and therefore

$$\int \|n\|_2^2 \mathrm{d}x \geq \int \|g * n\|_2^2 \mathrm{d}x.$$

So, at least to first approximation, we indeed have that $\tilde{\alpha} = \hat{\alpha}$. The benefit of this approach is a considerably simpler "direct" algorithm, and the cost is having changed the terms of the problem, or equivalently the model and the underlying assumption, from minimizing the residua error n, to minimizing a blurred version of it.

Of course, the devil is in the details, as real images and dictionaries are not defined on the entire real plane, and if we wish to keep the complexity of the coding step

manageable we will have to break down the image into patches, which raises the issue of boundary effects and scale, which causes $\tilde{\alpha} \neq \hat{\alpha}$. However, there is no reason why $\hat{\alpha}$ should be "better" than $\tilde{\alpha}$; they just represent different modeling assumptions. In the words of Box, "all models are wrong, some are useful." Ours is useful in the sense of yielding a particularly simple, direct algorithm, which we now derive for a partition of the image, explicitly taking the issues of boundaries and scale into account.

3.2.1 Continuum Formulation

In this section we formalize the problem of direct sparse deblurring. We find the formalization to be clearer when written in the continuum, so one knows on what domain each function is calculated. The previous claim, that the blurred image can be sparsely represented in the blurred basis by the same coefficients that the "ideal" image would have on the original basis, will become clear.

Let $u : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^+; x \mapsto u(x)$ be the "ideal image", corresponding to X in the discrete model. The procedure of extracting a small patch from an epsilon ball centered at x can be represented by

$$u_x(y) = \{u(y) : y \in \mathcal{B}_\epsilon(x)\} = \{u(x+y) : y \in \mathcal{B}_\epsilon(0)\}.$$

The function $u_x(\cdot)$ describes a mapping from an epsilon ball to a patch centered at x, which can be expressed by an indicator function $\chi_\epsilon(x-y)$ acting on the image $u(x)$.

Let $d_k : \mathcal{B}_\epsilon(0) \subset \mathbb{R}^2 \rightarrow \mathbb{R},\ k = 1, \ldots, K$ be a given overcomplete basis of $\mathcal{L}_{loc}(\mathcal{B}_\epsilon(0) \rightarrow \mathbb{R})$, and $\alpha_k(x)$ be the kth sparse coefficient of the patch centered at x. Then the sparse representation of one image patch is given by

$$\begin{aligned} u_x(y) &= \chi_\epsilon(y) u(x+y) \\ &= \sum_{k=1}^{K} d_k(y)\alpha_k(x) \doteq \mathbf{d}(y)\alpha(x) \ , \end{aligned} \tag{3.5}$$

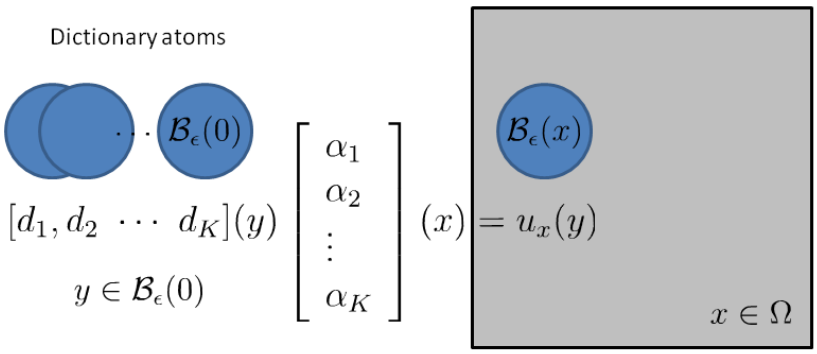


Figure 3.1: Diagram of the continuum formulation.

where $\mathbf{d}(y) = [d_1, d_2, \cdots, d_K](y), y \in \mathcal{B}_\epsilon(0)$ and $\alpha(x) = [\alpha_1, \alpha_2, \cdots, \alpha_K](x), x \in \Omega$. Fig. 3.1 illustrates the dictionary elements and how they match to local patches.

We want to use this local sparsity to enforce a global reconstruction prior in the sense that $u(x)$ is the minimizer of the sparse representation error for all the local patches.

$$\hat{u}(x) = \arg\min J(u) \ ,$$

where $J(u)$ is defined to be

$$\begin{aligned} & \int_{x\in\Omega} \int_{y\in\mathcal{B}_\epsilon(0)} \|\chi_\epsilon(y)u(x+y) - \mathbf{d}(y)\alpha(x)\|^2 \mathrm{d}y\mathrm{d}x \\ = & \iint_{\Omega\times\Omega} \chi_\epsilon(y)\|u(x+y) - \mathbf{d}(y)\alpha(x)\|^2 \mathrm{d}y\mathrm{d}x \\ = & \iint_{\Omega\times\bar{\Omega}} \chi_\epsilon(z-x)\|u(z) - \mathbf{d}(z-x)\alpha(x)\|^2 \mathrm{d}z\mathrm{d}x \ . \end{aligned} \tag{3.6}$$

where $z = x + y$ and $\bar{\Omega} = \Omega + \mathcal{B}_\epsilon(0)$.

To solve for $u(x)$, we compute its Euler-Lagrange equation

$$\begin{aligned} & \partial_u J(u)(z) \\ = & \int \chi_\epsilon(z-x)\Big(u(z) - \mathbf{d}(z-x)\alpha(x)\Big)\mathrm{d}x \\ = & \Big[\int \chi_\epsilon(z-x)\mathrm{d}x\Big]u(z) - \int \chi_\epsilon(z-x)\mathbf{d}(z-x)\alpha(x)\mathrm{d}x \ . \end{aligned} \tag{3.7}$$

There is a closed-form solution for $u(x)$ w.r.t $\alpha(x)$ that minimizes the objective function $J(u)$. It is obtained by setting the Euler-Lagrange equation to zero:

$$\hat{u}(x) = \frac{1}{\omega}\int \chi_\epsilon(x-y)\mathbf{d}(x-y)\alpha(y)\mathrm{d}y \ , \tag{3.8}$$

where $\omega = \int \chi_\epsilon(x-y)\mathrm{d}y$ is the area of the $\epsilon-$ball. Now, the measured image is, by assumption

$$\begin{aligned} f(x) & \doteq \int g(x-\bar{x})u(\bar{x})\mathrm{d}\bar{x} + n(x) \\ & = \frac{1}{\omega}\int g(x-\bar{x})\int \chi_\epsilon(\bar{x}-y)\mathbf{d}(\bar{x}-y)\alpha(y)\mathrm{d}\bar{x}\mathrm{d}y + n(x) \\ & = \frac{1}{\omega}\iint \big[g(x-\bar{x})\chi_\epsilon(\bar{x}-y)\mathbf{d}(\bar{x}-y)\mathrm{d}\bar{x}\big]\alpha(y)\mathrm{d}y + n(x) \ , \end{aligned} \tag{3.9}$$

where g is a space-invariant blurring kernel and $n(x)$ is the additive noise, whose variance is σ^2. The blurred basis is easily defined as

$$b_k(z) \doteq \int g(z-\bar{x})\chi_\epsilon(\bar{x})d_k(\bar{x})\mathrm{d}\bar{x} \ .$$

The characteristic function χ_ϵ implies that the boundary condition for the convolution is zero-padding. Denote with r the support of the blurred basis, in particular $r = \epsilon + supp(g)$. Therefore the measured image is a sparse representation under this blurred basis:

$$f(x) = \frac{1}{\omega}\int \chi_r(x-y)\mathbf{b}(x-y)\alpha(y)\mathrm{d}y + n(x) \ .$$

We solve for the sparse coefficients in the following,

$$\begin{aligned} & \hat{\alpha}(x) = \arg\min \int \|\alpha(x)\|_0 \mathrm{d}x \ , \\ & \text{s.t.} \int \|f(x) - \omega^{-1}\int \chi_r(x-y)\mathbf{b}(x-y)\alpha(y)\mathrm{d}y\|^2\mathrm{d}x \leqslant T \ . \end{aligned} \tag{3.10}$$

This optimization problem (a) is finite-dimensional (the only unknown is α) and (b) *does not involve de-blurring.* All that is required is to find the finite-dimensional sparse set of coefficient that best approximates the given image. Note that this is accomplished by *blurring the base*. In other words, one is only required to solve a *direct* problem, rather than the inverse problem of deblurring. Once the coefficients $\hat{\alpha}$ are obtained, we can compute the "deblurred" image $\hat{u}$ via eq. (3.8). Note that the deblurred image is a sparse combination of the (original, non-blurred) basis, and therefore – by construction – one should expect the reconstruction to exhibit the same spatial frequencies of the original (unblurred) data from which the overcomplete basis has been learned.

3.2.2 Boundary Issues

In practice, solving for α from (3.10) is not an easy task, since α at different y contribute to one value. Instead we minimize an upper bound.

$$\begin{aligned}&\int \| \int \chi_\epsilon(x-y) f(x) \mathrm{d}y - \int \chi_r(x-y)\mathbf{b}(x-y)\alpha(y)\mathrm{d}y\|^2 dx \\ &\leqslant \iint \chi_\epsilon(z)\|f(z+y) - \mathbf{b}(z)\alpha(y)\|^2 \mathrm{d}z\mathrm{d}y \ .\end{aligned}$$

The above equation suggests coding the blurry patch centered at y in terms of the blurred basis $\{b_i\}$. Furthermore, the characteristic function indicates that the blurry patch has to be zero-padded in order to be consistent with the dimension of the blurred basis. However, these two terms cannot match due to the boundary issue of convolution, as shown in Fig. 3.2. Instead we match them in the region where the convolution is computed without the zero-padded edges. In particular, we refine our blurred basis to be

$$\tilde{b}_k(z) = \begin{cases} b_k(z) & |z| < \epsilon - supp(g) \doteq \epsilon_0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.11}$$

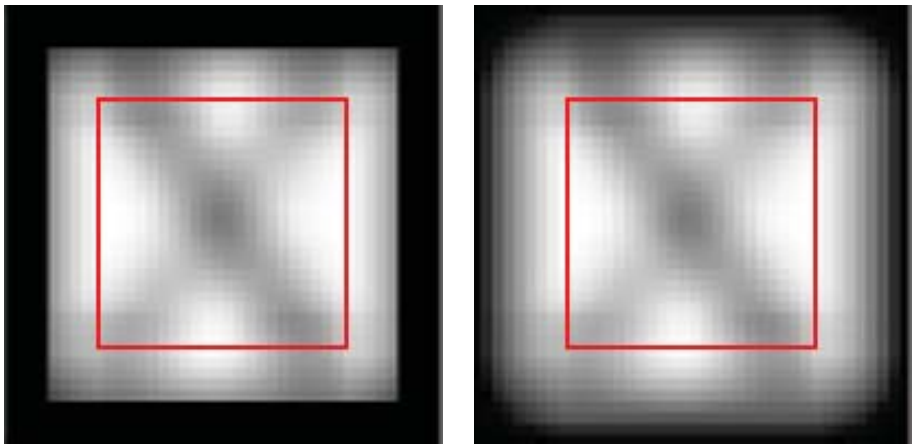


Figure 3.2: Left: the blurry patch with zero-padding. Right: the blur basis. The red square indicates the region for our refined dictionary.

Experimentally we find that it is better to tailor the clear basis to have the same domain size as the blurred one. Therefore we have a two step algorithm for sparse deblurring

1. Solve the coefficients from the measured image

$$\begin{aligned} &\hat{\alpha}(x) = \arg\min \|\alpha\|_0 \,, \\ &\text{s.t.} \int \chi_{\epsilon_0}(z) \|f(z+y) - \mathbf{b}(z)\alpha\|^2 \mathrm{d}y \leqslant T \,. \end{aligned} \tag{3.12}$$

2. Stitch all the patches by averaging

$$\hat{u}(x) = \frac{\int \chi_{\epsilon_0}(x-y)\mathbf{d}(x-y)\hat{\alpha}(x)\mathrm{d}y}{\int \chi_{\epsilon_0}(x-y)\mathrm{d}y} \,. \tag{3.13}$$

3.2.3 From Continuum to Discrete

In the discrete case, we assume the clear basis $\{d_k\}$ to be of size $a \times a$ and the blur kernel h be of size $c \times c$. It follows from (3.11) that the blurred basis $\{\tilde{b}_k\}$, which is the inner part of $\{b_k\}$, is of size $a_0 \times a_0$, where $a_0 = a - c + 1$. We crop the clear basis to be $a_0 \times a_0$ as well, denoted as $\{\tilde{d}_k\}$.

The first step (3.12) amounts to sparse coding for every x, or pixel (i, j) in the discrete sense. We use the Orthogonal Matching Pursuit (OMP) algorithm [131] to solve $\hat{\alpha}$ from (3.1) with y being the patch centered at (i, j) of size $a_0 \times a_0$ and dictionary D being comprised of $\{\tilde{b}_k\}$. The second step is to replace the blurred patch with the clear basis $\{\tilde{d}_k\}$ multiplying the sparse coefficient $\hat{\alpha}$. Finally, since each pixel is covered by different patches, the restored value is chosen to be the mean.

We use the L^0 solver OMP over a million of methods on L^1 minimization for two reasons. First, there is no additional parameter for L^0 and it has a natural stopping criterion, *i.e.* stops when the residual is smaller than the standard deviation of the addition noise. Second, it takes more iterations for L^1 to get reasonable sparse coefficients.

3.2.4 Weighted Averaging

One drawback is that the averaging of overlapping patches also degrades the reconstruction. As an alternative, one could use a weighted average or median to combine the results from local sparse coding. Accordingly, we perform sparse coding as described to get the coefficients $\alpha(y)$, then use weighted averaging to combine the results

$$\hat{u}(x) = \frac{\int_{y \in \mathcal{B}_\epsilon(x)} \mathbf{d}(x - y)\alpha(y)w(y)dy}{\int_{y \in \mathcal{B}_\epsilon(x)} w(y)dy} .$$

When the dictionary is rich enough, each patch can be ideally represented by single atom. It is reasonable to assume that the smaller the L^0 norm of the sparse coefficients, the better representation of this patch. Therefore, the weight is chosen to penalize large L^0 norm of the coefficients, for example,

$$w(y) = \exp\left\{\frac{-|\alpha(y)|_0}{s}\right\} ,$$

where s is a control parameter.

3.3 Experiments

In this section we compare our algorithm to alternate methods such as ROF [114], the wavelet-based "ForWaRD" approach [107], regularized kernel regression-based deblurring (AKTV) [123] and BM3D-based image restoration [31]. The optimal method parameters for both ROF and ForWaRD are chosen from a series of values with wide range. Publicly available code was used for comparison with AKTV and BM3D, including the suggested parameter values. The parameters in our algorithm are determined by the data: the size of the dictionary is proportional to the width of the blurring kernel and the stopping criterion for the sparse coding stage is when the residual is below the variance of the noise.

We use root-mean-square (RMS) as a means of judging performance, $RMS(u, I) = \sqrt{\int_{x\in\Omega} (u(x) - I(x))^2 \, \mathrm{d}x}$, where $I(x)$ is the original image and $u(x)$ is the recovery.

3.3.1 Binary Text Images

We synthesize a template with all the alphanumeric characters and common punctuation as well as 5 text images from different categories of CNN news. The dictionary is comprised of 10×10 image patches randomly sampled from the three images and the template, all shown in Fig. 3.3. The template contains all individual characters, while the training images serve to represent meaningful pairs. We test the deblurring on the other two text images. The data are corrupted by convolution with a 5×5-pixel Gaussian kernel with $\sigma = 1$ and additive noise whose standard deviation is 5. For direct sparse deblurring, the visual quality is for the most part satisfactory except for the smoothing effects around some letters. This is mostly attributed to the limitations of the dictionary.

We also measure the effect of the number of the elements in the dictionary on

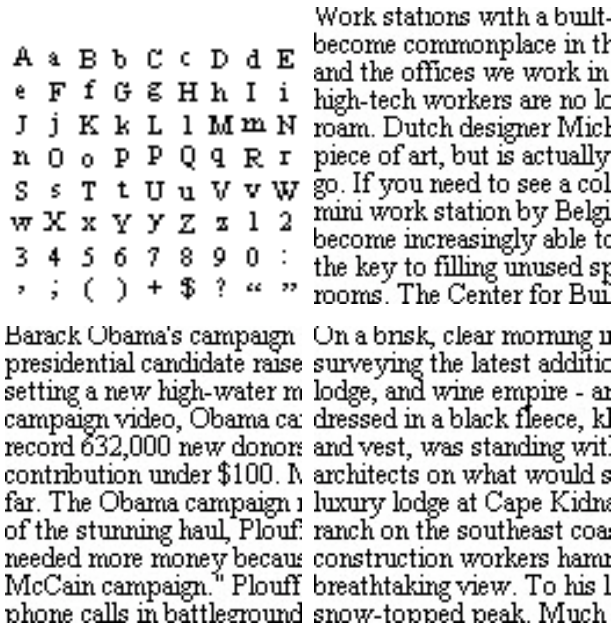


Figure 3.3: The training data. The dictionary is obtained by randomly sampling raw 10×10 patches from the text images as well as the template shown on the top left. All the text images are from different categories of CNN news.

the deblurring performance. Fig. 3.5 shows the results averaged from ten different experiments of randomly sampled elements in the dictionary. In general, increasing the number of elements in the dictionary improves the results, but with a diminishing return.

3.3.2 Blind Deconvolution of the Text Images

For blind deconvolution, we convolve the clear basis with 3 different Gaussian kernels (same size, different σ_{dict}). Now the blurred basis has 3 times more elements than in the non-blind case. There still exists correspondence between blurred basis and clear basis. Therefore, sparse deblurring follows the same procedure as the non-blind case.

The blurry noisy data is obtained by convolving the image with 5×5 Gaussian ker-

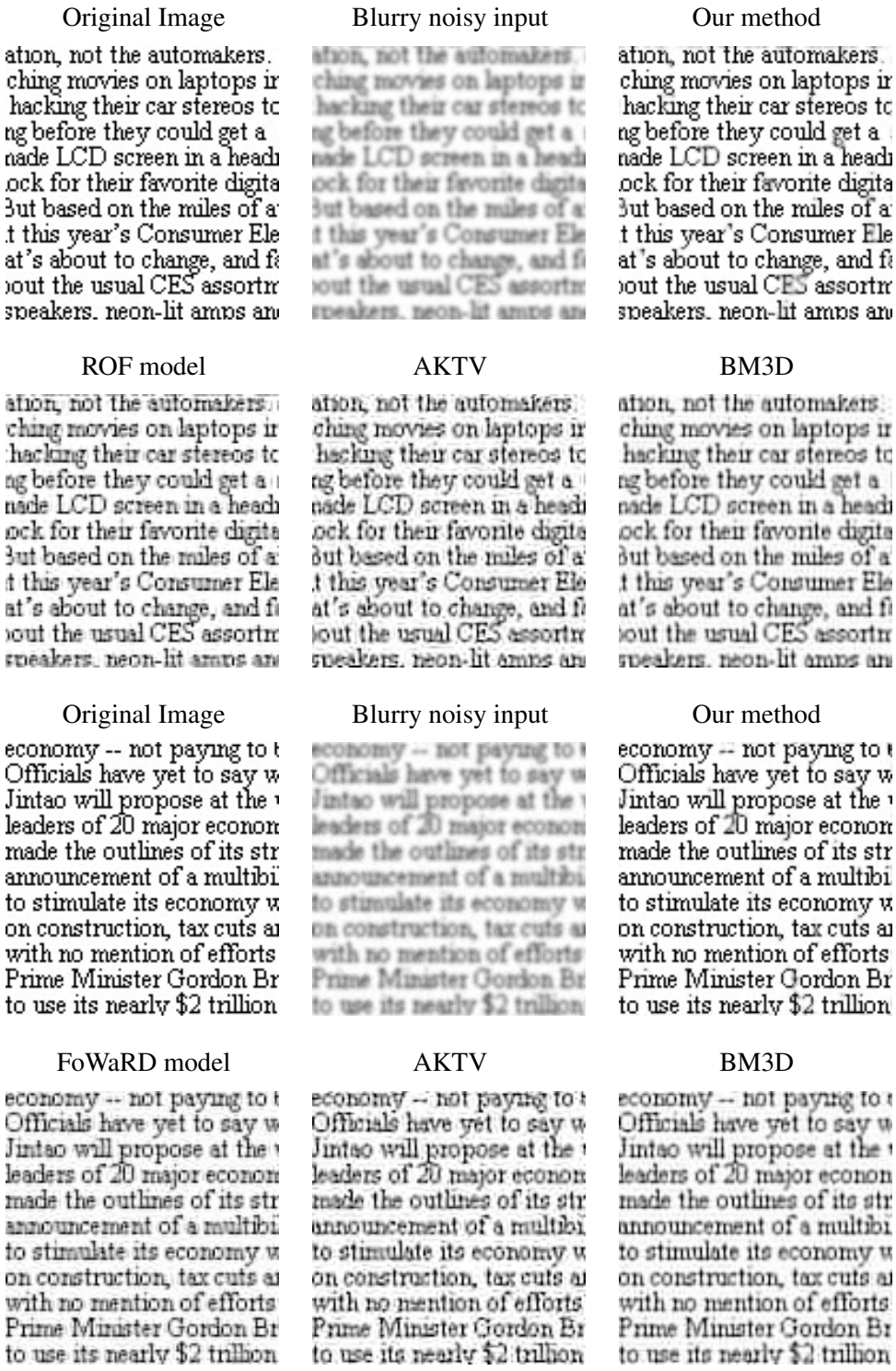


Figure 3.4: Text Deblurring with 20,000 dictionary elements.

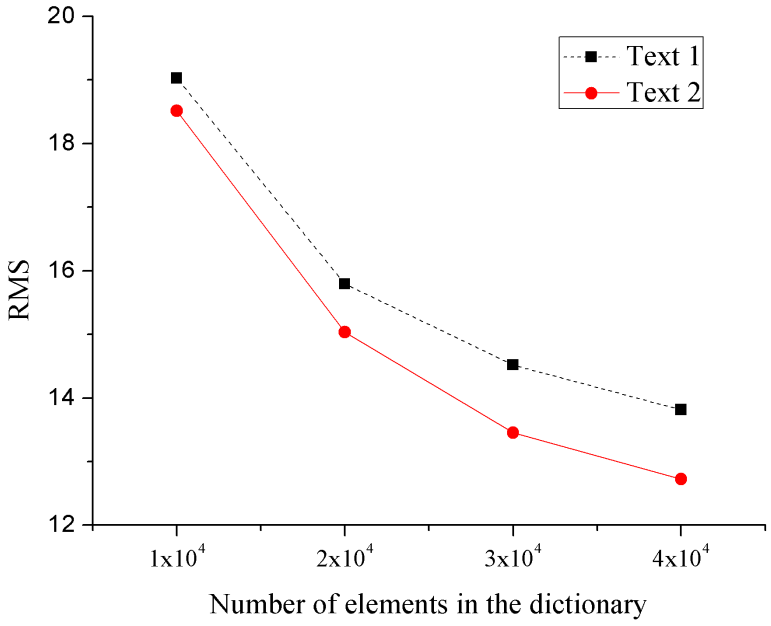


Figure 3.5: Influence of the number of the elements in the dictionary on the deblurring performance. The average of 10 experiments for each column is reported.

nel with $\sigma_{\text{data}} = 1$ plus noise. The clear dictionary is comprised of randomly sampling 10,000 patches from the training set. There are two cases for the blurring kernels to construct the blurred basis.

Case A $\sigma_{\text{dict}} = 0.5, 1, 1.5$: one of them happens to be the exactly same as σ_{data}.

Case B $\sigma_{\text{dict}} = 0.6, 0.9, 1.2$.

The results for both cases are presented in Fig. 3.6, along with the non-blind deconvolution. Case A is almost as good as non-blind with slightly worse RMS.

3.3.3 General Case

As in the case of super-resolution [138], the dictionary consists of random samples from the training images, which have statistics similar to the test image. Here we con-

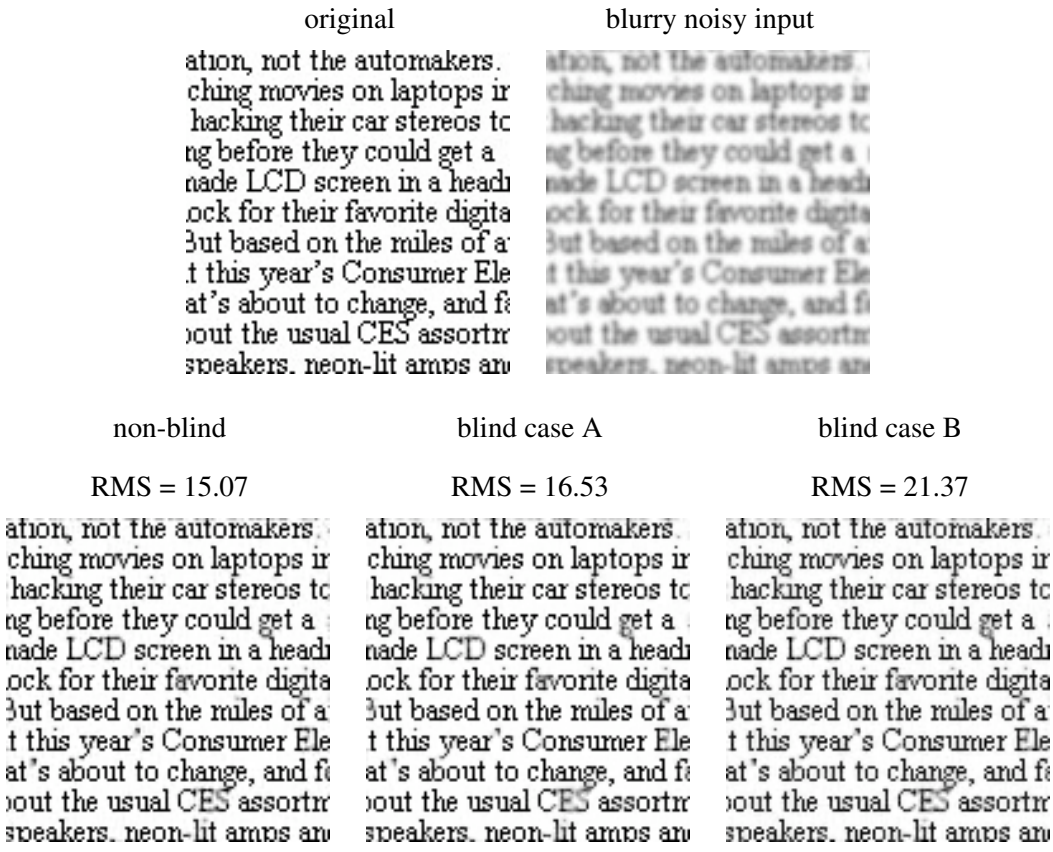


Figure 3.6: Blind deconvolution with comparison to non-blind case.

sider three images: "Rose," "Koala," and "Castle." The training images are taken from the image datasets of flowers, animals and architecture respectively, while excluding the test ones. Fig. 3.7 shows several examples in each training set. Flower images are from the Internet, while the other images are from the Berkeley Segmentation Dataset [95]. For each category we randomly sample 20,000 patches of size 16×16 to form the dictionary.

The input data are corrupted by convolving with a 9×9 Gaussian kernel of $\sigma = 1$ with additive noise whose standard deviation is 5. As shown in Fig. 3.8, ROF returns piecewise constant images, while ForWaRD produces noticeable artifacts in the

Figure 3.7: Examples in the training images.

reconstruction.

3.3.4 Trained Dictionary

We conduct a deblurring experiment of a trained dictionary. We cut a texture image into half, one as training and the other as testing. We take all the overlapping 16×16 patches in the training image to train a dictionary that has 1024 atoms. The dictionary, as shown in Fig. 3.9, is trained via KSVD [1]. We blur the test image with a Gaussian kernel of $\sigma = 2$ plus additive noise. As a comparison, we also construct a generic dictionary, which is comprised of 20,000 random samples from the training image. Fig. 3.10 summarizes the results. Our method using either dictionary improves upon traditional methods, with the generic dictionary providing further improvement over the trained one. This is because that training a dictionary introduces blurring effects

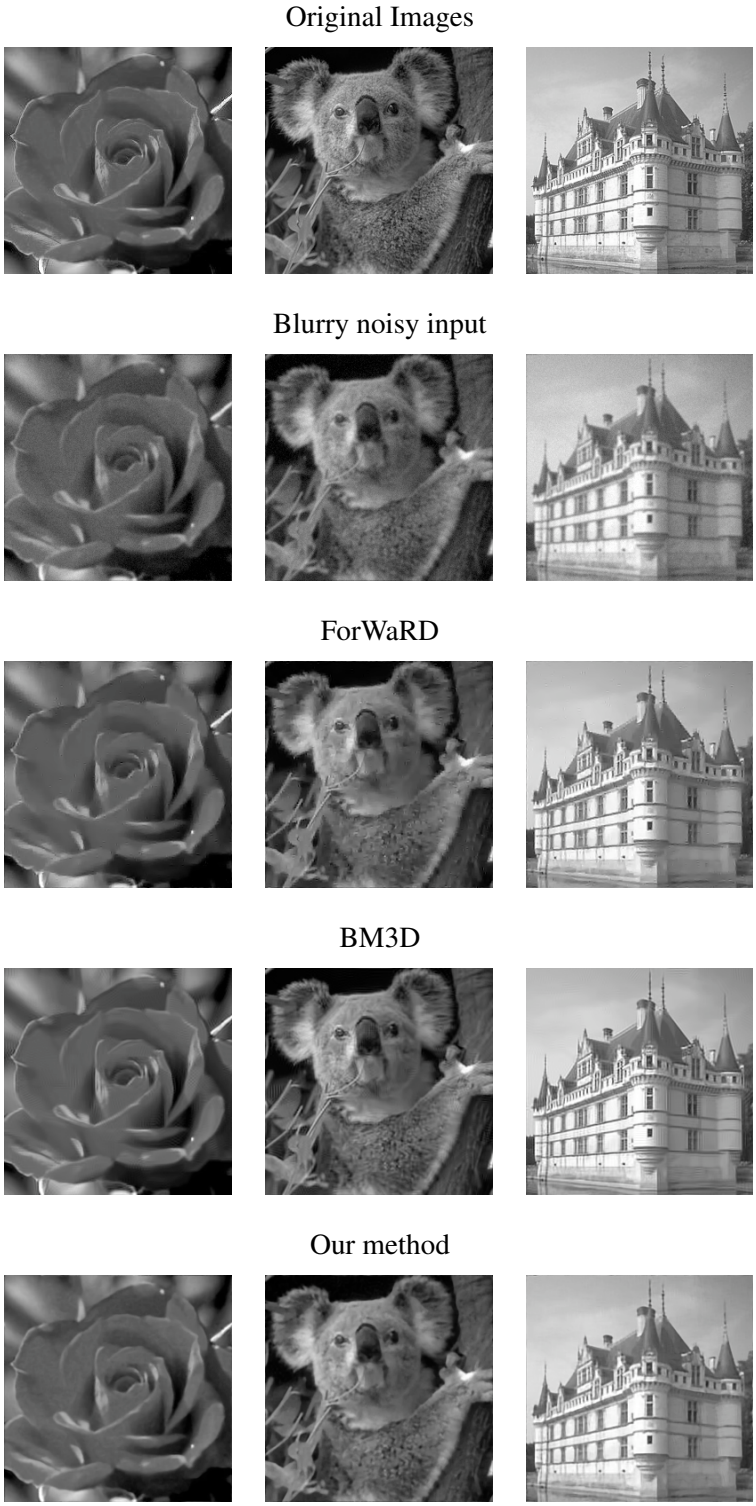


Figure 3.8: Grayscale image deblurring with 20,000 dictionary elements.

Table 3.1: RMS errors for different methods. (G) and (T) indicate what kind of dictionary to use for our method with (G) for generic and (T) for trained. In some case, BM3D is marginally better than our method (by about 10%), while in other cases our method fares significantly better (three times better in the Text examples and 50% better in the Texture).

Image	ROF	ForWaRD	AKTV	BM3D	Our method
Text 1	65.43	62.75	60.68	69.87	**14.60**
Text 2	63.59	61.78	57.93	70.33	**13.81**
Rose	6.68	5.75	4.90	**4.83**	5.62
Koala	9.80	8.98	8.45	**7.97**	8.64
Castle	14.23	12.82	12.51	**11.77**	13.62
Texture	14.46	14.54	12.91	12.13	**8.72** (G) 10.37 (T)

on the dictionary elements.

A quantitative comparison is provided in Table 3.1. In some case, BM3D is marginally better than our method (by about 10%), while in other cases our method fares significantly better (three times better in the Text examples and 50% better in the Texture).

3.4 Discussion

3.4.1 Coherence

The precision and stability of our "direct" approach depends on the smoothness of the blurring kernel and the geometry of the dictionary. The latter is roughly measured by the concept of *coherence*, which is defined to be the maximum absolute inner product

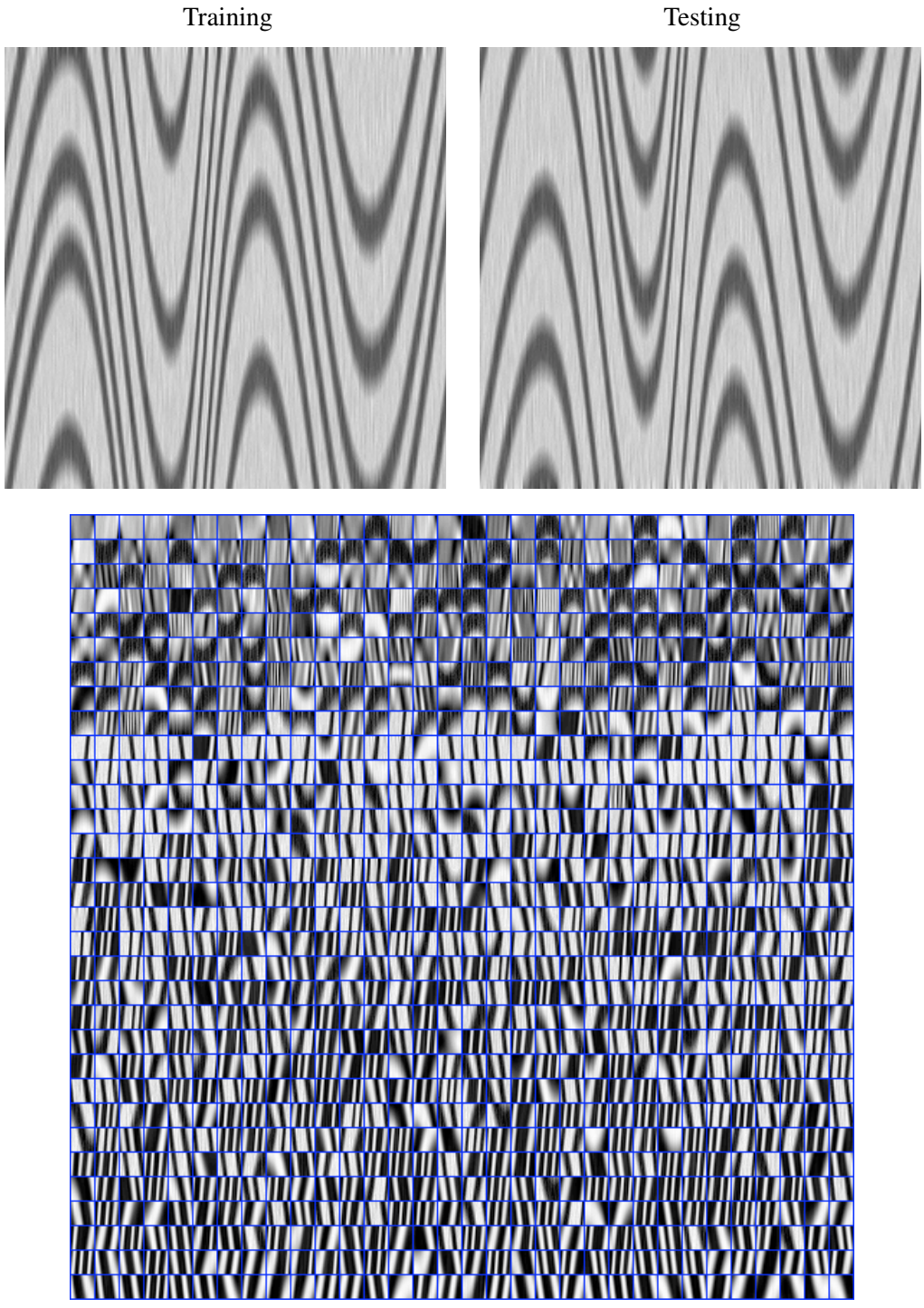


Figure 3.9: The dictionary is trained using all the 16×16 patches in the training image (top left), which has a similar structure to the test one.

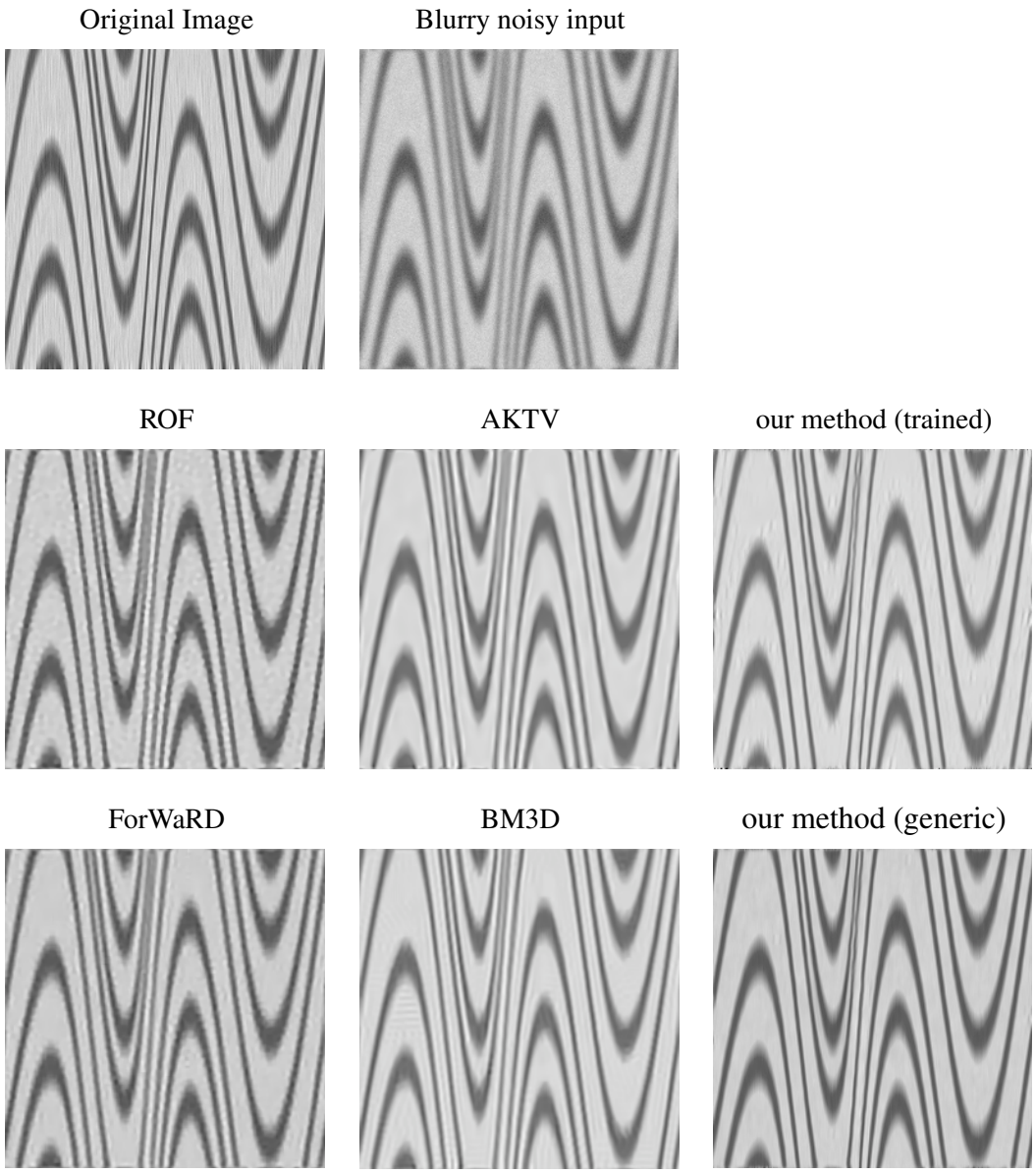


Figure 3.10: The dictionary is either trained or generic (comprised of random samples from the training image). Our method using either dictionary improves upon traditional methods, with the generic dictionary providing further improvement over the trained one.

between two distinct vectors in the dictionary [131]. If the coherence of a dictionary is large, it is difficult for the sparse coding algorithms to choose the best atoms.

The coherence of a clear dictionary is usually larger than the one of the blurred dictionary. For example, it is very likely that two blurred atoms b_i and b_j are similar or even identical, but their clean versions d_i and d_j are completely different. In this case, the algorithm may confuse d_i with d_j, leading to a large deblurring error. As a result, the coefficients $\hat{\alpha}$ recovered by sparse coding the blurred image f relative to the blurred basis $\{b_k\}$ could be very different from the true coefficients α of the clean image, and the deblurring estimation error can be thus inaccurate.

The amount of blur we can handle is limited by how distinctive the dictionary atoms are. For example, deblurring text images and texture yields very good results, since the dictionaries of these two cases are distinctive and the coherence of the blurred dictionary is more or less the same to the one of the clear dictionary. On the other hand, the results of "Rose," "Koala," and "Castle" imply that the associated dictionaries can only deal with smaller smoothing kernels.

3.4.2 Domain Overlapping

We want to point out the problem in minimizing the upper bound to the original formulation. Ideally, pixels that overlap with many blobs should be jointly coded, but it is computationally expensive. Instead, we adapt the procedure in [1, 91] to code the pixels multiple times and average each encoding. However, it is the averaging that in turn degrades the image reconstruction. One could code non-overlapping patches independently, but there is no guarantee for the smooth transition between neighboring patches. Yang *et. al.* [138] process the patches in a raster-scan order with one additional constraint in the sparse coding step, that is, to enforce the overlap between the current target patch to match with previously reconstructed ones. This amounts to

adding a linear equation into the optimization, thus easy to solve. However, it is not satisfactory since the results depend on the order of the scan. A better approach would consist in using a partition of unity of the domain of the image to trade off boundary artifacts while avoiding multiple encoding of the same pixel. We intend to pursue this approach as part of our future work.

CHAPTER 4

A Nonlocal Framework

In computer vision, we are used to more explicit models of the underlying scene, and even simple ones such as "cartoon models" [104, 140], occlusion "layers" [135], multi-resolution and scale-space processes [82] have ramifications in image processing. However, one could argue that the image formation process is unduly complex, and modelling it explicitly just to remove noise or increase the resolution is overkill. This philosophy is at the core of so-called "exemplar-based methods," [60]: Instead of explicitly modelling the image-formation process, one can just "sample" its effects and manipulate the samples to yield the desired inference result. In the simpler forward problem, that of image synthesis, this philosophy has yielded so-called "procedural methods" in computer graphics, that have been rather successful especially in synthesizing complex textures (see [136] and references therein). The basic idea is that – given a sample in the form of an image patch – one can generate new textures, or expand the sample, by searching portions of it that match the periphery, then translating them and "appending" them to the given sample. In the inverse problem of image analysis, one would search for patches similar to a given one, then transform them to overlap and then compute some statistic out of these samples, for instance the average or median, to perform denoising, or to resample the grid to obtain a super-resolution image. This is the basic idea underlying nonlocal image denoising approaches that have recently surged in popularity in the image processing community [20]. Its advantage is to exploit similar patches in the same image, without an explicit model of the

image formation process.

In order to denoise a pixel, it is better to average the nearby pixels with similar structures (patches). The resemblance is regarded in terms of a patch centered at each pixel, not just the intensity of the pixel itself. The mathematical formula of nonlocal means (NLM) is given as follows,

$$NL_v(x) := \frac{1}{C(x)} \int_\Omega w_v(x,y) v(y) dy, \tag{4.1}$$

where v is the reference image, the weight function $w_v(x,y)$ and the normalizing factor $C(x)$ have the form:

$$w_v(x,y) = \exp\left(-\frac{(G_a * |v(x+\cdot) - v(y+\cdot)|^2)(0)}{h^2}\right), \tag{4.2}$$

$$C(x) = \int_\Omega w_v(x,y) \mathrm{d}y, \tag{4.3}$$

where G_a is the Gaussian kernel with standard deviation a and h is a filtering parameter. The parameter a defines the dimension of the patch where we measure the similarity of two patches, while the parameter h regulates how strict or relaxed we are in considering patches similar. In general h corresponds to the noise level; usually we set it to be the standard deviation of the noise. The weights are significant only if the window around y looks like the corresponding window around x. Thus self-similarity is used to reduce the noise. Aimed at denoising, the authors [20] consider using the noisy data f as the reference image to construct the weight and by this weighed averaging, the structures, *e.g.* edges, are reinforced, while the noise gets canceled out. There are several variants based on the idea of the nonlocal means filter such as [89, 76, 75, 61, 17, 28]. Nonetheless, all the methods interpret the concept of "similarity" only up to translation, while we extend it to a more general similarity transformation, *i.e.*, scaling and rotation in Sect. 4.2.1.

The application of the non local means filter to different image processing tasks is an active research area [21, 62, 14]. However, it remains challenging to extend previ-

ous non local models to general inverse problems successfully. For example, the same authors use neighborhood filters to stabilize the inverse heat equation [21], following the direction in the PDE community to formulate Gaussian blurring as diffusion [77, 72, 73]. But this method does not work very well, since observed data and original images do not necessarily have same similarity distribution and structures. We propose a key step for nonlocal methods to work efficiently in the general inverse problem, which is to use a crude solution to build the weight function. In addition, we follow the framework [62] using nonlocal operator to solve the general problem.

This chapter is organized as follows. A mathematic framework for nonlocal operators is studied in Sect. 4.1. Sect. 4.2 is devoted to the weight computation, which involves two extensions to the original NLM. Finally Sect. 4.3 contains experiments from various applications including denoising, super-resolution, image deconvolution and tomographic reconstruction.

4.1 Nonlocal Operators

Here we review the work from [62]. Let $\Omega \subset \mathrm{R}^2, x, y \in \Omega,\ w(x,y)$ be a weight function, which is nonnegative and symmetric. Assuming the weights are pre-determined and considered as constants, we define

- nonlocal gradient $\nabla_w u : \Omega \to \Omega \times \Omega$

$$(\nabla_w u)(x,y) := (u(y) - u(x))\sqrt{w(x,y)}.$$

- nonlocal divergence $\mathrm{div}_w \overrightarrow{v} : \Omega \times \Omega \to \Omega$

$$(\mathrm{div}_w \overrightarrow{v})(x) := \int_\Omega (v(x,y) - v(y,x))\sqrt{w(x,y)}\mathrm{d}y.$$

It reduces to traditional gradient and divergence when the weight is the inverse square distance between neighboring pixels.

Two types of regularization functionals are designed based on the nonlocal operators.

$$J_{NL/H^1}(u) = \frac{1}{4}\int |\nabla_w u|^2 = \iint_{\Omega\times\Omega} (u(x)-u(y))^2 w(x,y)\mathrm{d}x\mathrm{d}y, \tag{4.4}$$

$$J_{NL/TV}(u) = \int |\nabla_w u| = \int_\Omega \sqrt{\int_\Omega (u(x)-u(y))^2 w(x,y)\mathrm{d}y}\mathrm{d}x. \tag{4.5}$$

Note that the functional in (4.4) is analogous to the standard H^1 semi-norm, so it is denoted as NL/H^1; similarly the one in (4.5) is denoted as NL/TV.

We calculate the Euler-Lagrange of the functionals above, thus obtaining

$$L_{NL/H^1}u = -\int_\Omega (u(y)-u(x))w(x,y)\mathrm{d}y, \tag{4.6}$$

$$L_{NL/TV}u = -\int_\Omega (u(y)-u(x))w(x,y)\left[\frac{1}{|\nabla_w u(x)|}+\frac{1}{|\nabla_w u(y)|}\right]\mathrm{d}y. \tag{4.7}$$

We use the nonlocal regularization functionals in (4.4) and (4.5) to perform image reconstruction by defining the total energy as

$$E(u) = J(u) + \frac{\lambda}{2}\int (\mathcal{K}u - f)^2, \tag{4.8}$$

As usual, one can resort a steepest descent method to compute the solution,

$$u_t = -Lu + \lambda\tilde{\mathcal{K}}^*(f-\mathcal{K}u), \tag{4.9}$$

where the operator L is the corresponding gradient flow with respect to the functional J, that is either (4.6) or (4.7), and $\mathcal{K}^*$ is the adjoint of $\mathcal{K}$. There are a lot of solvers for this minimization, such as [146] and similar technique to split Bregman for L^1 minimization [64].

The basic properties of the linear operator L_{NL/H^1} are studied in [61], making it a continuous generalization of graph Laplacian. Therefore, most well-established methods in the PDE community can be extended naturally to a nonlocal manner. In fact, $J_{NL/TV}$ is the nonlocal extension of total variation.

This nonlocal framework is associated with a fixed weight. The dependence of w on the unknown image u is possible to adapt. The difficulty lie in the complicated expression of gradient operator L. Several groups of researchers develop optimization schemes to update the weight during the iteration, such as [17, 18, 14]. Although computing the weight is the most time-consuming part, it is worth the effect especially in the applications of compressed sensing [146].

4.2 Computing the Weights

The weight computation is crucial in the nonlocal framework. We will discuss two independent approaches to compute the weights, as extension to the original Non-local means [20]. Sect. 4.2.1 considers a similarity-invariance weight, while in Sect. 4.2.2 preprocessed data is used to compute the weights in the applications of image deconvolution and tomographic reconstruction.

4.2.1 Similarity-Invariant Weights

It is standard to consider the reference image v in eq. (4.2) to be the input data f in the application of image denoising. In the next section, we discuss an alternative. In the discrete case, the weight can be interpreted as the L^2-norm of the difference of f_x (*i.e.*, f centered in x) and f_y (*i.e.*, f centered in y), weighted against a Gaussian window G_a. In other words, the distance $\|f_x - f_y\|$ measures how similar are two patches of f centered at x and y. If two patches are similar, then the corresponding weight $w_f(x, y)$ will be high. Vice versa, if the patches are dissimilar, the weight $w_f(x, y)$ will be small (but positive). The final result of the nonlocal means filter is that several (similar) patches are used to reconstruct another one.

Notice that the similarity of patches is defined up to translation. In other words,

we can only match patches that are simply in different locations, but otherwise unchanged – with the same orientation and scale. This motivates us to consider the larger class of similarity measures that discounts scale and rotation changes, *i.e.*, a similarity-invariant measure. In theory, defining this measure is just a matter of introducing two more integrals and an inverse similarity-transformation in eq. (4.2) to align the patches being averaged. In practice, however, because this similarity has to be computed multiple times for each patch, this introduces considerable computational burden that makes the ensuing algorithm all but impractical. One way to address this problem is to find a function that estimates a rotation and a scale at each patch with respect to a common reference system, so that each patch can be transformed into a "canonical" patch. Once this is done, one can apply the original nonlocal means filter.

The idea of determining when two regions are similar up to a similarity transformation has been widely explored in the past to solve several tasks including object recognition, structure from motion, wide-baseline matching, and motion tracking [115, 87, 132, 96, 134]. We will exploit the same idea of matching similarity-invariant regions for the purpose of image denoising.

One of the most successful methodologies to match regions up to a similarity transformation is the Scale Invariant Feature Transform (SIFT) [87] (Please refer to Chapter 2.3.1). There is a fundamental difference in how SIFT is commonly used and how it is employed in our algorithm. In our case the *keypoint localization* step is not implemented as we are interested in computing a SIFT descriptor and in obtaining some consistent estimate of scale and orientation at each pixel. From now on, therefore, we will define our SIFT filter to estimate scale $\rho(x) : \Omega \mapsto [0, \infty)$ and orientation $\theta(x) : \Omega \mapsto [0, \pi]$ respectively.

In Figure 4.1, we illustrate step-by-step how we align the patch to its canonical form: for each pixel x,

1. Take a patch with size $\sim 10\rho(x)$ around the pixel;

2. Rotate this patch with the angle $\theta(x)$;

3. Extract the middle part of size $\sim 7\rho(x)$ for the boundary problem after rotating;

4. Down-sample to a uniform size (the smallest size among all patches) and save as $P(x)$.

In this way we can extract more meaningful patches than in previous nonlocal means methods, as shown in Figure 4.2. Since we assume additive Gaussian white noise, noise is invariant to rotation and scaling if the image is considered to be a continuous function. When aligning the patches, there are interpolation errors, but they are negligible two-pixels away from the center, if bilinear interpolation is used. We mitigate scale errors by using only patches that are larger, and therefore at higher resolution, than the reference patch.

We reformulate the weight function to be similarity-invariant,

$$w_f(x,y) = \exp\{-\frac{\|P(x) - P(y)\|^2}{h^2}\}\ , \tag{4.10}$$

where $P(x)$ is the canonical form of the patch center at x and h is a parameter as in the Non-local means.

4.2.2 Preprocessing the Data by Linear Models

The application of nonlocal means to a general inverse problem is not direct since observed data and original images do not necessarily have same similarity distribution and structures. In the case of deconvolution, based on the hypothesis that the deblurred image must maintain the same similarities as the blurry image, Buades *et. al.*. proposed a non-local deblurring model in [21]:

$$u = \arg\min_u \int (u - NL_f(u))^2 + \frac{\lambda}{2}\int (f - \mathcal{K}u)^2,$$

where NL_f is defined in eq. (4.1) with the reference to be the blurry noisy data, *i.e.* $v = f$. This hypothesis has a strong limitation since it forces the deblurred image to maintain the same coherence as the given blurred one.

We propose a key step for nonlocal methods to work effectively in the general case, which is to use a crude solution of the inverse problem to construct the weight. This solution can be obtained by any fast image restoration method, e.g. Tikhonov regularization [128] for image deconvolution and Filtered Back Projection (FBP) [3] for tomography. This step amounts to preprocessing the data to exploit the useful spatial information in image. It is more important for tomography, since the observed data and the image lie in different spaces.

We consider the weight function as eq. (4.2), which is the same formula in [20] and [21], except that we use a preprocessed image as the reference to construct the weight. Another difference is that we do not require normalization of the weight since we formulate a minimization problem. The details of the preprocessing are given separately, since it varies from applications.

4.2.2.1 Image deconvolution

In this case, the general model $\mathcal{K}u$ amounts to the convolution between the image u and a circular shift-invariant blur kernel g. Furthermore, periodic boundary conditions for the convolution are assumed, thus the fast Fourier transform can be applied to realize the convolution.

A better way to construct the weight function (4.2) is to use a preprocessed image instead of the blurry and noisy data f. Ideally the preprocessed image should be sharper than the blurry image and easy to compute. For example, there are two common examples of linear regularizations to solve the image recovery problem. One is called Tikhonov regularization with identity [128], which is to solve the following

minimization problem:

$$u = \arg\min \int (f - g * u)^2 + \mu \int u^2 \,, \tag{4.11}$$

where $\mu > 0$ is a regularizing parameter. The variational formulation gives us

$$g^* * (g * u - f) + \mu u = 0 \,,$$

where $*$ is the conjugate operator. An alternative regularization is to replace the L^2 norm in the last term in (4.11) with H^1 semi-norm. The second approach is called Wiener filter [2] for image deconvolution:

$$u = \arg\min \int (f - g * u)^2 + \nu \int |\nabla u|^2 \,. \tag{4.12}$$

These linear methods are very simple and fast to implement, since they only involve one step of calculation, *e.g.*, the fast Fourier transform. However, in order to compensate for the noise, a large regularizing parameter μ or ν is necessary, which tends to smear out the edges.

Although these methods suffer from amplifying the noise frequencies, the nonlocal framework can automatically handle their side-effects. Since the noise statistics of the preprocessed image is changed, we should alter the parameter h in the weight function to compensate for the noise accordingly. We choose Tikhonov regularization to be the preprocessor considering that it produces sharper results than Wiener filter, which is to minimize the H^1 semi-norm.

The minimizer of the linear model eq. (4.11) can be obtained in the frequency domain (ξ, η) :

$$\hat{u}(\xi, \eta) = \frac{\hat{g}^*(\xi, \eta)\hat{f}(\xi, \eta)}{|\hat{g}|^2(\xi, \eta) + \mu} \,, \tag{4.13}$$

where $\wedge$ denotes the Fourier transform.

In the case where we know some statistics of additive noise, *e.g.* the standard deviation of noise, a proper parameter μ is chosen such that the restored image u in eq.

(4.13) satisfies

$$\iint (f - g * u)^2 dxdy = \iint n^2 dxdy = \sigma^2 \ .$$

By Parseval's theorem, this is equivalent to finding μ satisfying

$$\sigma^2 = \iint (f - g * u)^2 dxdy = \iint |\hat{f} - \hat{g}\hat{u}|^2 d\xi d\eta \ \ = \ \ \iint \frac{\mu \hat{f}(\xi, \eta)}{|\hat{g}|^2 + \mu} d\xi d\eta \ .$$

Notice that the right-hand side is monotonically increasing function in μ, hence there exist a unique solution μ, which can be determined via bisection.

Since the above procedure describes a linear operator applied to f, we can denote Inv_g. It follows from the image degradation model (1.1) that

$$\tilde{f} \ \ = \ \ \text{Inv}_g \circ f = \text{Inv}_g \circ (g * u + n) \ . \tag{4.14}$$

Note that after preprocessing the data, the noise has L^2 norm

$$||\text{Inv}_g \circ n||_{L^2} \leqslant ||\text{Inv}_g||_{L^2} ||n||_{L^2} = \sigma ||\text{Inv}_g||_{L^2} \ .$$

Hence a good choice is

$$h = \sigma ||\text{Inv}_g||_{L^2}. \tag{4.15}$$

In summary, the reference image in the weight function (4.2) is $\tilde{f}$ defined in eq. (4.14), and the parameter h is in eq. (4.15).

4.2.2.2 Tomographic reconstruction

We provide another example of the non-local functional model in the context of Computerized Tomography (CT). In a simplified parallel tomographic problem, an observed body slice is modelled as a two-dimensional distribution $(x, y) \mapsto u(x, y)$ of the x-ray attenuation constant, and a line integral called projection $p(r, \theta)$ represents the total attenuation of a beam of x-rays parameterized by (r, θ), where $r \in \mathbb{R}$ is the signed perpendicular distance from the line to the origin and $\theta \in [0, \pi)$ is the angle

between the perpendicular vector and the x-axis. As shown in Fig. 4.3, a projection obtained by illuminating the object along the line is given by

$$Ru : p(r,\theta) = \iint_{-\infty}^{\infty} u(x,y)\delta(x\cos\theta + y\sin\theta - r)dxdy,$$

where δ denotes the Dirac delta-distribution. The linear operator $R : u(x,y) \mapsto p(r,\theta)$ is called Radon transform. The tomographic reconstruction problem is to estimate the distribution $u(x,y)$ from a finite number of measured line integrals $p(r,\theta)$. The standard reconstruction algorithm in clinical applications is the Filtered Back Projection (FBP), which is a direct discretization of an analytical formula for the inverse of the Radon transform. As the name indicates, this method can be decomposed into two steps: (1) the one-dimensional projection data along each orientation θ is applied via a *ramp filter*; (2) each pixel of the image u is obtained by summing on those filtered projections passing through this pixel. In fact, the last step is actually an adjoint transform of the Radon transform [3], which is called *back projection*. In the presence of noise, this problem becomes extremely unstable since the inverse of the Radon transform is unbounded. Traditionally, a low-pass filter has to be applied to compensate the noise amplification by ramp filter. However this step smears out edges, where important structures are located.

The weight function $w(x,y)$ is required to measure the similarity of image features between pixels x and y. Since the observed projection data are in the Radon domain, which do not directly reflect the similarity information of the image, we need a more accurate reference image to compute the weight (4.2). We consider the FBP image as a crude and fast solution to build the weight. Moreover, we choose the filter parameter h to be the estimated noise variance in the filtered back projection image. Here we use a wavelet-based noise estimation model introduced by Donoho *et. al.* [41] in 1994. More precisely, we first apply a wavelet transform on the filtered back projection image, then the noise variance of this image is estimated by using the medial absolute

deviation of the wavelet coefficients at its finest scale, *i.e.*:

$$\hat{\sigma}^2 = \frac{\text{median}(|y_i|)}{0.6745}, \tag{4.16}$$

where y_i are wavelet coefficients in the finest subband of the filtered back projection image. Therefore, the parameter h is chosen to be $h = \hat{\sigma}$. An alternative choice is to compute the norm of an discrete inverse of the Radon transform.

4.3 Applications

We will discuss various applications of this nonlocal framework including image denoising, super-resolution, image deconvolution and tomographic reconstruction. The first two are solved by the similarity-invariant weight as established in Sect. 4.2.1, while the rest are based on the preprocessing data as in Sect. 4.2.2.

4.3.1 Denoising

The total energy for the denoising model is defined in the following

$$\hat{u} = \arg\min_u J(u) + \frac{\lambda}{2}\int (f(x) - u(x))^2 \,\mathrm{d}x \,, \tag{4.17}$$

where J is the NL/H^1 functional in (4.4) for simplicity. To minimize the energy (4.17), we apply a gradient descent flow:

$$u_t(x) = -\int (u(x) - u(y))w(x,y)\mathrm{d}y + \lambda(f(x) - u(x)) \,.$$

Notice that the above equation is linear in $u(x)$, so it is easy to implement an implicit time difference scheme, which makes the iterations more stable.

$$\frac{u^{n+1}(x) - u^n(x)}{dt} = \\ -\int (u^{n+1}(x) - u^n(y))w(x,y)\mathrm{d}y + \lambda(f(x) - u^{n+1}(x)) \,.$$

We can also extend this model to color image denoising in which the input image $\mathbf{f} := (f^R, f^G, f^B)$ is a three-channel signal. In a similar way, we can compute the weight $w_{\mathbf{f}}(x, y)$ using high-dimensional patches so that the weight is the same for all color channels. We express the total energy as follows,

$$\hat{u} = \arg\min_{u} \sum_{j=R,G,B} \int \left(u^j(x) - u^j(y)\right)^2 w_{\mathbf{f}}(x, y) \mathrm{d}x \mathrm{d}y + \frac{\lambda}{2} \int \left(f^j(x) - u^j(x)\right)^2 \mathrm{d}x \ .$$

Notice that we can perform color image denoising by treating the three color channels independently.

We first compare the performance of our method to that of the PDE-based method [25], the wavelet-based method [110] and the original nonlocal means [20]. Other denoising methods are examined and compared in [20].

We present the nonlocal similarity filtering on two synthetic images which are corrupted by additive Gaussian noise with standard deviation $\sigma = 20$ (Fig. 4.4) and $\sigma = 40$ (Fig. 4.5) respectively. For each method, the residual image $f - u$ is shown. The residual is called *method noise* in [20]. It is a visual measurement of the denoising scheme, since it should be as similar to a white noise as possible. Both the PDE-based method [25] and the wavelet based method [110] fail to preserve structures as they are left in the residual image. The traditional nonlocal method fails to denoise the central part in Fig 4.4 since these regions in the residual image are almost flat.

In Fig. 4.6 we also compare our approach with NLM and its iterated version by T. Brox and D. Cremers [17]. We crop the noisy image and the results of iterated nonlocal mean directly from their paper and perform denoising on this noisy input by the original nonlocal means and our approach. We do a better job on the fish and lena examples, however we cannot beat them for textured images since the scale and orientations are inaccurate in this case.

An example of color image denoising is presented in Fig. 4.7. In computing the

Table 4.1: RMS errors for the input images and different denoising methods.

RMS	Input	PDE-based [25]	wavelet-based[110]	NLM [20]	ours	Iterated NL [17]
fish	30.24	17.31	**13.03**	17.34	14.53	16.01
Lena	30.95	17.82	17.21	12.65	**11.97**	12.57
brodatz	39.77	32.06	23.35	**21.92**	23.66	22.93
testpat	20.00	14.65	12.56	9.80	**6.74**	N/A
letter	40.00	20.05	13.57	12.42	**11.66**	N/A
flag (color)	40.00	13.44	12.83	10.43	**9.11**	N/A

weight, the L^2 distance between 3-D patches (RGB) is used. As for denoising, we treat the three color bands independently. The results are presented in Fig. 4.7, which shows that our approach works better for stripes, while it is comparable to the original NLM for the stars. We compare the quantitative evaluation of various denoising methods. Table 4.1 lists the root mean square (RMS) error of each method.

4.3.2 Super-resolution

In this section, we investigate super-resolution as an application of our similarity non-local model. The low resolution noisy image f_{LR} is corrupted by

$$f_{\text{LR}}(x) = D \circ u(x) + n(x) \, , \tag{4.18}$$

where u is the original image, n is white noise and D is the downsampling operator. Our goal is to reconstruct a high resolution image u from the low resolution noisy one. We use linear interpolation of f_{LR} to get a high resolution image f_{HR}. We take the same procedure as denoising to build the weight function $w_{f_{\text{HR}}}(x, y)$. Now the data fidelity

term becomes

$$E_{\text{data}} = \int_{\Omega_{\text{L}}} (f_{\text{LR}}(x) - D \circ u(x))^2 \mathrm{d}x \ , \tag{4.19}$$

where Ω_{L} is the low resolution image domain. Its Euler-Lagrange equation is calculated

$$\partial E_{\text{data}} = 2S \circ (f_{\text{LR}}(x) - D \circ u(x)) \ , \tag{4.20}$$

where S is the transpose of the operator D, *i.e.*, upsampling. Notice that $S \circ f_{\text{LR}}$ is nothing but f_{HR}. Defining $T = S \circ D$, the overall gradient flow is

$$u_t(x) = -\int (u(x) - u(y)) w_{f_{\text{HR}}}(x, y) \mathrm{d}y + \lambda (f_{\text{HR}}(x) - T \circ u(x)) \ , \tag{4.21}$$

and its steady state yields a high resolution and denoised image.

We present the results of super-resolution by both the original nonlocal means and the similarity nonlocal filtering in Figure 4.8. The low-resolution noisy image is downsampled by a factor of two and then corrupted by additive Gaussian noise with $\sigma = 20$. Our method returns sharper edges than the original NLM.

4.3.3 Image deconvolution

In this section we present some numerical results using the nonlocal functionals: NL/H^1 and NL/TV. We compare these to traditional methods, such as Tikhonov regularization [128], ROF [114] and the NL-means deblurring model [21]. As for the nonlocal weight, we compute it from either the noisy blurry image or the preprocessed image.

We define the total energy as

$$E(u) = J(u) + \frac{\lambda}{2} \int (g * u - f)^2, \tag{4.22}$$

We use gradient descent to update the solution by the Euler-Lagrange of (4.22)

$$u_t = -Lu + \lambda \tilde{g} * (f - g * u), \tag{4.23}$$

Table 4.2: The statistics of blur and noise we add to the images.

Image	blur kernel	Gaussian noise
Shape	Gaussian with $\sigma_b = 2$	$\sigma_n = 10$
Barbara	Gaussian with $\sigma_b = 1$	$\sigma_n = 5$
Cameraman	9×9 box average	$\sigma_n = 3$

where the operator L is the corresponding gradient flow with respect to the functional J, that is either (4.6) or (4.7), and $\tilde{g}$ is the adjoint of g.

We use signal-to-noise ratio (SNR) as a means of judging performance

$$SNR(u, f) = 20 \log_{10} \left\{ \frac{||f - \bar{f}||_{L^2}}{||f - u||_{L^2}} \right\},$$

where u is the original image, f is the recovery and $\bar{f}$ is its mean value.

We test all the methods on three images: a synthetic image, Barbara and Cameraman with various kinds of blur and noise as listed in Tab 4.2. The synthetic one is referred to as Shape since it contains geometric features. The Cameraman image is high contrast and has many edges, while the image of Barbara contains more textures.

We use Tikhonov regularization to preprocess the data with the method parameter μ determined by the standard deviation of the noise. Then we use this preprocessed data to compute the weight function with filter parameter h chosen to compensate the amplified noise. Please refer to Sect. 4.2.2 for details. Once we have the weight, we solve the minimizer of (4.22) with either NL/H^1 or NL/TV as regularization via gradient descent. For each method, we present our result with an optimal parameter λ chosen from a series of values with wide range.

Regardless of which way we compute the nonlocal weights, visually NL/H^1 returns a smoother image than NL/TV does. We list the SNR of all the methods in Tab. 4.3. For Fig. 4.9 and Fig. 4.11, the best reconstruction is the one that combines a

Table 4.3: SNR for different methods

Image	Shape	Barbara	Cameraman
noisy input	11.6213	10.0363	8.4600
Tikhonov [2]	13.1970	10.5680	11.2940
ROF [114]	14.2192	11.4257	12.3448
NLM deblur [21]	16.3289	12.0903	12.4269
NL/H^1	16.8084	11.7952	12.4599
NL/TV	17.5251	12.0066	12.7495
Tikhonov + [21]	17.5269	**12.7369**	13.5099
Tikhonov + NL/H^1	19.6982	12.1182	13.5185
Tikhonov + NL/TV	**20.9401**	12.4616	**13.6194**

preprocessor by Tikhonov filter with NL/TV regularization functional both in terms of SNR and direct inspection as well. Although Tikhonov+NLM deblur achieves the highest SNR in Barbara example, Tikhonov+NL/TV restores most textures especially in the bottom left corner as zoomed in Fig. 4.10.

For computational time, it takes about 2.9 seconds for a dual core desktop with 2.99GHz processor and 1.99GB memory to construct the weight function of a 256×256 image in MATLAB. Once we have constructed the weight, the iteration of NL/H^1 and NL/TV is comparable to ROF in speed (NL/H^1 is more efficient than NL/TV). The computation speed depends on the number of iterations. In general, it takes around 200 seconds for 500 iterations. For some examples, the energy converges very quickly. Also, codes could be further optimized.

4.3.4 Tomographic reconstruction

We have tried the algorithms on a synthetic Shepp-Logan head phantom and a real brain image, for which projection data have been computed using a Radon transform matrix.

The size of each image is 128×128. The number of projection angles is chosen to be the size of the image and there are $\sqrt{2} * 128 \approx 185$ X-ray parallel beams for each angle in order to cover the whole image domain. Therefore the size of projection data for both images is 185×128. Then we add white noise of $\sigma_n = 100$ to the projection data of both images. Since the projection data lies in a different space to the image does, [21] can not be directly applied in this case. Therefore, the weight function for the non local methods is estimated from the image reconstructed by FBP, which is also used as the initial guess for all the variational models. As we discussed above, we compute the weight function $w(x, y)$ from an initial filtered back projection image and estimate noise level to determine the value of parameter h in the weight.

Fig. 4.12 and Fig. 4.13 show the simulation results in the presence of Gaussian white noise in the projection of a phantom and a brain image. The output of FBP is very poor. TV-regularization performs well but the details of the images are slightly blurred. Similar to deconvolution, NL/H^1 gives a visually smoother image than NL/TV does, but NL/TV archives a slightly better SNR. Furthermore, NL/H^1 gives smoother results but details are well-preserved. For both images, non local functionals outperform the traditional methods.

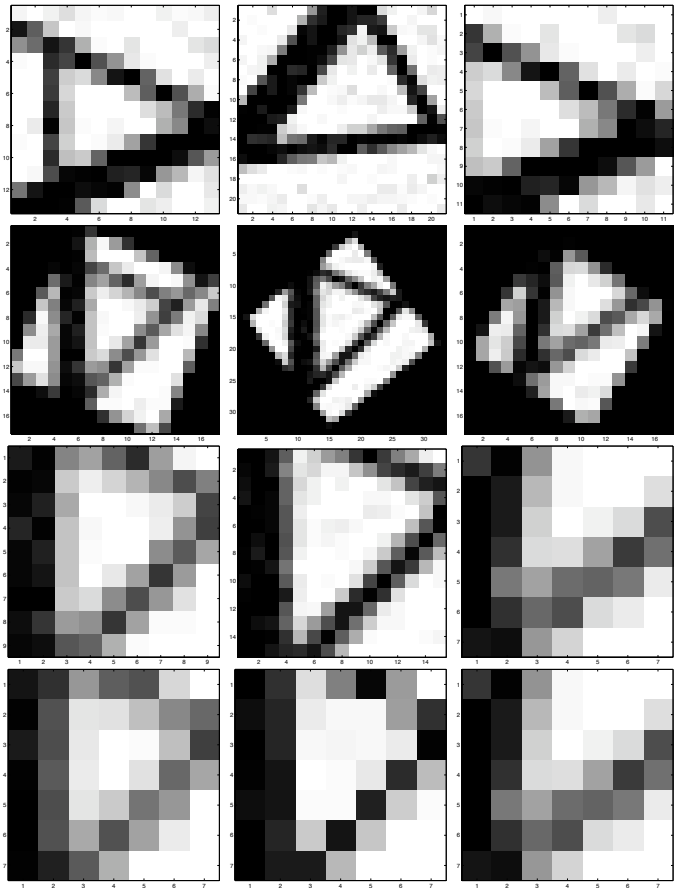


Figure 4.1: Procedure to align patches. Three patches are selected to illustrate the alignment, as shown in the first row. From top to bottom: (1) noisy patches whose size corresponds to the scale of its center; (2) rotate the patch with the angle assigned by SIFT; (3) crop the black boundary due to the rotation; (4) down-sample to a uniform size patch 7×7.

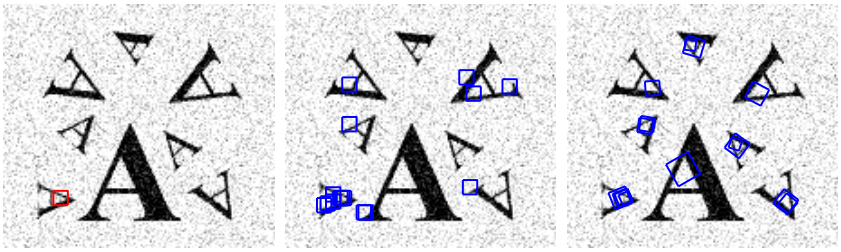


Figure 4.2: Fifteen most similar patches to the target one (red square on the left) are selected (middle) and aligned via similarity (right). On the right, the pose of the patch corresponds to the scale and orientation of its center as obtained by SIFT.

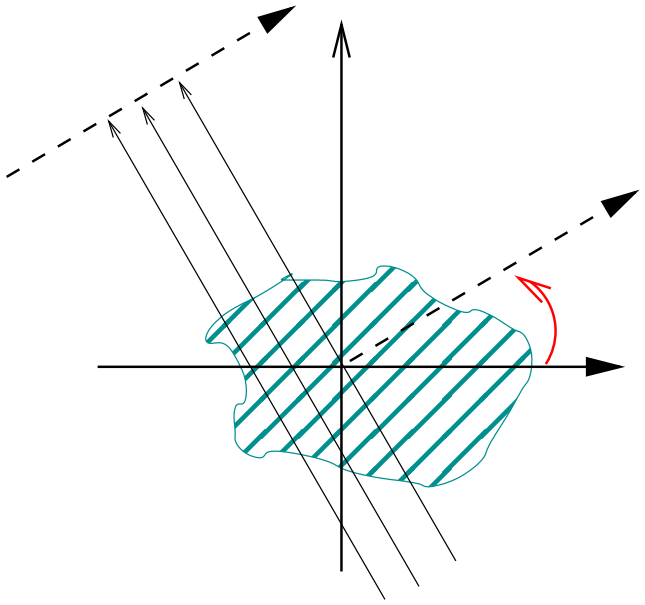


Figure 4.3: Radon transform in $\mathbb{R}^2$.

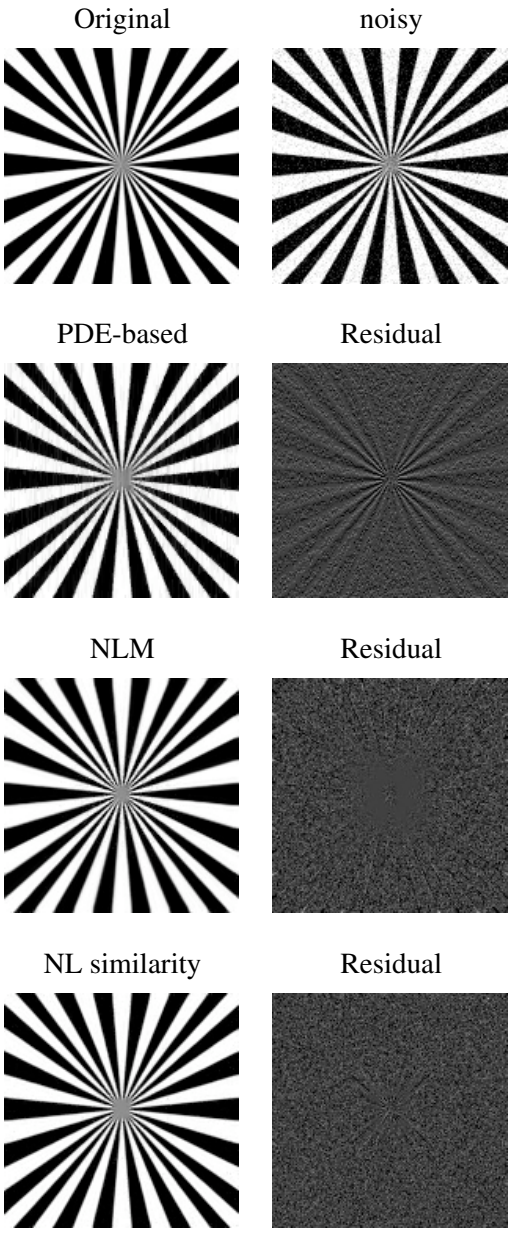


Figure 4.4: Experiment with Gaussian noise: $\sigma = 20$. The flat regions in the residual of NLM show that the central part has not been denoised.

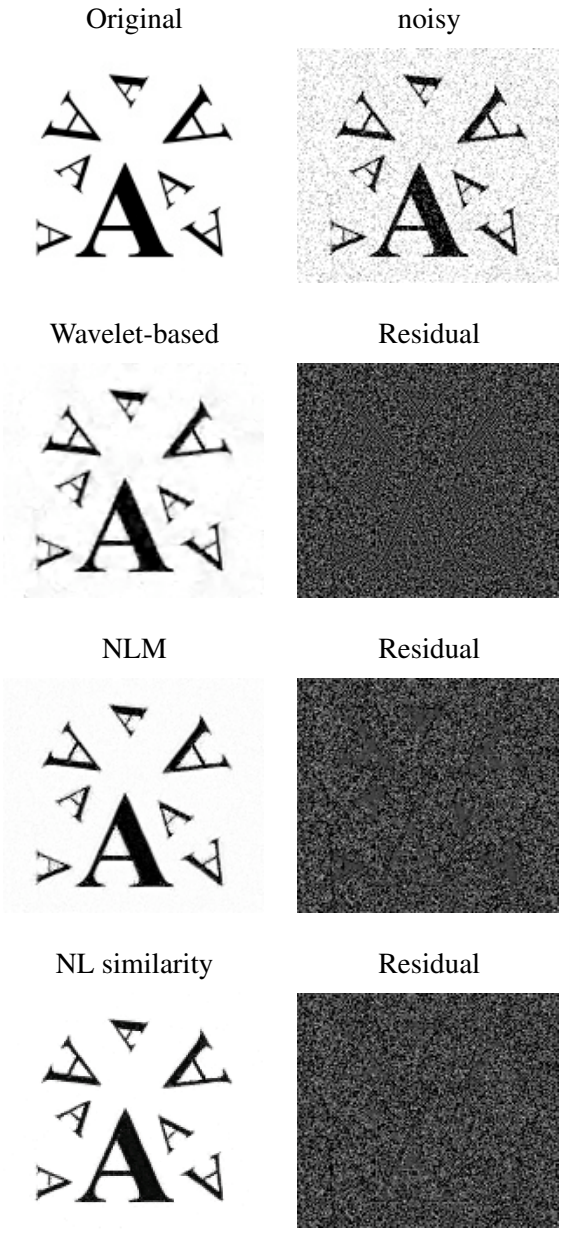


Figure 4.5: Experiment with Gaussian noise: $\sigma = 40$. The flat regions in the residual of NLM show that the serifs of the A characters have not been captured.

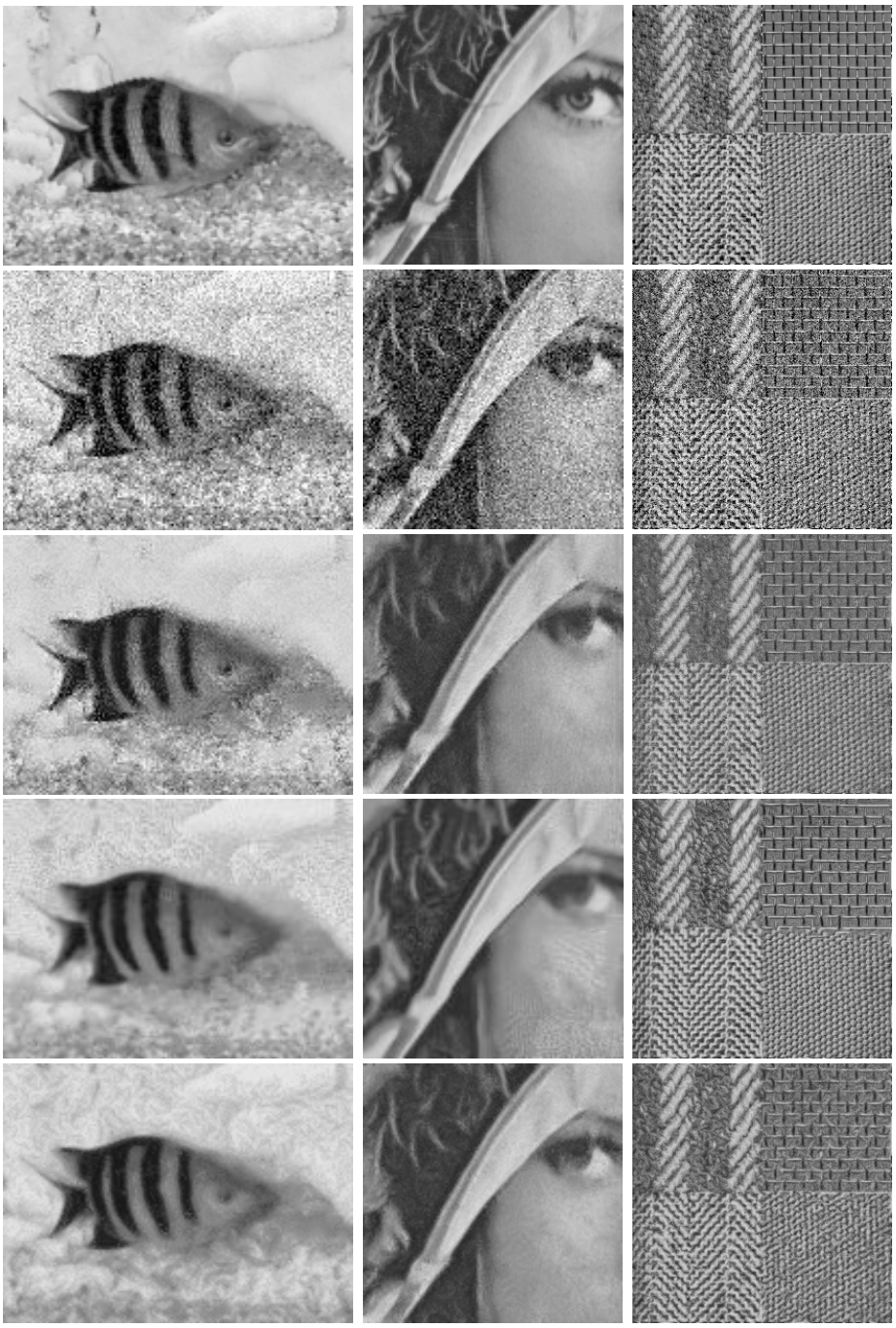
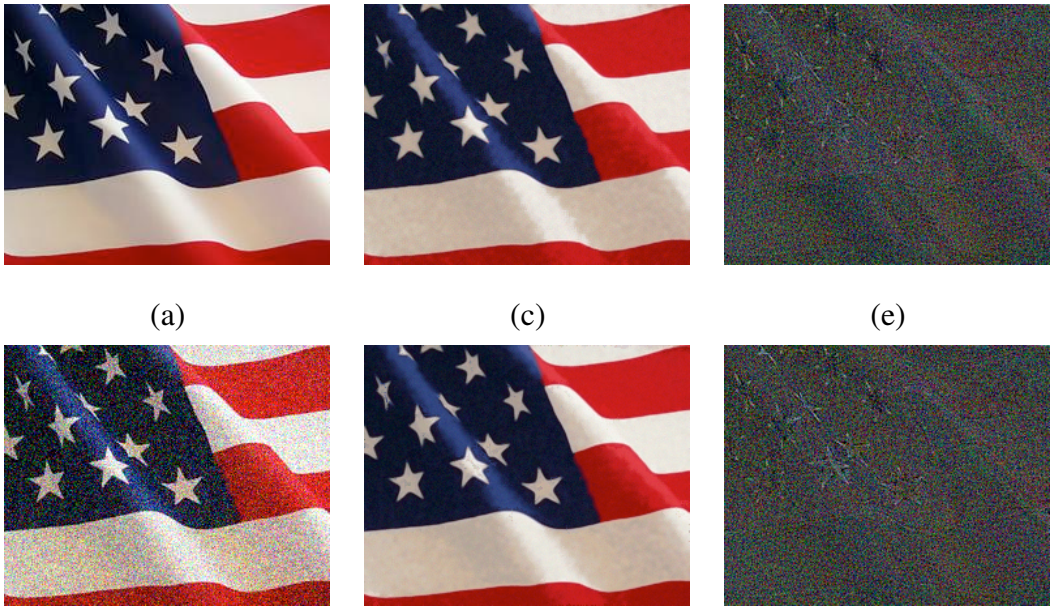


Figure 4.6: Denoising examples in T. Brox's paper [17]. From top to bottom: original image, noisy image (cropped from his paper), nonlocal means, iterated NL (his), NL similarity (ours).



(a) (c) (e)

(b) (d) (f)

Figure 4.7: Color image denoising with Gaussian noise with $\sigma = 40$. From left to right, top to bottom: (a) original image, (b) noisy input f, (c) nonlocal means u_1, (d) nonlocal Similarity u_2, (e) NL method noise $f - u_1$ and (f) NL similarity method noise $f - u_2$. The stripes tend to be restored better in (f) than in (e).

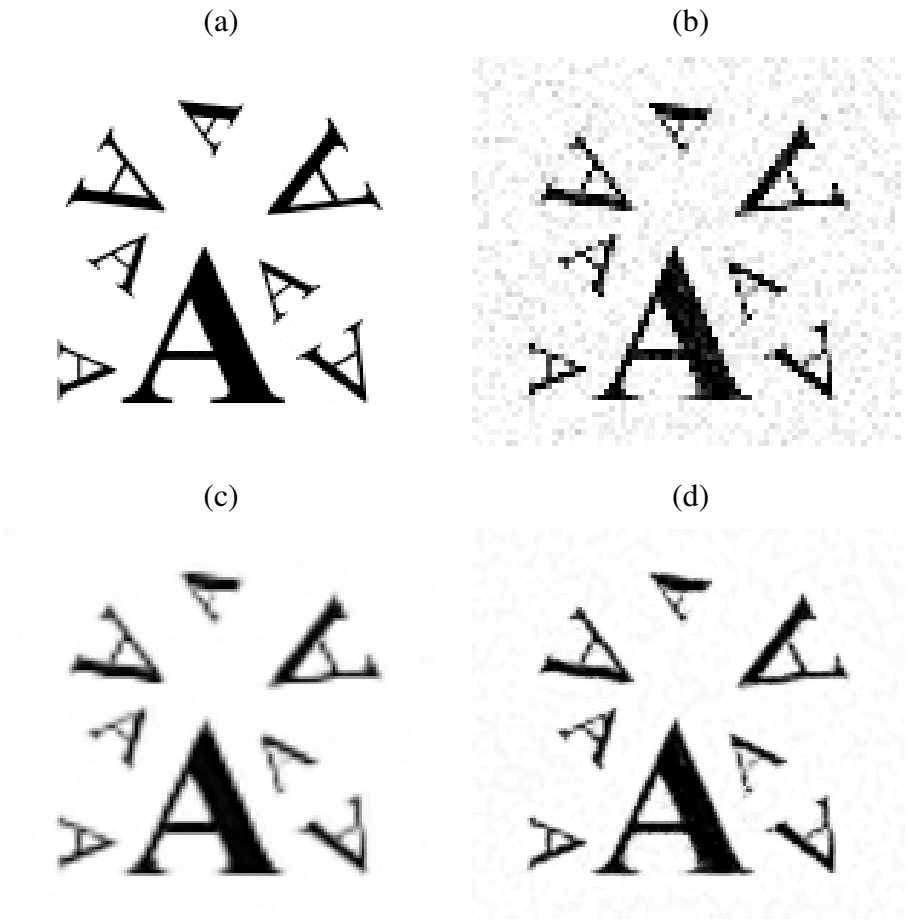


Figure 4.8: Super Resolution. From top to bottom and left to right: (a) original image, (b) low-resolution noisy image), (c) standard nonlocal means and (d) nonlocal similarity (ours). The characters are sharper in (d) than (c).

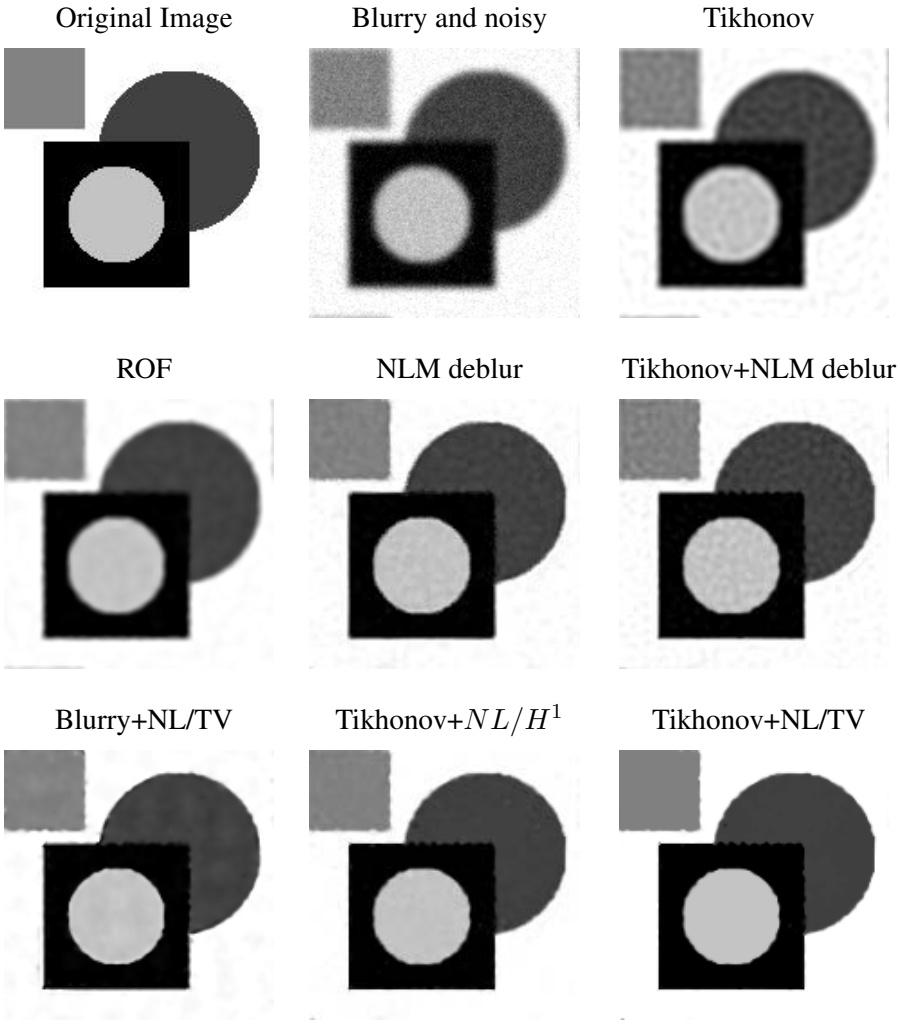


Figure 4.9: A 150×150 image with Gaussian blur $\sigma_b = 2$ and Gaussian noise $\sigma_n = 10$.

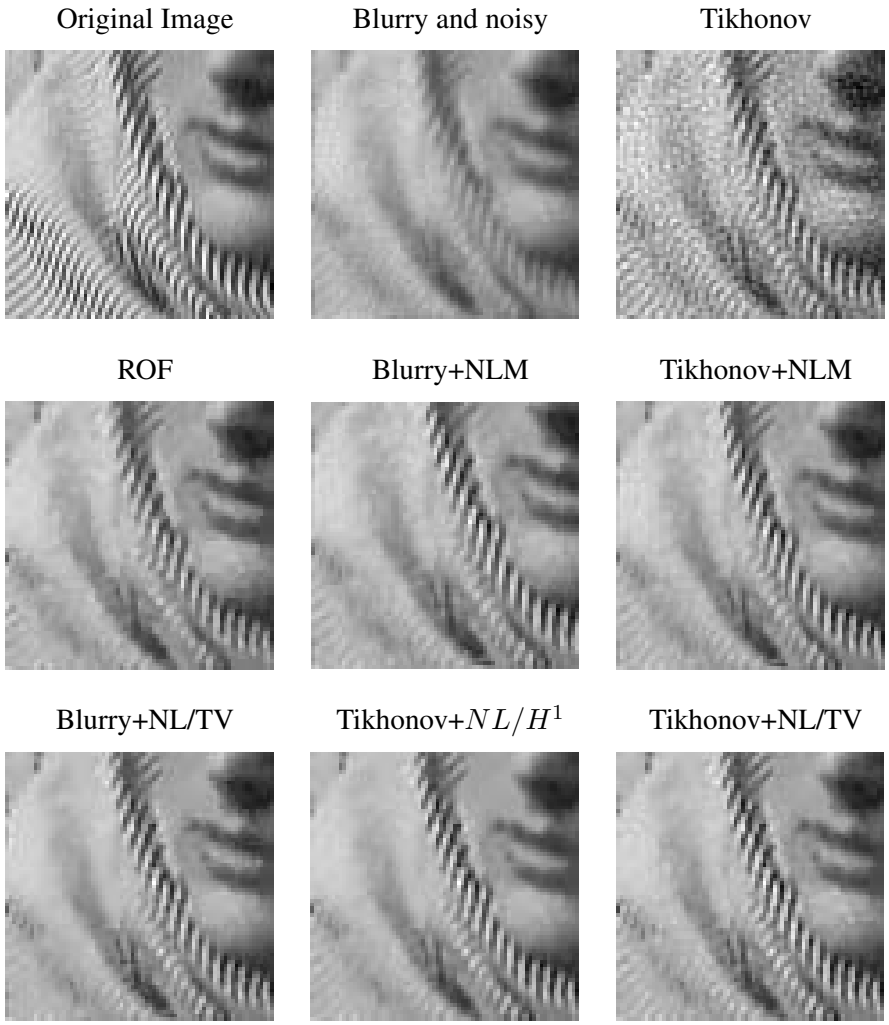


Figure 4.10: A 200×200 image with Gaussian blur $\sigma_b = 1$ and Gaussian noise $\sigma_n = 5$ cropped to 75×75 pixels.

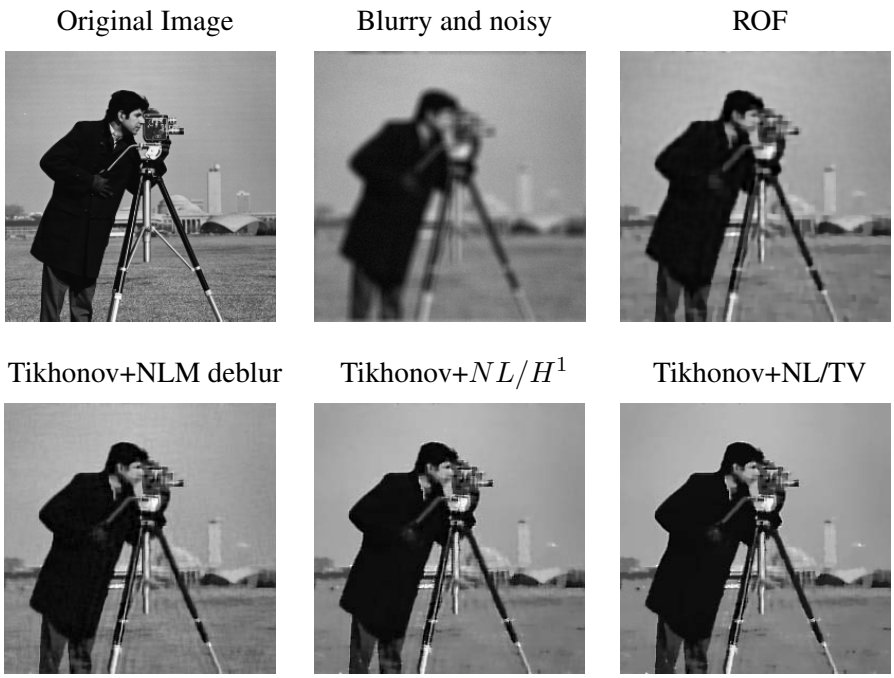


Figure 4.11: A 256×256 image with box average kernel 9×9 and Gaussian noise $\sigma_n = 3$.

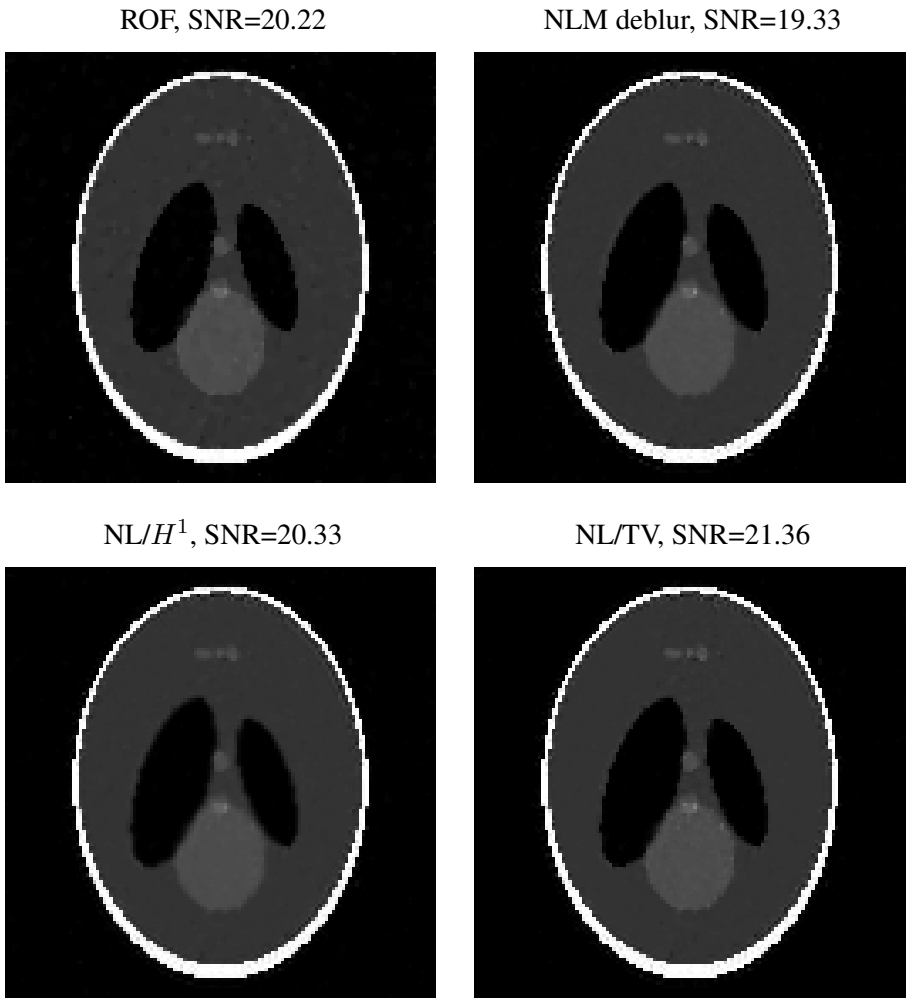


Figure 4.12: Results of reconstruction from noisy projection data with SNR=28.1db.

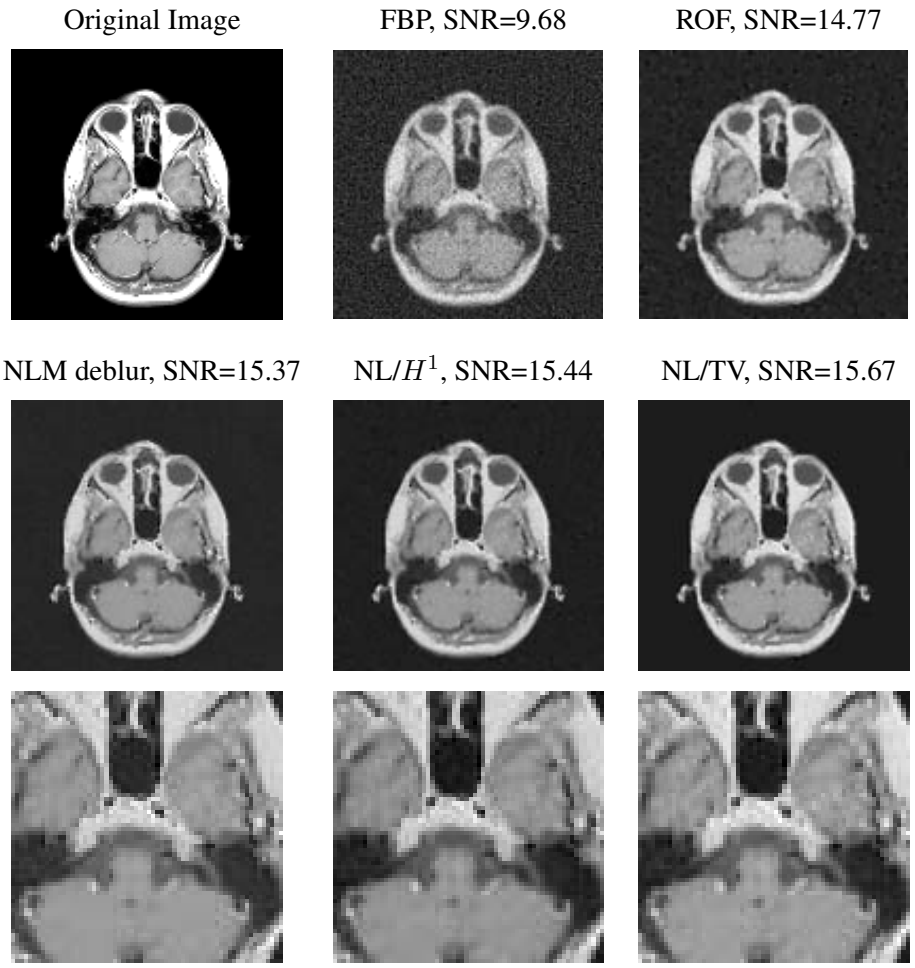


Figure 4.13: Results of reconstruction from noisy projection data with SNR=26.04db. On the last row is the enlarged central part of each reconstruction.

CHAPTER 5

A Global Approach for Multi-image Denoising

It is a frustrating experience, even for professional photographers, to take pictures under bad lighting conditions with hand-held camera. If the camera is set to a long exposure time, the photograph gets motion blur. If it is taken with short exposure, the image is noisy. This dilemma can be solved by taking a burst of images, each with short-exposure time, as shown in Fig. 5.1. But then, as classical in video processing, an accurate registration technique is required to align the images. Denote by $u(x)$ the ideal non noisy image color at a pixel x. Such an image can be obtained from a still scene by a camera in a fixed position with a long exposure time. The observed value for a short exposure time τ is a random Poisson variable with mean $\tau u(x)$ and standard deviation proportional to $\sqrt{\tau u(x)}$. Thus, the SNR increases with the exposure time proportionally to $\sqrt{\tau}$. The core idea of the *burst denoising* method is a slight extension of the same law. The only assumption is that the various values at a cross-registered pixel obtained by a burst are i.i.d.. Thus, averaging the registered images amounts to averaging several realizations of these random variables. An easy calculation shows that this increases the SNR by a factor proportional to $\sqrt{n}$, where n is the number of shots in the burst. (We call SNR of a given pixel the ratio of its temporal variance to its temporal mean). Fig. 5.1 summarizes the possibilities offered by an image burst. A long exposure image is exposed to motion blur. The short exposure image is noisy, but sharp. Finally, the image obtained by averaging the images of the burst after registration is both sharp and noiseless. In this real example the burst taken in a gallery

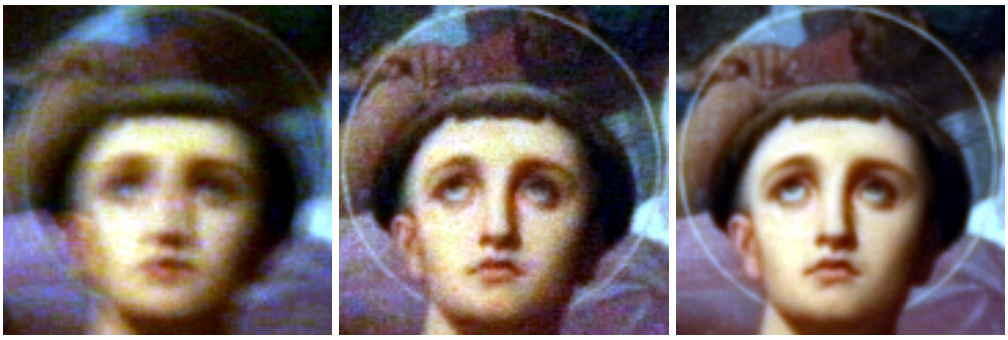


Figure 5.1: From left to right: one long-exposure image (time = 0.4 sec, ISO=100), one of 16 short-exposure images (time = 1/40 sec, ISO = 1600) and the average after registration. All images have been color-balanced to show the same contrast. The long exposure image is blurry due to camera motion. The middle short-exposure image is noisy, and the third one is some 4 times less noisy, being the result of averaging 16 short-exposure images.

had 16 images. The noise should therefore be divided 4.

The idea of combining multiple images to get a desired one is called image fusion. Most recent works on fusion use a pair of pictures taken with different camera parameters. Yuan *et. al.* [143] combine a blurred image with long-exposure time, and a noisy one with short-exposure time for the purpose of both denoising the second and deblurring the first. Beltramio and Levine [10] propose on the similar direction that two images (one underexposed and one overexposed) are combined into the one with the bright color in the overexposed image and sharp details contained in the underexposed one. Combining two snapshots, one with and the other without flash, is investigated by Eisemann *et. al.* [101] and Fattal *et. al.* [51]. Both papers report spectacular results. In contrast, we shall only consider classic image bursts, taken with the very same camera parameters. The number of images ranges from 9 to 36, thus promising a division of the noise by 3, 4 or 6. As is apparent in the above numbers, the denoising power of *burst denoising* is eventually hemmed by the low growth of the

square root. On the other hand, dividing the noise by the mentioned factors and getting an artifact free image is in no way a negligible ambition. Indeed, even the best state of the art denoising methods can create slightly annoying artifacts, such as adhesion effects and shocks in NL-means [20] or patterns in the transform thresholding methods [42], [32]. Simple accumulation instead is the essence of photography. The first Nicephore Niepce photograph [22] was obtained after a several hours exposure. The only objection to long exposure is the variation of the scene. The more this variation can be compensated, the longer the exposure can be.

There is a strong argument in favor of denoising by simple averaging of the registered samples instead of block-matching strategies. If a fine non-periodic texture is present in an image, it is virtually indistinguishable from noise, and actually contains a flat spectrum part which has the same Fourier spectrum as the white noise. Such fine textures can be distinguished from noise only if several samples of the same texture are present in other frames and can be accurately registered. Now, state of the art denoising methods are based on nonlocal block matching. In the case of a burst, the block matching would ideally find only one block in each image. But it doesn't. Precisely because of the noise, low contrasted textures are at risk of being mismatched across frames. The experimental section will show that this can cause a loss of resolution for such textures. A registration process more global than block matching, using strong features elsewhere in the image, should permit a safer denoising by accumulation.

Yet, this method rises serious technical objections. The main technical objection is: how to register globally the images of a burst? Fortunately, there are several situations where the series of snapshots are indeed related to each other by a homography, and we shall explore these situations first. The homography assumption is actually valid if one of the assumptions is satisfied:

1. the only motion of the camera is an arbitrary rotation around its optic center;

2. the photographed objects share the same plane in the 3D scene;

3. the whole scene is far away from the camera.

In those cases, image registration is equivalent to computing the underlying image homography. But this registration should be sub-pixel accurate. To this aim we will introduce a precise variant of SIFT [87] and a generalization of ORSA (Optimized Random Sampling Algorithm, [102]) to register all the images together.

Yet, in general, the images of 3D scene are **not** related by a homography, but by an epipolar geometry [68]. Even if the camera is well-calibrated, 3D point-to-point correspondence is impossible to obtain without knowing the depth of the 3D scene. Therefore, we should not expect that a simple homography will work everywhere in the image, but only on a significant part. On this part, we shall say that we have a dominant homography.

To go further, we shall need several tools whose list follows. The main one is the accurate estimation of the noise model from a partial registration.

- **High accurate keypoint detection:** By canceling the subsampling in SIFT, a subpixel precision of the key point detection will be reached. As a result, the dominant homography will be computed accurately from the matching points.

- **Noise estimation:** At each pixel that is well-registered, the registered samples are i.i.d. samples of the same underlying Poisson model. As a result, a signal dependent noise model will be accurately estimated for each colour channel. This model simply is a curve of image intensity versus the standard deviation of the noise.

- **Color equalization** However, the noise estimation will require an extra step, the histogram equalization of all images. Indeed, the images taken with indoor

lights often show fast variations of the contrast and brightness. It is only after this equalization that the empirical standard deviation of the samples becomes a measurement of the noise standard deviation.

- **Hybrid denoising scheme:** Averaging does not work at the mis-registered pixels, and block matching methods are at risk on the fine image structures. Thus they will be combined. The simple combination used here will be a convex combination of them, the weight function being based on the noise curve and on the observed standard deviation of the values for the accumulation at a certain pixel. If this standard deviation is compatible with the noise model, the denoised value will be the mean of the samples. Otherwise, the standard deviation test will imply that the registration at this point is inaccurate, and a conservative denoising will be preferred at the pixel. (More prudently, the denoised value will be a weighted average of both denoised values, the weights being steered by the test.)

References and preliminaries on the used techniques are given in Sect. 5.1. The tentative algorithm is described in Sect. 5.2 including how to register all the images, estimate the noise and combine two denoising schemes. Experiments on various kinds of real data sets are examined in Sect. 5.3.

5.1 Preliminaries, Anterior Works

5.1.1 Image Matching

To find key points in images and match them is a fundamental step for many computer vision and image processing applications. One of the most robust is the Scale Invariant Feature Transform (SIFT) [87]. There are other attempts to match key points in a more invariant fashion [99, 100, 96, 59, 105, 103]. Applications of image matching include

scene parsing[85], object/image retrieval [120] and motion estimation [86]. The image stitching [15, 16] generates a panorama from several images of the same landscape. The underlying technical problems are basically the same as for the burst denoising problem. In particular, the registration accuracy is a key issue in image stitching. In [15], bundle adjustment is used to minimize the homography projection error. This technique requires a knowledge of the camera internal parameters for initialization. Because wrong matches occur in the SIFT method used here for the registration, an accurate estimate of the dominant homography will require the elimination of outliers. The standard method to eliminate outliers is RANSAC (RANdom SAmple Consensus) [56]. However, it is efficient only when outliers are a small portion of the whole matching set. There are other variants of RANSAC to improve the performance of outlier elimination and the estimation of fundamental matrix, such as [125, 147, 130, 108]. We choose the method based on a contrario model proposed by Moisan and Stival [102]. It has zero parameter and is effective even the matching set contains up to 90% of outliers.

5.1.2 Noise Estimation

Most computer vision algorithms should adjust their parameters to the image noise level. Surprisingly, there are few papers dealing with the noise estimation problem and most of them only estimate a signal-independent noise. The standard procedure is the following: (1) compute the mean and standard deviation for each $N \times N$ block in the image (N is small, *e.g.* $N = 3$ or $N = 5$); (2) classify the standard deviations according to their mean, and (3) take the median value of all standard deviations for each mean. Instead of computing the variance of patches, Olsen [109] and posteriorly Rank *et. al.* [112] consider the patches of the image derivative, since it is more robust to the noise. As a variant, Donoho *et. al.* [42] propose to estimate the noise standard

deviation as the median of absolute values of wavelet coefficients at the finest scale. All the algorithms mentioned above usually give a reasonable estimation of the standard deviation when the noise is uniform. Yet, when applying these algorithms to estimate signal dependent noise, the results are poor. An exception is the work of C. Liu *et. al.* [84], which estimates the upper bound on the noise level from a single image. However, the real CCD camera noise is not simply additive, neither is it uniform over the gray levels. For obvious compression requirements, our experiments will treat JPEG bursts that have undergone an unknown contrast change (gamma-correction). As we shall see, the resulting estimated curve model is strongly image dependent and cannot be estimated by a parametric method.

5.1.3 Image/Video Denoising Algorithms

Image denoising methods are based on various models of the original noise-free image, which permit to separate it from noise. One of the assumptions is the sparsity in an basis, orthogonal or over-complete. Sparsity is widely used in the many applications of image processing, such as denoising [46], color denoising and inpainting [90] and super-resolution [138, 36]. Non-Local means [20] assumes an image self-similarity and restores an unknown pixel using other similar pixels. The similarity is considered in terms of a patch centered at each pixel, not just the intensity of the pixel itself. In order to denoise a pixel, it is better to average the nearby pixels with similar structures (patches). This idea is extended to movie denoising [19, 127, 126]. The denoising algorithm by Dabov *et. al.* [33] combines self-similarity block matching, and threshold in the transform domain. The sparse representation is enhanced in transform domain by grouping similar 2D image patches into a 3D block. The weighted averaging of all the block-wise estimates are aggregated for the final output. Extensions to other applications are discussed by the same group of the authors, such as color denoising

[32], grayscale video denoising [30], image sharpening [34] and restoration [31]. So far BM3D represents the state of the art for stand alone denoising. G. Boracchi and A. Foi [12] extend BM3D or V-BM3D to signal-dependent noise. They assume a parametric noise model, in which the parameters can be estimated using [58]. Then BM3D is applied on the images after a variance-stabilizing transformation to make noise homogeneous and post-processing follows.

The present chapter can be understood as an extension and explanation of the multiple image denoising attempt by Zhang *et. al.* [145]. These authors propose a global registration of an image burst before applying a block matching multiimage strategy to the registered images. They remark that their denoising performance stalls when the number of frames grows and write that this difficulty should be overcome. Yet, their observed denoising performance 3 curves grow like the square root of the number of frames, which indicates that their algorithm relies on accumulation. Thus, this performance is in fact optimal. The only non-synthetic experiments are made by these authors on a flat static real scene, actually a white board. The method proposed here is definitely an extension: It uses a hybrid scheme which chooses the best of accumulation or block denoising, depending on the reliability of the match. Without the accurate nonparametric noise estimation, this strategy would be unreliable.

5.2 The Main Tools of the Burst Denoising Chain

In this section, we discuss how to register all the images into one in the image sequence, which is taken as template. The average of the registered images gives a desired denoising result, but this only works at well-registered pixels. As for the pixels that are not well-registered, classic state of the art denoising (NLM, BM3D) will be tested. The decision maker, *i.e.* whether to use averaging or a denoising algorithm, will be based on the noise model, which will be estimated from the samples of each

well-registered pixel along time. Thus having an accurate noise model obtained from the burst itself in crucial in the strategy. In summary, burst denoising is a relatively complex chain that:

- registers the images of the burst by subpixel accurate SIFT and estimation of the best dominant homography;
- equalizes the histograms of the registered images to remove lighting effects;
- estimates accurately from the many samples offered after registration the noise for each channel and at each level;
- thanks to this estimation, proceeds to denoising by averaging at all pixels where the correct registration is confirmed, and applies a state of the art denoising elsewhere.

Short preliminary discussion: is that safe? In spite of its complexity the chain is safe. Indeed, the dominant registration yields many samples permitting robust estimation of the noise. The averaging is applied only at pixels where the observed standard deviation after registration is close to the one predicted by the noise model. Thus, there is no risk whatsoever with averaging. At the other pixels, standard state of the art video denoising is applied. Block matching is only made safer by the previous registration and equalization, even if inaccurate. The experimental section will confirm the safety of the method by showing that the final result always is better than classic video denoising alone.

5.2.1 Registration of an Image Sequence

We shall use SIFT as the tool for the key point detection. A sub-pixel precision for denoising is required but, unfortunately, the precision of the SIFT points decreases

through the octaves. Indeed, SIFT simulates the scale space by sub-sampling the images by factor two through each octave. Thus the sub-pixel key point detection, which is sub-pixel accurate in the first octave, can be several pixels inaccurate in the last octaves. To maintain a constant precision through the octaves, the SIFT sub-sampling between the octaves was simply canceled.

This cancelation of the sub-sampling entails two adjustments of SIFT. The first one is to adjust the Laplacian threshold, an important parameter in the SIFT method removing key points due to noise. Canceling the sub-sampling between octaves is equivalent to up-sampling the images by a power of two. Thus the Laplacian of the pixel on the twice up-sampled image is four times smaller than the corresponding one on the original image, because

$$\triangle \left(u(\frac{x}{2}, \frac{y}{2}) \right) = \frac{1}{4} \triangle u(x, y) \tag{5.1}$$

where $u(x, y)$ is the image and $\triangle$ is the Laplace operator.

The second adjustment after the cancelation of the SIFT sub-sampling is the construction of the descriptors. In our case, the blur is increasing through octaves, and so is the size of the domaing associated with each descriptor. To keep the scale invariance, the domain of each descriptor in the n-th octave is therefore sub-sampled by a 2^{n-1} ratio.

In summary, the precision of SIFT key points is improved by canceling the sub-sampling through octaves. The SIFT descriptor construction and the Laplacian threshold are adapted to keep them as in the original SIFT. As will be proved in simulations, the accurate SIFT retains a rather constant precision through octaves.

5.2.2 Reliable Dominant Homography Estimation

An adaptation to multi-images of the Moisan-Stival ORSA algorithm [102] will be used. We adapt their notations here. Assume the set of match pairs is

$$S = \Big(\mathbf{x}_i = (x_i, y_i), \mathbf{x}'_i = (x'_i, y'_i)\Big)_{i=1\ldots n},$$

We are interested in the homography matrix $\mathbf{H}$, that is best compatible with these matches (and not in the fundamental matrix itself [69, 68]). Also, we want to keep a safe subset of inliers T in S, with size k ($4 < k \leq n$). Following [102] define the rigidity of T associated with $\mathbf{H}$ by

$$\alpha_{\mathbf{H}}(T) = \frac{\pi}{A'}\Big(\max_{(\mathbf{x},\mathbf{x}')\in T} \text{dist}(\mathbf{x}', \mathbf{H}\mathbf{x})\Big)^2, \tag{5.2}$$

where A' the area of the second image domain. The rigidity is in fact a geometric probability. It is obtained by dividing the area of a disk with radius the maximal $\mathbf{H}$-projection error for T, by the image area A'. Following the *a contrario* method, if the rigidity is too small to be explained by randomness, the deduction is that there are only "inliers" in T. It is difficult to compute the probability $P(\inf_{\mathbf{H}} \alpha_{\mathbf{H}}(T) < t)$ to select the best subset T, even if we assume all the points are uniformly distributed in images. Instead of computing this probability directly, Moisan and Stival [102] use a Bonferroni-like estimate, namely the expected number of false alarms (**NFA**), also referred to as the meaningfulness:

$$\epsilon(\alpha, n, k) := (n-4) \cdot {}^n_k\mathbf{C} \cdot {}^k_4\mathbf{C} \cdot \alpha^{(k-4)}. \tag{5.3}$$

This number incorporates the size of the matching set, the size of the subset and the rigidity. This algorithm has zero parameter and does not require any assumptions on the camera motion or the estimation of noise variance.

In burst denoising, the ideal way would be to partition the image domain into different regions, each of which shares the same homography, compute homography on

each of them and finally register each image to the reference one by inverting the homography for each region. But, if we apply directly ORSA on each pair of two images, it is not guaranteed that the same region with a dominant homography will be chosen for each pair. A natural solution is to find a region and a homography common to all pairs of images. Therefore, ORSA is adapted by defining a "joint meaningfulness" as indicated in Algorithm 1.

Algorithm 1 multiple ORSA

Input The set S_0 of the common SIFT points in the template and the corresponding matching points in the $j-$th image, denoted as S_j.

Set $\epsilon = +\infty$

while the number of trials does not exceed N **do**

Pick up 4 random points from S_0

for (each $j > 0$) **do**

Compute the homography using these 4 points and the corresponding ones in S_j

Find the most meaningful subset of S with respect to S_j under this homography, save the meaningfulness parameter as ϵ_j

end for

Compute the joint meaningfulness $\epsilon_{joint} = \sum \epsilon_j$

If $\epsilon_{joint} < \epsilon$, then $\epsilon = \epsilon_{joint}$, and save the meaningful subset for each pair of images as T_j and the 4 points, $P4$.

end while

Return ϵ_{joint}, T_j and $P4$.

It is impossible to try all 4-points combinations. Instead, an optimized random sampling algorithm (ORSA)is used as suggested in [102]. The algorithm stops once $\epsilon_{joint} < 1.0$. Then it is iterated for a small number of trials, typically $N/10$.

5.2.3 Video Equalization

There still is an extra step before noise estimation: the burst equalization! The images taken under indoor lights usually consist of fast variations of the contrast and brightness. We want to make them consistent through all the images, so that the standard deviation along time is indeed due to the noise, not to the changes of lights. This is done by a joint histogram equalization of all images. The best exponent of joint equalization is the Midway method [37, 38] which is summarized in a simple and elegant formula. Let $v : \Omega \to [0, 1]$ be an image and h its intensity histogram. The cumulative histogram of v is

$$H(x) = \int_0^x h(s)ds.$$

Starting with a series of images $v_j,\ j = 1, \cdots, N$ with cumulative histogram $H_j(x)$, the Midway cumulative histogram H is defined as a compromise of all H_j by $H = \left(\frac{1}{N}(\sum_j H_j^{-1})\right)^{-1}$. Once H is computed, each image v_j is replaced by $\phi_j(v_j) = H^{-1}(H_j(v_j))$. The necessity of histogram equalization to get a reliable noise model is illustrated in Fig. 5.4.

5.2.4 Signal-Dependent Noise Estimation

Here is the crucial step of the chain. The sequence of registered images is used to estimate the signal-dependent noise curve. If one pixel is well-registered, its values along the time give samples permitting to estimate the noise model. Therefore the standard deviations are classified according to their mean. The main question is to have an estimate robust to the wrongly registered pixels. The histogram of the mean of each pixel along time is constructed, with $n = 100$ uniform bins. Inside each bin, the median value of the standard deviations of all pixels is computed. This yields a curve of mean versus standard deviation. The median is robust to outliers by itself,

but several precautions can be taken to make the estimate still more reliable. First, all edge points on which the interpolation error is stronger are simply ruled out. This can be done by Canny edge detector. Second, the pixels whose standard deviation is too large are also ruled out. The threshold is set to be the double of the peak value in the histogram of all standard deviations. Finally, bins that contain less than 100 items are simply not retained. The noise curves for the three color channels are estimated separately, but show a striking coincidence up to a multiplicative factor.

5.2.5 Hybrid Denoising Scheme

The noise estimate is crucial to meet a safe decision about which kind of denoising can be applied at each pixel. Suppose we have two denoising results: the one from averaging u_{A} and the other from NLM or BM3D, u_{BM3D}, the hybrid scheme will return

$$u_{\mathrm{H}} = \alpha \cdot u_{\mathrm{BM3D}} + (1 - \alpha) \cdot u_{\mathrm{A}}$$

For each registered pixel x in the template image, the average $u(x)$ of its samples after registration is looked up in the noise model. The noise curve gives the expected standard deviation $\sigma(x) := \sigma(u(x))$. At the same registered pixel the empirical standard deviation $\hat{\sigma}(x)$ of the samples is also computed. If this pixel is correctly registered, $\hat{\sigma}$ should close to σ, in which case a small value should be given to the weight α. A simple choice for α uses the sigmoid function:

$$\alpha(x) := \frac{1}{1 + \exp(c - \hat{\sigma}(x)/\sigma(x))}.$$

To avoid any impulse noise created by a local conflict between estimates, the weight function $\alpha(x)$ is slightly smoothed out by a 3×3 spatial average. Algorithm. 2 summarizes the steps of the proposed multi-image denoising.

Algorithm 2 multi-image denoising

Input: ImageSequence $V = \{V_0, \cdots, V_N\}$

compute SIFT points on V_0, saved as CommonSIFTpts

for $(j > 0)$ **do**

compute SIFT points on V_j

save the matching points in V_j and V_0 as currSIFTpts

update CommonSIFTpts = CommonSIFTpts $\cap$ currSIFTpts

end for

Apply multiple ORSA (Algo. 1) on the set of CommonSIFTpts to get the most meaningful 4 points P_4

for $(j > 0)$ **do**

compute homography between V_j and V_0 using P_4

$V\text{reg}_j$ = register V_j back to V_0 by this homography

end for

Video equalization

Noise estimation

Hybrid denoising scheme combining the average and BM3D applied on $V\text{reg}$

5.3 Experiments

5.3.1 Accurate SIFT

A check was made on the accuracy gain of the accurate SIFT described in Sect. 5.2.1. We applied SIFT and accurate SIFT on two images respectively. One of the images was generated from the other image by a simple rotation+translation, as shown in Fig. 5.2. The key points in both images were matched by using Lowe's classic matching method. After eliminating the outliers by ORSA the homograghy from one image to the other image was estimated. This homography allows us to project the key points

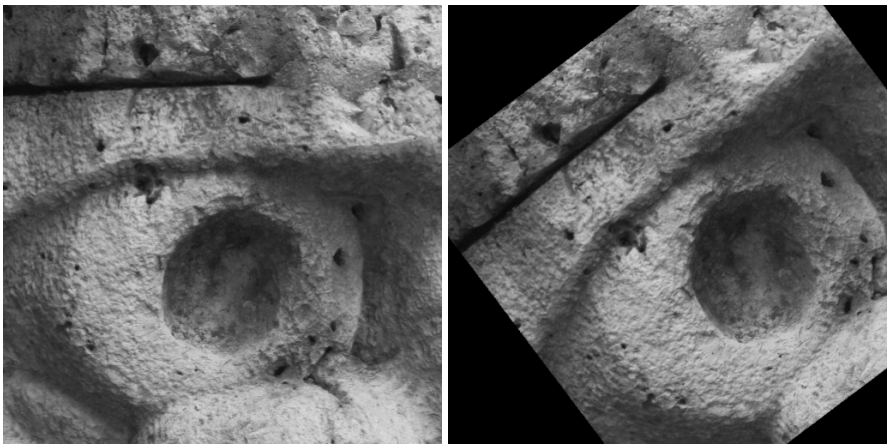


Figure 5.2: Two images used to test the accurate SIFT. The right image is generated from the left one by a translation+rotation.

of one image on the other image, and to estimate the average error. Table. 5.1 shows the average error estimated in each octave in the scale space underlying SIFT. The experiment confirms that the precision for the classical SIFT decreases when the octave index increases. For accurate SIFT, the precision remains stable through octaves.

5.3.2 Multi-image Registration

Video data provided by the company DxO Labs capture a series of images of a rotating pattern with a fixed pedestal. We show three images from the sequence and the ones after registration in Fig. 5.3. In this easy case the dominant homography is on the plane of the rotationing pattern, which contains more SIFT points than the pedestal region. As a result we observe the rotating pedestal and its background after registration.

5.3.3 Video Equalization

Fig. 5.4 shows the efficiency of video histogram equalization. The images were taken under ceiling lights with changing illumination.

average error		
	classical SIFT	improved SIFT
octave -1	0.036	0.036
octave 0	0.064	0.032
octave 1	0.263	0.033
octave 2	no keypoints	0.040

Table 5.1: The average error in each octave for Lowe's classical SIFT and for accurate SIFT. The precision decreases for Lowe's classical SIFT, while accurate SIFT remains stable through octaves. This is essentially obtained by removing the subsampling step in the SIFT method.

5.3.4 Noise Estimation

In the real scenario, the noise is inherent to the image, each pixel being modelled as a Poisson process. This model is valid except in the very dark regions where thermal and electronic noise dominate, and in the bright regions because the sensor gain is anyway nonlinear. The original image was simulated as a Poisson noise whose mean was a good quality image, after geometric homographies simulating the camera shaking. The noise estimation algorithm is demonstrated on three examples: Barbara, Couple and Hill. As shown in Fig. 5.5, the standard deviation (Y-axis) of the noise curves follows nicely the square root of the intensity (X-axis). The noise curves of the real datasets are given in Fig. 5.6.

5.3.5 Multi-image Denoising

For the experiments on synthetic data, the quantitative measurement of the denoising performance will be measured by the root-mean-square (RMS) errors of different de-

Figure 5.3: Multi-image registration. Top: three frames from an image sequence with a rotating pattern and a fixed pedestal. Bottom: the corresponding ones after registration. The dominant homography we find is on the plane of the rotating pattern, since it contains more SIFT points than the pedestal region. As a result we observe the rotating pedestal and its background after registration. The images are a courtesy of DxO Labs, Boulogne.

noising methods in Tab. 5.2. The accumulation is based on 16 images, thus yielding a theoretical noise reduction by 4. A 3.5 noise reduction is experimentally attained in the images. In all cases, the difference between the theoretical factor 4 and the observed one is probably due to the fact that the simulated images are seriously aliased, which caused interpolation errors after registration.

The denoising results are now given for several real data sets, each of which consists of 16 JPEG images bursts. For a better illustration, the comparison shows the intermediate steps: the simple average, Non-Local Means on the registered images,

Figure 5.4: Video Equalization. Top: three frames from an image sequence with different illuminations. Bottom: after registration and equalization.

BM3D on the registered images, and the result of the hybrid scheme. These results are shown on several well-chosen zoomed-in regions.

Since the proposed algorithm only finds a dominant homography, which is the rotating pattern in Fig. 5.3, the simple average fails to denoise the region of the fixed pedestals. It also fails to remove some dust that was incidentally stick to the camera objective, as zoomed-in and shown in Fig. 5.7. On the other hand, fine texture details are dramatically lost by Non-Local Means, which instead gives good denoising on contrasted regions such as the pedestals. The hybrid scheme with NLM, combining both averaging and NLM captures the virtue of each method. As expected the result is still better with a hybrid scheme using BM3D: Indeed BM3D is the best denoising algorithm and is actually quite close in performance to the direct averaging, as has been shown in Table 5.2.

The images in Fig. 5.4 captured a 3D scene with a single-lens reflex (SLR) camera, Canon EOS 30D. The scene consists of 2 books, a newspaper and a moving mouse. We enlarge three illustrative parts in Fig. 5.8, in which the structure lines on the book

Figure 5.5: Noise curve. From top to bottom: the original image, one of the simulated images by moving the image and adding Poisson noise, and the noise curve from our algorithm using 16 images. The standard deviation of the noise (Y-axis) fits to the square root of the intensity (X-axis).

and letters in the newspaper are smoothed out by non-local means. In contrast, the letters turn out to be readable when averaging. As for the moving mouse, the average fails completely, while block-matching succeeds, since it uses the similarity patches in the template image itself.

Finally we show a burst of images of a painting. This is a good direct application for our algorithm, since the images of the painting are in principle related by homog-

Table 5.2: RMS for different methods

	Barbara	Couple	Hill
noisy	11.30	11.22	10.27
NLM	10.83	5.43	6.73
BM3D	4.33	3.39	3.90
AR	**3.55**	**3.03**	**2.73**
GT	**2.85**	**2.89**	**2.63**

Table 5.3: RMSE on synthetic data with 16 images. AR and GT stand for "average after registration" and "ground-truth" in the sense of registration back by the ground-truth motion. In principle GT divides the RMSE by 4, while AR is very close but higher than GT due to misregistration and interpolation errors. In all cases video BM3D gets close to the ratio 4 limit, but still is overcome by AR for all the images.

raphy if the painting is flat and the camera distortion-free. As a result, the average is always favored by the hybrid scheme. The details are compared in Fig. 5.9, where the dynamics of the patches are equalized for a fair comparison.

Figure 5.6: Noise curves of the real data sets. Left: one of the images in the sequence; right: the noise curves of the three color channels.

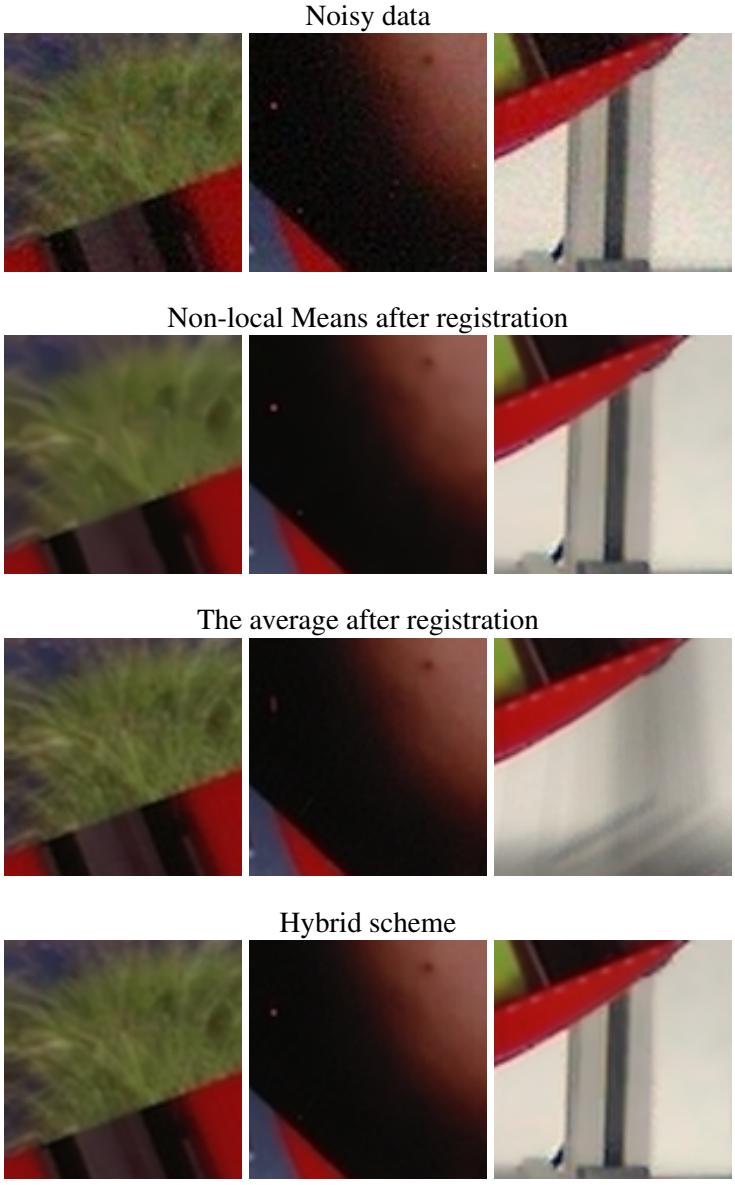


Figure 5.7: Real dataset from DxO Labs, Boulogne. Due to mis-registration, the simple average fails to denoise the region of the pedestals. In the middle example, it does not remove some dust stick to the camera objective. The hybrid scheme works everywhere and gives roughly the same result with NLM and BM3D.

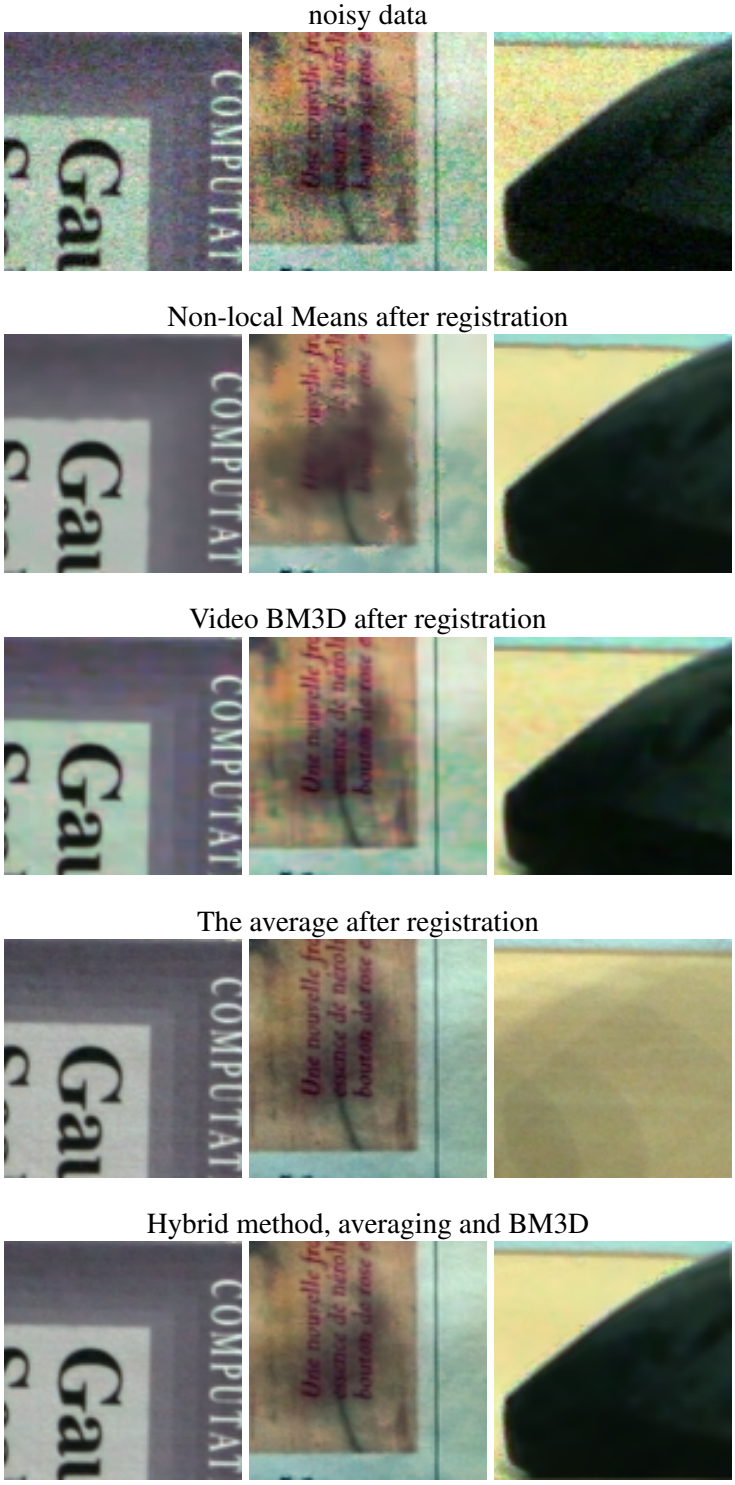


Figure 5.8: Real dataset of an indoor setting. The average fails completely with the moving mouse on the right example, while block matching succeeds since it uses the similarity patches in the template image itself.

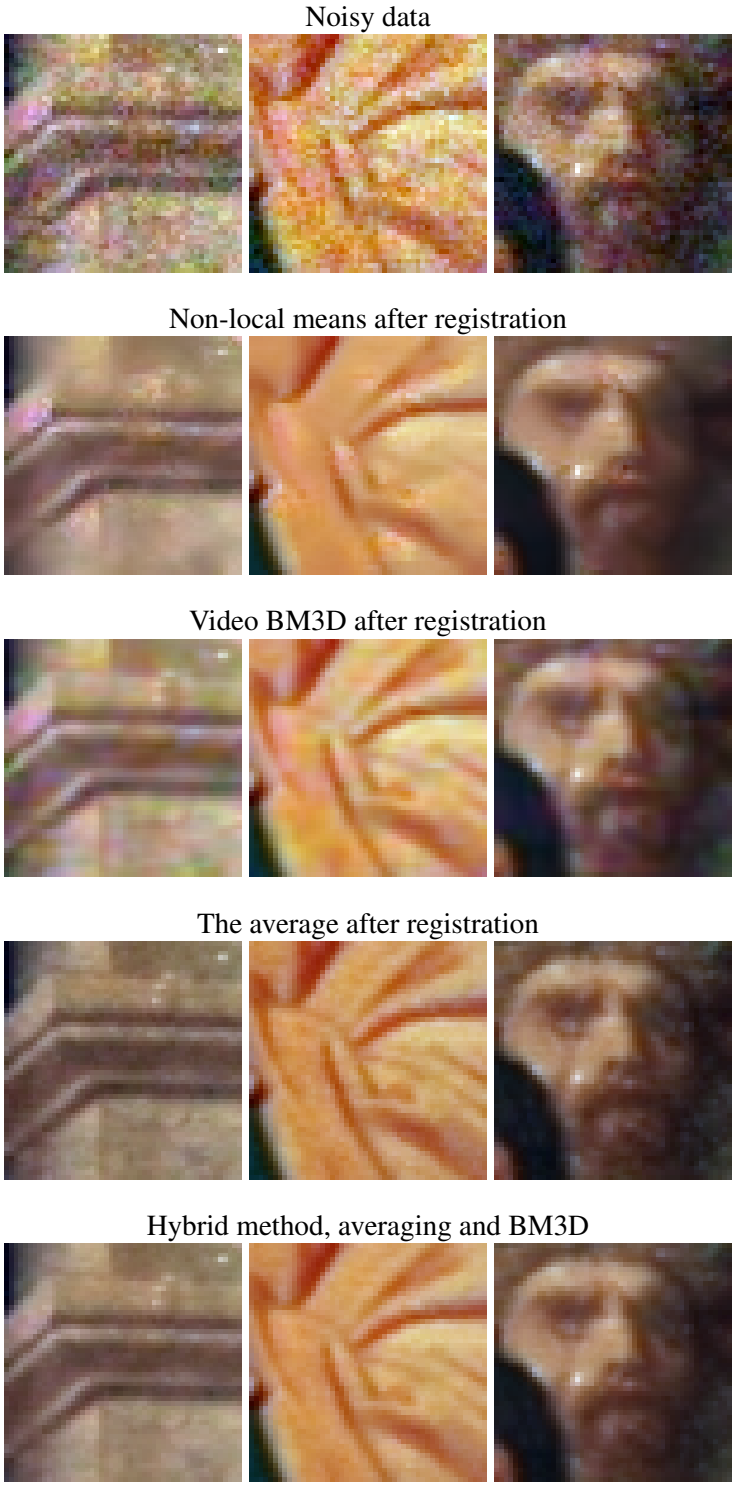


Figure 5.9: A burst of images of a painting. The last two results are almost identical, which indicates that the registration has been detected correct almost everywhere.

CHAPTER 6

Conclusion

We have presented a simple deblurring/deconvolution algorithm that exploits the assumption of sparse representation for natural image statistics. It is a "direct" algorithm, in the sense that it performs inference by synthesis relative to an explicit generative model (which acts as a regularizer), without the need to solve an ill-conditioned inverse problem.

Nonlocal functionals have recently been introduced for the case of image denoising. We have extended their utility to more general inverse problems. We discuss two applications of the general model: image deconvolution and tomographic reconstruction. We found the nonlocal functionals to give superior results, provided that the data is preprocessed using older simple techniques to construct the nonlocal weight. Furthermore, we extended the nonlocal means filtering by a more general similarity measurement. In particular, we applied SIFT to estimate a rotation and a scale at each patch so that it can be transformed to a canonical form. Then we construct the weight based on the canonical form so that we could exploit more similar patches to help denoising. We also investigated a super-resolution model as an application of the nonlocal framework. Not only does the reconstruction enhance the resolution, but it denoises the image as well.

Image deblurring is discussed using both local sparsity and nonlocal similarity. The comparison between these two methods is described as follows. First, natural images usually contain repetitive patterns, and thus similarity may result in a more reliable

regularizer than sparsity. Second, it takes time to compute the weight function, while enforcing the local sparsity constraints is highly parallelizable. Finally similarity relies on the data fidelity term, while sparsity itself is a very strong assumption. Therefore, blind deconvolution can be achieved as long as the dictionary is good enough.

Although the main purpose of Chapter 5 is denoising, it involves a series of useful techniques. We modified SIFT for keypoint detection with sub-pixel precision and adapted a multi-image ORSA for reliable homography estimation. On the technical side, the method can already be used to estimate a non parametric camera noise model from any image burst. Possible future work is to partition the image domain into different regions, each of which shares the same homography. It will involve image segmentation and depth map estimation of the 3D scene.

REFERENCES

[1] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Process.*, 54(11):4311–4322, 2006.

[2] H. C. Andrews and B. R. Hunt. *Digital Image restoration*. Prentice-Hall, Englewood Cliffs, NJ, 1977.

[3] C. Avinash and S. Malcolm. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.

[4] S. P. Awate and R. T. Whitaker. Unsupervised, information-theoretic, adaptive image filtering for image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(3):364–376, 2006.

[5] L. Baboulaz and P. L. Dragotti. Exact feature extraction using finite rate of innovation principles with an application to image super-resolution. *IEEE Trans. Image Process.*, 18(2):281–298, 2009.

[6] S. Baker and T. Kanade. Limits on super-resolution and how to break them. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(9):1167–1183, 2002.

[7] B. Bascle, B. Bascle, R. Deriche, R. Deriche, and P. Robotvis. Region tracking through image sequences. In *International Conference on Computer Vision (ICCV)*, pages 302–307, 1995.

[8] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-resolution from an image sequence. In *European Conference on Computer Vision (ECCV)*, pages 573–582, 1996.

[9] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *European Conference on Computer Vision (ECCV)*, pages 237–252, 1992.

[10] M. Bertalmio and S. Levine. Fusion of bracketing pictures. In *Conference for Visual Media Production*, pages 25–34, 2009.

[11] C. M. Bishop, A. Blake, and B. Marthi. Super-resolution enhancement of video. In *Artificial Intelligence and Statistics*, 2003.

[12] G. Boracchi and A. Foi. Multiframe raw-data denoising based on block-matching and 3-d filtering for low-light imaging and stabilization. In *International Workshop on Local and Non-Local Approximation in Image Processing (LNLA)*, 2008.

[13] S. Borman and R. Stevenson. Super-resolution from image sequences-a review. In *Midwest Symposium on Circuits and Systems*, pages 374–378, Aug 1998.

[14] S. Bougleux, G. Peyré, and L. Cohen. Non-local regularization of inverse problems. In *European Conference on Computer Vision (ECCV)*, 2008.

[15] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision*, pages 59–73, 2007.

[16] M. Brown, R. Szeliski, and S. Winder. Multi-image matching using multi-scale oriented patches. In *IEEE Computer Vision and Patter Recognition (CVPR)*, pages 510–517, 2005.

[17] T. Brox and D. Cremers. Iterated nonlocal means for texture restoration. In *International Conference on Scale Space and Variational Methods in Computer Vision*, volume 4485 of *LNCS*, pages 13–24, Ischia, Italy, May 2007. Springer.

[18] T. Brox, O. Kleinschmidt, and D. Cremers. Efficient nonlocal means for denoising of textural patterns. *IEEE Trans. Image Process.*, 17(7):1083–1092, July 2008.

[19] A. Buades, B. Coll, and J. Morel. Nonlocal image and movie denoising. *International Journal of Computer Vision*, 76(2):123–139, 2008.

[20] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling and Simulation*, 4(2):490–530, 2005.

[21] A. Buades, B. Coll, and J. M. Morel. Image enhancement by non-local reverse heat equation. Technical Report 22, CMLA, 2006.

[22] G. R. C. Chevalier and J. Niepce. Guide du photographe., 1854.

[23] J. Cai, H. Ji, C. Liu, and Z. Shen. Blind motion deblurring from a single image using sparse approximation. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, USA, 2009.

[24] D. Capel and A. Zisserman. Automated mosaicing with super-resolution zoom. In *Computer Vision and Pattern Recognition (CVPR)*, pages 885–891, 1998.

[25] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging Vision*, 20:89–97, 2004.

[26] T. F. Chan and J. Shen. Mathematical models for local non-texture inpainting. In *SIAM J. Appl. Math*, volume 62, pages 1019–1043, 2001.

[27] T. F. Chan and C. K. Wong. Total variation blind deconvolution. *IEEE Trans. Image Process.*, 7(3):370–375, March 1998.

[28] P. Chatterjee and P. Milanfar. A generalization of non-local means via kernel regression. In *SPIE Conf. on Computational Imaging*, 2008.

[29] A. Criminisi, P. Perez, and K. Toyama. Region filling and object removal by exemplar-based inpainting. *IEEE Trans. on Image Process.*, pages 1200–1212, 2004.

[30] K. Dabov, A. Foi, and K. Egiazarian. Video denoising by sparse 3D transform-domain collaborative filtering. In *Proc. European Signal Process. Conf., EUSIPCO*, 2007.

[31] K. Dabov, A. Foi, and K. Egiazarian. Image restoration by sparse 3D transform-domain collaborative filtering. In *Proc. SPIE Electronic Imaging*, volume 6812-07, San Jose, CA, USA, January 2008.

[32] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In *International Conference on Image Processing (ICIP)*, 2007.

[33] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3D transform-domain collaborative filtering. *IEEE Trans. Image Process.*, 16(8), 2007.

[34] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Joint image sharpening and denoising by 3D transform-domain collaborative filtering. In *Int. TICSP Workshop Spectral Meth. Multirate Signal Process., SMMSP*, 2007.

[35] A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian. Image and video super-resolution via spatially adaptive block-matching filtering. In *International Workshop on Local and Non-Local Approximation in Image Processing (LNLA)*, 2008.

[36] D. Datsenko and M. Elad. Example-based single document image super-resolution: a global map approach with outlier rejection. In *Multidim System Signal Processing*, number 18, pages 103–121, 2007.

[37] J. Delon. Midway image equalization. *Journal of Mathematical Imaging and Vision*, 21(2):119–134, 2004.

[38] J. Delon. Movie and video scale-time equalization application to flicker reduction. *IEEE Trans. Image Process.*, 15(1):241–248, Jan. 2006.

[39] J. Dias. Fast GEM wavelet-based image deconvolution algorithm. In *International Conference on Image Processing (ICIP)*, volume 3, pages 961–964, 2003.

[40] J. A. Dobrosotskaya and A. L. Bertozzi. A Wavelet-Laplace variational technique for image deconvolution and inpainting. *IEEE Trans. Image Process.*, 17(5):657–663, 2008.

[41] D. Donoho and I. M. Johnstone. Ideal spatial adaption via wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[42] D. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90:1200–1224, 1995.

[43] M. Ebrahimi and E. Vrscay. Solving the inverse problem of image zooming using self-examples. *Image Analysis and Recognition*, pages 117–130, 2007.

[44] M. Ebrahimi and E. Vrscay. Multi-frame super-resolution with no explicit motion estimation. In *International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, pages 455–459, Las Vegas, NV, USA, 2008.

[45] A. A. Efros and T. K. Leung. Texture synthesis by non-parameteric sampling. In *International Conference on Computer Vision (ICCV)*, volume 2, pages 1033–1038, 1999.

[46] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.*, 15(12):3736–3745, 2006.

[47] M. Elad and A. Feuer. Super-resolution reconstruction of image sequences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21:459–463, 1999.

[48] S. Esedoglu. Blind deconvolution of bar code signals. *Inverse Problems*, 20(1):121–135, February 2004.

[49] S. Farsiu, M. Elad, and P. Milanfar. Multiframe demosaicing and super-resolution of color images. *IEEE Trans. Image Process.*, 15(1):141–159, Jan. 2006.

[50] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar. Fast and robust multiframe super resolution. *IEEE Trans. Image Process.*, 13(10):1327–1344, october 2004.

[51] R. Fattal, M. Agrawala, and S. Rusinkiewicz. Multiscale shape and detail enhancement from multi-light image collections. *ACM Trans. GRAPH (SIGGRAPH)*, page 51, 2007.

[52] P. Favaro, S. Soatto, L. Vese, and S. J. Osher. 3D shape from anisotropic diffusion. In *Computer Vision and Pattern Recognition (CVPR)*, pages I–179–186, June 2003.

[53] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph. (SIGGRAPH)*, 25(3):787–794, 2006.

[54] M. Figueiredo and R. Nowak. An EM algorithm for wavelet-based image restoration. *IEEE Trans. Image Process.*, 12:906–916, August 2003.

[55] M. Figueiredo and R. Nowak. A bound optimization approach to wavelet-based image deconvolution. In *International Conference on Image Processing (ICIP)*, volume 2, pages 782–785, September 2005.

[56] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[57] A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian. Shape-adaptive DCT for denoising and image reconstruction. *Proc. of SPIE Electronic Imaging, Image Processing: Algorithms and Systems V*, 6064, 6064N, 2006.

[58] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single image raw-data. *IEEE Trans. Image Process.*, 17(10):1737–1754, 2008.

[59] P. M. Fr, P. Musé, and F. Sur. Unsupervised thresholds for shape matching. In *International Conference on Image Processing (ICIP)*, pages 647–650, 2003.

[60] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. *IEEE Computer Graphics and Applications*, 2002.

[61] G. Gilboa and S. Osher. Nonlocal linear image regularization and supervised segmentation. *Multiscale Modeling and Simulation*, 6(2):595–630, 2007.

[62] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling and Simulation*, 7(3):1005–1028, 2008.

[63] D. Goldfarb and W. Yin. Second-order cone programming methods for total variation based image restoration. *Journal of Scientific Computing*, 27(2):622–645, 2005.

[64] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[65] U. Grenander. *General Pattern Theory: A Mathematical Study of Regular Structures*. Cambridge University Press, England, 1993.

[66] J. Guerrero-Colon, L. Mancera, and J. Portilla. Image restoration using space-variant gaussian scale mixtures in overcomplete pyramids. *IEEE Trans. Image Process.*, 17(1):27–41, January 2008.

[67] C. J. Harris and M. Stephens. A combined corner and edge detection. In *Alvey Vision Conference*, pages 147–151, 1988.

[68] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[69] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(6):580–593, 1997.

[70] L. He, A. Marquina, and S. Osher. Blind deconvolution using TV regularization and Bregman iteration. *International Journal of Imaging Systems and Technology*, 15(1):74–83, 2005.

[71] J. Huang and D. Mumford. Statistics of natural images and models. In *Computer Vision and Pattern Recognition (CVPR)*, pages 541–547, 1999.

[72] R. A. Hummel, B. Kimia, and S. W. Zucker. Gaussian blur and the heat equation: Forward and inverse solutions. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 416–421, 1983.

[73] R. A. Hummel, B. Kimia, and S. W. Zucker. Deblurring Gaussian blur. In *Computer Vision, Graphics, and Image Processing*, volume 38, pages 66–80, 1987.

[74] M. Irani and S. Peleg. Motion analysis for image enhancement: Resolution, occlusion, and transparency. *Journal of Visual Communication and Image Representation*, 4:324–335, 1993.

[75] C. Kervrann and J. Boulanger. Optimal spatial adaptatio for patch-based image denoising. *IEEE Trans. Image Process.*, 15(10):2866–2878, 2006.

[76] S. Kindermann, S. Osher, and P. Jones. Deblurring and denoising of images by nonlocal functionals. *SIAM Multiscale Modeling and Simulation*, 4(4):1091–1115, 2005.

[77] H. P. Kramer and J. B. Bruckner. Iterations of a non-linear transformation for enhancement of digital images. In *Pattern Recognition*, volume 7, pages 53–58, 1975.

[78] D. Kundur and D. Hatzinakos. Blind image deconvolution. *IEEE Signal Processing Magazine*, 13(3):43–64, May 1996.

[79] D. Kundur and D. Hatzinakos. Blind image deconvolution revisited. *IEEE Signal Processing Magazine*, 13(6):61–63, November 1996.

[80] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2009.

[81] M. S. Lewicki and B. A. Olshausen. Inferring sparse, overcomplete image codes using an efficient coding framework. In *Advances in Neural Information Processing (NIPS)*, pages 815–821, 1998.

[82] T. Lindeberg. Scale-space theory: A basic tool for analysing sturctures at different scale. *Journal of Applied Statistics*, 21(2):224–270, 1994.

[83] M. Lindenbaum, M. Fischer, and A. M. Bruckstein. On Gabor's contribution to image enhancement. *Pattern Recognition*, 27(1):1–8, 1994.

[84] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang. Noise estimation from a single image. *IEEE Computer Vision and Patter Recognition (CVPR)*, 1:901–908, 2006.

[85] C. Liu, J. Yuen, and A. Torralba. Nonparametric scene parsing: label transfer via dense scene alignment. In *IEEE Computer Vision and Patter Recognition (CVPR)*, 2009.

[86] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. Sift flow: dense correspondence across different scenes. In *European Conference on Computer Vision (ECCV)*, 2008.

[87] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[88] J. Ma and F.-X. Le Dimet. Deblurring from highly incomplete measurements for remote sensing. *IEEE Trans. on Geoscience and Remote Sensing*, 47(3):792–802, 2009.

[89] M. Mahmoudi and G. Sapiro. Fast image and video denoising via nonlocal means of similiar neighborhoods. *IEEE Signal Processing Letter*, 12(12):839–842, 2005.

[90] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Trans. on Image Process.*, 17(1):53–69, 2008.

[91] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *SIAM Multiscale Modeling and Simulation*, 7(1):214–241, 2008.

[92] S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Trans. on Signal Process.*, 41:3397–3415, 1993.

[93] A. Marquina. Nonlinear inverse scale space methods for total variation blind deconvolution. *SIAM Journal on Imaging Sciences*, 2(1):64–83, 2009.

[94] A. Marquina and S. Osher. Image super-resolution by TV-regularization and Bregman iteration. *Journal of Scientific Computing*, 37(3):367–382, 2008.

[95] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Image Processing (ICIP)*, volume 2, pages 416–423, July 2001.

[96] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*,

volume 1, pages 384–393, 2002.

[97] M. Mignotte. A segmentation-based regularization term for image deconvolution. *IEEE Trans. Image Process.*, 15(7):1973–1984, July 2006.

[98] M. Mignotte. Image denoising by averaging of piecewise constant simulations of image partitions. 16(2):523–533, 2007.

[99] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *European Conference on Computer Vision (ECCV)*, volume 1, pages 128–142, 2002.

[100] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–68, 2004.

[101] E. E. Mit, E. Eisemann, and F. Durand. Flash photography enhancement via intrinsic relighting. *ACM Trans. Graph. (SIGGRAPH)*, 23:673–678, 2004.

[102] L. Moisan and B. Stival. A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *International Journal of Computer Vision*, 57(3):201–218, 2004.

[103] J. M. Morel and G. Yu. Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009.

[104] D. Mumford and J. Shah. Optimal approximation by piecewise smooth optimal approximation by piecewise smooth functions and associated variational problems. *Commun. Pure Appl. Math*, 42(577-685), 1989.

[105] P. Musé, F. Sur, F. Cao, Y. Gousseau, and J.-M. Morel. An a contrario decision method for shape element recognition. *International Journal of Computer Vision*, 69(3):295–315, 2006.

[106] J. Nagy and Z. Strakos. Enforcing nonnegativity in image reconstruction algorithms. In *SPIE Mathematical Modeling Estimation, and Imaging*, pages 182–190, 2000.

[107] R. Neelamani, H. Choi, and R. G. Baraniuk. ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems. *IEEE Trans. Signal Processing*, 52(2):418–433, Feb. 2004.

[108] D. Nistér. Preemptive RANSAC for live structure and motion estimation. *Mach. Vision Appl.*, 16(5):321–329, 2005.

[109] S. I. Olsen. Estimation of noise in images: an evaluation. *CVGIP: Graph. Models Image Process.*, 55(4):319–323, 1993.

[110] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Trans. Image Process.*, 12(11):1338–1351, 2003.

[111] M. Protter, M. Elad, H. Takeda, and P. Milanfar. Generalizing the non-local-means to super-resolution reconstruction. *IEEE Trans. Image Process.*, 18(1):36–51, 2009.

[112] K. Rank, M. Lendl, and R. Unbehauen. Estimation of image noise variance. In *Vision, Image and Signal Processing*, volume 146, pages 80–84, 1999.

[113] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 860–867, 2005.

[114] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[115] C. Schmid and R. Mohr. Local greyvalue invariants for image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19:530–535, 1997.

[116] Y.-M. Seong and H. Park. Superresolution technique for planar objects based on an isoplane transformation. *Opt. Eng.*, 47(5), 2008.

[117] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. *ACM Transactions on Graphics (SIGGRAPH)*, 2008.

[118] E. Shechtman, Y. Caspi, and M. Irani. Increasing space-time resolution in video. In *European Conference on Computer Vision (ECCV)*, pages 753–768, 2002.

[119] E. Shechtman, Y. Caspi, and M. Irani. Space-time super-resolution. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4):531–545, 2005.

[120] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *International Conference Computer Vision (ICCV)*, volume 2, pages 1470–1477, 2003.

[121] S. M. Smith and J. M. Brady. SUSAN—a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.

[122] H. Takeda, S. Farsiu, and P. Milanfar. Kernel regression for image processing and reconstruction. *IEEE Trans. Image Process.*, 16(2):347–364, 2007.

[123] H. Takeda, S. Farsiu, and P. Milanfar. Deblurring using regularized locally adaptive kernel regression. *IEEE Trans. Image Process.*, 17(4):550–563, April 2008.

[124] H. Takeda, P. Milanfar, M. Protter, and M. Elad. Super-resolution without explicit subpixel motion estimation. *IEEE Trans. Image Process.*, 18(9):1958–1975, 2009.

[125] C.-K. Tang, G. G. Medioni, and M.-S. Lee. N-dimensional tensor voting and application to epipolar geometry estimation. *IEEE Trans. on Pattern Anal. Mach. Intell.*, 23(8):829–844, 2001.

[126] M. Tico. Adaptive block-based approach to image stabilization. In *IEEE International Conference on Image Processing (ICIP)*, 2008.

[127] M. Tico. Multiframe image denoising and stabilization. In *European Signal Processing Conference (EUSIPCO)*, 2008.

[128] A. Tikhonov and V. Arsenin. *Solution of Ill-Posed Problems*. New York: Wiley, 1977.

[129] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *International Conference Computer Vision (ICCV)*, pages 839–846, 1998.

[130] P. H. S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156, 2000.

[131] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. Inform. Theory*, 50:2231–2242, 2004.

[132] T. Tuytelaars and L. V. Gool. Wide baseline stereo based on local, affinely invariant regions. In *British Machine vision conference*, pages 412–422, 2000.

[133] P. Vandewalle, S. Süsstrunk, and M. Vetterli. A frequency domain approach to registration of aliased images with application to super-resolution. *EURASIP Journal on Applied Signal Processing*, 2006:1–14, March 2006.

[134] A. Vedaldi and S. Soatto. Local features, all grown up. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 1753–1760, 2006.

[135] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Trans. on Image Process. Special Issue: Image Sequence Compression*, 3(5):625–638, 1994.

[136] L.-Y. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In *Computer Graphics and interactive techniques*, pages 479–488, 2000.

[137] Y. Weiss and W. T. Freeman. What makes a good model of natural images. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[138] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2008.

[139] L. P. Yaroslavsky. *Digital Picture Processing, an Introduction*. Springer-Verlag, Berlin, 1985.

[140] W. Yin, D. Goldfarb, and S. Osher. Image cartoon-texture decomposition and feature selection using the total variation regularized L1 functional. In *Variational, Geometric, and Level Set Methods in Computer Vision*, volume 3752, pages 73–84, 2005.

[141] G. Yu and S. Mallat. Sparse super-resolution with space matching pursuit. In *Signal Processing with Adaptive Sparse Structured Representation*, 2009.

[142] G. Yu and S. Mallat. Super-resolution with sparse mixing estimators. *to appear in IEEE Trans. on Image Processing*, 2010.

[143] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Image deblurring with blurred/noisy image pairs. *ACM Trans. GRAPH (SIGGRAPH)*, page 1, 2007.

[144] Z. Yuan, P. Yan, and S. Li. Super resolution based on scale invariant feature transform. In *International Conference on Audio, Language and Image Processing*, pages 1550–1554, 2008.

[145] L. Zhang, S. Vaddadi, H. Jin, and S. Nayar. Multiple view image denoising. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2009.

[146] X. Zhang, M. Burger, X. Bresson, and S. Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *to appear in SIAM Journal on Imaging Sciences*, 2010.

[147] Z. Zhang, R. Deriche, O. Faugeras, and Q. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1-2):87–119, 1995.

[148] W. Zhao and H. S. Sawhney. Is super-resolution with optical flow feasible? In *European Conference on Computer Vision (ECCV)*, pages 599–613, 2002.

[149] D. Zhou and B. Scholkopf. A regularization framework for learning from graph data. In *ICML Workshop on Stat. Relational Learning and Its Connections to Other Fields*, 2004.