

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

PATCH-BASED IMAGE DENOISING AND ITS PERFORMANCE LIMITS

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

by

Priyam Chatterjee

June 2011

The Dissertation of Priyam Chatterjee
is approved:

Professor Peyman Milanfar, Chair

Professor Benjamin Friedlander

Professor Boaz Nadler

Tyrus Miller
Vice Provost and Dean of Graduate Studies

Copyright © by
Priyam Chatterjee
2011

Table of Contents

List of Figures	vi
List of Tables	ix
Abstract	x
Dedication	xii
Acknowledgments	xiii
1 Image Denoising and the State-of-the-Art	1
1.1 Introduction	1
1.2 Contributions	5
1.3 Review of Image Denoising Methods	6
1.3.1 Spatial-domain Denoising	7
1.3.2 Transform-domain Methods	18
1.4 The State-of-the-Art	21
2 Fundamental Limits for Image Denoising	25
2.1 Introduction	25
2.2 Bias in Denoising	28
2.3 Optimal Bias Bayesian Cramér-Rao Lower Bound	35
2.4 Lower Bound on the MSE	38
2.4.1 Deriving the Bayesian MSE bound	39
2.4.2 Fisher Information Matrix	42
2.5 Bounds for General Images	46
2A Mathematical Justification for Affine Bias	49

2B	Optimal Parameters for Affine Bias Function	50
2C	Higher Order Bias Model	51
3	Estimation of Denoising Bounds	56
3.1	Introduction	56
3.2	Estimating Denoising Bounds from Ground Truth	58
3.2.1	Practical Geometric Clustering	59
3.2.2	Covariance Estimation from Ground Truth	62
3.2.3	Calculating Patch Redundancy	64
3.3	Bounds Estimation for Noisy Images	66
3.3.1	Covariance Estimation	67
3.3.2	Photometric Redundancy from Noisy Images	70
3.4	Denoising Bounds and State-of-the-Art	71
4	Information Theoretic Interpretations of the MSE Bound	85
4.1	Introduction	85
4.2	Denoising Bounds and Mutual Information	88
4.3	Relationship between Denoising Bounds and Entropy	93
4A	Entropy Estimation	101
4B	Relation between Mutual Information and MMSE Matrix	104
4C	Derivation of Overall Entropy	107
5	Patch-based Locally Optimal Wiener (PLOW) Denoising	109
5.1	Introduction	109
5.2	Patch-based Wiener filter	111
5.3	Patch-based Locally Optimal Wiener Filter (PLOW)	113
5.4	Parameter Estimation for Denoising	117
5.4.1	Geometric Clustering and Moment Estimation	118
5.4.2	Calculating Weights for Similar Patches	120
5.4.3	Aggregating Multiple Pixel Estimates	122
5.5	Experimental Results	125
5A	Derivation of Noise Covariance for Similarity Model	137
5B	Derivation of Redundancy Exploiting Wiener Filter	139
5C	Derivation of Approximate Similarity Measure	140

6	Conclusions and Future Work	143
6.1	Conclusions	143
6.2	Future Works and Extensions	151
6.2.1	Extending PLOW to Different Degradation Models	151
6.2.2	Guided Filtering with Image Pairs	159
6.2.3	Accounting for Intensity Dependent Noise	161
	Bibliography	162

List of Figures

1.1	Image formation model illustrating the various noise sources.	2
1.2	Illustration of the concept of search window, patches and similar patches	7
1.3	Illustration of distance measures of different weight functions	10
1.4	Framework for denoising with locally learned dictionaries	15
1.5	Clustering of a simple image based on geometric similarity.	16
1.6	Principal operations in shrinkage-based denoising methods	18
1.7	Some popular benchmark images used for different experiments in this thesis.	20
1.8	State-of-the-art denoising performance: Is denoising dead?	22
2.1	Example of bias in denoising produced by some modern denoising methods	29
2.2	Some images consisting of geometrically similar patches that we use for our study.	30
2.3	Visual comparison of the method bias and predicted bias for BM3D & K-SVD for the grass image	32
2.4	The spatial distribution of N_i values for a patch size of 11×11 on (a) house image, and (b) Barbara image, shown in Fig. 1.7.	46
2.5	Clustering results by K-Means algorithm on the box image.	47
3.1	Outline of the bounds estimation process.	57
3.2	Steering kernels at different locations of the house image.	59
3.3	Clustering using K-Means for the box and house images.	61
3.4	Some query patches and their respective least similar neighbors as defined by (3.4) with various values of p found from a dictionary of approximately 450,000 noise-free patches from 4 different images.	64

3.5	The spatial distribution of N_i values for a patch size of 11×11 on (a) house image, and (b) Barbara image, shown in Fig. 1.7.	66
3.6	Clustering of noisy and noise-free Barbara images into 5 clusters based on geometric structure of patches	67
3.7	Some images consisting of geometrically similar patches that we use for our study.	72
3.8	MSE bounds for noise standard deviation 25 as a function of (a) varying patch size with $K = 1$ for the grass and cloth images (Fig. 3.7), and $K = 5$ for the house and Barbara images; and (b) varying number of clusters with patch size 11×11	73
3.9	MSE bounds computed on simulated images and compared with the performance of some state-of-the-art methods.	75
3.10	Bounds for texture images compared to denoising performance of some state-of-the-art denoising methods.	76
3.11	Comparison of some state-of-the-art methods with our bounds formulation for some general images.	77
3.12	MSE bounds estimated from a given noisy image compared to the ground truth where the bounds are calculated from clean images	81
4.1	Illustration of the modified data model considering all patches that are photometrically similar to any given reference patch \mathbf{z}_i in the image. . .	86
4.2	Effect of noise on different parts of the House image.	91
4.3	Density estimation of points sampled from an unknown pdf at a reference point.	94
4.4	Illustration of the relation between N_i and cluster complexity	99
4.5	Estimation of entropy for data sampled from a multidimensional Gaussian density function as a function of dimensions and number of samples.	103
5.1	Illustration of the data model formed by expressing all photometrically similar patches.	113
5.2	Outline of our patch-based locally optimal Wiener (PLOW) filtering method.	117
5.3	An illustration of how a pixel is estimated multiple times due to overlapping patches.	121
5.4	Comparison of denoising results for the house image corrupted by WGN of $\sigma = 25$	127

5.5	Comparison of denoising results for the Barbara image corrupted by WGN of $\sigma = 25$	128
5.6	Comparison of denoising performance with leading denoising methods for Lena, man and stream images (Fig. 1.7) corrupted by $\sigma = 25$	129
5.7	Comparison of (cropped) denoising results for color images corrupted by 5% WGN.	132
5.8	Denoising of some real noisy color images.	134
5.9	Restoration of images with non-Gaussian noise profiles.	135
6.1	Comparison of denoising results with MSE bounds for some benchmark images corrupted by varying levels of additive WGN.	144
6.2	Comparison of denoising results with MSE bounds for some benchmark images corrupted by varying levels of additive WGN.	147
6.3	Image formation model showing the different degradation steps that the image goes through due to camera hardware limitations.	151
6.4	Illustration of correlation among the red, green and blue color channels.	153
6.5	Illustration of patch formation in Bayer patterned raw images.	156

List of Tables

2.1	R^2 values for the affine model fit of the bias produced by different methods for different images.	34
3.1	Some images ranked according to the predicted denoising bounds showing their relative denoising difficulty. The noise standard deviation is 25 and the bounds are calculated using 11×11 patches.	79
3.2	Comparison of bounds from noisy and noise-free images considered to be ground truth.	82
4.1	Clustering of the house image and the cluster-wise mutual information estimates when corrupted by various levels of WGN.	90
4.2	Ranking of images based on denoising difficulty as indicated by the MI, compared to the entropy, the denoising bound and MSE of BM3D denoising algorithm for WGN.	93
5.1	Denoising performance of some popular methods (NLSM, BM3D) under WGN corruption, compared to PLOW, with and without oracle information. Results noted are average PSNR (top), SSIM (middle) & Q -measure (bottom) over 5 independent noise realizations for each σ	131
6.1	Some images ranked according to improvement in denoising yet to be achieved, as predicted by our bounds. The noise standard deviation is 25 and the bounds are calculated using 11×11 patches.	146

Abstract

Patch-based Image Denoising and its Performance Limits

by

Priyam Chatterjee

Recently there has been considerable increase in the casual and commercial uses of image and video capturing devices. Apart from their applications in photography, the captured data are often inputs to sophisticated object detection and tracking, and action recognition methods, applications of which permeate different industries. Captured images are often not of desired quality and need to be enhanced by software. One of the major causes of the performance degradations for most methods is the presence of noise. Noise removal, therefore, forms a critical first step for many applications. In this thesis, we concentrate on this important problem of image denoising.

Image denoising has been an active field of research with literature dating back to the 1970s. However, given the importance of the problem, considerable effort still continues to be channeled to bettering the state-of-the-art. Surprisingly, performance improvement in recent years has been somewhat limited. In this thesis, we first study the possible causes of such restricted improvement. To do so we analyze the problem of image denoising in a statistical framework. Since the best performing methods are patch-based, we frame the problem of denoising as that of estimating the underlying image patches from their noisy observations. The performance limits of this class of methods is then studied in a Bayesian Cramér-Rao lower bound framework. We show that the denoising bounds depend on the image content as well as the noise statistics, and are related to different information-theoretic measures. In this

thesis, we also show how the denoising bound can be calculated for any given image. Finally, we use the insights gained in our analysis to develop a practical denoising algorithm (FLOW) that is designed to achieve the theoretical limits of the denoising. Through various experiments, we show that our proposed method achieves state-of-the-art performance, both visually and quantitatively.

In loving memory of my grandfather, Sameer Rai Chaudhury.

Acknowledgments

I can no other answer make, but, thanks, and thanks. - William Shakespeare

As the curtains draw on my journey as a PhD student, and I look forward to embarking on my career, I have many people to thank. Foremost of them is my advisor, Professor Peyman Milanfar, without whom this thesis would not have been possible. You provided me with the necessary direction and encouragement that any student may hope for from an advisor. Words cannot express my gratitude to you for your patience and support. Our interactions have helped transform me into the researcher that I am today.

I am also thankful to Professor Benjamin Friedlander and Professor Boaz Nadler for being a part of my thesis committee and providing useful comments and constructive criticisms that have helped improve this thesis. I am also grateful to Dr. Sing Bing Kang, Dr. Neel Joshi and Dr. Yasuyuki Matsushita for a very interesting summer at Microsoft Research. It was a pleasure working with you. I also thank Professor Michael Elad for his involvement in the initial stages of my thesis, and for providing useful feedback on many of my papers. I have had the pleasure of sharing the lab with some of the most interesting friends and colleagues. Thank you, Aryn, Hae Jong, Hiro, Xiang, Erik, Hossein and Chelhwon for making the lab a fun place to work over the years. I also take this opportunity to thank Carol Mullane for her careful guidance on all official requirements throughout the program.

Santa Cruz has played a very important role in my years at UCSC, and I thank my friends for making my stay here enjoyable. I cannot ever forget the endless meaningless discussions that continue to define Baskin-tea. Sumanth, Anita, Avik, Reema, Suchit, Dhananjay, Anindya, Rajeswari, Suresh, Ashok, Shirish, Subhash, Ambarish, Sangeetha, Pritam, Varun, Pranav, Shruti, Mainak, Sanchita, Mrunal, Neeraj, Michael,

Valerie, Sudha, Seinjuti, (pew!) thank you for all the cherished happy times. A special thanks goes out to Madhan for lifting my spirits by volunteering to drive us to West Cliff on numerous occasions, and to Aryn for continuing to be a readily available friend, philosopher and guide. I would also like to thank my friends over the hill who have made the move to Bay Area very welcoming. All work and no play would have made my days very dull indeed and I thank Morteza and Hae Jong for ensuring a non-sedentary lifestyle through various sporting activities.

While my friends have made my years in school fun, I cannot forget the support of my parents, and their sacrifices that have made this achievement possible. My mother has been a constant source of love and inspiration, instilling in me an attitude of staying positive and not giving up even when the chips are down. I thank my father for providing regular healthy doses of fun and laughter while growing up and for standing behind all my decisions unconditionally. I also acknowledge the support of my brother for being there and taking up my share of responsibilities. Dadu, I owe my wanderlust nature to you. I wish you were here to witness this. Thank you Dida for caring too much. I also have the pleasure of thanking my in-laws who have always kept my best interests in mind. Last, and most definitely not the least, I want to thank my loving wife for standing by me through my years at school, for believing in me and making life wonderful in general. I am lucky to have you by my side as a constant source of encouragement.

Santa Cruz, California

June 9, 2011

Priyam Chatterjee

Chapter 1

Image Denoising and the State-of-the-Art

Abstract – This chapter introduces the problem of image denoising, and discusses the various sources and characteristics of noise corrupting images and why denoising is an important problem. We then briefly study the similarities and dissimilarities among the various approaches that have been proposed over the last decade. Finally, we discuss the state-of-the-art in image denoising and its improvement over the years by comparing the performance of some of the best denoising methods.

1.1 Introduction

In recent years, images and videos have become integral parts of our lives. Applications now range from casual documentation of events and visual communication, to the more serious surveillance and medical fields. This has led to an ever-increasing demand for accurate and visually pleasing images. However, images captured by modern cameras are invariably corrupted by noise [1]. Apart from the obvious reduction in image quality, such noise usually also hinders the performance of

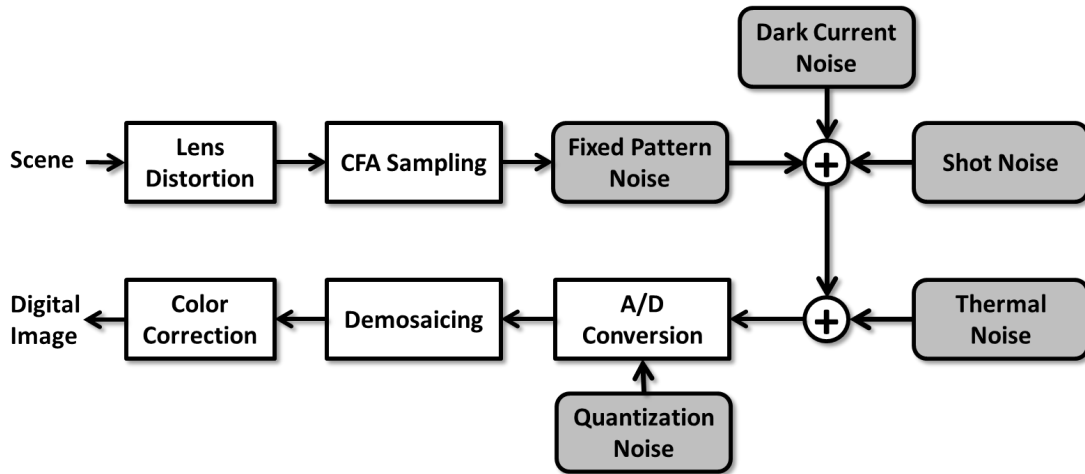


Figure 1.1: Image formation model illustrating the various noise sources. Adapted from [2].

subsequent computer vision applications, such as tracking, object detection, etc. Suppressing such noise is, thus, the usual first step. However, before this can be performed, it is imperative to understand the source and characteristics of the corrupting noise. To do so, we briefly study the image formation pipeline (Fig. 1.1).

The image capturing process of digital cameras is based on light from the scene entering the camera and being focused on a sensor where it is converted to digital information. These light rays undergo various processing stages (see Fig. 1.1) before the final digital image is produced. They are first distorted by the lens which focuses the incident light on the camera sensor. In modern commercial cameras, the (CCD or CMOS) sensor usually includes a color filter array (CFA) in which each sensor element reacts to a particular range of light wavelengths. The incident rays, or photons, reaching each sensor element are then accumulated and converted to electrical voltage which is then read and stored in digital form. The CFA data has incomplete color information at each location (pixel) and, thus, must be interpolated by a demosaicing

process. Further adjustments such as white balancing, gamma correction and color tone-mapping are then performed before the final image is produced.

Noise corrupting the final image is introduced in different forms in various stages of the image formation pipeline [2], illustrated in Fig. 1.1. Some of these noise sources stem from the camera characteristics. For example, the fixed pattern noise appears due to non-uniform response of the sensor elements and dark current noise is a result of aberrant charges appearing at the sensors even without any incident photons. These noise sources can be effectively modeled for a given camera and, hence, controlled. Thermal noise appears due to heating of electronic components of the camera with use. This noise increases as a function of the exposure time and duration of use. In general, for short exposures the effect of thermal noise is minimal. Effects of quantization noise, arising as a result of analog-to-digital conversion of the signal, can also be mitigated by using sufficient number of bits to store pixel information.

On the other hand, shot noise is due to the photonic nature of light. Since photons do not hit the sensors uniformly even for a uniform scene, the resultant observation is always noisy. This photon counting process makes the noise signal-dependent and is usually modeled to be Poisson distributed [3]. When the image is well-exposed, that is, a large number of photons are incident on the camera sensor, the Poisson probability density function (pdf) closely resembles a Gaussian pdf. Moreover, as the photons are accumulated in each sensor element independent of neighboring elements, the noise can also be assumed to be spatially uncorrelated. As a result, image noise

is popularly modeled to be zero mean independent and identically distributed (iid) Gaussian¹, or white Gaussian noise (WGN).

Under photon-limited low light conditions, the noise is known to be dominated by the Poisson distributed shot noise [1, 4]. However, in such cases variance stabilization methods, such as Anscombe root transformation [5], can be applied to obtain an approximation of a Gaussian distributed signal. Any denoising algorithm based on the Gaussian assumption of noise can then be applied, following which an inverse transformation is performed to obtain the denoised image [6]. Thus, denoising methods addressing Gaussian noise are practically applicable even in such cases [7].

The image denoising problem can be posed as that of estimating the noise-free pixel intensity z_i from its noisy observation² y_i at each location i where

$$y_i = z_i + \eta_i. \tag{1.1}$$

Here η_i is the corrupting noise, assumed to be WGN with a certain standard deviation σ . With increasing pixel resolution and limited sensor sizes, fewer photons are available at each sensor element, leading to more pronounced noise effects. Noise suppression has, thus, become more relevant. While advances in optics and hardware try to mitigate such undesirable effects, software-based denoising approaches are more popular as they are usually device independent and widely applicable. In this thesis we focus on such software-based approaches for image denoising.

¹In practice, demosaicing of the noisy CFA image corrupts the spatial independence as well as the Gaussian structure of the noise. However, this color interpolation is usually performed within a small neighborhood, thus making the iid Gaussian assumption a reasonable approximation.

²For color images, y_i represents each of the color components, while for grayscale images y_i is the observed intensity.

1.2 Contributions

Image denoising is a very basic problem that is of wide interest to the image processing and computer vision communities. In this thesis, we perform a thorough statistical analysis of the image denoising problem leading to a practical denoising method that achieves near-optimal performance. The remainder of this thesis is organized as follows:

- **Chapter 2 – Fundamental Limits for Image Denoising**

In this chapter, we present an expression for the performance limits of image denoising. Considering the superior performance of patch-based methods, we cast the problem of denoising as that of estimating the unknown noise-free patch intensities at each image location. A lower bound on the mean squared error (MSE) of the estimate is then formulated in a Bayesian Cramér-Rao bound framework.

- **Chapter 3 – Estimation of Denoising Bounds**

This chapter deals with the estimation of the bounds from a given image. For this work, we assume the noise to be WGN. We estimate the bounds by independently estimating the different parameters of the denoising bounds. We first present methods of estimating the parameters assuming the noise-free image to be available. These methods are then generalized to account for the presence of noise in the input image. Through experiments, we show that the estimated parameters are accurate enough to obtain an estimate of the performance limits for denoising any given noisy image.

- **Chapter 4 – Information Theoretic Interpretations of the MSE Bound**

Here, we analyze the bounds formulation of Chapter 2 from an information theo-

retic point of view. We show that the bounds are related to information theoretic measures and that parameters of the bound are connected through such information theoretic measures. We also show how information theoretic measures can be used to determine relative denoising difficulty among noisy images.

- **Chapter 5 – Patch-based Locally Optimal Wiener (PLOW) Denoising**

In this chapter, the insights gained in Chapters 2 & 3 are used to realize a practical patch-based image denoising algorithm that achieves or improves on the current state-of-the-art. As an added advantage, the proposed method has a sound statistical basis that justifies its performance.

Although developed for denoising, a more generalized model of our denoising algorithm developed in Chapter 5 can be applied to many other image processing problems. We point out such directions and conclude the findings of our research in Chapter 6. Before proceeding with the analysis of the denoising problem and its performance limits, we provide a brief overview of the various existing approaches for denoising in Sec. 1.3 followed by a discussion of the current state-of-the-art in Sec. 1.4.

1.3 Review of Image Denoising Methods

Image denoising has been a well-studied problem. The challenge facing any denoising algorithm is to suppress noise artifacts while retaining finer details and edges in the image. Over the years, researchers have proposed many different methods that attempt to achieve these contradictory goals. These methods vary widely in their approaches. Broadly, these denoising filters can be categorized based on their domain

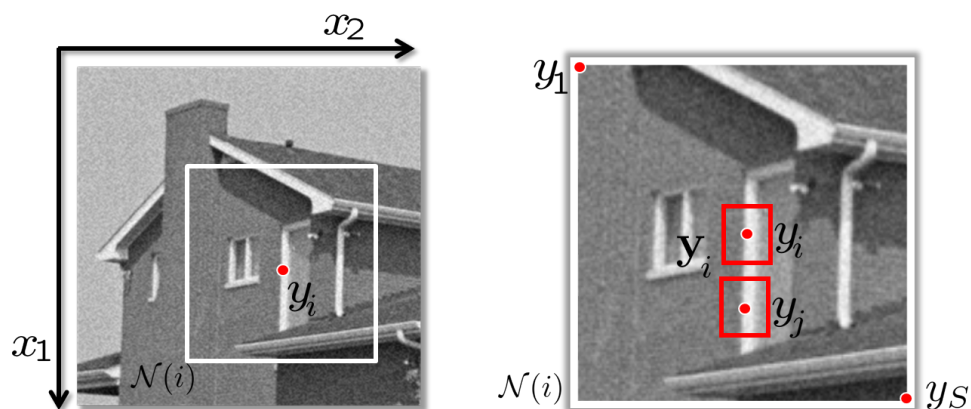


Figure 1.2: Illustration of the concept of search window $\mathcal{N}(i)$, patches and similar patches. The square region bounded in red with noisy pixel of intensity y_i at the center is the patch y_i . If two pixels y_i and y_j have similar neighborhoods, we term the patches y_i and y_j as similar.

of denoising - spatial or transform domain. Below, we briefly outline some of the most popular approaches within each category.

1.3.1 Spatial-domain Denoising

Denoising methods where the pixel intensities are used directly in the denoising process are said to be spatial-domain filters. Even within this class of denoising methods, the actual approaches can vary significantly. In general, the most successful approaches can be classified as being either a process where denoising is performed by a weighted averaging of pixel intensities; or an explicit model-based approach where parameters of the data model are usually learned from the noisy image itself.

Weighted Averaging Methods:

The underlying concept behind many spatial-domain denoising filters is to suppress noise through a weighted averaging process which can be mathematically

written as

$$\hat{z}_i = \sum_j W_{ij} y_j, \quad \text{such that } \sum_j W_{ij} = 1. \quad (1.2)$$

Here \hat{z}_i is the denoised estimate of the pixel at i , and W_{ij} is the normalized weight that controls the influence of y_j in the denoising of y_i . The simplest weighted averaging process for denoising is averaging pixels within a local neighborhood (or search window) with a normalized weight function

$$W_{ij} = \frac{w_{ij}}{\sum_j w_{ij}} \quad \text{where } w_{ij} = \begin{cases} 1, & \text{if } j \in \mathcal{N}(i) \\ 0, & \text{otherwise.} \end{cases} \quad (1.3)$$

Here $\mathcal{N}(i)$ denotes a small $\sqrt{S} \times \sqrt{S}$ neighborhood centered at i (see Fig. 1.2). This simple averaging can be considered to be a special case of the Yaroslavsky filter [8] where pixels closer to the pixel of interest exert greater influence over the denoising process. The weight function in [8] is designed as a Gaussian:

$$w_{ij} = \exp \left\{ -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{h_s^2} \right\}, \quad (1.4)$$

where $\mathbf{x}_i = [x_1 \ x_2]_i$ denotes the position of the i -th pixel in a two dimensional coordinate system, as shown in Fig. 1.2. Here h_s^2 is the bandwidth of the Gaussian filter that controls the level of smoothing. The filters in Equations 1.2 & 1.4 do not take into account intensity information. As a result, pixels across edges are averaged in the denoising process, leading to loss of detail and edge sharpness.

To restrict such loss of detail in the image, it is important to ensure that the averaging is performed only over photometrically similar pixels. For example, to denoise the pixel y_i of Fig. 1.1, we should use the intensity information of y_j , but not y_1 . However, in the presence of noise, identifying such similar pixels can be challenging. One of the first approaches making use of a data adaptive weight function is attributed

to Smith *et al.* (SUSAN [9]) and Tomasi *et al.* (bilateral filter [10]). The non-linear weight function proposed by them takes the form

$$w_{ij} = \exp \left\{ -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{h_s^2} \right\} \exp \left\{ -\frac{(y_i - y_j)^2}{h_r^2} \right\}. \quad (1.5)$$

The weight function here takes into account the local intensity information as well. An additional smoothing parameter h_r^2 is introduced and this intensity bandwidth needs to be tuned based on the corrupting noise level. The added photometric term ensures that similar pixels in the neighborhood are preferred in the averaging, thus avoiding smoothing across edges. However, as the noise level increases, the filter's ability to distinguish between similar and dissimilar pixels degrades quickly [11].

Buades *et al.* [12] and Awate *et al.* [13] independently proposed a simple modification that lends robustness to this weight function. There the authors generalized the photometric part of the weight function by taking into account information from the vicinity of the pixels as well (see Fig. 1.2). That is to say, instead of comparing intensities of a pair of pixels, intensities of local groups of pixels (patches) are compared. Thus, the denoising process in [12] uses a patch-based weight function

$$w_{ij} = \exp \left\{ -\frac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{h_r^2} \right\}, \quad (1.6)$$

where \mathbf{y}_i is a group of pixels within a small (usually rectangular) vicinity of y_i with its center at i . Referring to Fig. 1.2, the weight for y_j in denoising y_i is high as patches \mathbf{y}_i and \mathbf{y}_j are also similar. Since a group of pixels are compared, the weight calculation scheme is considerably better at rejecting dissimilar pixels from the averaging process.

An interesting aspect of Eq. 1.6 is that there is no spatial weight component. As a result, the search for similar patches can be conducted over the entire image leading to a *non-local* formulation. However, doing so can be computation-

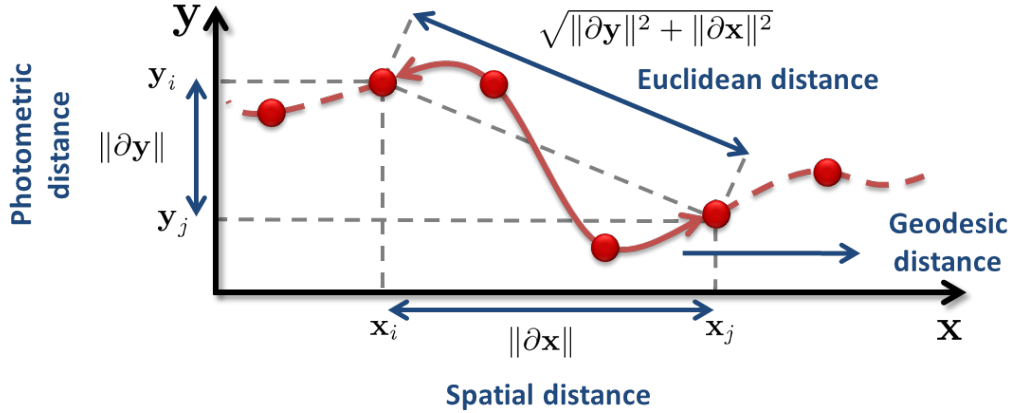


Figure 1.3: Illustration of distance measures of different weight functions, adapted from [21]. The measure of distance between two points used by the Yaroslavsky filter [8] is $\|\partial\mathbf{x}\|^2$ while NLM [12] uses $\|\partial\mathbf{y}\|^2$. The patch-based bilateral filter [9,10] measures the Euclidean distance, while the steering kernel of [22] measures the geodesic distance along the solid red curve.

ally prohibitive [12]. Although many methods have been proposed to speed up this process [14–17], it was argued in [18] that an exhaustive search can often lead to performance loss. Consequently, performing the averaging over a smaller search window $\mathcal{N}(i)$ is usually preferable. The size of this search window thus plays a role in the denoising performance of the non-local means (NLM) filter. For example, when the underlying image is smooth, a larger search window is effective in suppressing the noise, while textural regions gain from using a smaller neighborhood. Kervrann *et al.* [19,20] exploited this observation, along with a slightly modified weight function:

$$w_{ij} = \exp \left\{ -\frac{(\mathbf{y}_i - \mathbf{y}_j)^T (\mathbf{V}_i^{-1} + \mathbf{V}_j^{-1}) (\mathbf{y}_i - \mathbf{y}_j)}{h_r^2} \right\}. \quad (1.7)$$

Here, the diagonal matrix \mathbf{V}_i contains the variance of each pixel of the patch \mathbf{y}_i as its entries. Such optimal spatial adaptation of the search window leads to considerable improvement in denoising performance of the NLM filter.

The weight measures that we have discussed until now can be generally con-

sidered to be the exponential of the negative distance between two points. In Fig. 1.3 we illustrate the different distance measures used by different methods. Considering a simplified 1-dimensional representation of the image where the horizontal and vertical axes denote the intensity and spatial location of pixels (patches), we can see that the Yaroslavsky filter measures the positional distance between two point at \mathbf{x}_i and \mathbf{x}_j . NLM, on the other hand, measures the intensity distance, while the (patch-based) bilateral filter uses the Euclidean distance metric.

In [22], Takeda *et al.* proposed a very different approach to computing the distance between two pixels. The authors compute the approximate geodesic distance which is the distance along the signal curve [21], denoted by the solid red line in Fig. 1.3. The weight function used there is

$$w_{ij} = \frac{\sqrt{|\mathbf{C}_j|}}{2\pi h^2} \exp \left\{ -\frac{(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{C}_j (\mathbf{x}_i - \mathbf{x}_j)}{h^2} \right\}, \quad (1.8)$$

where h^2 is a global smoothing parameter, \mathbf{C}_j is the structure tensor [23] at location j measured from the local gradient information, and $|\cdot|$ denotes the determinant. With a robust estimate of the local structure matrix \mathbf{C}_j , this weight measure has proved to be effective in discriminating similar and dissimilar pixels even in the presence of considerable noise [11]. A more detailed analysis of this locally adaptive regression kernel (LARK) is provided in [11, 24, 25].

As opposed to the methods discussed earlier, the normalized denoising weights W_{ij} of [22] are obtained in a higher order kernel regression framework [26]. There, the authors assume the underlying noise-free image to be a regression function of which only noisy samples are observed. This regression function is assumed to be sufficiently smooth locally to allow fitting of some low (usually 0, 1 or 2) degree polynomial. This

leads to a local patch-based³ model as

$$\mathbf{y}_i = \mathbf{z}_i + \boldsymbol{\eta}_i = \boldsymbol{\Phi}\boldsymbol{\beta}_i + \boldsymbol{\eta}_i, \quad (1.9)$$

where $\boldsymbol{\Phi}$ is the basis containing polynomial functions in its columns as shown in Eq. 1.10, $\boldsymbol{\beta}_i$ is the vector of coefficients and $\boldsymbol{\eta}_i$ is the noise patch. The polynomial matrix $\boldsymbol{\Phi}$ for a 2nd order polynomial model (as used in [22]) has the form

$$\boldsymbol{\Phi} = \begin{bmatrix} 1 & (\mathbf{x}_j - \mathbf{x}_i) & \text{vech}\{(\mathbf{x}_j - \mathbf{x}_i)(\mathbf{x}_j - \mathbf{x}_i)^T\} \\ \vdots & \vdots & \vdots \end{bmatrix}, \quad (1.10)$$

where $\text{vech}(\mathbf{A})$ denotes the raster scanned lower triangular form of the matrix \mathbf{A} .

To allow for discontinuities at edges, this polynomial model fitting is restricted to only pixels that are similar to the reference pixel in the center of the patch. For this, the weight function of Eq. 1.8 is used. The final estimate of the denoised pixel is obtained as the weighted least squares solution:

$$\begin{aligned} \hat{\boldsymbol{\beta}}_i &= \arg \min_{\boldsymbol{\beta}_i} (\mathbf{y}_i - \boldsymbol{\Phi}\boldsymbol{\beta}_i)^T \mathbf{K}_i (\mathbf{y}_i - \boldsymbol{\Phi}\boldsymbol{\beta}_i) \\ &= (\boldsymbol{\Phi}^T \mathbf{K}_i \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T \mathbf{K}_i \mathbf{y}_i, \end{aligned} \quad (1.11)$$

$$\text{and } \hat{z}_i = \mathbf{e}_1^T \hat{\boldsymbol{\beta}}_i, \quad (1.12)$$

where $\mathbf{K}_i = \text{diag}([\dots w_{ij} \dots])$ and \mathbf{e}_1 is the first column of an identity matrix. In the special case where a zero-th order polynomial model is considered, the polynomial matrix becomes $\boldsymbol{\Phi} = [\dots 1 \dots]^T$. This leads to the estimator taking the form of Eq. 1.2 with $W_{ij} = \frac{w_{ij}}{\sum_j w_{ij}}$.

In general, a higher order polynomial forms a better local model. However, if too high an order is chosen, the local image model starts fitting to noise. Consequently,

³In this case, the search window $\mathcal{N}(i)$ and patch \mathbf{y}_i (see Fig. 1.2) are the same.

the authors in [22] used a 2nd order polynomial. It is important to note that such a generalization is not restricted to the locally adaptive regression kernel (LARK) weights of Eq. 1.8 and can be applied to other methods as well. In [27–30] it was shown that the higher order bilateral [9, 10], NLM [12] and OSA [19] filters improved on the performance of the respective zero-th order filters. As such, most weighted averaging methods can be cast in a kernel regression framework where the main difference among the methods lies in the choice of respective weight functions. However, irrespective of the distance metric (weight function) chosen, the weight matrices share many common characteristics. A very detailed study of such intrinsic properties is presented in [21].

The kernel regression framework has also been shown to have an equivalent variational formulation. Brox *et al.* [31] analyzed this relation to derive an iterative framework involving the NLM filter. Another variational interpretation of the NLM filter was used for image denoising and segmentation by Gilboa *et al.* [32]. In [33], Barash illustrated the relation between the bilateral filter and anisotropic diffusion [34], which in turn also has a variational interpretation [35]. The bilateral filter was also shown to have a Bayesian formulation in [36]. While the above equivalence analyses were performed for specific kernels, parallel relations can be drawn for other weighting functions as well [21].

The steering kernel regression (SKR) framework of [22] uses a local polynomial model for the data. This polynomial model arises from making use of a local Taylor expansion of the underlying image which is assumed to be a locally smooth regression function. However, the framework admits other basis functions as well. As such, alternate choices of bases lead to different local models (Φ) for the image. Next,

we briefly discuss some of the spatial-domain approaches where more explicit models are used to perform denoising.

Denoising through Data Modeling:

While the kernel regression-based framework typically employs an implicit local model of the image for denoising, many spatial-domain methods employ a more explicit model-based approach. In most of these methods the models act as prior information about the clean image and are either learned a priori from noise-free natural images or directly from the noisy image. Denoising is then performed by enforcing these priors on the noisy image.

One of the most popular model-based methods is the K-SVD algorithm [37]. There the authors propose a patch-based framework where each patch in the image is represented as a linear combination of patches from some over-complete set of bases. Building on the observation that noise-free image patches are sparse-representable [38], the authors enforce a constraint on the number of basis patches (or atoms) that can be used in estimating any given patch. Mathematically, the problem can be formulated as

$$\begin{aligned} \hat{\mathbf{z}}_i &= \Phi \hat{\beta}_i, \text{ where} \\ \hat{\beta}_i &= \arg \min_{\beta_i} \|\mathbf{y}_i - \Phi \beta_i\|^2 \text{ subject to } \|\beta_i\|_0 \leq \tau. \end{aligned} \quad (1.13)$$

Here τ is some small threshold that controls the level of sparsity, thus ensuring that the data model does not fit to noise. In practice, the optimization framework of Eq. 1.13 is replaced by a convex approximation where the ℓ_0 norm is replaced by an ℓ_1 norm.

The over-complete basis set (or dictionary) Φ can be formed either offline from some parametric basis functions (say, DCT), or patches from clean images; or online using patches from the noisy image itself. In [37], the authors note that the

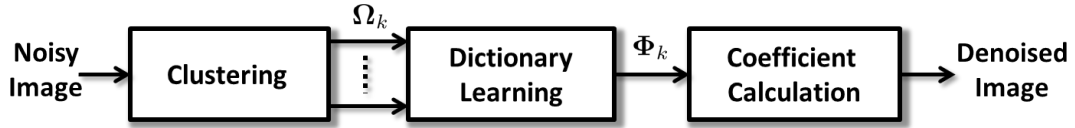


Figure 1.4: Framework for denoising with locally learned dictionaries [24].

latter approach is more practical and also leads to better denoising performance. In that case, the dictionary Φ also needs to be learned. In [37], the Φ and the coefficients β_i are estimated alternately in an iterative framework.

The sparsity constraint on β_i ensures that each estimated patch lies in some low dimensional subspace spanned by few atoms. In [37, 38] these atoms are normalized to have unit norm. Consequently, one of the atoms in the support of any given patch is the constant patch which can be scaled to match the mean intensity of the patch being restored. It is then reasonable to assume that *structurally* similar patches share the same support, although the coefficients β_i may vary. The global over-complete dictionary can then be thought to be a union of multiple smaller *local* dictionaries where locality is defined by geometric or structural similarity among patches.

This observation was exploited in [24] where we proposed a denoising framework using multiple locally learned dictionaries. The basic outline of the framework is illustrated in Fig. 1.4. Denoising there follows three steps. In the first step, the image is clustered based on patch structure. An example of such clustering is shown in Fig. 1.5 where the simulated box image is clustered based on whether the patches contain flat regions, horizontal or vertical edges, or corners. In the next step, a dictionary Φ_k is learned for each cluster Ω_k . Finally, the coefficients β_i for each patch in the cluster are estimated. In [24], the principal components of the member patches were used as dictionary atoms for each cluster. The number of components used were adaptively

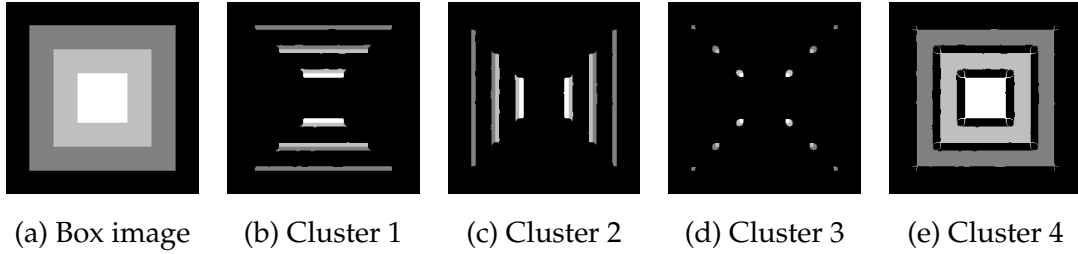


Figure 1.5: Clustering of a simple image based on geometric similarity. Note how pixels in any particular cluster can have quite different intensities but similar geometric structure (edge, corner, flat regions, etc.)

chosen based on the complexity of the image patches within each cluster, taking care to avoid over-fitting. The coefficients were learned in the kernel regression framework of Eq. 1.11 using the LARK weights of [22].

The framework proposed in [24] incorporates flavors of regression as well as dictionary-based approaches. It is easy to see that when Φ_k is restricted to polynomials of a given maximum degree, we obtain the method outlined in [22]. On the other hand, without any clustering, the learned dictionary is global and the framework becomes similar to that of K-SVD [37]. A clustering-based framework employing principal components for dictionary learning was also applied to image coding in [39], and to denoising and other applications in [40]. When the clustering is based on spatial proximity, this framework leads to the localized PCA-based denoising framework of [41, 42] with the latter approach exploiting *photometric* similarity among patches as well. Although the generic framework is similar, the exact processes employed by the methods within each step can be quite different.

The observation of similar patches sharing similar atoms was also recently exploited by Mairal *et al.* [43] to improve the performance of the K-SVD framework. There, the authors explicitly restrict similar patches to share the same support, and the

coefficients β_i for the group are estimated in a joint optimization framework. However, instead of using structural information, *photometric* similarity was used as the grouping criterion. Two noisy patches \mathbf{y}_i and \mathbf{y}_j were considered similar if they satisfied the condition

$$\|\mathbf{y}_i - \mathbf{y}_j\|^2 \leq \gamma^2, \quad (1.14)$$

where γ is a threshold whose value is usually dictated by the strength of the noise corrupting the observed image. Photometric similarity is a more restrictive condition compared to structural similarity as patches need to have similar intensities as well. This can be seen from Fig. 1.5 where geometrically similar patches within each cluster can have different intensities. As a result, a large number of local models were used in [43].

Another model-based approach involving segmentation of the image was proposed in [44], where the authors segment the image into multiple local regions with similar intensities (or color). However, as opposed to the previous models, a parametric polynomial model was used to represent the underlying data in each segment. As with SKR [22], the authors there showed that a higher order (in their case, affine) model retains much more texture than a lower order polynomial (locally constant) fit.

The modeling-based methods discussed so far make use of either parametric or data adaptive models where the parameters are estimated from the noisy image without any prior information. On the other hand, many methods explicitly make use of clean (*noise-free*) images to learn either the model or a prior on the model parameters. Such methods require a separate (offline) training phase. For example, the two color model used for denoising by Joshi *et al.* [45] models the color of each pixel in a neighborhood as a linear combination of two principal colors, where the coefficients

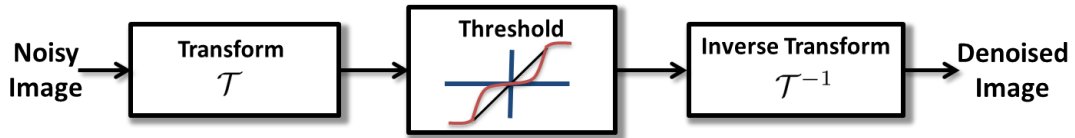


Figure 1.6: Principal operations in shrinkage-based denoising methods, as presented in [48].

are estimated in an optimization framework that makes use of a learned prior. Another such method presented in [46] models the image gradients as a generalized Gaussian distribution. The parameters of the distribution were learned from a database of noise-free images. In [47] Roth and Black proposed the Field of Experts (FoE) model for natural images. There the authors proposed a parametric model for natural images based on Markov random fields (MRF) and sparse coding techniques.

The advantage of *learning* image priors from training samples is that the learned model can be used in many image processing applications, including denoising. However, the training phase is usually time consuming, requiring a large vocabulary of images to ensure substantial variety. Care must also be taken to ensure that images chosen as training samples are not themselves noisy or degraded in any other way.

Until now we have summarized various spatial-domain methods. While such methods have shown promise in suppressing noise in natural images, a large section of denoising literature is devoted to transform-domain methods. In the next section, we discuss some of those approaches.

1.3.2 Transform-domain Methods

The main motivation of denoising in some transform domain is that in the transformed domain it may be possible to separate image and noise components. The basic principle behind most transform-domain denoising methods is *shrinkage* - trun-

cation (hard thresholding) or scaling (soft thresholding) of the transform coefficients to suppress the effects of noise, as shown in Fig. 1.6. For such thresholding, the challenge is to develop a suitable coefficient mapping operation that does not sacrifice the details in the image. The final denoised image is obtained by performing an inverse transform on the *shrunk* coefficients. Apart from the choice of the thresholding operator, the choice of the transform domain is also critical. In the image processing literature, a variety of such transform domains or bases have been proposed. Examples of such bases include two dimensional extensions of the well-studied discrete cosine (DCT) bases used in [49], as well as those developed specifically for image modeling purposes, namely curvelets [50], ridgelets [51], contourlets [52], etc. Of the many transform bases used in literature, the space-frequency localization property of the wavelet domain makes it the most popular choice.

Since the seminal work by Donoho and Johnstone [53], the wavelet basis has been at the core of many transform-domain denoising methods [54–58]. Of these, the denoising method proposed by Portilla *et al.* [59] has shown considerable promise. There the authors proposed a denoising approach based on the scale mixture of Gaussians (GSM) model for the wavelet coefficients [60]. The noisy image is first broken into multiple sub-bands in the wavelet domain, and in each sub-band the wavelet coefficients within a local neighborhood are modeled as a Gaussian scale mixture [61]. A Wiener filter is then used to denoise the wavelet coefficients in a Bayesian least squares framework. The denoised coefficients across sub-bands are then inverse transformed to form the final denoised image. Recently, Lyu and Simoncelli [62] extended this local framework by incorporating a global model for natural images using Gaussian MRFs



Figure 1.7: Some popular benchmark images used for different experiments in this thesis.

to form a Field of GSMs (FoGSM). Such a global model was shown to improve upon the performance of the BLS-GSM method of [59].

The Wiener filter forms the basis of another celebrated denoising method proposed by Dabov *et al.* [49]. There the authors proposed BM3D - a two step denoising method which exploits both spatial and frequency information of an image. The first step involves a shrinkage-based transform domain operation. The initial denoised image is then used as a guide or pilot estimate of the ground-truth for a Wiener filtering operation. What makes this approach unique is that in each step it exploits patch redundancy within the image to improve performance. This is done by first identifying photometrically similar patches in the spatial domain. This group is then used to perform adaptive thresholding in the shrinkage step. This allows them to process the entire group of patches simultaneously. A similar grouping on the pilot estimate is used to perform a transform-domain Wiener filtering. The transform domain of choice for strong noise was the DCT basis, although the wavelet basis was recently shown to

improve performance somewhat [63]. Use of a group of patches to adaptively estimate the threshold and parameters of the Wiener filter lends robustness to the process in presence of strong noise. As such, this hybrid approach showcases the performance benefits of the *non-local* framework, first presented in [12].

In our survey of image denoising literature, we distinguished between the different approaches based on their domain of denoising. However, we point out here that many of these so-called transform-domain denoising methods have equivalent spatial-domain interpretations. Specifically, in [54,64,65], many wavelet domain methods have been shown to have a maximum a posteriori (MAP) based pixel-domain interpretation. A thorough analysis showing such equivalence for a more general class of shrinkage-based estimators was presented in [48]. More recently, Milanfar [21] also cast the hybrid approach of BM3D in a spatial-domain weighted averaging framework. Consequently, such distinction based on the domain of denoising can often be based on how a specific filter is implemented.

1.4 The State-of-the-Art

As discussed in the previous section, a plethora of methods have been proposed with widely varying approaches. In keeping with the traditional *estimation* framework, the performance of these algorithms are quantified using mean squared error (MSE) or peak signal-to-noise ratio ⁴ (PSNR). The MSE of the denoised image is computed as

$$\text{MSE} = E [(z - \hat{z})^2] \approx \frac{1}{M} \sum_{i=1}^M (z_i - \hat{z}_i)^2, \quad (1.15)$$

⁴PSNR, measured in decibels (dB), is related to the MSE as $10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right)$ for images with an intensity range of [0 – 255]. An improvement of 1 dB corresponds to approximately 20% reduction in MSE.

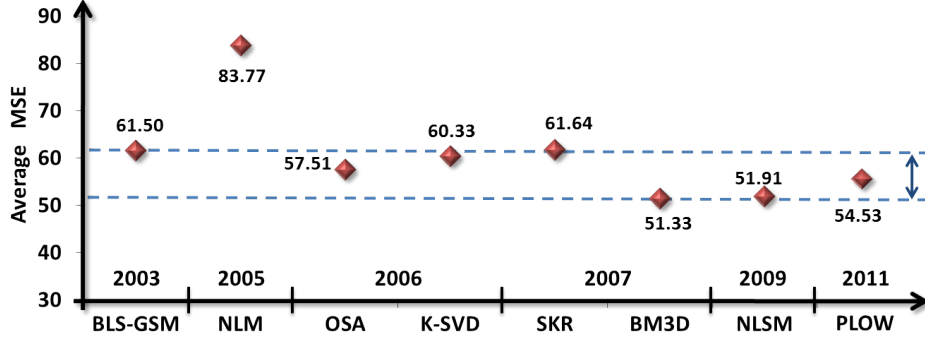


Figure 1.8: State-of-the-art in image denoising: MSEs obtained by some popular methods (BLS-GSM [59], NLM [12], OSA [19], K-SVD [37], SKR [22], BM3D [49], NLSM [43], PLOW [68, 69]) averaged over 4 different images corrupted by WGN of standard deviation 25 show that performance improvement in terms of MSE is not phenomenal. Is denoising dead?

where \hat{z}_i is the estimate of the pixel z_i at location i , and M is the total number of pixels in the image. Although MSE is a useful fidelity measure, it does not always serve as a good indicator of the visual quality of the output image. Consequently, considerable research has been devoted to quantifying perceptual quality, leading to the SSIM measure [66] and the Q -metric [67], to name a few. For evaluation of denoising quality in this thesis, we will use the MSE (PSNR) in conjunction with SSIM [66], Q -metric [67] and visual inspection. However, in keeping with convention, we study the performance of various methods based on output MSE (PSNR).

Since it is impractical to evaluate the vast number of methods addressing the image denoising problem, we restrict ourselves to a few prominent ones proposed over the last decade. The first method that advanced the state-of-the-art considerably can be attributed to BLS-GSM [59] proposed in 2003. In Fig. 1.8 we study the denoising performance of some popular denoising methods proposed since then. The plot there shows the average MSE obtained on 4 benchmark images (house, peppers, Lena and boat images of Fig. 1.7) corrupted with WGN of standard deviation 25. In the subse-

quent years, various other methods (OSA [19], K-SVD [37], SKR [22], to name a few) were proposed. These can be seen to perform comparably to BLS-GSM. Building on the non-local concept of [12], the BM3D [49] method improved upon these to advance the state-of-the-art. More recently, Mairal *et al.* [43] incorporated a non-local formulation within the K-SVD [37] framework to achieve similar denoising performance.

Recently, in [68, 69], we proposed a patch-based spatial-domain method that extends the popular Wiener filter by exploiting patch redundancies. In our method, both geometrically and photometrically similar patches are used to infer different parameters of the locally optimal filter. Our patch-based locally optimal Wiener (PLOW) filtering method achieves denoising performance that is quantitatively and visually on par or even better than the other recently proposed methods. The average MSE achieved over 4 test images are shown in Fig. 1.8. As an added benefit, our method has a sound statistical basis that explains such performance. In [69] and Chapter 5, we present our proposed method in detail, drawing motivation from our study presented in the following chapters of this thesis.

The comparisons in Fig. 1.8 illustrate that the non-local frameworks (OSA, BM3D, NLSM, PLOW) enjoy a distinct advantage over other localized approaches, even though the original method of NLM [12] does not fare well for strong noise. The plot also demonstrates that there has been some performance improvement, in terms of MSE, over the last decade. Surprisingly, however, this improvement, an average gain of 10 in terms of MSE, has not been very significant. Naturally, this raises the question: Is denoising dead? Have we reached the limits of denoising performance? [70]. We explore the answers to these questions in the following chapter.

Summary – In this chapter, we discussed why image denoising is still a relevant problem even though it has been studied quite extensively. We discussed a variety of algorithms that have been proposed in the last decade and pointed out their inherent similarities and dissimilarities. We pointed out some of the measures that are used to evaluate denoising performance and showed that the state-of-the-art has improved over the years, albeit not substantially. This motivates us to analyze the problem of denoising further in an effort to understand if we have reached some theoretical limits. We present our findings in the next chapter.

Chapter 2

Fundamental Limits for Image Denoising

Abstract – In this chapter we study the fundamental limits of denoising any given image. We derive a general expression to lower bound the performance (in terms of MSE) of any patch-based method. Our bounds predict that denoising performance is limited by the noise statistics, the complexity of image patches, as well as the level of patch redundancy that is observed in the given image. We also analyze the MSE bounds formulation in relation to the performance of some well-known estimators.

2.1 Introduction

In the previous chapter, through a simple experiment we demonstrated that many of the recently proposed methods achieve comparable denoising performance, even though the approaches are quite varied. This begs the following questions: Has the problem of denoising already been solved? If not, how much improvement can be expected? In this chapter, we answer these questions through a statistical analysis of the denoising problem. Specifically, we pose the denoising problem as that of esti-

imating the underlying image patches and formulate lower bounds on the MSE for this estimation in a Bayesian Cramér-Rao bound framework [71]. The developed lower bound is independent of any particular denoising method. The bounds framework presented here offer some nice insights that will be exploited later in Chapter 5 where we design a practical denoising method to achieve near-optimal performance.

Although literature on performance limits exists for some of the more complex image processing tasks, very few works address the problem of denoising. One of the earliest such studies was presented by Unser *et al.* [72] where the authors study the best signal-to-noise ratio that can be obtained from multiple noise-corrupted electron micrographs. In [73], Voloshynovskiy *et al.* briefly analyzed the performance of MAP estimators to define performance limits. Another study presented in [74] analyzed the bounds of denoising performance under a simplistic model where any given image is modeled as a union of constant regions separated by sharp edges. These works were, however, limited to analyzing bounds for pixel-wise restoration and do not take into account the patch-based frameworks that the best performing methods employ. Performance limits to object or feature recovery in images in the presence of pointwise degradation was studied by Treibitz *et al.* [75]. There, the effect of noise corruption is studied along with other degradations to formulate optimal filtering parameters that define the resolution limits to recovering any given image feature. Although interesting, their work does not define any fundamental limits to denoising general images. Moreover, none of these works account for the performance improvement from using a good image prior.

Recently, Levin and Nadler [76] introduced a non-parametric method of estimating the (lower and, in certain cases, upper) Bayesian MSE bounds for image de-

noising. In contrast to other analyses discussed so far, they specifically accounted for the effect of prior information. In their work, the prior distribution of image patches was learned from a vast collection of patches from noise-free natural images. However, learning such priors can be cumbersome [76]. While all these works do address the issue of finding limits for denoising performance, they do not readily apply to patch-based non-local methods where patch redundancies within the specific image are exploited to improve performance. As these methods have been shown to achieve the best denoising results (see Fig. 1.8), it is important to account for such advantages. Our bounds formulation [70] takes into account such *non-local* redundancies and are, therefore, developed in a more general setting. To the best of our knowledge, no such study currently exists for the problem of denoising.

We study the performance limits for patch-based approaches as they define the state-of-the-art. The problem of denoising can then be framed as that of estimating the underlying image patches \mathbf{z}_i from their respective noisy observations

$$\mathbf{y}_i = \mathbf{z}_i + \boldsymbol{\eta}_i, \quad \forall i = 1, \dots, M, \quad (2.1)$$

where $\boldsymbol{\eta}_i$ is a noise patch assumed to be independent of \mathbf{z}_i , and M is the number of patches in the image. Typically, such patches are overlapped so as to share pixels between neighbors. This is done to avoid artifacts at the estimated patch boundaries. The pixels in the overlapping regions are thus estimated more than once, and the final estimate is usually obtained from a second averaging process, resulting in further suppression of noise. However, for the purposes of this theoretical study, we will neglect this secondary averaging process and treat the patches as non-overlapping. This simplification allows us to assume that each of the \mathbf{z}_i , $\boldsymbol{\eta}_i$ and, as a result, \mathbf{y}_i patches are independent realizations of random vectors \mathbf{z} , $\boldsymbol{\eta}$ and \mathbf{y} , respectively. For the sake

of clarity of presentation, we assume that the \mathbf{z}_i vectors are all sampled from some unknown probability density function (pdf) $p(\mathbf{z})$. For the purposes of our study, namely the calculation of performance limits, we assume that the noise-free image is available and our aim then is to find out how well, in terms of mean squared error (MSE), the random variable \mathbf{z} can be estimated.

Our purpose in this chapter is to introduce the bounds formulation and identify parameters that need to be estimated from any given image. In the next chapter, we deal with how such parameters can be estimated accurately to compute the bounds. In Sec. 2.2, we show that most denoising methods produce a biased estimate of the \mathbf{z}_i vectors. There we study the bias characteristics of these successful methods and develop a simple but accurate model for the bias. In such a scenario, studying performance limits for unbiased estimators will not provide us with practical bounds on the MSE. Our MSE bounds are therefore developed in Sec. 2.4 through an Optimal Bias Bayesian Cramér-Rao Lower Bound (OB-CRLB) formulation for biased estimators, explained in Sec. 2.3. This requires us to model the bias for denoising methods and such models are structure specific, as will be clear shortly. The lower bound is, thus, initially developed assuming geometric homogeneity among patches in the latent image. Since patches in any given image can exhibit widely varying geometric structures, we extend our lower bound to general images in Sec. 2.5.

2.2 Bias in Denoising

In this section we study the bias in some of the most popular (non-linear) estimators used to date. In estimation theory, it is well known that unbiased estimators do not always exist. Even when they do exist, it is often advantageous to work with

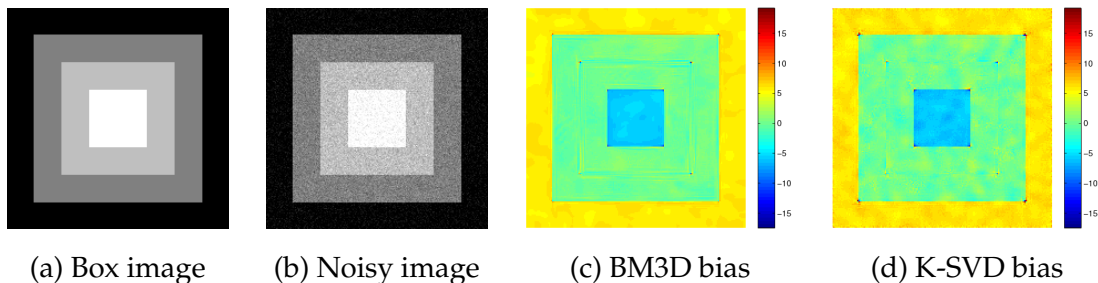


Figure 2.1: Example of bias in denoising produced by some modern denoising methods: (a) Box image, (b) noisy image of standard deviation 15, and bias produced by (a) BM3D [49], and (d) K-SVD [37].

biased estimators as they may result in a lower MSE [77,78]. Moreover, unbiased estimators for a difficult problem such as denoising will tend to have unacceptably large variance, and, therefore, result in visually unpleasant processed images. Hence, bias in high quality image denoising is to be expected. It is for these reasons that we focus our attention on general estimators that may be biased.

The MSE of an estimator is determined by the covariance of the estimate as well as its bias. For any particular patch, the (conditional) bias, defined as¹

$$\mathbf{b}(\mathbf{z}) = E[\hat{\mathbf{z}} - \mathbf{z} | \mathbf{z}], \quad (2.2)$$

is, in general, a function of the parameter to be estimated. This is easily verified through a simple experiment in Fig. 2.1, where we show the bias from denoising a synthetic box image corrupted by WGN of $\sigma = 15$ with two very different approaches (BM3D [49] and K-SVD [37]). Observe that the bias in each case is a function of the underlying image patches. Interestingly, the *structure* of the bias is in keeping with that of the underlying patches. That is to say that, in flat regions, the bias is largely

¹At first glance, the definition of the conditional bias used here may appear to be different from that used in [70] where we defined $\mathbf{b}(\mathbf{z}) = E[\hat{\mathbf{z}}|\mathbf{z}] - \mathbf{z}$. However, Eq. 2.2 is a more general definition of the bias which, for a specific realization of \mathbf{z} , is indeed identical to the expression used in [70]. That is to say, for a particular realization \mathbf{z}_i , $\mathbf{b}(\mathbf{z}_i) = E[\hat{\mathbf{z}}_i - \mathbf{z}_i | \mathbf{z}_i] = E[\hat{\mathbf{z}}_i | \mathbf{z}_i] - \mathbf{z}_i$.

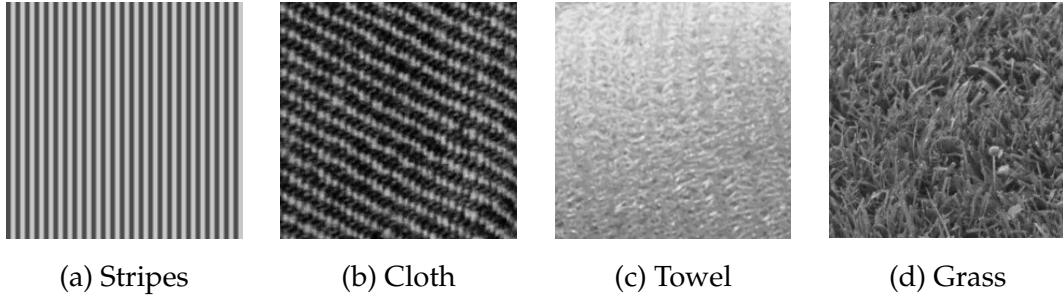


Figure 2.2: Some images consisting of geometrically similar patches that we use for our study.

flat, with discontinuities appearing at the edges and corners. We can therefore study the bias through a localized model, where locality is defined by (not necessarily contiguous) regions of similar geometric structure, as shown in Fig. 2.1. We claim that it is reasonable to approximate this local behavior of the bias function as affine. Namely,

$$\mathbf{b}(\mathbf{z}) = \mathbf{F}\mathbf{z} + \mathbf{u}, \quad (2.3)$$

where the matrix \mathbf{F} and the vector \mathbf{u} are parameters of the affine bias model. Such a model for the bias has been justified and used to study the MSE bound for estimation problems in [79]. In Appendix 2A, we provide further mathematical justification for using such an affine model of the bias.

For general images made up of geometrically non-homogeneous patches, we have to use a different \mathbf{F} and \mathbf{u} for each geometrically similar region or cluster. That is to say, the bias is modeled as a different affine function for each cluster. For clarity of presentation, we will, for now, restrict our study to only consider images containing *geometrically* similar patches, a few (synthetic and natural) examples of which are shown in Fig. 2.2. The developed theory is later generalized to natural images of varied patch structures in Sec. 2.5.

Dealing with patches that are geometrically similar also allows us to consider

\mathbf{z} as a random vector that has a particular (as of yet unknown) pdf $p(\mathbf{z})$ such that the model of Eq. 2.3 holds for every instance of \mathbf{z}_i sampled from this (unknown) distribution. That is to say, for any particular patch within the structurally homogeneous image, the bias model $\mathbf{b}(\mathbf{z}_i) = \mathbf{F}\mathbf{z}_i + \mathbf{u}$ holds. As we will demonstrate, this model, while simple, is reflective of the behavior of essentially all the leading state-of-the-art algorithms. So this provides us a good starting point. In Appendix 2C, we study the case where the bias function is modeled with higher order terms. There, we show that such a generalization makes little difference to our bounds formulation under certain reasonable and physically meaningful assumptions on $p(\mathbf{z})$.

To further substantiate the claim that the bias can be modeled to be approximately affine, we perform experiments where the model parameters (\mathbf{F} and \mathbf{u}) are estimated to fit to the bias from some leading denoising methods. This is done by solving the system of equations obtained using Eq. 2.3 for each of the \mathbf{z}_i vectors. Before describing this experimental demonstration, it is worth noting that our interest here does not lie specifically with the actual values of the bias function for such leading algorithms. Rather, we simply aim to convince the reader that the affine model is a reasonable overall local model for the bias.

As can be expected, different denoising methods will have different bias characteristics (that is, different \mathbf{F} and \mathbf{u}). Fig. 2.3 shows the bias of the denoised intensity estimates obtained using 10 runs of BM3D [49] and K-SVD [37] respectively and illustrates how well the model, learned individually, fits the actual bias. In these experiments, we simulate noisy images by corrupting the 512×512 textured grass image with 10 different realizations of WGN with standard deviation 25. The noisy images are then denoised with each of the methods (using the default parameter settings in

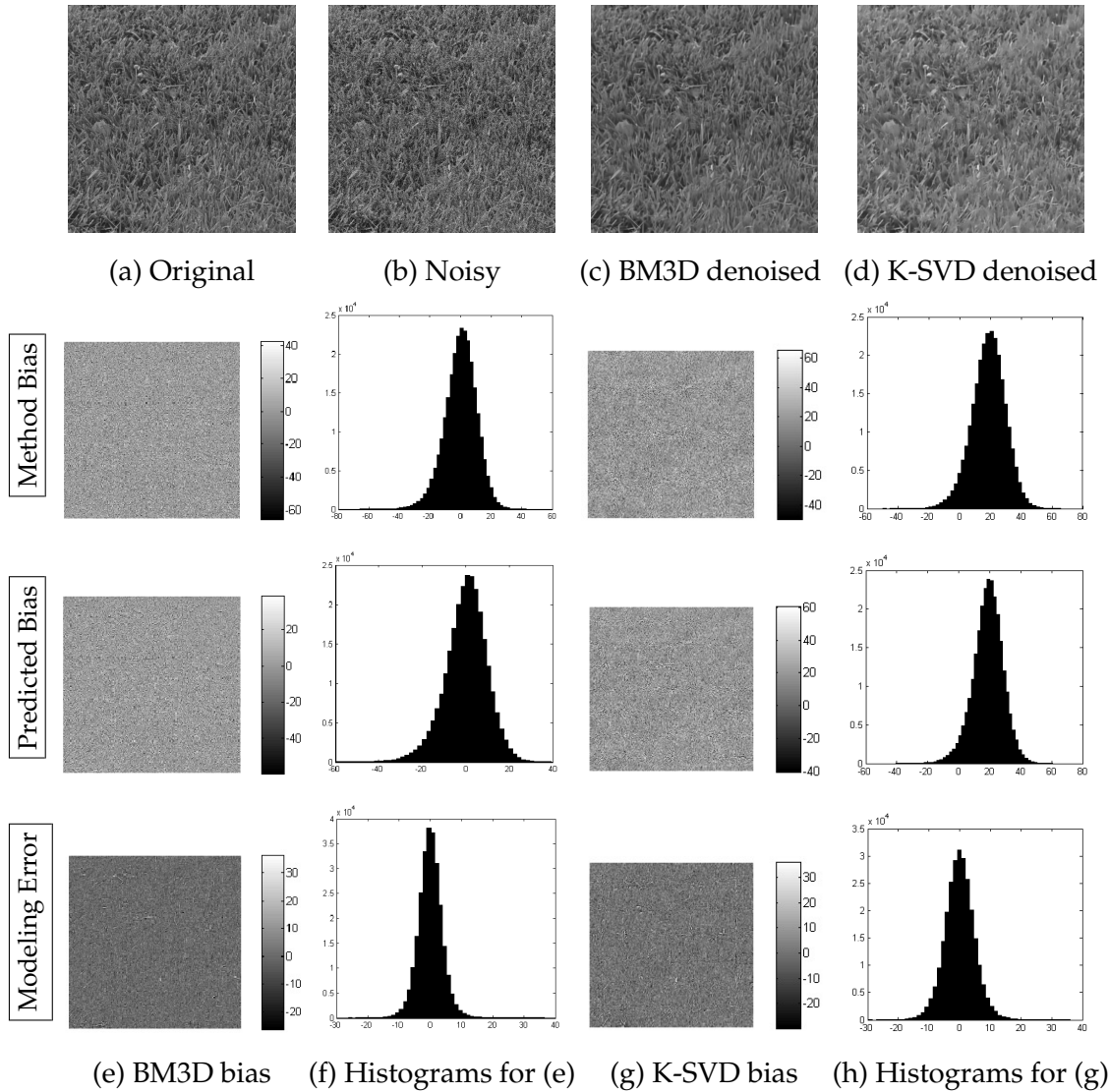


Figure 2.3: Visual comparison of the actual bias obtained from BM3D [49] & K-SVD [37] and reconstructed bias using affine model fit for the 512×512 grass image using 11×11 patches. Note how the histograms for the modeling errors are centered around zero and have short tails.

each case) and the mean denoised image is obtained for each method. From this, the bias vectors $\mathbf{b}(z_i)$ are obtained for each method using non-overlapping 11×11 patches. The bias vectors of all such patches are tiled to form the *method bias* images shown in Fig. 2.3. The bias for each method is then modeled by Eq. 2.3 and the model parameters

($\hat{\mathbf{F}}$ and $\hat{\mathbf{u}}$) are fit using least squares. The predicted bias patches $\hat{\mathbf{b}}(\mathbf{z}_i)$ are then computed for each patch in each case. These vectors are tiled to form the predicted bias images in Fig. 2.3. The difference between the actual and predicted bias is also shown as the error in modeling in each case. For a good fit, the difference between the actual bias and that predicted by the model can be expected to be a random variable sampled from some short tailed distribution centered around zero. This can be qualitatively verified by examining the histogram of the difference.

While the model performs quite well visually, we also present a quantitative measure for the goodness of fit of the model. For the quantitative evaluation, we use the coefficient of determination [80] which can be defined as

$$R^2 = 1 - \frac{\sum_i \|\mathbf{b}(\mathbf{z}_i) - \hat{\mathbf{b}}(\mathbf{z}_i)\|_2^2}{\sum_i \|\mathbf{b}(\mathbf{z}_i) - \bar{\mathbf{b}}(\mathbf{z})\|_2^2}, \quad (2.4)$$

where i indexes all patches in the image, $\mathbf{b}(\mathbf{z}_i)$ is the actual bias of the estimated intensity of the i -th patch, $\bar{\mathbf{b}}(\mathbf{z})$ is the mean bias obtained by the denoising method across all patches in the image and $\hat{\mathbf{b}}(\mathbf{z}_i) = \hat{\mathbf{F}}\mathbf{z}_i + \hat{\mathbf{u}}$ is the predicted bias obtained from the estimated parameters $\hat{\mathbf{F}}$ and $\hat{\mathbf{u}}$ of the affine model. As such, the R^2 value indicates the level of variability in the data that is explained effectively by the regression model. A higher value of R^2 thus indicates a higher level of predictability of the bias by the affine model. In Table 2.1 we obtained high R^2 values for the examples in Fig. 2.3 with various denoising methods [22,24,37,49]. Our experiments with these denoising methods on other images² have yielded comparable results that confirm the goodness of the affine model (Table 2.1).

²For general images, such as the house image (Fig. 1.7), that contain patches of diverse geometric structure, the R^2 values are computed separately on clusters of geometrically similar patches. This will become apparent later in Sec. 2.5 where we discuss the bound calculation process for natural images. The mean R^2 values across 5 such clusters are reported in Table 2.1.

Table 2.1: R^2 values for the affine model fit of the bias produced by different methods for different images.

Image	BM3D [49]	K-SVD [37]	SKR [22]	K-LLD [24]
Grass	0.863	0.801	0.833	0.810
Towel	0.913	0.928	0.864	0.880
House	0.916	0.955	0.959	0.963

To provide further empirical evidence that the affine model is a good fit for the bias and that it holds true only when the patches considered have roughly similar geometric structure, we performed experiments with general images such as those shown in Fig. 1.7, where we randomly selected patches from a given image and tried to model the bias for such patches by estimating a single set of parameters (\mathbf{F} and \mathbf{u}). For such images, we consistently obtained much lower values ($R^2 < 0.6$) for the goodness of fit. However, when only patches of similar structure were considered for the same images, the R^2 values for the fit were considerably higher (Table 2.1). These experiments indicate that the affine model is a good *local* fit, where locality is defined by similarity in patch geometry.

However, the question of higher order models still remains. For the sake of completeness, we refer the interested reader to Appendix 2C where we show that the MSE bounds formulation for a more sophisticated (higher order) bias model remains unchanged from the affine case under certain symmetry constraints on the density $p(\mathbf{z})$. In the remainder of this chapter, we will assume an affine model for the bias to derive the theoretical performance limits of denoising.

2.3 Optimal Bias Bayesian Cramér-Rao Lower Bound

For the purposes of deriving bounds on denoising performance, we pose the problem of denoising as that of estimating a multivariate random vector. In the statistics and signal processing literature, a number of bounds exist to evaluate performance limits of such estimation. While some bounds were developed for the estimation of a deterministic parameter (for instance, those proposed by Seidman [81], Cramér [82] and Rao [83,84]), others, such as the Ziv-Zakai bound [85], address the Bayesian setting where the parameter of interest is a random variable. One primary difference between the two cases lies in the meaning of MSE for which the lower bound is established. In the deterministic case, the bound is a function of the parameter of interest, whereas in the Bayesian case it is a numerical value obtained by integrating over the random parameter [78] (\mathbf{z} in our case). As a result, Bayesian versions have been derived for many of the bounds developed for the deterministic case [86]. In our work, we build on a Bayesian version of the classical Cramér-Rao lower bound (CRLB) [71].

In its simplest form, the CRLB is a lower bound on the variance of any unbiased estimator of \mathbf{z} , subject to the regularity condition

$$E \left[\frac{\partial \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}} \right] = \mathbf{0}, \quad \forall \mathbf{z} \quad (2.5)$$

on the conditional probability density function $p(\mathbf{y}|\mathbf{z})$. Here, we assume that $\ln p(\mathbf{y}|\mathbf{z})$ is twice differentiable with respect to \mathbf{z} . An important point to note here is that our CRLB formulation differs from that defined by van Trees [86, 87] where the joint pdf $p(\mathbf{y}, \mathbf{z})$ is directly used. The two pdf's are related by

$$p(\mathbf{y}, \mathbf{z}) = p(\mathbf{y}|\mathbf{z})p(\mathbf{z}), \quad (2.6)$$

where $p(\mathbf{z})$ is the probability density function on \mathbf{z} . We work with the conditional pdf

$p(\mathbf{y}|\mathbf{z})$ to formulate a bound on the MSE in the conditional sense and integrate it to get the overall (Bayesian) MSE, as we illustrate below in (2.11). Assuming, for now, that an unbiased estimator $\hat{\mathbf{z}}$ exists, the bound on the (conditional) covariance $\mathbf{C}_{\hat{\mathbf{z}}|\mathbf{z}}$ of the estimate is given by the CRLB as

$$\mathbf{C}_{\hat{\mathbf{z}}|\mathbf{z}} = E \left[(\hat{\mathbf{z}} - E[\hat{\mathbf{z}}|\mathbf{z}]) (\hat{\mathbf{z}} - E[\hat{\mathbf{z}}|\mathbf{z}])^T | \mathbf{z} \right] \geq \mathbf{J}^{-1}, \quad (2.7)$$

where the operator \geq in the matrix case implies that the difference of the two matrices has to be positive semi-definite. Here \mathbf{J} is the conditional Fisher information matrix (FIM) given by

$$\mathbf{J} = -E \left[\frac{\partial^2 \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z} \partial \mathbf{z}^T} \right]. \quad (2.8)$$

The estimator which achieves this lower bound is said to be efficient. While this provides us with a simple method for evaluating performance limits for an estimation problem, it cannot be applied directly to our denoising problem. As illustrated previously, most denoising methods are biased in nature and this bias needs to be taken into account to obtain a useful lower bound. For such cases, the CRLB on the covariance of the biased estimate $\hat{\mathbf{z}}$ is given by

$$\mathbf{C}_{\hat{\mathbf{z}}|\mathbf{z}} \geq \left(\frac{\partial E[\hat{\mathbf{z}}|\mathbf{z}]}{\partial \mathbf{z}} \right) \mathbf{J}^{-1} \left(\frac{\partial E[\hat{\mathbf{z}}|\mathbf{z}]}{\partial \mathbf{z}} \right)^T = (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T, \quad (2.9)$$

where \mathbf{I} denotes the identity matrix and (2.9) follows from making use of our affine bias model of Eq. 2.3. It is useful to note here that the estimator covariance for the affine model is only influenced by the parameter \mathbf{F} (which can also be interpreted as the gradient of the bias) and not by the constant term \mathbf{u} . As such, a negative definite gradient on the bias lowers the minimum achievable estimator variance compared to that of the unbiased case given by (2.7). Placing constraints on the bias gradient, Fessler *et al.* [88]

used this property to explore the performance limits for image restoration problems in a biased CRLB setting.

Using the relation in (2.9), we can calculate a lower bound on the conditional MSE in estimating \mathbf{z} as

$$E [\|\mathbf{z} - \hat{\mathbf{z}}\|^2 | \mathbf{z}] = \text{Tr} [\mathbf{C}_{\hat{\mathbf{z}}|\mathbf{z}}] + \|\mathbf{b}(\mathbf{z})\|^2 \geq \text{Tr} [(\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T] + \|\mathbf{b}(\mathbf{z})\|^2, \quad (2.10)$$

where $\text{Tr}[\cdot]$ denotes the trace of a matrix. Now, by the law of total expectation, the overall Bayesian MSE can be expressed as

$$\begin{aligned} E [\|\mathbf{z} - \hat{\mathbf{z}}\|^2] &= \int_{\mathbf{z}} E [\|\mathbf{z} - \hat{\mathbf{z}}\|^2 | \mathbf{z}] p(\mathbf{z}) d\mathbf{z} \\ &\geq \underbrace{\int_{\mathbf{z}} \left[\text{Tr} \left\{ (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T \right\} + (\mathbf{F}\mathbf{z} + \mathbf{u})^T (\mathbf{F}\mathbf{z} + \mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z}}_Q. \end{aligned} \quad (2.11)$$

It is interesting to note that in the above formulation the pdf $p(\mathbf{z})$ can be thought of as the prior information on \mathbf{z} . Most denoising methods make use of informative priors in the form of smoothness or sparsity penalties and other constraints to achieve improved performance. Our Bayesian approach thus takes into account the effect of such priors in calculating the lower bound on the MSE. When $p(\mathbf{z})$ is known a priori, the expression in (2.11) can be used directly to evaluate the Bayesian MSE bound for estimating \mathbf{z} , as was done by Young *et al.* [89]. The scalar parameter of interest there was assumed to lie within a known interval. Their results were later generalized for an unconstrained vector case by Ben-Haim *et al.* in [90].

It would appear that the effective calculation of the above Bayesian bound necessitates the complete knowledge of the prior density $p(\mathbf{z})$, as is the case for [89, 90]. This is related to the subject of statistical modeling of images, which has seen much activity [39, 62, 91–95] and is still the subject of some controversy. Levin and

Nadler [76] took an alternate non-parametric approach where they *learn* the prior (pdf) from a vocabulary of image patches. However, such an approach is cumbersome as the database needs to contain sufficient variety to cover the gamut of natural patches. Moreover, as with any such prior learning technique, care must be taken to ensure that patches used are themselves free of degradations from noise and blur. Happily, as described in Sec. 2.4.1 below, we are able to avoid the need for complete knowledge of such priors. More specifically, only a few low order moments of the density $p(\mathbf{z})$ are needed for our calculations, and, as we will show, these can be effectively estimated directly from a given (noise-free) image.

An important point to note is that the bound formulation of (2.11) is related to those used in [89,90] but differs from the Bayesian CRLB (B-CRLB) of van Trees [86,87], as alluded to earlier. The FIM used in the B-CRLB formulation there is calculated from the joint pdf $p(\mathbf{y}, \mathbf{z})$ whereas in our case (and also [89, 90]) it is calculated from the conditional pdf $p(\mathbf{y}|\mathbf{z})$. Hence, the B-CRLB of [87] is more restrictive in the sense that $p(\mathbf{y}, \mathbf{z})$ has to be twice differentiable. In our case, twice differentiability is necessary only for the conditional pdf. To disambiguate the two, we refer to our formulation as the Optimal Bias B-CRLB (OB-CRLB). We calculate the lower bound on the MSE based on the OB-CRLB formulation in the next section.

2.4 Lower Bound on the MSE

In this section, we derive the bound using expressions for the bias model parameters (\mathbf{F} and \mathbf{u}) that minimize Q in (2.11). We also derive an analytical expression for the FIM and discuss how we estimate the covariance of image patches that is needed to derive the MSE bound.

2.4.1 Deriving the Bayesian MSE bound

The MSE of any estimator is a function that depends on the variance as well as the bias term. To obtain the lower bound on the MSE, we thus need to establish optimal values for \mathbf{F} and \mathbf{u} that minimize (2.11). This is in line with the approach advocated in [90]. We can thus obtain the optimal \mathbf{F} and \mathbf{u} (denoted as \mathbf{F}^* and \mathbf{u}^* respectively) by solving the optimization problem

$$\{\mathbf{F}^*, \mathbf{u}^*\} = \arg \min_{\{\mathbf{F}, \mathbf{u}\}} \int_{\mathbf{z}} \left[\text{Tr} \{ (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T \} + (\mathbf{F}\mathbf{z} + \mathbf{u})^T (\mathbf{F}\mathbf{z} + \mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z}. \quad (2.12)$$

The optimum parameters \mathbf{F}^* and \mathbf{u}^* can be obtained by differentiating Q (defined in (2.11)) with respect to \mathbf{F} and \mathbf{u} and solving the simultaneous system of equations

$$\frac{\partial Q}{\partial \mathbf{u}} = \mathbf{0}, \quad \frac{\partial Q}{\partial \mathbf{F}} = \mathbf{0}. \quad (2.13)$$

Solving these simultaneous equations results in the following expressions for the optimal parameters

$$\mathbf{F}^* = -\mathbf{J}^{-1} [\mathbf{J}^{-1} + \mathbf{C}_{\mathbf{z}}]^{-1}, \quad (2.14)$$

$$\mathbf{u}^* = -\mathbf{F}^* E[\mathbf{z}] = \mathbf{J}^{-1} [\mathbf{J}^{-1} + \mathbf{C}_{\mathbf{z}}]^{-1} E[\mathbf{z}]. \quad (2.15)$$

The derivations are detailed in Appendix 2B. It is important to note that the covariance is not of any estimated \mathbf{z} vectors but the second moment from the pdf $p(\mathbf{z})$ of the random vector \mathbf{z} . Thus, we are able to obtain expressions for \mathbf{F}^* and \mathbf{u}^* that result in the theoretical lower bound on the MSE for any affine-biased denoiser³. Note that it is not necessary that any denoiser with the said bias and variance characteristics actually exist. That is to say, no ‘‘Bayes-efficient’’ estimator that achieves this derived lower

³It is interesting to note that this optimization indeed yields a negative definite \mathbf{F}^* (see Eq. 2.14).

bound may actually exist. Next, we obtain an expression for the lower bound on the MSE using the optimized parameters for our bias model by inserting \mathbf{F}^* and \mathbf{u}^* in the expression for Q (Eq. 2.12).

Once we obtain expressions for the FIM and the parameters for the affine model of the optimal bias function, we can proceed to find an expression for the optimal lower bound on the MSE. We rewrite the right hand side of (2.11) by plugging in the obtained expressions of the parameters from Equations 2.14 and 2.15 as

$$\begin{aligned}
Q_{\min} &= \int_{\mathbf{z}} \left[\text{Tr} \{ (\mathbf{I} + \mathbf{F}^*) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F}^*)^T \} + (\mathbf{F}^* \mathbf{z} + \mathbf{u}^*)^T (\mathbf{F}^* \mathbf{z} + \mathbf{u}^*) \right] p(\mathbf{z}) d\mathbf{z} \\
&= \text{Tr} \{ (\mathbf{I} + \mathbf{F}^*) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F}^*)^T \} + \int_{\mathbf{z}} (\mathbf{z} - E[\mathbf{z}])^T \mathbf{F}^{*T} \mathbf{F}^* (\mathbf{z} - E[\mathbf{z}]) p(\mathbf{z}) d\mathbf{z} \\
&= \text{Tr} \{ (\mathbf{I} + \mathbf{F}^*) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F}^*)^T \} + E \left[(\mathbf{z} - E[\mathbf{z}])^T \mathbf{F}^{*T} \mathbf{F}^* (\mathbf{z} - E[\mathbf{z}]) \right] \\
&= \text{Tr} \{ (\mathbf{I} + \mathbf{F}^*) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F}^*)^T \} + E \left[\text{Tr} \left\{ \mathbf{F}^* (\mathbf{z} - E[\mathbf{z}]) (\mathbf{z} - E[\mathbf{z}])^T \mathbf{F}^{*T} \right\} \right] \\
&= \text{Tr} \{ (\mathbf{I} + \mathbf{F}^*) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F}^*)^T \} + \text{Tr} \left\{ E \left[\mathbf{F}^* (\mathbf{z} - E[\mathbf{z}]) (\mathbf{z} - E[\mathbf{z}])^T \mathbf{F}^{*T} \right] \right\} \\
&= \text{Tr} \{ (\mathbf{I} + \mathbf{F}^*) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F}^*)^T \} + \text{Tr} \left\{ \mathbf{F}^* E \left[(\mathbf{z} - E[\mathbf{z}]) (\mathbf{z} - E[\mathbf{z}])^T \right] \mathbf{F}^{*T} \right\} \\
&= \text{Tr} \left\{ \mathbf{F}^* \mathbf{J}^{-1} \mathbf{F}^{*T} + 2\mathbf{F}^* \mathbf{J}^{-1} + \mathbf{J}^{-1} + \mathbf{F}^* \mathbf{C}_{\mathbf{z}} \mathbf{F}^{*T} \right\} \\
&= \text{Tr} \left\{ \mathbf{F}^* (\mathbf{J}^{-1} + \mathbf{C}_{\mathbf{z}}) \mathbf{F}^{*T} + 2\mathbf{F}^* \mathbf{J}^{-1} + \mathbf{J}^{-1} \right\} \\
&= \text{Tr} \left\{ \mathbf{J}^{-1} (\mathbf{J}^{-1} + \mathbf{C}_{\mathbf{z}})^{-1} \mathbf{J}^{-1} - 2\mathbf{J}^{-1} (\mathbf{J}^{-1} + \mathbf{C}_{\mathbf{z}})^{-1} \mathbf{J}^{-1} + \mathbf{J}^{-1} \right\} \\
&= \text{Tr} \left\{ \mathbf{J}^{-1} - \mathbf{J}^{-1} (\mathbf{J}^{-1} + \mathbf{C}_{\mathbf{z}})^{-1} \mathbf{J}^{-1} \right\} \tag{2.16} \\
&= \text{Tr} \left\{ (\mathbf{J} + \mathbf{C}_{\mathbf{z}}^{-1})^{-1} \right\}, \tag{2.17}
\end{aligned}$$

where the last equality is derived from the matrix inversion lemma [96]. Note that Eq. 2.17 can be computed only when $\mathbf{C}_{\mathbf{z}}$ is well-conditioned. For smoother patches, this may not be the case. In practice, we use the expression in Eq. 2.16 to compute the bounds. Eq. 2.17, however, allows us to get a neat expression for the lower bound and

provides useful insights that we discuss later in this section and in Chapters 4 and 5. For its simplicity, we will denote the bound on the MSE for the denoising problem by the expression in Eq. 2.17 as

$$E [\|\mathbf{z} - \hat{\mathbf{z}}\|^2] \geq \text{Tr} \left[(\mathbf{J} + \mathbf{C}_{\mathbf{z}}^{-1})^{-1} \right]. \quad (2.18)$$

It is interesting to analyze the implications of the obtained expression. This lower bound is a function of both the FIM \mathbf{J} and the covariance of the parameter vector \mathbf{z} . Within a cluster of geometrically similar patches, the covariance of \mathbf{z} is an indication of the variability of the geometric structures encountered in the image. For images that are mostly smooth, we can expect \mathbf{z} to have a smaller variance whereas images containing more geometric variability will yield larger $\mathbf{C}_{\mathbf{z}}$. This is also in keeping with our expectations and experimental findings that smooth images lacking much detail are easier to denoise than those containing much texture.

Our bounds are derived assuming an affine-biased estimator. One type of estimator having this bias is an affine estimator which, in the case of Gaussian noise, can be shown to be the only class of estimators having an affine bias function [79]. Moreover, the expression for the lower bound is precisely that of the linear minimum mean square error (LMMSE) estimate for the problem [71]. In theory, this bound is achievable by an affine estimator with exact knowledge of the first and second order moments of $p(\mathbf{z})$. Later, in Chapter 5, we use this insight to design a practical denoising method aimed at achieving the lower bounds. However, as the moments can only be *estimated* from the given noisy image, loss of efficiency is observed.

Interestingly, the expression for the lower bound corresponds to the MSE of the Bayesian minimum mean square error (BMMSE) estimate of \mathbf{z} when the prior pdf $p(\mathbf{z})$ is assumed to be Gaussian [71]. We, of course, make no such assumption on the

prior. Moreover, the bounds formulation does not even assume complete knowledge of the entire distribution of \mathbf{z} , unlike the Bayesian MSE bound derived by Ben-Haim *et al.* [90]. Our affine model of the bias allows us to assume only the availability of the first and second order moments of \mathbf{z} for the computation of the lower bound. In Appendix 2A, we show that the bias of any patch-based weighted averaging filter is necessarily an affine function of the image patches. Moreover, our experiments in Sec. 2.2 show that the affine-bias model is quite general, and provides a good fit for the bias of more complicated denoising methods (namely, K-SVD [37] and BM3D [49]) as well. Therefore, our performance analysis of affine-biased patch-based methods is quite general.

Of course, we cannot altogether rule out the possibility of an optimal denoising method for which the affine-bias model may be inadequate. Extending our approach to the case where the bias is higher order will incorporate correspondingly higher order moments of the distribution of \mathbf{z} , as shown in Appendix 2C. However, for natural images, the higher order moments such as skewness are typically small [97] and, therefore, have less significant effects on the denoising bounds. As a result, in spite of assuming a lower order bias model, the MSE bounds expression in (2.18) depending on only the first and second order moments applies to a broad class of patch-based denoising methods. While the moments of \mathbf{z} are dependent on image patches, the noise statistics dictates the analytical expression for the FIM, which we derive next.

2.4.2 Fisher Information Matrix

The expression for the MSE bound in (2.18) holds true for any noise distribution. Noise statistics influence denoising performance. Its effect on the bound is

captured by the FIM \mathbf{J} which takes into account the noise pdf. Hence, our framework can be used to derive bounds for any noise distribution. In this thesis, however, we only consider the case of additive white Gaussian noise⁴ (WGN). Although we assume the noise to be iid pointwise, this does not allow us to immediately claim statistical independence of all the noise *patches* across the entire image. In fact, if the patches are allowed to overlap, data from one patch may be duplicated in neighboring patches. To make our derivation of the FIM simple, we will assume the image patches to be non-overlapping. This allows us to assert that the $\boldsymbol{\eta}_i$ noise patches are mutually independent. Since the corrupting noise patches of size $\sqrt{n} \times \sqrt{n}$ are sampled from a multivariate Gaussian, we can write the pdf as

$$p(\mathbf{y}|\mathbf{z}) = \frac{1}{(\sqrt{2\pi}\sigma^n)^M} \exp \left\{ \sum_{j=1}^M \frac{-\|\mathbf{y}_j - \mathbf{z}_j\|^2}{2\sigma^2} \right\}, \quad (2.19)$$

where M is the total number of (non-overlapping) patches.

As explained earlier, \mathbf{z} is a random variable and \mathbf{z}_i vectors are instances of the variable sampled from a certain (unknown) distribution. In the denoising problem, one is required to estimate each of the \mathbf{z}_i instances in an image and, hence, the FIM is calculated on a per patch basis. Many denoising algorithms [12, 20, 49, 69] infer information about a single patch by taking into account multiple *photometrically similar*⁵ patches that exhibit similar pixel intensities in addition to geometric structure. Such algorithms, in essence, estimate each \mathbf{z}_i vector from multiple photometrically similar noisy \mathbf{y}_j vectors, assuming each of these \mathbf{y}_j vectors to be a different noisy observation

⁴WGN assumption will be used in different aspects of our study in Chapters 3 and 4. In Chapter 5, we develop a practical denoising method motivated by the bounds framework presented here. Although designed specifically for WGN, we will show that the method is quite effective in removing noise from images where the noise characteristics are unknown.

⁵We use the notation $\mathbf{z}_i \approx \mathbf{z}_j$ to denote the photometric similarity between two patches \mathbf{z}_i and \mathbf{z}_j . Later in this chapter, we provide a more formal definition for such similarity (2.22), and also describe how to identify similar patches to compute denoising bounds.

of \mathbf{z}_i . The dissimilarities between \mathbf{y}_j (\mathbf{y}_i included) and \mathbf{z}_i are then mainly due to noise.

In such a scenario, we obtain an expression for the FIM as

$$\frac{\partial \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}_i} = \sum_j \frac{(\mathbf{y}_j - \mathbf{z}_j)}{\sigma^2} \quad \text{where } \mathbf{z}_j \approx \mathbf{z}_i \quad (2.20)$$

$$\Rightarrow \mathbf{J} = -E \left[\frac{\partial^2 \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}_i \partial \mathbf{z}_i^T} \right] = N \frac{\mathbf{I}}{\sigma^2}, \quad (2.21)$$

assuming that N similar patches are taken into account in denoising any given patch. Note that Eq. 2.21 is only an approximate expression for the FIM. The FIM takes this exact form only when N *identical* patches are considered. It is also important to reiterate that Eq. 2.21 holds only when we assume that the patches are non-overlapping. In the case where the image patches are considered to be overlapping, the calculation of the FIM becomes more complicated and the issue of it being singular arises. In this paper, we only deal with the non-overlapping case where the noise patches can be considered to be iid.

The expression for the FIM and, hence, the bound in (2.18) takes into account the strength of the noise as well as the number of photometrically similar patches (N) that are considered in denoising any given patch. In general, the level of such redundancy will vary widely from image to image, and also from patch to patch within the same image. For example, the corner regions of the box image (Fig. 2.1(a)) have fewer matching patches than the smoother regions. Consequently, using a fixed value of N for the entire image is unwise. This value thus needs to be calculated patch-wise by identifying the number of patches that are photometrically similar to each reference patch in the image.

To determine the level of photometric redundancy, we first need to define a measure of similarity between two patches. We consider two patches \mathbf{z}_i and \mathbf{z}_j to be

similar if they can be expressed as

$$\mathbf{z}_j = \mathbf{z}_i + \boldsymbol{\epsilon}_{ij} \quad \text{such that } \|\boldsymbol{\epsilon}_{ij}\|^2 \leq \gamma^2, \quad (2.22)$$

where γ is a small threshold. Later, in Chapter 3, we detail how such a threshold is chosen so as to ensure few false positives and negatives. Denoting the number of patches similar to a reference \mathbf{z}_i as N_i , we obtain a patch-wise FIM

$$\mathbf{J}_i = N_i \frac{\mathbf{I}}{\sigma^2}, \quad (2.23)$$

where N_i photometrically similar patches are taken into account in denoising a noisy patch \mathbf{y}_i . The MSE bound can then be calculated with a corresponding FIM for each patch, and the MSE bound for the entire set of M structurally similar patches can be calculated as the aggregate of the patch-wise MSE bounds as

$$E [\|\mathbf{z} - \hat{\mathbf{z}}\|^2] \geq \frac{1}{M} \sum_{i=1}^M \text{Tr} \left[(\mathbf{J}_i + \mathbf{C}_z^{-1})^{-1} \right]. \quad (2.24)$$

Although the FIM is derived for non-overlapping patches, to be more realistic, we consider overlapping patches in our calculation of N_i . This leads to a practical estimate of the number of patches that is available to any denoising algorithm. Fig. 2.4 shows the spatial distribution of N_i values for the house and Barbara images (shown in Fig. 1.7) calculated with 11×11 patches and a suitable γ threshold (discussed in Chapter 3). As can be expected, N_i takes much larger values for the smoother regions than the edge and textured regions.

Until now, we have assumed that the image patches we deal with are geometrically similar across the entire image (that is, samples from a single $p(\mathbf{z})$), although the patch intensities may differ. This was necessary only for the sake of clarity of the presentation. In the next section, we extend our bounds framework to general images that usually contain varied patch structures.

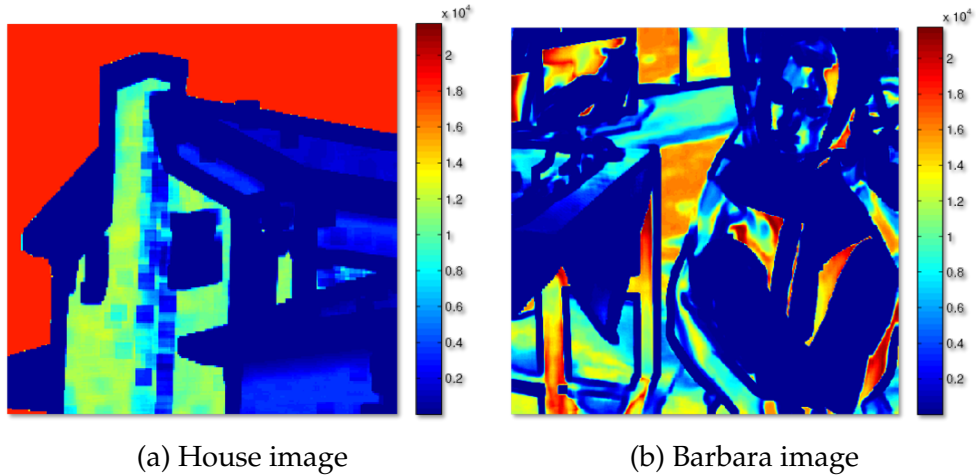


Figure 2.4: The spatial distribution of N_i values for a patch size of 11×11 on (a) house image, and (b) Barbara image, shown in Fig. 1.7.

2.5 Bounds for General Images

Although natural images do not exhibit uniform structural composition, the diversity of geometric structure of patches within an image is usually limited. To make our formulation applicable to such geometrically heterogeneous images (Fig. 1.7), we can group together patches of similar patterns and the analysis of denoising performance can be considered independently for each such segment. The performance limits on denoising a particular image can then be calculated as an aggregate of the MSE bounds for each of the clusters.

The first step in computing the denoising bounds is thus a clustering mechanism that groups patches based on their geometric similarities. Fig. 1.5 illustrates an example of such clustering for a synthetic box image consisting mainly of smooth regions, separated by horizontal and vertical edges and corners. In Chapter 3, we give a brief description of our automatic “geometric” clustering first developed for denoising in [24]. In the present scenario, however, we are chiefly interested in extending



Figure 2.5: Clustering results by K-Means algorithm on the box image. Notice how edges and patterns of a certain kind are clustered together even though the patches may have different intensities.

our bounds framework to general images. Therefore, we will assume an ideal “oracle” clustering method (which may be user-defined) to characterize the various clusters in a given image. Assuming availability of such a clustering, we proceed to calculate the MSE bound for any general image that is composed of K such (not necessarily contiguous) clusters.

Clustering the given image decomposes it into multiple segments such that patches of similar geometric structure are captured in the same cluster (see Fig. 2.5.) In such a case, we can assume that the \mathbf{z}_i vectors corresponding to patches belonging to a particular cluster (say Ω_k) are realizations of a random vector sampled from an unknown pdf $p_k(\mathbf{z})$. This allows us to model the bias to be an affine function of \mathbf{z} in each cluster resulting in cluster-wise optimal bias model parameters \mathbf{F}_k^* and \mathbf{u}_k^* . Consequently, the bounds formulation of (2.24) holds separately for each cluster. The final bound on the MSE for the entire image can then be obtained as a weighted average of the bounds for each cluster. Mathematically, this can be derived by observing the sum of squared error (SSE) for the entire image:

$$\begin{aligned}
 \text{SSE} &= \sum_{k=1}^K \text{SSE}_k = \sum_{k=1}^K M_k Q_k \\
 \Rightarrow Q &= \frac{1}{M} \text{SSE} = \sum_{k=1}^K \frac{M_k}{M} Q_k = \sum_{k=1}^K \alpha_k Q_k,
 \end{aligned} \tag{2.25}$$

where \hat{Q} is the estimate of the bound for the entire image, \hat{Q}_k and SSE_k are the estimates of the bounds on the MSE and the SSE respectively for the k -th cluster, M_k denotes the cardinality of the set Ω_k and Ω denotes the set of all patches in the image, and α_k is the weight corresponding to the k -th cluster in the averaging process.

We now have an expression for the MSE bound from an independent analysis of each cluster. Referring to our discussions on the achievability of the bound in Sec. 2.4.1, we expect the bound to be theoretically achievable by a linear MMSE estimator in each cluster. Later in Chapter 5, we will use this as the motivation for our patch-based denoising framework. However, in addition to perfect knowledge of the first and second order moments of $p_k(\mathbf{z})$, such an estimator now has to have access to “perfect” clustering as well. Moreover, the bounds formulation takes into account the effect of photometric redundancies. In practice, identifying such similar patches can be non-trivial, especially under strong noise. Due to all these nuances one can only hope to come up with an affine estimator with performance close to the bound, as will be demonstrated in Chapter 5. Thus, our formulation presents a valid lower bound.

The focus of our work in this chapter was restricted to developing an expression for the fundamental limits for denoising any given image. The parameters of this lower bound, however, need to be *estimated* from the image itself. In the next chapter we describe methods for estimating the different parameters, and hence, the bounds.

Summary – In this chapter, we developed a formulation to lower bound the performance of any patch-based denoising method. We showed that the bounds can be calculated even without knowledge of the exact distribution of the underlying patches.

The resultant bounds depend on the noise statistics, complexity of image patches to be estimated, as well as the number of photometrically similar patches that can be exploited to perform denoising. In the next chapter, we discuss how these parameters of the bounds can be estimated to compute bounds for denoising any given image.

2A Mathematical Justification for Affine Bias

Most current state-of-the-art methods perform denoising of any given patch \mathbf{y}_i by searching for similar patches \mathbf{y}_j in the noisy image. Here, we show that such class of non-linear denoising methods produce biased estimates and that the bias for such methods can be shown to be an affine function of the underlying patch \mathbf{z}_i . In this derivation we assume that for two patches $\mathbf{y}_i, \mathbf{y}_j$ to be similar, their noise-free versions will have to be similar and can be written as

$$\mathbf{z}_j = \mathbf{z}_i + \boldsymbol{\epsilon}_j \quad \text{such that } \|\boldsymbol{\epsilon}_j\|^2 \leq \gamma^2, \quad (2.26)$$

where γ is some small threshold and $\boldsymbol{\epsilon}_j$ is a vector. The denoised estimate $\hat{\mathbf{z}}_i$ of the patch \mathbf{z}_i is obtained by performing a weighted averaging over all (say N) such similar noisy patches. In general, this can be written as

$$\hat{\mathbf{z}}_i = \sum_{j=1}^N \mathbf{W}_{ij} \mathbf{y}_j, \quad (2.27)$$

where \mathbf{W}_{ij} is a (data-dependent) weight matrix that measures the similarity between patches \mathbf{y}_i and \mathbf{y}_j . Using the data model of Eq. 2.1, and Eq. 2.26 above, we can express Eq. 2.27 as

$$\hat{\mathbf{z}}_i = \sum_j \mathbf{W}_{ij} \mathbf{y}_j = \sum_j \mathbf{W}_{ij} (\mathbf{z}_j + \boldsymbol{\eta}_j) = \sum_j \mathbf{W}_{ij} (\mathbf{z}_i + \boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j). \quad (2.28)$$

The expected value of this estimate can then be written as

$$E[\hat{\mathbf{z}}_i | \mathbf{z}_i] = E \left[\sum_j \mathbf{W}_{ij} (\mathbf{z}_i + \boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j) \right] = E \left[\sum_j \mathbf{W}_{ij} \right] \mathbf{z}_i + \sum_j E [\mathbf{W}_{ij} (\boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j)]. \quad (2.29)$$

For a specific patch \mathbf{z}_i , the above expression allows us to calculate the (conditional) bias for such non-linear weighted averaging methods as

$$\begin{aligned} \mathbf{b}(\mathbf{z}_i) &= E[\hat{\mathbf{z}}_i - \mathbf{z}_i | \mathbf{z}_i] \\ &= \left(E \left[\sum_j \mathbf{W}_{ij} \right] - \mathbf{I} \right) \mathbf{z}_i + \sum_j E [\mathbf{W}_{ij} (\boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j)] = \mathbf{F}_i \mathbf{z}_i + \mathbf{u}_i, \end{aligned} \quad (2.30)$$

where $\mathbf{F}_i = \left(E \left[\sum_j \mathbf{W}_{ij} \right] - \mathbf{I} \right)$ and $\mathbf{u}_i = \sum_j E [\mathbf{W}_{ij} (\boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j)]$.

As can be seen from the above derivation, to first order, the bias is an affine function of \mathbf{z}_i . While the parameters of the affine bias (namely, \mathbf{F}_i and \mathbf{u}_i) are different for each patch, we make the simplifying assumption that the same \mathbf{F} and \mathbf{u} provide an adequate approximation of the bias for all patches exhibiting a common geometric structure. This assumption is also statistically justified in Sec. 2.2 of this chapter.

2B Optimal Parameters for Affine Bias Function

In this section we derive expressions for \mathbf{F} and \mathbf{u} that minimize the cost function of Eq. 2.12. This can be obtained by solving a system of simultaneous equations (Eq. 2.13). To do this we first solve for \mathbf{u}

$$\begin{aligned} \frac{\partial Q}{\partial \mathbf{u}} &= \frac{\partial}{\partial \mathbf{u}} \int_{\mathbf{z}} \left[\text{Tr} \{ (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T \} + (\mathbf{F} \mathbf{z} + \mathbf{u})^T (\mathbf{F} \mathbf{z} + \mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z} = 0 \\ \Rightarrow \int_{\mathbf{z}} \left[\frac{\partial}{\partial \mathbf{u}} \text{Tr} \{ (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T \} + \frac{\partial}{\partial \mathbf{u}} (\mathbf{F} \mathbf{z} + \mathbf{u})^T (\mathbf{F} \mathbf{z} + \mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z} &= 0 \\ \Rightarrow \int_{\mathbf{z}} \left[\frac{\partial}{\partial \mathbf{u}} (\mathbf{F} \mathbf{z} + \mathbf{u})^T (\mathbf{F} \mathbf{z} + \mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z} &= 0 \end{aligned}$$

$$\begin{aligned}
&\Rightarrow \mathbf{u} \int_{\mathbf{z}} p(\mathbf{z}) d\mathbf{z} = -\mathbf{F} \int_{\mathbf{z}} \mathbf{z} p(\mathbf{z}) d\mathbf{z} \\
&\Rightarrow \mathbf{u} = -\mathbf{F} \int_{\mathbf{z}} \mathbf{z} p(\mathbf{z}) d\mathbf{z} = -\mathbf{F} E[\mathbf{z}].
\end{aligned} \tag{2.31}$$

Similarly, taking the derivative with respect to \mathbf{F} , we get

$$\begin{aligned}
&\frac{\partial Q}{\partial \mathbf{F}} = \frac{\partial}{\partial \mathbf{F}} \int_{\mathbf{z}} \left[\text{Tr} \{ (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} (\mathbf{I} + \mathbf{F})^T \} + (\mathbf{F}\mathbf{z} + \mathbf{u})^T (\mathbf{F}\mathbf{z} + \mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z} = 0 \\
&\Rightarrow \int_{\mathbf{z}} \left[2(\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} + 2(\mathbf{F}\mathbf{z} + \mathbf{u}) \mathbf{z}^T \right] p(\mathbf{z}) d\mathbf{z} = 0 \\
&\Rightarrow (\mathbf{I} + \mathbf{F}) \mathbf{J}^{-1} \int_{\mathbf{z}} p(\mathbf{z}) d\mathbf{z} + \int_{\mathbf{z}} (\mathbf{F}\mathbf{z} + \mathbf{u}) \mathbf{z}^T p(\mathbf{z}) d\mathbf{z} = 0 \\
&\Rightarrow \mathbf{F} \left[\mathbf{J}^{-1} + \int_{\mathbf{z}} \mathbf{z}\mathbf{z}^T p(\mathbf{z}) d\mathbf{z} \right] = - \left[\mathbf{J}^{-1} + \mathbf{u} \int_{\mathbf{z}} \mathbf{z}^T p(\mathbf{z}) d\mathbf{z} \right] \\
&\Rightarrow \mathbf{F} = - \left[\mathbf{J}^{-1} + \mathbf{u} E[\mathbf{z}\mathbf{z}^T] \right] \left[\mathbf{J}^{-1} + E[\mathbf{z}\mathbf{z}^T] \right]^{-1}.
\end{aligned} \tag{2.32}$$

Now, substituting \mathbf{u} from Eq. 2.31 in Eq. 2.32, we get the optimal \mathbf{F}^* as

$$\begin{aligned}
&\mathbf{F} \left[\mathbf{J}^{-1} + E[\mathbf{z}\mathbf{z}^T] \right] = - \left[\mathbf{J}^{-1} - \mathbf{F} E[\mathbf{z}] E[\mathbf{z}]^T \right] \\
&\Rightarrow \mathbf{F} \left[\mathbf{J}^{-1} + E[\mathbf{z}\mathbf{z}^T] - E[\mathbf{z}] E[\mathbf{z}]^T \right] = -\mathbf{J}^{-1} \\
&\Rightarrow \mathbf{F}^* = -\mathbf{J}^{-1} \left[\mathbf{J}^{-1} + \mathbf{C}_z \right]^{-1},
\end{aligned} \tag{2.33}$$

where $\mathbf{C}_z = (E[\mathbf{z}\mathbf{z}^T] - E[\mathbf{z}] E[\mathbf{z}]^T)$ is the covariance of \mathbf{z} . Thus, we obtain the optimal bias parameters that minimize the function Q as

$$\mathbf{F}^* = -\mathbf{J}^{-1} \left[\mathbf{J}^{-1} + \mathbf{C}_z \right]^{-1}, \tag{2.34}$$

$$\mathbf{u}^* = -\mathbf{F}^* E[\mathbf{z}] = \mathbf{J}^{-1} \left[\mathbf{J}^{-1} + \mathbf{C}_z \right]^{-1} E[\mathbf{z}]. \tag{2.35}$$

2C Higher Order Bias Model

In Sec. 2.2, we assumed that the bias can be modeled reasonably well by an affine function of \mathbf{z} . This allows us to derive the corresponding *optimal bias function*

in Sec. 2.4.1 and, finally, an expression for the MSE bound. Although we have shown experimentally that the bias from some of the recent denoising methods can be effectively modeled as affine, the question about the effect of higher order models remains. In this section, we briefly study the implications of such a higher order model for the bias. For simplicity, we model the bias function to be a restricted second order model:

$$\begin{aligned} \mathbf{b}(\mathbf{z}) &= [b_1(\mathbf{z}) \dots b_l(\mathbf{z}) \dots b_n(\mathbf{z})]^T, \quad \text{and} \\ b_l(\mathbf{z}) &= a_l \mathbf{z}^T \mathbf{z} + \mathbf{f}_l^T \mathbf{z} + u_l, \end{aligned} \quad (2.36)$$

where a_l is a scalar, \mathbf{f}_l^T is the l -th row from a matrix \mathbf{F} , u_l is the l -th entry of a vector \mathbf{u} and n is the number of pixels in a patch. Now, we can express the Bayesian bound as

$$\begin{aligned} Q &= \int_{\mathbf{z}} \left[\text{Tr} \left\{ \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right) \mathbf{J}^{-1} \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right)^T \right\} + \mathbf{b}^T(\mathbf{z}) \mathbf{b}(\mathbf{z}) \right] p(\mathbf{z}) d\mathbf{z} \\ &= \int_{\mathbf{z}} \left[\frac{1}{\sigma^2} \text{Tr} \left\{ \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right) \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right)^T \right\} + \sum_{l=1}^n b_l^2(\mathbf{z}) \right] p(\mathbf{z}) d\mathbf{z} \\ &= \int_{\mathbf{z}} \left[\frac{1}{\sigma^2} \text{Tr} \left\{ \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right) \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right)^T \right\} + \sum_{l=1}^n b_l^2(\mathbf{z}) \right] p(\mathbf{z}) d\mathbf{z}, \end{aligned} \quad (2.37)$$

assuming $\mathbf{J} = \sigma^{-2} \mathbf{I}$ without any loss of generality. Next, it can be seen that

$$\begin{aligned} \text{Tr} \left\{ \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right) \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right)^T \right\} &= \text{Tr} \left\{ \begin{bmatrix} \vdots \\ \left[c_l + \frac{\partial b_l(\mathbf{z})}{\partial \mathbf{z}} \right]^T \\ \vdots \end{bmatrix} \begin{bmatrix} \vdots \\ \left[c_l + \frac{\partial b_l(\mathbf{z})}{\partial \mathbf{z}} \right]^T \\ \vdots \end{bmatrix}^T \right\} \\ &= \sum_{l=1}^n (2a_l \mathbf{z} + \mathbf{f}_l + \mathbf{c}_l)^T (2a_l \mathbf{z} + \mathbf{f}_l + \mathbf{c}_l), \end{aligned} \quad (2.38)$$

where \mathbf{c}_l is the l -th column of the identity matrix containing all zeros except a one at the l -th position and $\frac{\partial b_l(\mathbf{z})}{\partial \mathbf{z}} = (2a_l \mathbf{z} + \mathbf{f}_l)$. We can then write Eq. 2.37 as

$$Q = \int_{\mathbf{z}} \sum_{l=1}^n \left[\frac{1}{\sigma^2} (2a_l \mathbf{z} + \mathbf{f}_l + \mathbf{c}_l)^T (2a_l \mathbf{z} + \mathbf{f}_l + \mathbf{c}_l) + (a_l \mathbf{z}^T \mathbf{z} + \mathbf{f}_l^T \mathbf{z} + u_l)^2 \right] p(\mathbf{z}) d\mathbf{z}. \quad (2.39)$$

As before, we take the derivatives of the right hand side of Eq. 2.39 with respect to the unknown parameters (a_l , \mathbf{f}_l and u_l) and solve the equations to get expressions for the optimal parameters that minimize Q . Differentiating the right hand side of Eq. 2.39 with respect to a_l , \mathbf{f}_l and u_l , we get three simultaneous equations

$$u_l = -a_l E[\mathbf{z}^T \mathbf{z}] - \mathbf{f}_l^T E[\mathbf{z}], \quad (2.40)$$

$$\mathbf{f}_l = - \left(\frac{\mathbf{I}}{\sigma^2} + E[\mathbf{z}\mathbf{z}^T] \right)^{-1} \left(\frac{1}{\sigma^2} \mathbf{c}_l + 2 \frac{a_l}{\sigma^2} E[\mathbf{z}] + a_l E[\mathbf{z}\mathbf{z}^T \mathbf{z}] + E[\mathbf{z}] u_l \right) \quad (2.41)$$

$$a_l = - \left(\frac{4}{\sigma^2} E[\mathbf{z}^T \mathbf{z}] + E[\mathbf{z}^T \mathbf{z}\mathbf{z}^T \mathbf{z}] \right)^{-1} \left(\frac{2\mathbf{c}_l^T}{\sigma^2} + \mathbf{f}_l^T \left(\frac{2}{\sigma^2} + E[\mathbf{z}\mathbf{z}^T \mathbf{z}] \right) + u_l E[\mathbf{z}^T \mathbf{z}] \right). \quad (2.42)$$

Now, using the expression for u_l from Eq. 2.40 in Equations 2.41 and 2.42 we get the system of equations in two variables

$$\mathbf{f}_l = - \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \left(\frac{\mathbf{c}_l}{\sigma^2} + a_l \left(\frac{2}{\sigma^2} E[\mathbf{z}] + E[\mathbf{z}\mathbf{z}^T \mathbf{z}] - E[\mathbf{z}] E[\mathbf{z}^T \mathbf{z}] \right) \right), \quad (2.43)$$

$$a_l \left[\frac{4}{\sigma^2} E[\mathbf{z}^T \mathbf{z}] + E[\mathbf{z}^T \mathbf{z}\mathbf{z}^T \mathbf{z}] - (E[\mathbf{z}^T \mathbf{z}])^2 \right] = -\mathbf{f}_l^T \left[2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right] - \frac{2\mathbf{c}_l^T}{\sigma^2} E[\mathbf{z}], \quad (2.44)$$

where we denote $\mathbf{S}_z = E \left[(\mathbf{z} - E[\mathbf{z}]) (\mathbf{z} - E[\mathbf{z}])^T (\mathbf{z} - E[\mathbf{z}]) \right]$ to be the higher order moment that is related to the multidimensional skewness of the pdf of \mathbf{z} . Now, we use the expression for \mathbf{f}_l as given in Eq. 2.43 and plug it in Eq. 2.44 to obtain

$$\begin{aligned} & a_l \left[\frac{4}{\sigma^2} E[\mathbf{z}^T \mathbf{z}] + E[\mathbf{z}^T \mathbf{z}\mathbf{z}^T \mathbf{z}] - (E[\mathbf{z}^T \mathbf{z}])^2 \right] \\ &= - \left[-\frac{\mathbf{c}_l^T}{\sigma^2} - a_l \left\{ 2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right\} \right] \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \left[2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right] \\ &\quad - \frac{2\mathbf{c}_l^T}{\sigma^2} E[\mathbf{z}] \\ &= \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \left[2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right] - \frac{2\mathbf{c}_l^T}{\sigma^2} E[\mathbf{z}] \\ &\quad + a_l \left[2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right]^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \left[2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right] \end{aligned}$$

$$\begin{aligned}
&\Rightarrow a_l \left[- \left(2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right)^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \left(2 \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] + \mathbf{S}_z \right) \right. \\
&\quad \left. + \frac{4}{\sigma^2} E[\mathbf{z}^T \mathbf{z}] + E[\mathbf{z}^T \mathbf{z} \mathbf{z}^T \mathbf{z}] \right] = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z \\
&\Rightarrow a_l \left[\frac{4}{\sigma^2} E[\mathbf{z}^T \mathbf{z}] + E[(\mathbf{z}^T \mathbf{z})^2] - 4E[\mathbf{z}^T] \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] \right. \\
&\quad \left. - 4E[\mathbf{z}^T] \mathbf{S}_z - \mathbf{S}_z^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z \right] = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z. \quad (2.45)
\end{aligned}$$

This equation can be written in a much simpler form by making use of the relation

$$\begin{aligned}
\frac{4E[\mathbf{z}^T \mathbf{z}]}{\sigma^2} + E[\mathbf{z}^T \mathbf{z} \mathbf{z}^T \mathbf{z}] - 4E[\mathbf{z}^T] \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right) E[\mathbf{z}] - 4E[\mathbf{z}^T] \mathbf{S}_z \\
= \text{Tr}(\mathbf{K}_z) + \text{Tr}^2(\mathbf{C}_z), \quad (2.46)
\end{aligned}$$

where \mathbf{K}_z is related to the multidimensional kurtosis (fourth order moment) of the pdf $p(\mathbf{z})$ and is defined as

$$\mathbf{K}_z = E[(\mathbf{z} - E[\mathbf{z}])(\mathbf{z} - E[\mathbf{z}])^T (\mathbf{z} - E[\mathbf{z}])(\mathbf{z} - E[\mathbf{z}])^T]. \quad (2.47)$$

This allows us to rewrite Eq. 2.45 as

$$a_l \left[\text{Tr}(\mathbf{K}_z) + \text{Tr}^2(\mathbf{C}_z) - \mathbf{S}_z^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z \right] = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z, \quad (2.48)$$

which leads to the expression for the optimal a_l parameter as

$$a_l^* = \frac{\mathbf{c}_l^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z}{\sigma^2 \left\{ \text{Tr}[\mathbf{K}_z] - \mathbf{S}_z^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathbf{C}_z \right)^{-1} \mathbf{S}_z + \text{Tr}^2[\mathbf{C}_z] \right\}}. \quad (2.49)$$

Skewness is a very good indicator of reflectance properties of surfaces such as albedo and gloss [98,99]. As such, the image of a well-exposed scene will generally have small skew such that the histogram of the image is more or less symmetric [97]. This principle is, in fact, behind the tried and true method of histogram equalization which is used often to improve contrast in images. So, for typical natural images,

the term S_z related to the skewness is close to zero and the optimal bias model then collapses to the affine model that we have used earlier.

Chapter 3

Estimation of Denoising Bounds

Abstract – In the previous chapter, we formulated an expression for the fundamental limits for denoising a given image using any patch-based method and identified the parameters for the bound. In this chapter we present practical methods using which the various parameters, and, hence, the bounds can be estimated. We consider both noise-free and noisy cases and show that the bounds can be estimated quite accurately from a single noisy image, even under considerable noise corruption. The bounds computed for various images are compared to the state-of-the-art in image denoising to show that the formulation provides meaningful lower bounds for denoising performance.

3.1 Introduction

In the previous chapter, we analyzed the performance limits of patch-based denoising methods. Since such methods currently achieve state-of-the-art performance (see Fig. 1.8), the expression in (2.18) can be considered to formulate the lower bounds

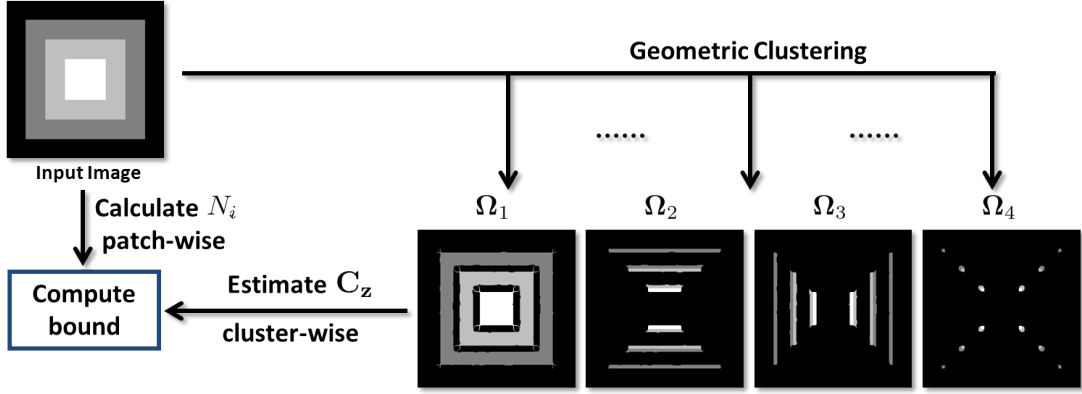


Figure 3.1: Outline of the bounds estimation process.

on the MSE for denoising any given image. The performance limits were shown to be dictated by the strength and characteristics of the corrupting noise (captured by the FIM), as well as image content (captured by the covariance C_z) as

$$E [\|\mathbf{z} - \hat{\mathbf{z}}\|^2] \geq \text{Tr} \left[(\mathbf{J}_i + \mathbf{C}_z^{-1})^{-1} \right], \quad (3.1)$$

where $\mathbf{J}_i = N_i \frac{\mathbf{I}}{\sigma^2}$ when WGN is considered. The influence of image content is captured by patch complexity (C_z) as well as photometric redundancy (N_i). In this chapter, we deal with the issue of accurately estimating the denoising bounds through estimation of these parameters. Since the bounds are defined by the latent (noise-free) image, we first consider the case of estimating the bounds for noise-free images. Later, in Sec. 3.3, we discuss the more practical issue of estimating bounds given any noisy image. However, in either case, the overall framework for bounds estimation remains the same, as illustrated in Fig. 3.1.

As mentioned in Chapter 2, structurally similar patches are assumed to be sampled from the same pdf. The covariance matrix C_z of Eq. 3.1 is, thus, estimated from the entire set (or cluster) of geometrically similar patches. Consequently, the first step in estimating the bounds is identifying patches of similar structure and grouping

them together. Once such a *geometric clustering* is performed, we estimate the covariance matrix \mathbf{C}_z for each cluster Ω_k . Then, using each patch in the image as reference, we identify all photometrically similar patches. Considering this number to be N_i , the bound for denoising each patch is computed using Eq. 3.1. These patch-based bounds are then averaged to obtain the overall bound for the cluster, and, hence, the image itself. In estimating the bounds, we will assume that the noise variance is known a priori or is estimated accurately using any of the methods outlined in [44, 100, 101].

Although the framework for estimating the bounds remains the same for noisy and noise-free images, the effect of noise must be taken into account for the latter case. In the next section, we detail these estimation processes for noise-free images. They are then extended to the more practical case of noise-contaminated images in Sec. 3.3.

3.2 Estimating Denoising Bounds from Ground Truth

The bound in denoising any given image is dependent on noise statistics as well as image content. This image complexity is essentially defined by the latent *noise-free* image. Although in practice access to such clean images cannot be guaranteed, in this section we assume availability of such ground truth from which the parameters that define the denoising bounds are estimated. As mentioned earlier, the input image is first clustered into geometrically similar regions. In deriving the bounds we assumed such segmentation to be provided by some oracle. Next, we describe how such clustering can be achieved in practice.

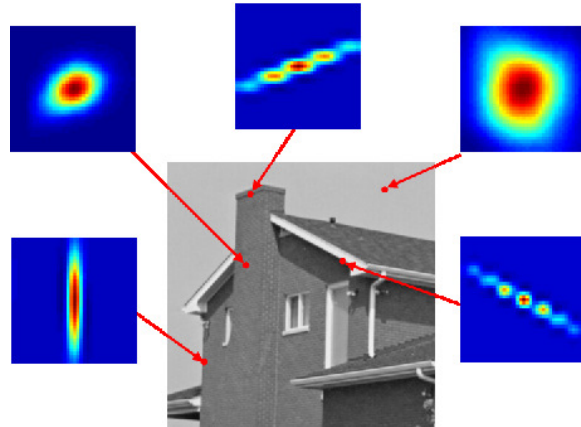


Figure 3.2: Steering kernels at different locations of the house image. The patch size is chosen to be 11×11 . Note how the kernels adapt to the underlying image structure.

3.2.1 Practical (“Non-Oracle”) Geometric Clustering

Clustering is a much studied research problem that has led to a number of different methods [102, 103] (see [104] for a nice survey). Image clustering is a subset of this huge field where researchers have devoted considerable attention to the choice of a clustering method, the features to use to achieve the intended segmentation, as well as choosing a meaningful distance metric that captures similarities between the chosen features. The choice of features to work with is particularly important as they need to effectively portray the property on which the clustering is to be based. For our purposes, we need to identify features that capture the underlying geometry of the image patches, without regard to their absolute intensities. For this, we make use of the locally adaptive regression kernels (LARK) developed by Takeda *et al.* [22]. As an added benefit, such features are also quite robust to the presence of noise. This property of the LARK features makes it very useful for clustering noisy images, as is required in estimating the bounds for noisy images, and later for our work in Chapters

4 & 5. In Fig. 3.2, we see that the LARK features are indicative of the underlying image geometry. These kernels, normalized such that the elements sum to one, form excellent descriptors of the underlying image geometry [24, 25]. We use these as feature vectors to perform clustering.

Before we proceed to perform clustering on the weights, we need to specify a metric to calculate the distance between LARK features. The easiest measure of distance to use is the ℓ_1 or ℓ_2 distance between the features. Other measures such as the Mahalanobis, some weighted distance metric, Kullback-Leibler divergence [105] or matrix cosine similarity [25] are equally applicable. The choice of the optimal distance metric to use for the LARK features remains an open question. Our experiments with few of the above mentioned metrics show that the final bounds estimate for different images are largely insensitive to the metric chosen. As a result, we refrain from delving deep into this metric selection problem and use the simple ℓ_2 metric to compute distances between the normalized LARK features.

The next question in performing our geometric clustering is to select a particular clustering method. Although many clustering methods are suitable for our clustering purposes, we use K-Means [102] due to its simplicity and efficiency. K-Means requires as input the features (normalized LARK vectors in our case) and the number of clusters. For our work, we require the user to specify the number of clusters to segment the image into. The number of clusters will vary across images based on the variance in the edge orientation and strength that an image exhibits. The choice of the number of clusters is important to us since too few members in a particular cluster will lead to erroneous estimation of the covariance matrix for \mathbf{z} and as a result an erroneous MSE bound. On the other hand, too few clusters will result in patches of widely vary-



(a) Clustering of box image



(b) Clustering of house image

Figure 3.3: Clustering using K-Means for the box and house images. Notice how edges and patterns of a certain kind are clustered together even though the patches may have different intensities.

ing geometric structures being grouped together, again resulting in an inaccurate MSE bound. This is especially true for K-Means which tends to prefer similar sized clusters. Fortunately, variations in patch patterns within a given natural image are usually limited and our bounds can be computed effectively with a fixed number of clusters (K) for most natural images. The effect of K on the predicted MSE bounds is discussed later in Sec. 3.4.

Fig. 3.3 illustrates the effectiveness of using K-Means with our choice of LARK features and the ℓ_2 distance metric. There we see that regions similar in geometry are clustered together, even though the pixel intensities may widely differ in any particular cluster. Note how even finer textures such as the facade of the house are correctly captured by the features and are, hence, differentiated from the largely smooth background. This shows that with our choices of features and distance metric, a simple clustering method such as K-Means is able to perform geometric clustering accurately.

3.2.2 Covariance Estimation from Ground Truth

Once the image is segmented into regions of similar structure, we compute the covariance of \mathbf{z} within each cluster. This covariance is the second moment of $p_k(\mathbf{z})$ from which the geometrically similar latent patches \mathbf{z}_i are assumed to be sampled. The sample covariance matrix is known to be the maximum likelihood estimate (MLE) for the second moment, approaching the actual covariance as the number of sample patches tends to infinity. However, presence of sufficiently large number of geometrically similar patches cannot always be guaranteed. This occurs, for example, in the cluster containing the corner regions for the box image in Fig. 3.3(a) where only a few patches are present when compared to the other clusters. Consequently, we need to use an estimator that is robust with respect to possibly limited number of samples. Estimating the covariance matrix from a limited sample set has been an active field of research with applications spanning diverse disciplines (see [106] and references therein). Of them, perhaps one of the best studied methods is bootstrapping [107]. In our work, we use this approach to estimate the distribution moments.

Bootstrapping is a method of estimating parameters of an unknown distribution from its empirical distribution formed from a finite set of samples (\mathbf{z}_i in our case). This well-studied statistical method performs sampling with replacement from the set of observed samples to form multiple empirical distributions. The parameters of interest (in our case, the first and second order moments) are then calculated from each such empirical distribution. The final estimate of the covariance is then obtained as an average of all the calculated parameters. This final estimate converges to the actual second moment when re-sampling is performed sufficiently many times [108]. Since the covariance itself is calculated through an estimation process, it has associated with

it a *confidence interval*. This means that ultimately our lower bound is, in practice, a stochastic one with a corresponding confidence interval. Since the parameter of interest is the covariance matrix \mathbf{C}_z , the associated confidence interval itself will be of similar dimensions. To simplify matters, we instead use the bootstrapping mechanism to directly estimate the MSE bound (Q_{\min}) from each empirical distribution and obtain an associated confidence interval for it. This is done using the following steps :

1. Given the noise-free image, make non-overlapping patches \mathbf{z}_i .
2. Generate \mathcal{M} samples ($\mathbf{z}_{B,j}$) with replacement from the pool of available \mathbf{z}_i samples (empirical distribution) to generate a bootstrap sample set \mathbf{B} .
3. Estimate \mathbf{C}_z from the bootstrap sample set using the formula

$$\hat{\mathbf{C}}_z = \frac{1}{(\mathcal{M} - 1)} \sum_{j=1}^{\mathcal{M}} (\mathbf{z}_{B,j} - \bar{\mathbf{z}}_B)(\mathbf{z}_{B,j} - \bar{\mathbf{z}}_B)^T, \quad (3.2)$$

where $\bar{\mathbf{z}}_B$ is the mean of all the $\mathbf{z}_{B,j}$ vectors that make up set \mathbf{B} .

4. Compute Q_{\min} with the estimated $\hat{\mathbf{C}}_z$ using (3.1).
5. Repeat steps 2 through 4, \mathcal{R} times.

In each of the \mathcal{R} iterations, an estimate of the covariance of \mathbf{z} and a corresponding estimate of Q_{\min} are obtained as the bootstrap estimates. Finally, these bootstrap estimates of Q_{\min} are averaged to obtain the estimated MSE bound (denoted as \hat{Q}_{\min}). The confidence interval of the MSE bound estimate can be readily calculated as the 95% confidence interval given by the Normal interval¹ formulation [109]:

$$\hat{Q}_{\min} \pm 2\sigma_Q, \quad (3.3)$$

¹This interval formulation is accurate only if the distribution of Q_{\min} is close to Normal. Our experiments indicate that the histograms of the bootstrapped Q_{\min} values for different images indeed closely approximate a Gaussian.

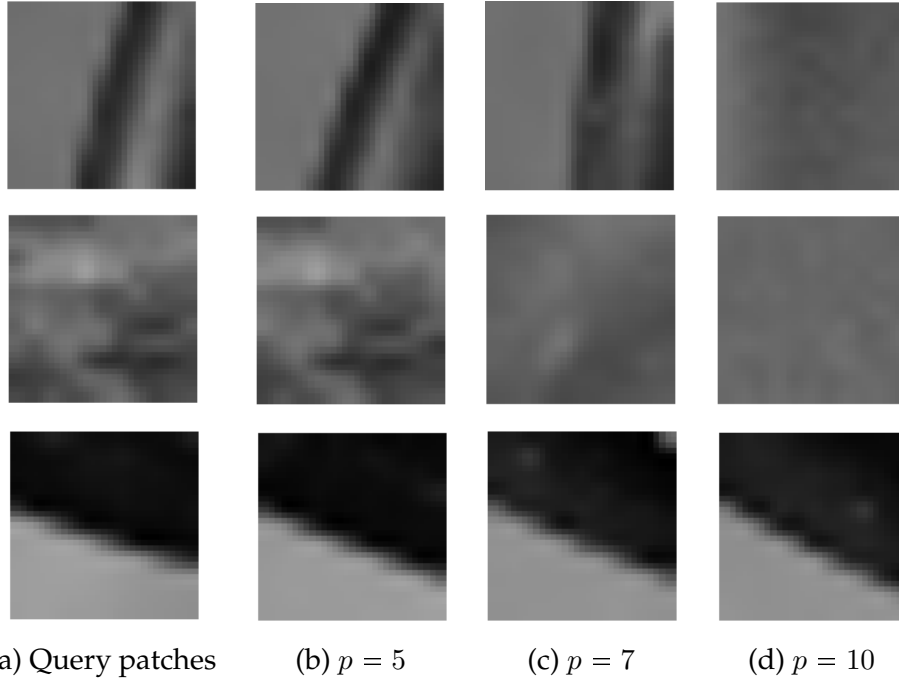


Figure 3.4: Some query patches and their respective least similar neighbors as defined by (3.4) with various values of p found from a dictionary of approximately 450,000 noise-free patches from 4 different images.

where σ_Q is the standard error of the bootstrapped estimate \hat{Q}_{\min} .

3.2.3 Calculating Patch Redundancy (N_i)

While the covariance matrices are learned on a cluster-by-cluster basis, the level of photometric redundancy N_i is learned patch-wise. To determine N_i , we first need to define a measure of similarity between two patches. We consider two patches \mathbf{z}_i and \mathbf{z}_j to be similar if they can be expressed as

$$\mathbf{z}_j = \mathbf{z}_i + \boldsymbol{\varepsilon}_{ij} \quad \text{such that} \quad \|\boldsymbol{\varepsilon}_{ij}\|^2 \leq \gamma^2, \quad (3.4)$$

where γ is a small threshold. This threshold needs to be chosen carefully, to ensure few false positives and negatives. Further, the threshold should also take into account the

number of pixels present in each patch. For our experiments, we choose γ to be such that all \mathbf{z}_j patches that are identified to be similar to \mathbf{z}_i differ (on average) in less than $p\%$ of the range of intensity values in each pixel location. Assuming this range to be within 0 to 255, an expression for the threshold is

$$\gamma^2 = \left(\frac{p \times 255}{100} \right)^2 \times n, \quad (3.5)$$

where n is the number of pixels in each image patch.

The value of p is empirically chosen such that photometric similarity of patches that satisfy (3.4) can be guaranteed for all patches. For this, we devised an experiment where 11×11 patches from 4 different images were used to form a database of approximately 450,000 photometrically (and geometrically) diverse patches. We then randomly chose some patches from the database and searched for similar patches using various values of p . Fig. 3.4 shows some reference patches with interesting structure and the corresponding *least* similar patches that satisfied (3.4) for different values of p . The results there show that $p = 5$ is a reasonable choice for the threshold. That is to say, *similar* patches are allowed to vary, on average, in less than 5% of the intensity range for each pixel. In what follows, we fix $p = 5$ throughout the rest of this thesis.

We have now described how to estimate the parameters from which the bounds can be obtained for each patch within a cluster and, hence, the cluster as a whole. The estimated cluster-wise bounds (\hat{Q}_k) can then be aggregated to obtain the bound \hat{Q} for the entire image as (Eq. 2.25)

$$\hat{Q} = \frac{1}{M} \text{SSE} = \sum_{k=1}^K \frac{M_k}{M} \hat{Q}_k = \sum_{k=1}^K \alpha_k \hat{Q}_k, \quad (3.6)$$

where $\alpha_k = \frac{M_k}{M}$. The covariance in each cluster being an *estimated* parameter, the estimated bound \hat{Q}_k has an associated confidence interval, as shown in Eq. 3.3. An ex-

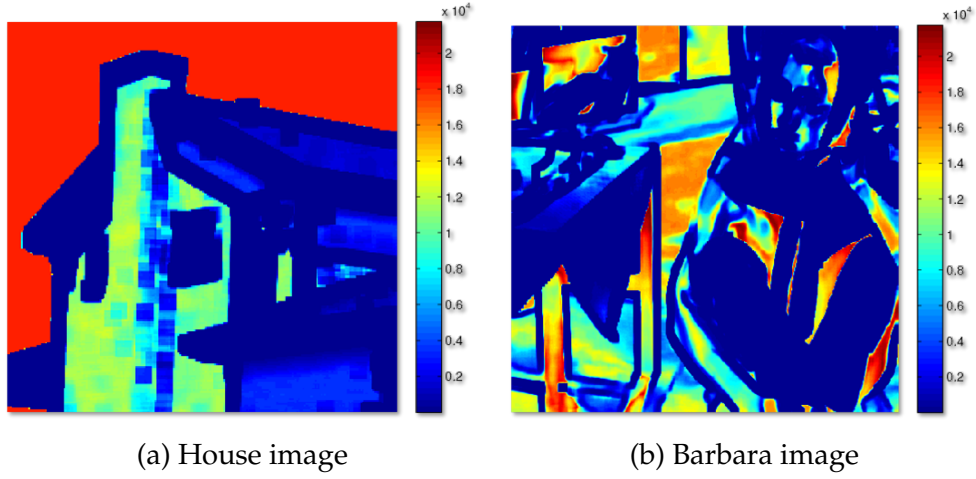


Figure 3.5: The spatial distribution of N_i values for a patch size of 11×11 on (a) house image, and (b) Barbara image, shown in Fig. 1.7.

pression for the 95% confidence interval for the overall bounds can then be obtained by calculating the standard deviation (σ_Q) of the \hat{Q} estimate as

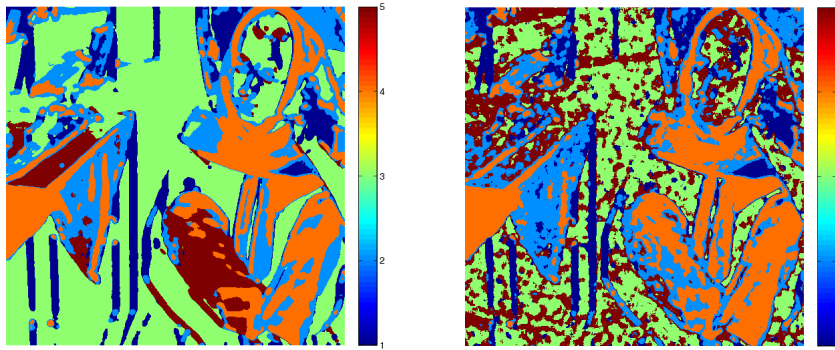
$$\sigma_Q = \sqrt{\sum_{k=1}^K \alpha_k^2 \sigma_{Q_k}^2}, \quad (3.7)$$

where σ_{Q_k} is the standard deviation of the \hat{Q}_k estimate. The 95% confidence interval, as shown before in (3.3), is then given by the Normal interval

$$\hat{Q} \pm 2\sigma_Q. \quad (3.8)$$

3.3 Bounds Estimation for Noisy Images

Until now, we have outlined a method of estimating the bounds from noise-free images by estimating the parameters of the bound. However, in practice, such ground truth is not available. To make our method practical, it is necessary to account for the presence of noise when estimating the bounds. In this section, we extend each



(a) Noise-free case

(b) Noisy case

Figure 3.6: Clustering of Barbara image into 5 clusters based on geometric structure of patches. Clustering is performed with features calculated from (a) clean image, and (b) noisy image of noise standard deviation 15. Note how the kernel features can capture structural information and thereby properly cluster majority of patches even in the presence of noise.

parameter estimation process by considering the effects of noise contamination. We begin by estimating the covariance matrix from noisy image patches.

3.3.1 Covariance Estimation

To estimate the covariance matrix, we need to first identify patches of similar structure from the given noisy image. As with the noise-free case, this requires us to cluster the image patches based on the underlying geometric structure. Such clustering from noisy data can be challenging. To avoid this, in [110] we presented a method of learning the covariance matrix without performing any explicit clustering on the noisy image. For that, we made use of a vast database of varied noise-free patches that were processed off-line to form a look-up table from which the covariance matrix for any noisy patch was estimated. However, methods employing such off-line databases are restrictive as they require a very large and relevant set of clean image patches, the (lack of) quality and variety of which can strongly influence the results.

Instead, here we compute the LARK features [22, 24, 25] for each patch in the noisy image and then perform clustering using K-Means [102], much in the same way as done for the noise-free case in Sec. 3.2. In [24], we demonstrated that such normalized kernels can be quite robust to the presence of noise, leading to relatively robust clustering performance. This is illustrated in Fig. 3.6 where we show the overall similarities between the (color-coded) clusters of the noise-free and noisy Barbara images. However, inaccuracies do appear, especially when dealing with strong noise. In such cases, one can pre-filter the noisy image to reduce the effect of noise and perform clustering on the denoised image. As will be apparent from the experimental results in Sec. 3.4, the covariance estimate obtained using such a clustering leads to quite accurate estimates of the denoising bounds, even for substantially noisy images ($\sigma = 25$).

Once the image has been clustered, we proceed to compute the covariance of the noisy patches in each cluster. For this, we employ the bootstrapping method of Efron [107], although other stable and computationally efficient methods (such as [106] and references therein) are equally applicable. This allows us to estimate the covariance \mathbf{C}_y of the noisy patches within a given cluster, from which we need to estimate the covariance matrix \mathbf{C}_z . From the data model of Eq. 2.1, it is easy to see that

$$\mathbf{C}_z = \mathbf{C}_y - \mathbf{C}_\eta, \quad (3.9)$$

where $\mathbf{C}_\eta = \sigma^2 \mathbf{I}$ is the covariance of the iid noise, assumed to be independent of the patch intensity. However, directly using Eq. 3.9 can lead to an estimate of \mathbf{C}_z that may not be positive semidefinite, a necessary property of covariance matrices. To avoid such problems in the estimation of the covariance, we use a modified plug-in estimator [55, 111, 112]

$$\hat{\mathbf{C}}_z = [\hat{\mathbf{C}}_y - \sigma^2 \mathbf{I}]_+, \quad (3.10)$$

where $\hat{\mathbf{C}}_y$ is the covariance estimated from the noisy image patches and $[\mathbf{X}]_+$ denotes a matrix with the negative eigenvalues of \mathbf{X} replaced by $\theta \approx 0$. Note that it may still be the case that $\hat{\mathbf{C}}_z$ is rank deficient and, hence, not invertible. Therefore, we compute the bounds using an alternate formulation based on the matrix inversion lemma [96] as

$$\begin{aligned} E [\|\mathbf{z}_i - \hat{\mathbf{z}}_i\|^2] &\geq \text{Tr} \left[\left(\mathbf{J}_i + \hat{\mathbf{C}}_z^{-1} \right)^{-1} \right] \\ &= \text{Tr} \left[\mathbf{J}_i^{-1} - \mathbf{J}_i^{-1} \left(\mathbf{J}_i^{-1} + \hat{\mathbf{C}}_z \right)^{-1} \mathbf{J}_i^{-1} \right], \end{aligned} \quad (3.11)$$

where the covariance estimate $\hat{\mathbf{C}}_z$ need not necessarily be invertible.

Another point to note is that we assume knowledge of the noise variance in our estimation of the covariance matrix in Eq. 3.10. However, in practice, this needs to be estimated from the given noisy image. In this case, one can employ methods outlined in [44, 100, 101] where it is shown that noise variance can be quite accurately estimated from a single noisy image.

One disadvantage of the shrinkage-based maximum likelihood estimator of Eq. 3.10 is that they can lead to inaccuracies when only a few patches (compared to the number of pixels in each patch) are present in a cluster, as demonstrated in [113]. In such cases, the estimation process can be modified based upon observations in [113, 114]. Luckily, such structural singularities are not very common among natural images. As a result, the estimation process outlined above allows us to estimate the bounds quite accurately, as we demonstrate in Sec. 3.4. There we also show that the estimated $\hat{\mathbf{C}}_z$ matrices are quite robust to minor inaccuracies in noise variance estimation, as well as to the presence of outliers that appear due to errors in clustering a noisy image. As such, these estimates are sufficiently accurate for us to estimate the bounds from any given noisy image.

3.3.2 Photometric Redundancy from Noisy Images

Next, we need to calculate the FIM from the noisy image. Considering the corrupting noise to be additive white Gaussian with known (or estimated [44, 100, 101]) variance and zero mean, estimating the FIM reduces to estimating the redundancy factor N_i for each patch. We obtain a k -nearest neighbor based estimate for N_i from the noisy input image, similar to the case where the MSE bounds were estimated from noise-free images [70]. However, the similarity measure of Eq. 3.4 needs to be modified to account for the effects of the corrupting noise. In the present context, given any noisy patch \mathbf{y}_i , we wish to identify patches \mathbf{y}_j in the noisy image, such that their corresponding noise-free counterparts \mathbf{z}_i and \mathbf{z}_j satisfy the similarity condition defined in Eq. 3.4. Thus, we define a measure of similarity between *noisy* patches as

$$\begin{aligned}
 \mathbf{z}_j &= \mathbf{z}_i + \boldsymbol{\varepsilon}_{ij} \\
 \Rightarrow \mathbf{y}_j - \boldsymbol{\eta}_j &= \mathbf{y}_i - \boldsymbol{\eta}_i + \boldsymbol{\varepsilon}_{ij} && \text{[from Eq. 2.1]} \\
 \Rightarrow \mathbf{y}_j &= \mathbf{y}_i + \underbrace{(\boldsymbol{\eta}_j - \boldsymbol{\eta}_i + \boldsymbol{\varepsilon}_{ij})}_{\tilde{\boldsymbol{\varepsilon}}_{ij}}, && (3.12)
 \end{aligned}$$

$$\begin{aligned}
 \text{where } \|\tilde{\boldsymbol{\varepsilon}}_{ij}\|^2 &= \|\boldsymbol{\varepsilon}_{ij}\|^2 + \|\boldsymbol{\eta}_j - \boldsymbol{\eta}_i\|^2 + 2\boldsymbol{\varepsilon}_{ij}^T(\boldsymbol{\eta}_j - \boldsymbol{\eta}_i) \\
 \Rightarrow E[\|\tilde{\boldsymbol{\varepsilon}}_{ij}\|^2] &= E[\|\boldsymbol{\varepsilon}_{ij}\|^2] + 2\sigma^2 n, && (3.13)
 \end{aligned}$$

considering $\sqrt{n} \times \sqrt{n}$ patch dimensions. The last expression of Eq. 3.13 is obtained assuming the noise patches are iid. A noisy patch \mathbf{y}_j can then be considered photometrically similar to \mathbf{y}_i if it satisfies the condition

$$\mathbf{y}_j = \mathbf{y}_i + \tilde{\boldsymbol{\varepsilon}}_{ij} \quad \text{such that} \quad \|\tilde{\boldsymbol{\varepsilon}}_{ij}\|^2 \leq \gamma^2 + 2\sigma^2 n, \quad (3.14)$$

where γ is the threshold defined in Eq. 3.4. Note that, as with the estimation of the covariance matrix, we make use of the known (or estimated [44, 100, 101]) noise variance

in identifying similar patches. With a similarity measure defined, we can now estimate N_i values for each patch within the given noisy image.

Once an estimate of the N_i values for each patch (denoted by \hat{N}_i) and its associated covariance matrix ($\hat{\mathbf{C}}_{\mathbf{z}}$) are obtained, we can estimate the MSE bound for denoising from the input noisy image as

$$E[\|\mathbf{z}_i - \hat{\mathbf{z}}_i\|^2] \geq \frac{1}{M} \sum_{i=1}^M \text{Tr} \left[\hat{\mathbf{J}}_i^{-1} - \hat{\mathbf{J}}_i^{-1} \left(\hat{\mathbf{J}}_i^{-1} + \hat{\mathbf{C}}_{\mathbf{z}} \right)^{-1} \hat{\mathbf{J}}_i^{-1} \right], \quad (3.15)$$

with $\hat{\mathbf{J}}_i = \hat{N}_i \frac{\mathbf{I}}{\sigma^2}$ and $M = \sum_k M_k$ is the total number of patches in the image. This proposed estimation method can be used to accurately predict the denoising bounds for images corrupted by considerable levels of noise, as we will demonstrate in the next section. However, as expected, the accuracy degrades when the input signal-to-noise ratio is severely low. In our experiments with different images (Fig. 1.7), this breaking point occurs when the corrupting noise has a standard deviation σ greater than 15. In such cases, it is useful to pre-filter the noisy image to reduce the effects of noise. The N_i values can then be estimated directly from the noise-suppressed version of the given image. Next, we compute the bounds for various (noisy and noise-free) images and compare them to the state-of-the-art denoising performance.

3.4 Denoising Bounds and State-of-the-Art

In this section we describe experimental results where we calculate the MSE bounds for various images and compare these to the performance of several state-of-the-art denoising algorithms. We begin with estimating the bounds from ground truth images to show how well a given image can be expected to be denoised. For this, we first perform experiments on simulated images of simple repeating patterns. We then

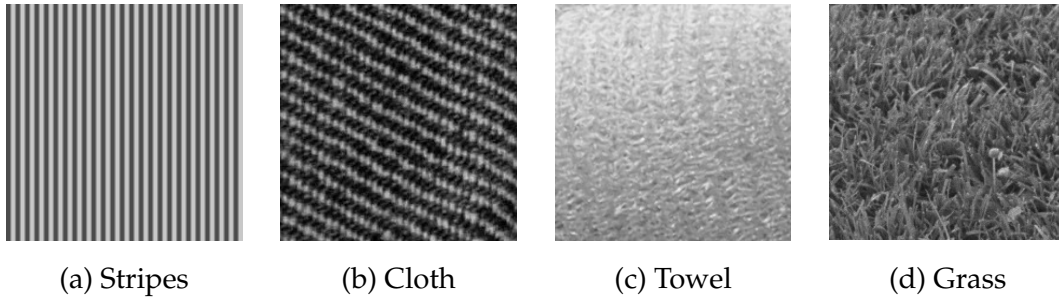


Figure 3.7: Some images consisting of geometrically similar patches that we use for our study.

show results obtained using uniform texture images and on more general images that consist of both smooth and texture regions. Finally, we show that the bounds can be estimated quite accurately even when presented with a single noisy image using the approach outlined in Sec. 3.3.

Denoising performance is often dependent on many intrinsic parameters of any given method. Similarly, for the purposes of estimating the bounds, we need to take into account the influence of certain implicit parameters such as the size of the patches and the number of clusters used. Therefore, before any meaningful bounds can be estimated, we need to identify suitable choices for such intrinsic parameters.

We begin with studying the effect of patch size which plays an important role in calculation of the MSE bounds. Too large a patch size might capture regions of widely varying geometric structure in a single patch and also result in fewer similar patches being present in the image. On the other hand, too small a patch size can lead to degraded denoising performance resulting from the lack of geometric structure captured by each patch. In practice, noise greatly impairs the search for nearest neighbors when too small a patch size is considered. In our work, search for similar patches is carried out on the noise-free image resulting in larger values of N_i when using smaller

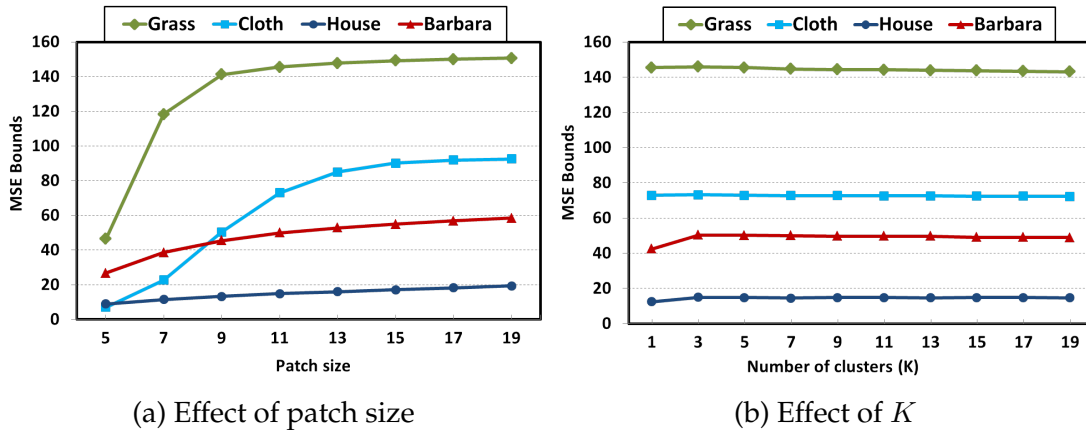


Figure 3.8: MSE bounds for noise standard deviation 25 as a function of (a) varying patch size with $K = 1$ for the grass and cloth images (Fig. 3.7), and $K = 5$ for the house and Barbara images; and (b) varying number of clusters with patch size 11×11 .

patches. As a result, the MSE bounds for smaller patches are quite small, which contradicts the performance we observe in practical denoising methods which only have access to the noisy observations. But this effect is typically stabilized with patch sizes of 11×11 or beyond. Fig. 3.8(a) illustrates this effect on different images. Note how the bound on the predicted MSE increases at different rates as the patch size grows from 5×5 to 19×19 for the images. In our comparisons, we will calculate the bounds with a fixed patch size of 11×11 which is a reasonable choice for denoising as it can capture the underlying patch geometry while offering sufficient robustness in the search for similar patches.

The other parameter that influences our predicted lower bound is the number of clusters. Clustering ensures that patches of similar geometric structure are grouped together. In Fig. 3.8(b), we show the effect of the predicted bounds as a function of increasing number of clusters. Note how, in most cases, the MSE bounds change little once the number of clusters is chosen to be $K = 5$ or higher. This may encourage one

to think that it might be best to use a much larger number of clusters ($K = O(M)$). However, with a smaller K , we can ensure the presence of enough patches in the clusters so as to obtain a reasonably accurate estimate of the covariance matrix for each cluster. At the same time, we do not compromise on the requirement that patches of similar geometric structure be grouped together in each cluster. On the other hand, choosing too small a value for K results in an erroneous bound as dissimilar patches may be clustered together and the covariance matrix is then learned assuming that all \mathbf{z}_i vectors are sampled from a single $p(\mathbf{z})$. For the natural images of Fig. 1.7 clearly $K = 1$ is not a good choice. As a general rule, choosing a value of K to lie within 5 and 10 leads to a stable estimate of the bound without incurring unnecessary time penalty in clustering. Such a choice is also roughly in keeping with the number of clusters used for denoising various images in [24, 40, 69].

In Fig. 3.8(b), we observe that for typical natural images such as the house and Barbara images that exhibit variations among patch patterns, using $K = 1$ always results in a lower bound than for higher values of K . This is somewhat contrary to intuition where one would expect the intra-cluster variation among the structurally dissimilar patches forced to lie within a single cluster to increase the average patch complexity resulting in a higher bound for smaller K . However, it should be noted that the bounds are also influenced by the level of photometric redundancies (N_i) for each patch. When the effect of N_i is nullified by forcing $N_i = 1$ for all \mathbf{z}_i , the bound obtained with $K = 1$ is indeed higher than those for $K > 1$.

Having established specific choices for patch size and number of clusters and their effects on the computed bounds, we now proceed with estimating the bounds for various images. We begin our experimental analysis of the bounds with the simulated

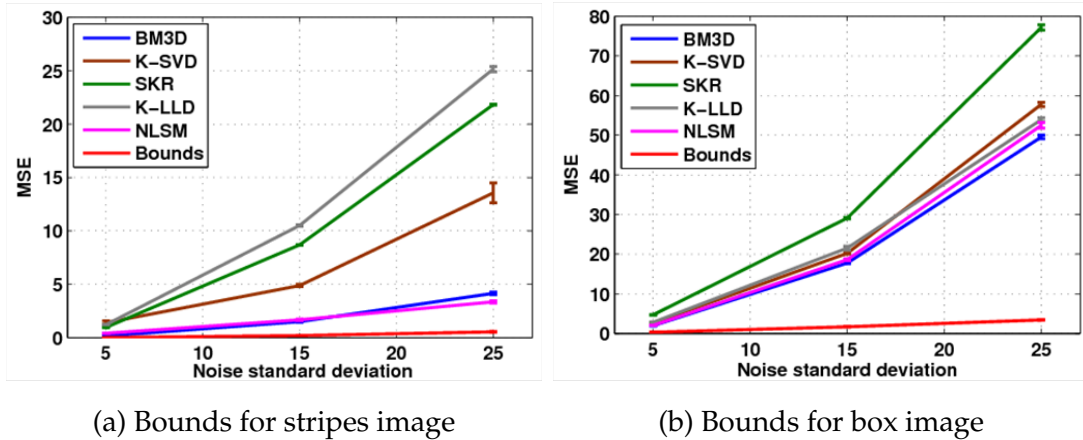
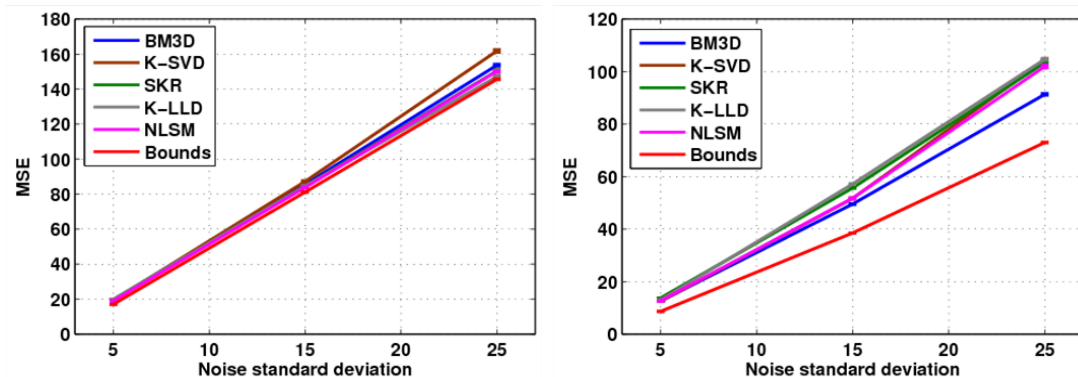


Figure 3.9: MSE bounds computed on simulated images and compared with the performance of some state-of-the-art methods (BM3D [49], K-SVD [37], SKR [22], K-LLD [24], NLSM [43]). The bounds are calculated using 11×11 patches.

stripes image (of size 220×220) that we generated to provide a proof of concept for our MSE bounds calculation. The image (shown in Fig. 3.7(a)) consists of simple repeating patterns (stripes), each 2 pixels wide, made up of two gray levels (75 and 200). It is very easy to see that for our choice of 11×11 patches, the image patches will all be similar in geometric structure and, hence, no clustering is necessary for this particular image. Fixing the patch size to be 11×11 , we calculate the performance bounds of denoising this particular image under WGN of different strengths ($\sigma = 5, 15, 25$). We compare the lower bound to the MSE obtained using various state-of-the-art denoising methods ([22, 24, 37, 43, 49]). From the plots in Fig. 3.9(a) we see that our MSE bound is quite small as a result of larger number of identical patches being available. Also, the image consists of a very simple repeating pattern leading to rather small variability in geometric structure of the image patches. This makes it easier to denoise as opposed to more complex natural images. Our bounds formulation takes into account these factors



(a) Bounds for grass image

(b) Bounds for cloth image

Figure 3.10: Bounds for texture images compared to denoising performance of some state-of-the-art denoising methods. A single cluster of 11×11 patches are considered for this experiment.

and predicts a lower bound on the MSE that is rather lower than the performance of the state-of-the-art denoising algorithms.

As a next step, we calculate the MSE bounds for another, more interesting, simulated image. Fig. 2.1(b) shows the box image (of size 200×200) where, as opposed to the stripes image, the edges vary in directionality. Clearly, such an image requires the use of multiple clusters to capture the different geometric structures. As shown earlier in Fig. 2.5(a), we make use of 4 clusters to capture the smooth, horizontal and vertical edges, and the corner regions. Fig. 3.9(b) shows the calculated MSE bounds for the box image for different noise standard deviations and compares them to the performance of denoising methods. This image is more difficult to denoise than the stripes image and the predicted MSE bound is also considerably lower than the MSE obtained by any of the state-of-the-art denoising methods.

We now present experimental results obtained using images containing relatively uniform natural texture. These images (example, the grass image) typically

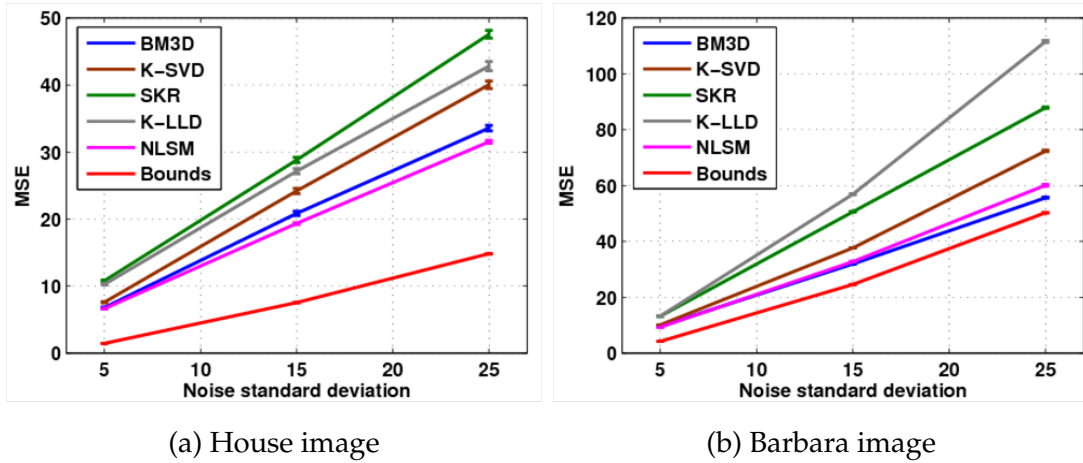


Figure 3.11: Comparison of some state-of-the-art methods with our bounds formulation for some general images. The patch size is fixed at 11×11 and the number of clusters (K) used is 5 in all the cases.

contain semi-stochastic repetitive patterns. Patches from such images can be considered to be geometrically similar and, hence, form a single cluster. However, the patches typically contain more natural variability in their structure than the synthetic stripes image. For such heavily textured images, most methods perform quite comparably to the predicted bounds (see Fig. 3.10(a)). The bound for another texture image (cloth) in Fig. 3.10(b) is lower than the best performing method (BM3D), but not significantly so. These seem to indicate that, theoretically, the performance cannot be vastly improved for such class of images. Also note that the MSE for each of the methods (and our bounds prediction) are much higher than those obtained for the simulated images. This is because the (semi-stochastic) variability in the image patches makes them harder to denoise than the simpler simulated images. This fact is captured by our bounds formulation as well.

Next, we evaluate the bounds for some natural images, namely the house and Barbara images. Such images typically consist of both smooth and textured regions.

Thus, clustering is needed to group together patches of similar geometric structure. In our experiments, we cluster each image into 5 clusters using the technique outlined in Sec. 3.2.1. The MSE bound is then calculated on a per cluster basis from which a final estimate of the MSE bound is obtained for the entire image, along with a confidence interval for the estimate. Fig. 3.11 shows the MSE bounds obtained for two natural images using a patch size of 11×11 and its comparison to performances of some state-of-the-art methods.

The bounds for different images also give us an indication of the relative difficulty in denoising images. In Table 3.1 we rank various images based on the predicted denoising bounds, considering WGN with $\sigma = 25$. The predicted bounds shown there agrees with the intuition that images that are relatively smooth and devoid of much texture (e.g. house image) are easier to denoise than those containing fair amounts of semi-stochastic textures (e.g. stream and Mandrill images). This relative difficulty is also largely in keeping with the performance of some recent denoising methods.

Apart from the relative denoising difficulty, the bounds when compared to the state-of-the-art can also serve as an indicator of the room for improvement in denoising performance that we can hope to achieve. Table 3.1 shows that images containing a fair amount of non-stochastic texture are denoised quite well as compared to the predicted bounds and little room for improvement exists. One probable reason for this is that for naturally occurring textures, few similar patches may exist. Moreover, identifying such patches under noise contamination is also not trivial. These contribute towards a higher MSE for the denoised estimate. As an extreme case, consider images where, on average, N_i is close to 1. Denoising then has to be performed from essentially a single observation of each patch and, hence, not much denoising can be

Table 3.1: Some images ranked according to the predicted denoising bounds showing their relative denoising difficulty. The noise standard deviation is 25 and the bounds are calculated using 11×11 patches.

Image	K-SVD [37]	SKR [22]	K-LLD [24]	BM3D [49]	NLSM [43]	Bound
Stripes	13.56	21.83	25.15	4.16	3.36	0.55
Box	57.78	77.17	53.93	49.56	52.49	3.42
House	40.05	47.57	42.82	33.57	31.56	14.82
Lena	48.09	44.09	46.02	40.46	42.57	19.66
Boats	78.39	78.44	77.45	67.17	69.20	38.70
Barbara	72.39	87.91	111.58	55.62	60.13	50.24
Cloth	104.36	103.42	104.68	91.33	101.97	72.98
Grass	161.74	150.39	147.13	153.64	150.16	145.58
Mandrill ²	185.60	196.20	195.75	188.84	178.94	181.61

expected. Our formulation also cannot be expected to predict an informative bound for such extreme cases. However, for most general images, our formulation predicts meaningful bounds, as justified by various experiments shown in this section. In Chapter 6, we analyze this further and show that improvement in denoising performance can still be expected, particularly for a class of smoother images.

While the predicted bounds were used to rank *images* based on their relative denoising difficulty in Table 3.1, the cluster-wise calculation process allows us to obtain such ranking for different clusters within the same image as well. This we demonstrate later in Chapter 4 where we present information-theoretic interpretations of the bounds formulation. The bound for each cluster can then be used to adaptively control the amount of smoothing to be performed by any denoising method based on the un-

²NLSM achieves an MSE that is slightly lower than the bounds for this case. This anomaly can be explained by analyzing the bias characteristics for this non-linear method, on such images. Our bounds are derived for affine-biased methods, and the method bias in *most* cases conform to this model. Further, the bounds here are estimated by restricting $N_i \leq 100$ as a practical consideration. With a more relaxed maximum (say, 1,000), we obtain a bound of 176.24 which is lower than the MSE for NLSM.

derlying image content. However, for such a system, the cluster-wise bounds needs to be accurately estimated from a single noisy image. Such a mechanism was presented in Sec. 3.3. Employing the estimation processes detailed there, we estimate the bounds for some *noisy* natural images next. To verify the accuracy of such comparisons we compare the bounds estimated from the noisy images to those computed from their corresponding clean versions (ground truth).

Since the bounds are estimated by estimating the parameters independently, we analyze the accuracy of estimating each parameter as well. As a first step, we consider the accuracy of the covariance estimates from a given noisy image. The covariance estimates also depend on the clustering performance, which in turn is also influenced by the presence of noise (see Fig. 3.6). However, our experiments reveal that the covariance estimation process is quite robust to the presence of outliers within each cluster. This can be inferred from Fig. 3.12 where we plot the bounds for the *covariance test* case with N_i values computed from the clean images. Even in the presence of strong noise ($\sigma = 25$) the estimated bounds are quite close to the ground truth computed from clean images. The small error bars representing the standard deviations about the mean for the bounds estimates over 5 different realizations of noise illustrate the fact that the covariance estimation process is quite robust to the presence of outliers that occur due to errors in clustering.

Next, we consider the case where the bounds are calculated entirely from the noisy image. That is to say that both N_i and C_z are estimated from the noisy image. The mean of the bounds estimates obtained for various images over 5 different realizations of noise are shown in Fig. 3.12. We observe that when the noise standard deviation $\sigma \leq 15$, the bounds are estimated quite accurately from the noisy image. This

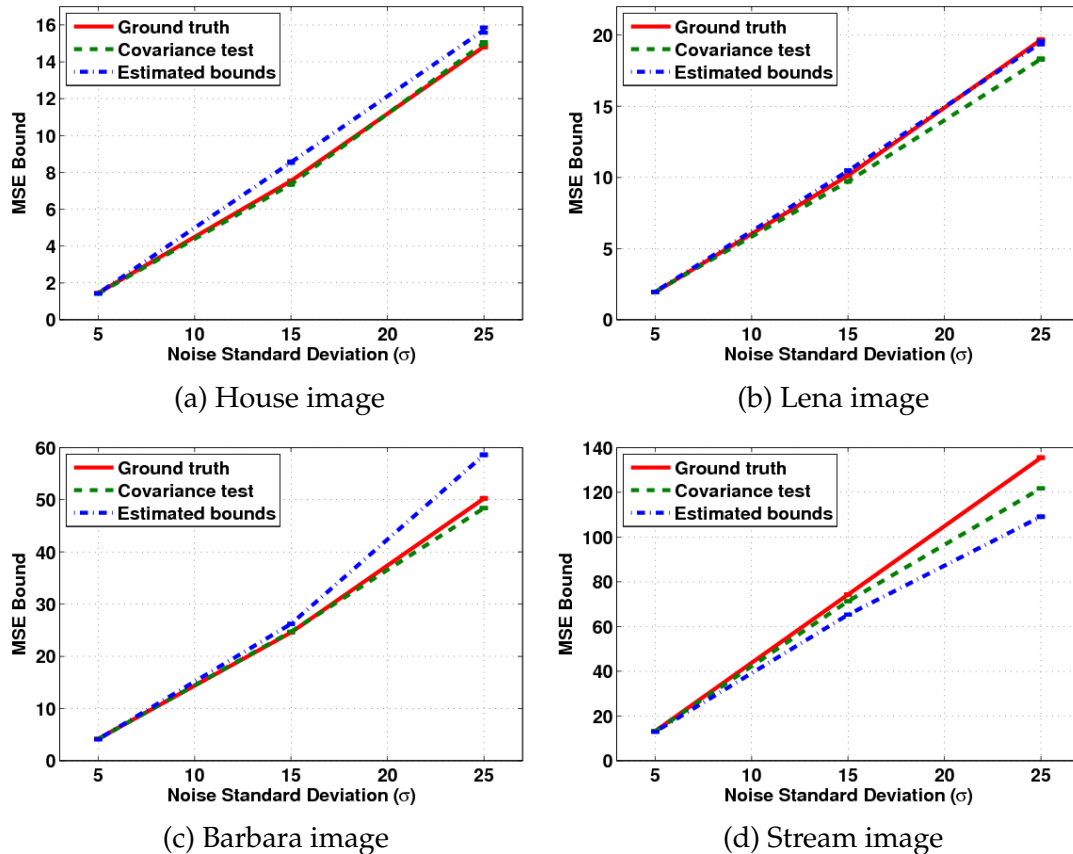


Figure 3.12: MSE bounds estimated from a given noisy image (labeled *estimated bounds*) compared to the ground truth [70] where the bounds are calculated from clean images. We also test the accuracy of covariance estimation (labeled *covariance test*) by calculating the bounds using N_i values estimated from the clean image. For all the images, the N_i estimates used to compute the *estimated bounds* are obtained directly from the noisy images for noise standard deviation $\sigma \leq 15$, and from the pre-filtered images for $\sigma = 25$.

is verified for a variety of natural images in Table 3.2 where the bounds estimated from noisy images are compared with those estimated from ground truth. However, when stronger noise is considered, our experiments indicate that the bounds estimates can be quite inaccurate. In particular, we noted that estimation of N_i is severely affected when strong noise corrupts the image. However, the same is not the case for the estimation

Table 3.2: Comparison of bounds from noisy and noise-free images considered to be ground truth. The noise is WGN with standard deviation 15. The mean bounds from 5 different realizations of noise are shown, along with standard deviations about the means in braces.

Image	Bounds from		Error
	Ground truth	Noisy image	Percentage
House	7.54	8.55 (0.042)	13.40
Peppers	9.93	9.53 (0.052)	4.03
Lena	10.13	10.55 (0.042)	4.15
Boats	19.68	19.41 (0.069)	1.37
Barbara	24.58	26.22 (0.041)	6.67
Man	33.56	28.32 (0.035)	15.61
Stream	74.30	65.25 (0.037)	12.18
Mandrill	92.56	83.78 (0.123)	9.49
Mean	34.04	31.45	7.61

of C_z . This is not surprising since the N_i values are estimated patch-wise, whereas the covariance matrices are computed from a much larger number of patches within each cluster.

Robust identification of photometrically similar patches is important for many denoising methods that rely on such redundancies to perform denoising. In fact, one of the most popular denoising algorithms, BM3D [49], performs an initial pre-filtering of highly noisy images to reduce the effects of noise before comparing patches to detect similarities. Along similar lines, we can pre-filter for our N_i estimation. However, strong denoising leads to considerable over-estimation of N_i values, especially for patches containing fine texture, resulting in considerable under-estimation of the bounds. To avoid this, we perform only mild pre-filtering in such a way so as to retain the texture in the image. For this preprocessing step we make use of the successful BM3D [49] algorithm, setting the parameter (input noise variance) of the algorithm such that the denoising process leaves behind sufficient noise so as to bring the pre-

filtered image to within effective range of the bound estimate ($\hat{\sigma} \leq 15$). In particular, using the residual of the estimate, we set the BM3D parameter so as to ensure that a noise-suppressed image is obtained for which the estimated noise standard deviation $\sigma \approx 5$ in the smoother regions of the image. Using such a method, the bounds are estimated more accurately even for images corrupted by strong noise ($\sigma = 25$), as shown in Fig. 3.12. For the case where the noise standard deviation $\sigma \leq 15$, we compute the bounds parameters directly from the noisy images. However, for the strong noise case ($\sigma = 25$) the pre-filtered images are used in estimating the N_i values. The patch covariance matrices, however, are still computed from the noisy images. The results show that using a pre-filtering step, we are able to estimate the bounds quite accurately even in the presence of strong noise.

We have thus demonstrated, with a variety of experiments, that (3.1) presents a meaningful lower bound on the performance (in terms of MSE) of denoising any given image. We also showed that the bounds can be estimated directly from any noisy image without the need for any explicit modeling of image patches. In our opinion, this makes the formulation practical and easily applicable to natural images. The bounds here were derived as that of the fundamental limits in estimating the image patches from their noisy observations. In the next chapter, we provide further analysis of the formulation and show that photometric redundancy is related to the patch covariance and these parameters, along with the bounds, have information-theoretic interpretations as well.

Summary – In this chapter we presented a method of estimating the lower bounds on denoising performance. First, we restricted ourselves to estimating the bounds from the latent image as they define the bounds on denoising. This was done through estimating the different parameters of the bound independently. These estimation processes were then generalized to account for the presence of noise in the input image for which the bounds need to be estimated. Experimentally, we showed that the formulation provides a meaningful lower bound that can be estimated directly from any noisy image. We also showed that these bounds can be used to predict the relative denoising difficulty between images and such rankings were largely in keeping with the denoising performance of current state-of-the-art denoising methods.

Chapter 4

Information Theoretic Interpretations of the MSE Bound

Abstract – In this chapter we continue our analysis of the MSE bound expression that we derived from an estimation theoretic point of view in Chapter 2. We show that the bound and its parameters have interesting information-theoretic interpretations. We also demonstrate that information-theoretic measures such as the mutual information and, in the limiting case, the entropy can be used to predict relative denoising difficulty between images that are similarly corrupted by noise.

4.1 Introduction

In Chapter 2, we derived a lower bound on the MSE for denoising any given image. The expression for the bound was derived as that of estimating the underlying noise-free patches \mathbf{z}_i from their noisy observations. We showed that the formulation is the performance of a linear MMSE estimator when the corrupting noise is known to be zero mean white Gaussian. However, such an LMMSE estimator requires us to

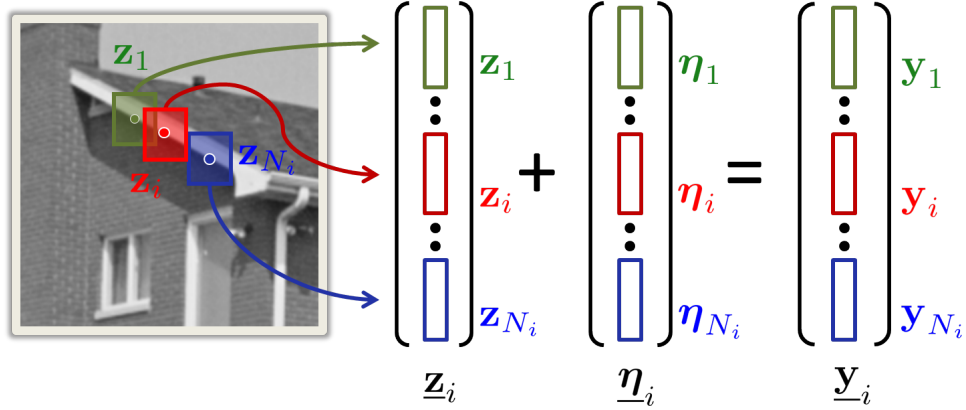


Figure 4.1: Illustration of the modified data model considering all patches that are photometrically similar to any given reference patch \mathbf{z}_i in the image.

take into account the contribution of all N_i patches that are photometrically similar to a given reference patch \mathbf{z}_i , as we shall see in Chapter 5. To derive the LMMSE estimator for this case, we first need to rewrite the patch-wise data model by considering all N_i similar patches as (see Fig. 4.1)

$$\underline{\mathbf{y}}_i = \underline{\mathbf{z}}_i + \underline{\boldsymbol{\eta}}_i, \quad (4.1)$$

where $\underline{\mathbf{y}}_i = [\mathbf{y}_1^T \dots \mathbf{y}_i^T \dots \mathbf{y}_{N_i}^T]^T \in \mathbb{R}^{nN_i \times 1}$,

$$\underline{\mathbf{z}}_i = [\mathbf{z}_1^T \dots \mathbf{z}_i^T \dots \mathbf{z}_{N_i}^T]^T \in \mathbb{R}^{nN_i \times 1},$$

$$\underline{\boldsymbol{\eta}}_i = [\boldsymbol{\eta}_1^T \dots \boldsymbol{\eta}_i^T \dots \boldsymbol{\eta}_{N_i}^T]^T \in \mathbb{R}^{nN_i \times 1}.$$

The above data model, written for each underlying patch \mathbf{z}_i , accounts for the N_i similar patches that exist for any given \mathbf{z}_i . The $\underline{\mathbf{y}}_i$ vector, as shown in Fig. 4.1, is formed by concatenating all \mathbf{y}_j vectors corresponding to the \mathbf{z}_j patches (\mathbf{z}_i included) that are similar to the reference patch \mathbf{z}_i , where similarity is defined in Eq. 3.4. Making a simplifying

assumption that these N_i patches are identical¹, we can express $\underline{\mathbf{z}}_i = \mathbf{A}_i \mathbf{z}_i$ and rewrite Eq. 4.1 as

$$\underline{\mathbf{y}}_i = \mathbf{A}_i \mathbf{z}_i + \underline{\boldsymbol{\eta}}_i, \quad (4.2)$$

where $\mathbf{A}_i = [\mathbf{I} \dots \mathbf{I}]^T \in \mathbb{R}^{nN_i \times n}$ with \mathbf{I} denoting the $n \times n$ identity matrix. The corresponding noise patch $\underline{\boldsymbol{\eta}}_i$ formed from independent $\boldsymbol{\eta}_j$ vectors then has covariance $\mathbf{C}_{\underline{\boldsymbol{\eta}}} = \sigma^2 \mathbf{I}_{nN_i}$ where \mathbf{I}_{nN_i} is the $nN_i \times nN_i$ identity matrix. The LMMSE estimator for each \mathbf{z}_i then has the form $\hat{\mathbf{z}}_i = E[\mathbf{z}_i | \underline{\mathbf{y}}_i]$. This estimator has zero expected error ($E[\mathbf{z}_i - \hat{\mathbf{z}}_i] = \mathbf{0}$) and an error covariance (discussed in more detail in Sec. 5.2)

$$\begin{aligned} \mathbf{Q}_i &= E[(\mathbf{z}_i - \hat{\mathbf{z}}_i)(\mathbf{z}_i - \hat{\mathbf{z}}_i)^T] = (\mathbf{C}_{\mathbf{z}}^{-1} + \sigma^{-2} \mathbf{A}_i^T \mathbf{I}_{nN_i} \mathbf{A}_i)^{-1} \\ &= \left(\mathbf{C}_{\mathbf{z}}^{-1} + N_i \frac{\mathbf{I}}{\sigma^2} \right)^{-1}. \end{aligned} \quad (4.3)$$

Note that the error covariance above is the MSE of the LMMSE estimate. Moreover, comparing to Equations 2.18 & 2.8, we observe that the trace of \mathbf{Q}_i is, in fact, the lower bound derived for the denoising problem, assuming WGN. The error covariance matrix \mathbf{Q}_i above is, thus, the MMSE matrix for the estimation of \mathbf{z}_i .

Interestingly, although derived purely from an estimation theoretic point of view, the MSE bounds for denoising can be shown to be related to information-theoretic measures. In Sec. 4.3, we show that, in the limiting case when considering noise-free ground truth images, the expression for the MSE bound is related to the entropy of the image. Further, in such cases, the parameters of the bound (N_i and $\mathbf{C}_{\mathbf{z}}$) that we estimated independently in the previous chapter, can be shown to be inter-dependent

¹By identical we mean $\mathbf{z}_i = \mathbf{z}_j$. The expression for the FIM in Eq. 2.21 makes an implicit assumption of N_i *identical* patches being present in the image, although in practice only *similar* patches satisfying Eq. 3.4 may actually exist. Later, in Chapter 5, we derive an LMMSE estimator for a more practical data model where this *identical* constraint is relaxed and the underlying patches are considered to be *similar*.

through their relation with the entropy. However, we start our study of such information-theoretic interpretations with the more practical case when the input image is corrupted by noise. In such cases, we can show that the MMSE matrix of Eq. 4.3 is related to the mutual information of the noisy \mathbf{y} and noise-free \mathbf{z} patches.

4.2 Denoising Bounds and Mutual Information

The mutual information (MI) of random variables \mathbf{y} and \mathbf{z} is a measure of the information that one variable contains about the other. Considering the patch-based data model of Eq. 2.1, the MI can be mathematically expressed as [115]

$$\begin{aligned}
 I(\mathbf{y}; \mathbf{z}) &= H(\mathbf{y}) - H(\mathbf{y}|\mathbf{z}) \\
 &= H(\mathbf{y}) - H(\mathbf{z} + \boldsymbol{\eta}|\mathbf{z}) \quad [\because \mathbf{y} = \mathbf{z} + \boldsymbol{\eta} \quad (\text{Eq. 2.1})] \\
 &= H(\mathbf{y}) - H(\boldsymbol{\eta}|\mathbf{z}) \\
 &= H(\mathbf{y}) - H(\boldsymbol{\eta}), \tag{4.4}
 \end{aligned}$$

where $H(\mathbf{y})$ and $H(\boldsymbol{\eta})$ denote the entropy of \mathbf{y} and the noise $\boldsymbol{\eta}$ respectively². The entropy of a random variable \mathbf{y} (or equivalently its pdf $p(\mathbf{y})$) is defined as

$$H(\mathbf{y}) = -E[\ln p(\mathbf{y})] = - \int p(\mathbf{y}) \ln p(\mathbf{y}) d\mathbf{y}. \tag{4.5}$$

While $H(\mathbf{y})$ can be estimated from the observed noisy image patches (see Appendix 4A), the noise entropy may be analytically calculated if the noise statistics are known. Specifically, for Gaussian noise with given covariance \mathbf{C}_η , the entropy is given by

$$H(\boldsymbol{\eta}) = \ln \left[(2\pi e)^{n/2} |\mathbf{C}_\eta|^{1/2} \right] = \frac{1}{2} \ln(|\mathbf{C}_\eta|) + \frac{n}{2} [1 + \ln(2\pi)], \tag{4.6}$$

²We will alternately denote the entropy of a random variable \mathbf{x} with a pdf $p(\mathbf{x})$ as $H(p)$ or $H(\mathbf{x})$, as necessary for clarity of presentation. The notation used will be clear from context.

where $|\cdot|$ denotes the determinant.

In [116], Palomar *et al.* studied the relationship of the mutual information between the noisy and noise-free image patches and the minimum mean squared error (MMSE) on the estimation of the input given the output of a Gaussian channel. For the multivariate case (Eq. 4.2), the authors show that the gradients of the MI with respect to the signal and noise covariance can be written in terms of the MMSE matrix of Eq. 4.3 as

$$\frac{d}{d\mathbf{C}_z} I(\mathbf{y}_i; \mathbf{z}_i) \mathbf{C}_z = \mathbf{A}_i^T \mathbf{C}_\eta^{-1} \mathbf{A}_i \mathbf{Q}_i, \quad \text{and} \quad (4.7)$$

$$\frac{d}{d\mathbf{C}_\eta} I(\mathbf{y}; \mathbf{z}) = -\mathbf{C}_\eta^{-1} \mathbf{A}_i \mathbf{Q}_i \mathbf{A}_i^T \mathbf{C}_\eta^{-1}. \quad (4.8)$$

When dealing with iid noise, where $\mathbf{C}_\eta = \sigma^2 \mathbf{I}_{nN_i}$, the above relations can be written for each cluster as (see Appendix 4B for derivation)

$$\frac{d}{d\mathbf{C}_z} I(\mathbf{y}; \mathbf{z}) \mathbf{C}_z = \frac{1}{M_k} \sum_{i=1}^{M_k} \frac{d}{d\mathbf{C}_z} I(\mathbf{y}_i; \mathbf{z}_i) \mathbf{C}_z = \frac{1}{M_k} \sum_{i=1}^{M_k} \frac{N_i}{\sigma^2} \mathbf{Q}_i, \quad \text{and} \quad (4.9)$$

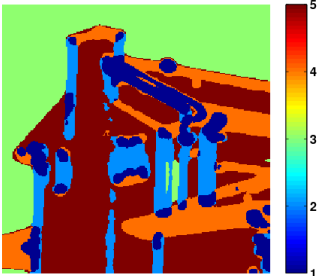
$$\frac{d}{d\sigma^2} I(\mathbf{y}; \mathbf{z}) = \frac{1}{M_k} \sum_{i=1}^{M_k} \frac{d}{d\sigma^2} I(\mathbf{y}_i; \mathbf{z}_i) = -\frac{1}{nM_k\sigma^4} \sum_{i=1}^{M_k} \text{Tr}(\mathbf{Q}_i). \quad (4.10)$$

These then establish direct relationships between the denoising bound of (2.24) and the MI (Eq. 4.4). As with the bound, the MI too is a function of both the input signal characteristics and the noise. This can be seen by further expansion of Eq. 4.9 as

$$\begin{aligned} \frac{d}{d\mathbf{C}_z} I(\mathbf{y}; \mathbf{z}) \mathbf{C}_z &= \frac{1}{M_k} \sum_{i=1}^{M_k} \frac{N_i}{\sigma^2} \left(\mathbf{C}_z^{-1} + N_i \frac{\mathbf{I}}{\sigma^2} \right)^{-1} \\ \Rightarrow \frac{d}{d\mathbf{C}_z} I(\mathbf{y}; \mathbf{z}) &= \frac{1}{M_k \sigma^2} \sum_{i=1}^{M_k} N_i \left(\mathbf{C}_z^{-1} + N_i \frac{\mathbf{I}}{\sigma^2} \right)^{-1} \mathbf{C}_z^{-1} \\ &= \frac{1}{M_k \sigma^2} \sum_{i=1}^{M_k} \left(\frac{\mathbf{I}}{N_i} + \frac{\mathbf{C}_z}{\sigma^2} \right)^{-1}, \end{aligned} \quad (4.11)$$

assuming invertibility of \mathbf{C}_z . A positive definite gradient with respect to the covariance here implies that the MI is an increasing function of patch complexity. Further, as patch

Table 4.1: Clustering of the house image and the cluster-wise mutual information estimates when corrupted by various levels of WGN.



σ	Noise Entropy	Estimated MI $\hat{I}(\mathbf{y}; \mathbf{z})$					Overall MI
		Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	
0	-	430.76	365.15	212.71	375.09	345.91	322.49
5	366.43	119.25	80.53	27.88	87.00	60.52	63.93
15	499.37	62.94	43.17	21.63	47.31	30.59	37.64
25	561.18	45.77	33.50	20.35	36.46	25.03	33.05
35	601.89	35.58	27.92	17.81	30.36	22.31	23.97
45	632.30	27.80	23.42	13.47	25.18	19.55	19.74
55	656.58	21.07	19.06	8.59	19.97	16.13	15.23

complexity captured by \mathbf{C}_z increases (with a corresponding drop in the expected N_i , as we shall see later in Sec. 4.3), the magnitude of the gradient decreases. This implies that the rate of increase of MI drops as the underlying patch complexity increases. However, with increase in noise strength, the MI can be expected to decrease, as is implied by the negative gradient of the MI with respect to the noise variance in Eq. 4.10.

Using Eq. 4.3 to expand Eq. 4.10 as

$$\begin{aligned}
 \frac{d}{d\sigma^2} I(\mathbf{y}; \mathbf{z}) &= -\frac{1}{nM_k} \sum_{i=1}^{M_k} \text{Tr} \left[\frac{1}{\sigma^4} \left(\mathbf{C}_z^{-1} + N_i \frac{\mathbf{I}}{\sigma^2} \right)^{-1} \right] \\
 &= -\frac{1}{nM_k} \sum_{i=1}^{M_k} \text{Tr} \left[\left(\sigma^4 \mathbf{C}_z^{-1} + \sigma^2 N_i \mathbf{I} \right)^{-1} \right], \quad (4.12)
 \end{aligned}$$

we can see that the rate of such decrease is also expected to drop as the noise strength increases.

We study this behavior of the mutual entropy as a function of the noise strength and patch complexity through a simple experiment. For this, we make use of the House image and estimate the MI $\hat{I}(\mathbf{y}; \mathbf{z})$ for each cluster containing geometrically similar patches (color-coded in Table 4.1) for various levels of additive WGN. For meaningful comparisons, we perform clustering on the noise-free image and use the same cluster membership in computing the MI estimates for the noisy cases. In Table 4.1, cluster Ω_3

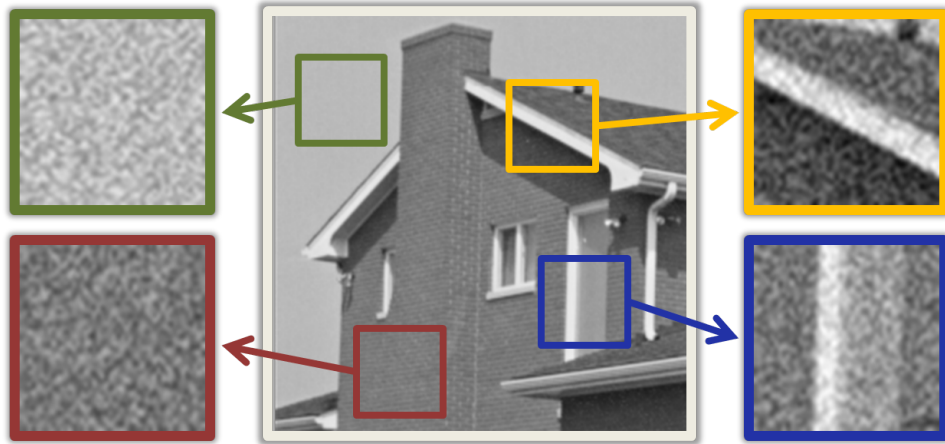


Figure 4.2: Effect of noise on different parts of the House image. Note how in regions of strong edges the underlying image content is visually discernible even when corrupted by strong noise, whereas regions lacking strong structure are largely indistinguishable from noise.

consisting of the much smoother background patches has a much lower complexity than that of clusters Ω_2 and Ω_4 which capture the edge regions. This relative complexity is also captured by the MI estimates for the clusters (see Table 4.1) as illustrated by clusters with higher complexity exhibiting higher MI. This is in keeping with Eq. 4.11 which implies an increase of MI with increase in patch complexity. Further, the MI of each cluster is decreasing as the noise increases, and the rate of such a decrease also drops with higher noise. This is in keeping with the relationship between the MI and noise variance captured by Eq. 4.12.

Although the MI is clearly related to the parameters of the bound, it is important to note that one cannot be used directly to predict the other. The formulation of Eq. 2.24 predicts an increase in the MSE bound for denoising as the image complexity and noise variance increases. However, the MI, which quantifies the relative information between a noisy patch y and its noise-free counterpart z , increases with increasing image complexity, but has quite the opposite effect as noise variance increases. This

is in keeping with intuition that as noise increases the noisy patches increasingly resemble noise, resulting in a reduction of information that \mathbf{y} conveys about \mathbf{z} (and vice versa). However, as the complexity of the noise-free patch increases, stronger noise is needed for the noise to overwhelm the patch characteristics (see Fig. 4.2), thus justifying an increase in MI. These relations are captured analytically in Equations 4.11 and 4.12 and experimentally in Table 4.1. It is also important to note that Equations 4.11 and 4.12 relate the *rate of change* of MI as a function of changing image complexity and noise variance respectively. Thus, it is the magnitude of the rate of change of MI (and not the MI itself) that is inversely related to the bounds. Consequently, with only a single noisy observation the MI cannot be used to predict the denoising bound. However, the MI can be used to study the relative denoising difficulty of different images that are corrupted by similar levels of noise.

As before, we consider additive Gaussian noise to illustrate the effectiveness of the MI measure in studying relative complexity of images containing patches of diverse geometric structure. For this we need to first estimate the entropy of the entire noisy image from its cluster-wise entropy estimates as (see Appendix 4C)

$$H(\mathbf{y}) = \sum_{k=1}^K \omega_k H(\mathbf{y} \in \Omega_k) - \sum_{k=1}^K \omega_k \ln \omega_k, \quad (4.13)$$

where $\omega_k = M_k/M$ is the fraction of total patches that belong to cluster Ω_k . For WGN, the noise entropy is calculated analytically (Eq. 4.6) using the known noise covariance matrix. The overall MI can then be estimated using Eq. 4.4. In Table 4.2, we show the estimated mutual information obtained for some images (Fig. 1.7) when corrupted by WGN of different strengths. The MI there is indicative of the relative denoising difficulty between images. This can be seen by comparing it to the relative ranking obtained by the MSE of one of the best performing denoising methods (namely, BM3D

Table 4.2: Ranking of images based on denoising difficulty as indicated by the MI, compared to the entropy, the denoising bound and MSE of BM3D denoising algorithm for WGN.

Images	Size	Noise-free Entropy	Mutual Information $I(y; z)$			Denoising Bounds [70]	BM3D MSE [49]
			$\sigma = 5$	$\sigma = 15$	$\sigma = 25$		
House	256 ²	322.49	63.92	37.65	33.06	14.82	33.57
Lena	512 ²	350.17	67.39	38.55	31.88	19.66	40.46
Peppers	512 ²	374.29	72.56	38.37	30.53	19.21	42.96
Barbara	512 ²	376.74	89.32	49.95	37.81	50.24	55.62
Boats	512 ²	398.36	89.75	45.75	35.04	38.70	67.17
Man	512 ²	407.16	94.28	43.49	29.25	62.97	96.46
Stream	512 ²	473.65	136.18	63.67	43.52	135.46	158.26
Mandrill	512 ²	498.75	153.67	74.50	51.59	181.61	185.60

[49]). In fact, this ranking (for $\sigma = 5$) is more in keeping with the relative denoising difficulty exhibited by the practical methods than that obtained from the denoising bounds calculated from the clean images.

In the limiting case when the image is noise-free, the mutual information becomes the same as the Shannon entropy of the noise-free image.³ In Table 4.2 we show that the relative denoising difficulty prediction of the entropy in that case is also in keeping with those obtained by the MI and the MSE of BM3D. This indicates that the entropy of the image is also related to the denoising bounds. In the next section, we explore this relationship further.

4.3 Relationship between Denoising Bounds and Entropy

The bounds formulation of Eq. 3.1 depends on two parameters, namely the FIM \mathbf{J}_i and the covariance matrix \mathbf{C}_z that corresponds to the cluster of which patch \mathbf{z}_i is a member. For WGN, estimating the FIM amounts to estimating the number (N_i) of

³Images considered to be “noise-free” can often contain noise as well [101]. However, the noise in such images is typically quite small and, hence, we consider them to be noise-free in our study.

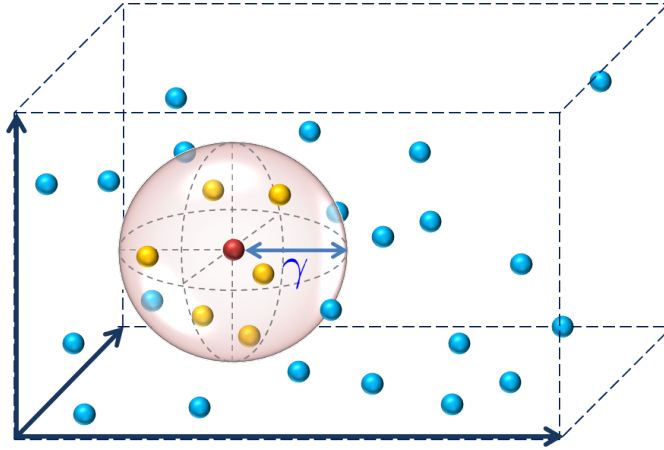


Figure 4.3: Density estimation of points sampled from an unknown pdf at a reference point (in red). N_i is the number of (orange) samples that lie within the ball of radius γ whereas the blue samples lie outside. The pdf at the reference can then be approximately evaluated by Eq. 4.14.

similar patches that exist for each patch \mathbf{z}_i , assuming accurate knowledge of the noise variance. In general, one can expect to find fewer similar patches in any given image if the variability between patches within a cluster is high. In Sec. 3.2 (and [70]), both these parameters were estimated from the noise-free image, in which case the MI of Eq. 4.4 reduces to the Shannon entropy of the noise-free image. In this section, we establish how each of these two parameters of the bounds formulation are related to the Shannon entropy, and as a result, to each other. Our interest here lies solely in analyzing the information-theoretic interpretations of the parameters and relating the two. As such, this relationship between the two does not translate to one being estimated from the other in practice, as will be apparent from the following discussions.

In Sec. 3.4, N_i is estimated for each patch $\mathbf{z}_i \in \Omega_k$ by searching over the entire image. This ensured that errors in clustering did not result in ignoring any photometrically similar patches. Assuming oracle clustering, one can then expect to obtain a good estimate of N_i by limiting the search for patches similar to any given $\mathbf{z}_i \in \Omega_k$ to

patches within the same cluster Ω_k . Let N_i then denote the number of similar patches that lie within the cluster Ω_k , where similarity is defined in Eq. 3.4. We then approximate N_i by performing a nearest neighbor search within patches in Ω_k with a search radius of γ . Considering $\mathbf{z}_i \in \mathbb{R}^n$, an estimate of the N_i -nearest neighbor probability density function can then be calculated as the fraction of total patches that are present within a ball of radius γ centered at \mathbf{z}_i , as shown in Fig. 4.3. The pdf at \mathbf{z}_i can then be approximated as [117]

$$p_k(\mathbf{z}_i) \approx \frac{N_i/(M_k - 1)}{V_i(\gamma)} = \frac{N_i}{(M_k - 1) \nu_n \gamma^n} = \frac{N_i \Gamma(1 + n/2)}{(M_k - 1) \pi^{n/2} \gamma^n}, \quad (4.14)$$

where $V_i(\gamma)$ is the volume of the ball centered at \mathbf{z}_i with radius γ and ν_n is the volume of the unit ball in \mathbb{R}^n . Solving for N_i we have

$$N_i \approx \frac{(M_k - 1) \pi^{n/2} \gamma^n}{\Gamma(n/2 + 1)} p_k(\mathbf{z}_i), \quad (4.15)$$

where $\Gamma(\cdot)$ denotes the Gamma function. Unfortunately, the relation of Eq. 4.14 is accurate only when a considerably large number of patches are present [117]. This is especially true when considering high dimensions (e.g. $n = 121$ arising from choosing patch sizes of 11×11 which have been shown in Sec. 3.4 to be a good choice for obtaining meaningful bounds.) Moreover, this requires us to know or estimate the multivariate pdf $p_k(\mathbf{z})$. However, Eq. 4.14 is still useful as it establishes a relation between the pdf $p_k(\mathbf{z})$ and the number of similar patches that exist within the cluster Ω_k .

We now extend the relationship of Eq. 4.15 by considering the average patch redundancy level within each cluster. Let $\bar{N}(k) = E[N_i \in \Omega_k | \gamma]$ be the conditional expected value of N_i for patches within the k -th cluster for a given value of γ , with the expectation taken over $\mathbf{z} \in \Omega_k$. From Eq. 4.15, we can then express $\bar{N}(k)$ as

$$\bar{N}(k) = E \left[\frac{(M_k - 1) \pi^{n/2} \gamma^n}{\Gamma(n/2 + 1)} p_k(\mathbf{z}_i) \right] = \frac{(M_k - 1) \pi^{n/2} \gamma^n}{\Gamma(n/2 + 1)} \int [p_k(\mathbf{z})]^2 d\mathbf{z}. \quad (4.16)$$

Interestingly, $\bar{N}(k)$ here is related to the Rényi α -entropy [118] which is defined as

$$R_\alpha(p_k) = \frac{1}{1-\alpha} \ln \left(\int p_k(\mathbf{z})^\alpha d\mathbf{z} \right). \quad (4.17)$$

Choosing $\alpha = 2$, we can then express Eq. 4.16 as

$$\begin{aligned} \ln(\bar{N}(k)) &= \ln \left(\frac{(M_k - 1) \pi^{n/2} \gamma^n}{\Gamma(n/2 + 1)} \right) - \left[-\ln \left(\int p_k(\mathbf{z})^2 d\mathbf{z} \right) \right] \\ &= \ln \left(\frac{(M_k - 1) \pi^{n/2} \gamma^n}{\Gamma(n/2 + 1)} \right) - R_2(p_k). \end{aligned} \quad (4.18)$$

This provides a relationship between $\bar{N}(k)$ and the Rényi entropy. Namely, as the Rényi entropy increases, the expected number of similar patches within a cluster decreases. The Rényi entropy being a measure of uncertainty of a random variable, Eq. 4.18 then fits with the intuition of lower patch redundancy in clusters with more complicated structure.

Alternately, we can think of the level of redundancy within any cluster to be measured by the mean distance from any patch to its most similar patch (nearest neighbor). An overall smaller distance would then indicate the presence of a larger number of similar patches. Generalizing this alternate measure by considering the distance to the $\tilde{N}(k)$ -most similar patch, one can then expect a smaller average distance for clusters exhibiting higher levels of redundancy for any fixed $\tilde{N}(k)$. Denoting $\gamma_{i, \tilde{N}(k)}$ as the distance from \mathbf{z}_i to its $\tilde{N}(k)$ -th nearest neighbor in Ω_k , we express the (conditional) mean distance to the $\tilde{N}(k)$ -th nearest neighbor using Eq. 4.15 as

$$\begin{aligned} E[\gamma_{i, \tilde{N}(k)} | \tilde{N}(k)] &= E \left[\left\{ \frac{\tilde{N}(k) \Gamma(1 + n/2)}{(M_k - 1) \pi^{n/2} p_k(\mathbf{z}_i)} \right\}^{1/n} \right] \\ &= \left(\frac{\tilde{N}(k) \Gamma(1 + n/2)}{(M_k - 1) \pi^{n/2}} \right)^{1/n} \int p_k(\mathbf{z})^{(1-\frac{1}{n})} d\mathbf{z}, \end{aligned} \quad (4.19)$$

where, as before, the expectation is taken over $\mathbf{z} \in \Omega_k$. Evans *et al.* [119] derived a more general expression for the distance to the $\tilde{N}(k)$ -th nearest neighbor as

$$E[\gamma_{i,\tilde{N}(k)}|\tilde{N}(k)] = \frac{\Gamma(\tilde{N}(k) + \frac{1}{n})}{[\nu_n(M_k - 1)]^{1/n}\Gamma(\tilde{N}(k))} \int p_k(\mathbf{z})^{(1-\frac{1}{n})} d\mathbf{z}, \quad (4.20)$$

where using the approximation

$$\frac{\Gamma(X + \frac{1}{n})}{\Gamma(X)} \approx X^{\frac{1}{n}} \quad (4.21)$$

one obtains the same relation as in Eq. 4.19. Note that in our case, we consider a fixed search radius of γ which is chosen independent of the image patches. Hence, we set $E[\gamma_{i,\tilde{N}(k)}|\tilde{N}(k)] = \gamma$ and evaluate the corresponding $\tilde{N}(k)$ for which the mean $\tilde{N}(k)$ -nearest neighbor distance is γ . We are, thus, interested in determining the value of $\tilde{N}(k)$ for which the mean distance to the $\tilde{N}(k)$ -nearest patch is γ . Intuitively, we can then expect a larger $\tilde{N}(k)$ for clusters with relatively simpler patches that are known to exhibit higher redundancy levels. Denoting

$$I_n(p_k) = \int p_k(\mathbf{z})^{(1-\frac{1}{n})} d\mathbf{z}, \quad (4.22)$$

we can then rewrite Eq. 4.19 as

$$\begin{aligned} \gamma &= \left(\frac{\tilde{N}(k) \Gamma(\frac{n}{2} + 1)}{(M_k - 1)} \right)^{1/n} \frac{I_n(p_k)}{\sqrt{\pi}} \\ \Rightarrow \tilde{N}(k) &= \frac{(M_k - 1)}{\Gamma(\frac{n}{2} + 1)} \left(\frac{\sqrt{\pi}\gamma}{I_n(p_k)} \right)^n. \end{aligned} \quad (4.23)$$

From Eq. 4.23, we see that the expected number of similar patches that exist within the given cluster is directly proportional to the total number of member patches and the radius of the ball of similarity; and inversely proportional to the n -th power of the integral $I_n(p_k)$. Eq. 4.17 shows that $I_n(p_k)$ is directly related to the Rényi entropy for

the pdf $p_k(\mathbf{z})$, where now $\alpha = (1 - \frac{1}{n}) < 1$. Denoting the Rényi entropy for this choice of α as $R_n(p_k)$, we obtain

$$\begin{aligned} R_n(p_k) &= \frac{1}{1 - (1 - \frac{1}{n})} \ln(I_n(p_k)) = n \ln(I_n(p_k)) \\ \Rightarrow \ln(\tilde{N}(k)) &= \ln\left(\frac{M_k - 1}{\Gamma(\frac{n}{2} + 1)}\right) + n \ln\left(\frac{\sqrt{\pi}\gamma}{I_n(p_k)}\right) \\ &= \ln\left(\frac{M_k - 1}{\Gamma(\frac{n}{2} + 1)}\right) + n \ln(\sqrt{\pi}\gamma) - R_n(p_k). \end{aligned} \quad (4.24)$$

Eq. 4.24 thus provides a direct relationship between the number of γ -similar patches that can be expected for patches within any given cluster, and the Rényi entropy for that cluster. We can then relate $\tilde{N}(k)$ to the Shannon entropy [120] by using the fact that as $\alpha \rightarrow 1$, the Rényi entropy closely approximates the Shannon entropy. For large n (such as $n = 121$), we obtain a value of $\alpha = (1 - \frac{1}{n}) \approx 0.992$ which is quite close to 1. Substituting the Shannon entropy, $H(p_k)$, for the Rényi entropy, we obtain a relation between $\tilde{N}(k)$ and $H(p_k)$ as

$$\ln(\tilde{N}(k)) \approx \ln\left(\frac{M_k - 1}{\Gamma(\frac{n}{2} + 1)}\right) + n \ln(\sqrt{\pi}\gamma) - H(p_k). \quad (4.25)$$

The higher the variability of patches within a cluster, the higher is its entropy. Keeping with intuition, Eq. 4.25 predicts an inverse relationship between the number of similar patches and the entropy of the cluster being considered. That is to say, when the entropy of \mathbf{z} within a particular cluster is high, a lower level of redundancy can be expected from the image patches. This is illustrated in Fig. 4.4 where for clusters lacking complex structure (for example, the background region), the average N_i tends to be higher than those containing patches of more complicated patterns.

The entropy of a pdf is also dependent on the second order moment that captures the variability between patches within a cluster. This relationship has been

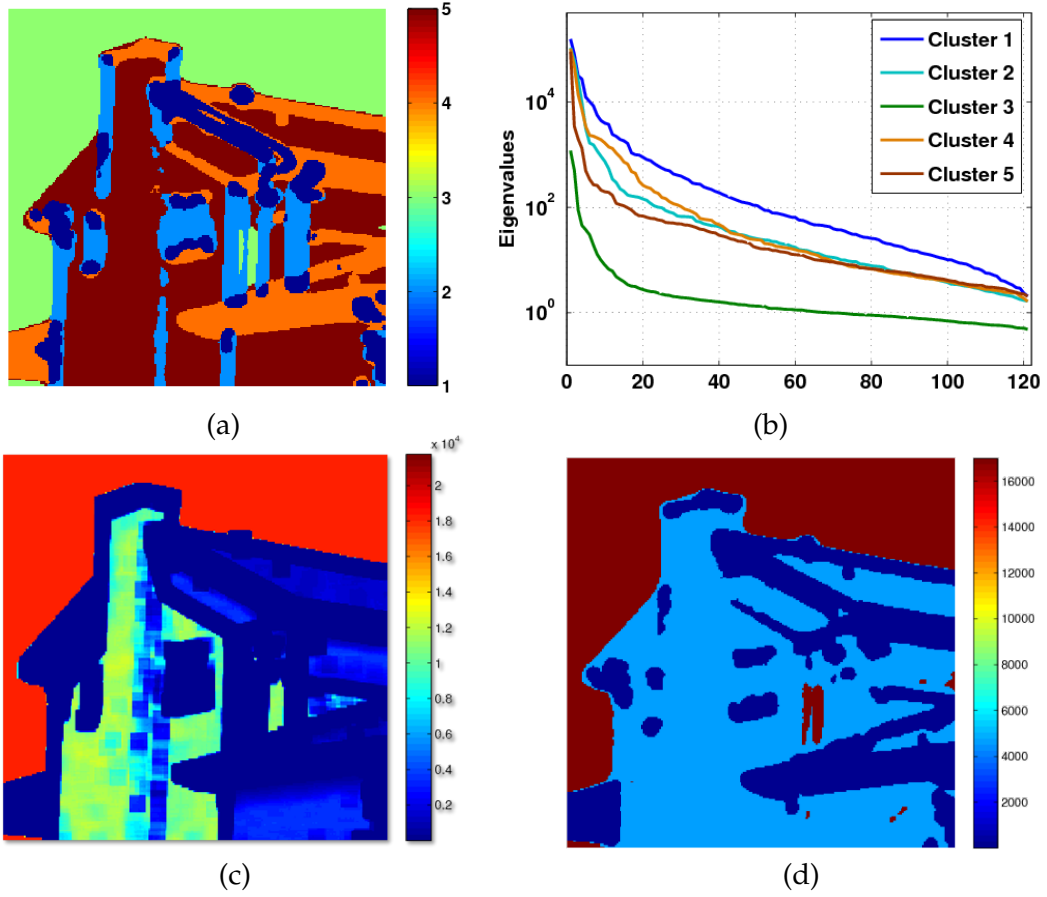


Figure 4.4: Illustration of the relation between N_i and cluster complexity: (a) Clustering of House image based on patch structure, (b) eigenvalues of \mathbf{C}_z for each cluster, (c) spatial distribution of N_i , and (d) cluster-wise average N_i . Note how the smoother background (Ω_3) has a higher average N_i than the finely textured facade (Ω_5) or the edge regions of Ω_4 . The eigenvalues of \mathbf{C}_z , on the other hand, are larger for the more complex clusters.

documented for many of the most popularly used multivariate density functions by Zografos *et al.* [121]. Specifically, for the entropy maximizing n -dimensional Gaussian density function $\mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$, the entropy can be expressed as a function of the covariance \mathbf{C} (see Eq. 4.6). Further, an increase in entropy corresponds to the existence of fewer similar patches (lower \tilde{N}_k in Eq. 4.25). Plugging this (maximum) entropy for the Gaussian pdf into Eq. 4.23 then provides us with an estimate of the minimum number of

similar patches within an expected distance γ as

$$\tilde{N}_{\min}(k) = \frac{(M_k - 1)\gamma^n}{(2e)^{\frac{n}{2}}\Gamma(\frac{n}{2} + 1)|\mathbf{C}|^{\frac{1}{2}}}. \quad (4.26)$$

This $\tilde{N}_{\min}(k)$ can then be taken to be the lower bound on $\tilde{N}(k)$ that can be expected for any cluster with a covariance \mathbf{C} . Eq. 4.26 also indicates that as the variance of the Gaussian increases in any of the n dimensions, the minimum number of similar patches that can be expected decreases. Further note that the redundancy measure is not dependent on the mean $\boldsymbol{\mu}$, which implies independence of the $\tilde{N}_{\min}(k)$ value from the mean intensity of the patches within the cluster.

For the case of any general (unknown) pdf $p_k(\mathbf{z})$, Eq. 4.25 establishes a relation between the number of similar patches that one can expect in a cluster and the corresponding covariance matrix \mathbf{C}_z that captures the cluster complexity. Eq. 4.25 provides the useful insight that the bounds formulation of (2.24) and the entropy are similarly related to patch redundancy and cluster complexity. Thus, the entropy can serve as a measure of denoising complexity when noise-free images are considered. Ranking of images according to their entropies is consistent with the relative denoising performance by practical methods (example, BM3D [49]), as shown in Table 4.2.

Although predominantly devised to support the theoretical analysis of the bounds formulation, our experiments point to some useful practical applications of the entropy measures by exploiting the relationship between the denoising bounds and the mutual information. Namely, the entropy (more generally, MI) can be used as indicators of the relative performance of denoising that one can hope to obtain for different images. In practice, this can be used to automatically set parameters in a denoising framework to control the level of smoothing required based on image content and the level of noise corruption. In general, such tasks can also be performed using

the bounds estimated from the noisy image. However, the entropy based approach can be computed faster and has been shown in Table 4.2 to be better representative of the practical difficulties in denoising any given noisy image.

Summary – In this chapter we presented further theoretical analysis of the bound formulation by relating it to information measures, namely mutual information and entropy. For noisy images, we established that the MSE bound is related to the mutual information between the noisy and the corresponding noise-free image. In the limiting case when clean images are considered, the mutual information reduces to the Shannon entropy of the image. We demonstrated the relation between the entropy and the parameters of the bound. In the process, we also established a connection between the two parameters.

4A Entropy Estimation

As mentioned earlier, the entropy for any given cluster is related to its complexity, and can, therefore, serve as a measure of denoising difficulty for that cluster. The entropy could be calculated if the prior pdf $p_k(\mathbf{z})$ could be ascertained or modeled accurately at all $\mathbf{z}_i \in \Omega_k$. Although many have proposed various models for natural images [93–95], they are not directly applicable to our case since we consider the patch vectors to be *geometrically* similar within each cluster. As a result, we need to *estimate* the entropy in each cluster from the available \mathbf{z}_i vectors. For this we make use of order statistics of the nearest-neighbor distances. Let $\gamma_{i,N}$ denote the distance between the

patch \mathbf{z}_i and its N -most similar (“nearest”) patch. We then obtain a set of $\gamma_{i,N}$ measures for $i = 1, \dots, M_k$ patches in the k -th cluster. An estimator for the entropy can then be obtained by using Eq. 4.25 for a fixed N as

$$\begin{aligned}\hat{H}_N(p_k) &= -\frac{1}{M_k} \sum_{i=1}^{M_k} \ln(\hat{p}_k(\mathbf{z}_i)) \\ &= \frac{1}{M_k} \sum_{i=1}^{M_k} \ln \left[\frac{(M_k - 1) \pi^{n/2} \gamma_{i,N}^n}{N \Gamma(1 + n/2)} \right].\end{aligned}\quad (4.27)$$

Such an estimator based on the nearest neighbor distance ($N = 1$) with added bias correction terms was proposed by Kozachenko *et al.* [122] as

$$\hat{H}_1(p_k) = \frac{1}{M_k} \sum_{i=1}^{M_k} \ln \left[\frac{(M_k - 1) \pi^{n/2} \gamma_{i,1}^n}{\Gamma(1 + n/2)} \right] + \psi,$$

where $\psi \approx 0.5772$ is the Euler constant. This was later extended by considering N -nearest neighbor distances in [123–125] (see also [126]) where the generalized bias-compensated entropy estimator takes the form

$$\begin{aligned}\hat{H}_N(p_k) &= \frac{1}{M_k} \sum_{i=1}^{M_k} \ln \left[(M_k - 1) \nu_n \gamma_{i,N}^n \right] - \Psi(N) \\ &= \frac{1}{M_k} \sum_{i=1}^{M_k} \ln \left[\frac{(M_k - 1) \pi^{n/2} \gamma_{i,N}^n}{\Gamma(1 + n/2)} \right] - \Psi(N).\end{aligned}\quad (4.28)$$

Here $\Psi(N) = \frac{d}{dN} \ln \Gamma(N)$ is the digamma function.

The entropy estimator of Eq. 4.28 is based on the N -th most similar patch and, thus, can vary with the choice of N . Estimators of the Shannon (and Rényi) entropy based on a combination of such estimates obtained using multiple values of N have been proposed in [127, 128]. However, such estimators require computation of distances to the N -most similar patches for each patch in the cluster, a process that can be quite time consuming. Instead, we make use of only the most similar patch, that is $N = 1$. In that case, the digamma function $\Psi(1) = -\psi$. The entropy estimate obtained

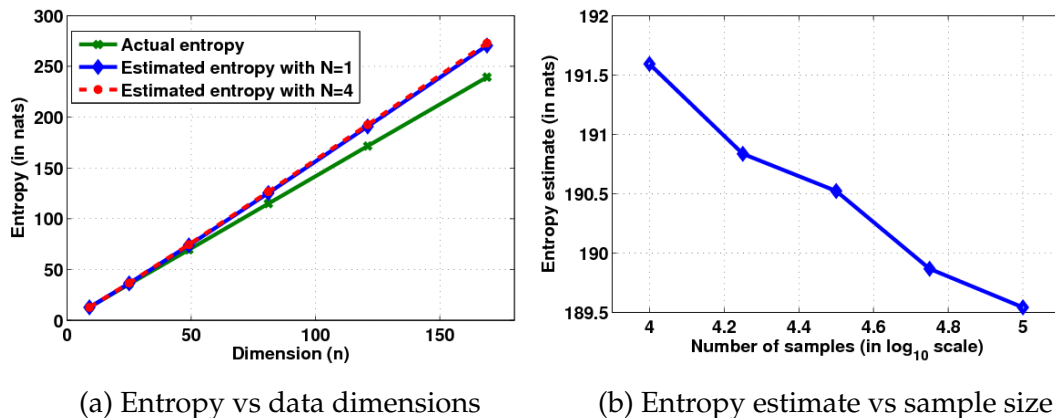


Figure 4.5: Estimation of entropy (Eq. 4.28) for data sampled from a multidimensional Gaussian density function $\mathcal{N}(\mathbf{0}, \mathbf{I})$ as a function of : (a) dimensions with 20,000 samples, and (b) number of samples with $n = 121$, where actual entropy is 171.64. These show that the nearest neighbor entropy estimate ($N = 1$) achieves a slightly better estimate of the entropy than using $N = 4$. Moreover, the entropy estimates are more accurate for lower dimensions. However, the estimate gets better as the number of samples increases.

using only the distance to the most similar patch is very similar to that obtained using larger N for the high dimensional case. We show this in Fig. 4.5(a) where we plot the entropy estimates obtained with different values of N for samples from Gaussian density functions $\mathcal{N}(\mathbf{0}, \mathbf{I})$ of various dimensions. Observe that most estimates are quite accurate when the data is relatively low dimensional.

The accuracy of the entropy estimator is also dependent on the sample size M_k . This is especially true for higher dimensions where exponentially increasing number of samples are required for stable entropy estimation. In Fig. 4.5(b) we demonstrates this for $n = 121$, by plotting density estimates using Eq. 4.28 with $N = 1$ as a function of sample size. As the number of samples increases, the estimate comes closer

to the actual entropy value⁴ of 171.64. However, as Fig. 4.5(b) illustrates, this convergence as a function of sample size is quite slow. This behavior is to be expected as these estimators are essentially *asymptotically* unbiased, converging to the true value of the entropy as $M_k \rightarrow \infty$.

4B Relation between Mutual Information and MMSE Matrix

In [116], the authors derive expressions for the gradient of the mutual information between the input and output of a general multivariate Gaussian channel. The gradients are derived with respect to the model parameters specifically for the Gaussian channel model of the form

$$\underline{\mathbf{y}}_i = \mathbf{A}_i \mathbf{z}_i + \underline{\boldsymbol{\eta}}_i, \quad (4.29)$$

where $\mathbf{z}_i \in \mathbb{R}^n$ and $\underline{\mathbf{y}}_i \in \mathbb{R}^q$ are the input and output of the Gaussian channel respectively, \mathbf{A}_i is a $q \times n$ deterministic matrix, and $\underline{\boldsymbol{\eta}}_i$ is iid Gaussian noise. The authors in [116] show that the gradients of the MI with respect to the signal and noise covariances \mathbf{C}_z and \mathbf{C}_η respectively are related to the MMSE matrix \mathbf{Q}_i as

$$\frac{d}{d\mathbf{C}_z} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) \mathbf{C}_z = \mathbf{A}_i^T \mathbf{C}_\eta^{-1} \mathbf{A}_i \mathbf{Q}_i, \quad \text{and} \quad (4.30)$$

$$\frac{d}{d\mathbf{C}_\eta} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) = -\mathbf{C}_\eta^{-1} \mathbf{A}_i \mathbf{Q}_i \mathbf{A}_i^T \mathbf{C}_\eta^{-1}. \quad (4.31)$$

The Gaussian channel model of Eq. 4.29 can be thought of as a generalization of the patch-wise data model in Eq. 2.1, where in Eq. 4.29 we account for the number (say N_i) of similar patches that exist in the cluster for each \mathbf{z}_i . The vector $\underline{\mathbf{y}}_i$ is then

⁴Our experiments with iid samples drawn from Gaussian pdfs with different covariance matrices indicate that the bias of the entropy estimator of Eq. 4.28 is a function of the dimensionality and the number of samples present, and is independent of the covariance matrix of the Gaussian pdf.

formed by concatenating all \mathbf{y}_j patches that are similar to any given \mathbf{y}_i , and \mathbf{A}_i takes the form of N_i identity matrices stacked together, as shown in Eq. 4.2 (and Fig. 4.1). Thus, in our case, $q = nN_i$. Assuming iid noise, we have the $nN_i \times nN_i$ noise covariance $\mathbf{C}_{\underline{\boldsymbol{\eta}}} = \sigma^{-2} \mathbf{I}_{nN_i}$, and

$$\mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{-1} = \sigma^{-2} [\mathbf{I} \dots \mathbf{I}] \mathbf{I}_{nN_i} = \sigma^{-2} [\mathbf{I} \dots \mathbf{I}] \quad (4.32)$$

$$\Rightarrow \mathbf{Q}_i \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{-1} = \sigma^{-2} [\mathbf{Q}_i \dots \mathbf{Q}_i] \quad (4.33)$$

$$\Rightarrow \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{-1} \mathbf{A}_i \mathbf{Q}_i \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{-1} = \sigma^{-4} \begin{bmatrix} \mathbf{I} \\ \vdots \\ \mathbf{I} \end{bmatrix} [\mathbf{Q}_i \dots \mathbf{Q}_i] = \sigma^{-4} \begin{bmatrix} \mathbf{Q}_i & \mathbf{Q}_i & \dots \\ \mathbf{Q}_i & \ddots & \\ \vdots & & \end{bmatrix}. \quad (4.34)$$

Now, we can rewrite Eq. 4.30 as

$$\frac{d}{d\mathbf{C}_{\mathbf{z}}} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) \mathbf{C}_{\mathbf{z}} = \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\eta}}}^{-1} \mathbf{A}_i \mathbf{Q}_i = \sigma^{-2} [\mathbf{I} \dots \mathbf{I}] [\mathbf{I} \dots \mathbf{I}]^T \mathbf{Q}_i = \frac{N_i}{\sigma^2} \mathbf{Q}_i, \quad (4.35)$$

and Eq. 4.31 as

$$\begin{aligned} \frac{d}{d\mathbf{C}_{\underline{\boldsymbol{\eta}}}} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) &= -\sigma^{-4} \begin{bmatrix} \mathbf{Q}_i & \mathbf{Q}_i & \dots \\ \mathbf{Q}_i & \ddots & \\ \vdots & & \end{bmatrix} \\ \Rightarrow \text{Tr} \left[\frac{d}{d\mathbf{C}_{\underline{\boldsymbol{\eta}}}} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) \right] &= \sum_{j=1}^{nN_i} \frac{d}{d\sigma^2} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) = -\sigma^{-4} N_i \text{Tr}(\mathbf{Q}_i) \\ \Rightarrow \frac{d}{d\sigma^2} I(\underline{\mathbf{y}}_i; \mathbf{z}_i) &= -\frac{1}{n\sigma^4} \text{Tr}(\mathbf{Q}_i). \end{aligned} \quad (4.36)$$

Until now we have shown how the MMSE matrix is related to the MI between \mathbf{z}_i and the vector $\underline{\mathbf{y}}_i$ that contains all patches similar to \mathbf{y}_i . To relate the MMSE matrix to $I(\underline{\mathbf{y}}_i, \mathbf{z}_i)$, we derive a relation between $I(\underline{\mathbf{y}}_i, \mathbf{z}_i)$ and $I(\mathbf{y}_i, \mathbf{z}_i)$ by writing

$$I(\underline{\mathbf{y}}_i; \mathbf{z}_i) = H(\underline{\mathbf{y}}_i) - H(\underline{\boldsymbol{\eta}}_i) \quad [\text{from Equations 4.4 \& 4.29}]$$

$$\begin{aligned}
&= H([\mathbf{y}_1 \dots \mathbf{y}_{N_i}]) - H([\boldsymbol{\eta}_1 \dots \boldsymbol{\eta}_{N_i}]) \\
&= H(\mathbf{y}_1) + \sum_{j=2}^{N_i} H(\mathbf{y}_j | \mathbf{y}_{j-1} \dots \mathbf{y}_1) - \left[H(\boldsymbol{\eta}_1) + \sum_{j=2}^{N_i} H(\boldsymbol{\eta}_j | \boldsymbol{\eta}_{j-1} \dots \boldsymbol{\eta}_1) \right]. \quad (4.37)
\end{aligned}$$

Now, for every \mathbf{y}_j similar to \mathbf{y}_i , we can relate their corresponding noise-free patches as $\mathbf{z}_j = \mathbf{z}_i + \boldsymbol{\varepsilon}_{ij}$. Using this relation and the data model of Eq. 2.1, we get

$$\begin{aligned}
H(\mathbf{y}_j | \mathbf{y}_{j-1} \dots \mathbf{y}_1) &= H((\mathbf{z}_j + \boldsymbol{\eta}_j) | (\mathbf{z}_{j-1} + \boldsymbol{\eta}_{j-1}), \dots, (\mathbf{z}_1 + \boldsymbol{\eta}_1)) \\
&= H((\mathbf{z}_j + \boldsymbol{\eta}_j) | (\mathbf{z}_j + \boldsymbol{\varepsilon}_{jj-1} + \boldsymbol{\eta}_{j-1}), \dots, (\mathbf{z}_j + \boldsymbol{\varepsilon}_{j1} + \boldsymbol{\eta}_1)) \\
&= H(\boldsymbol{\eta}_j | (\boldsymbol{\varepsilon}_{jj-1} + \boldsymbol{\eta}_{j-1}), \dots, (\boldsymbol{\varepsilon}_{j1} + \boldsymbol{\eta}_1)) \\
&= H(\boldsymbol{\eta}_j), \quad (4.38)
\end{aligned}$$

where the last step is a result of $\boldsymbol{\eta}_j$ vectors being independent of $\boldsymbol{\varepsilon}_{ij}$ and from each other. From the latter property, we also obtain

$$H(\boldsymbol{\eta}_j | \boldsymbol{\eta}_{j-1} \dots \boldsymbol{\eta}_1) = H(\boldsymbol{\eta}_j). \quad (4.39)$$

Plugging the above relations into Eq. 4.37, and replacing \mathbf{y}_1 and $\boldsymbol{\eta}_1$ with \mathbf{y}_i and $\boldsymbol{\eta}_i$ respectively (without loss of generality), we get

$$I(\underline{\mathbf{y}}_i; \mathbf{z}_i) = H(\mathbf{y}_i) - H(\boldsymbol{\eta}_i) = I(\mathbf{y}_i; \boldsymbol{\eta}_i). \quad (4.40)$$

Equations 4.35 & 4.36 can then be written as

$$\frac{d}{d\mathbf{C}_z} I(\mathbf{y}_i; \mathbf{z}_i) \mathbf{C}_z = \frac{N_i}{\sigma^2} \mathbf{Q}_i, \quad \text{and} \quad (4.41)$$

$$\frac{d}{d\sigma^2} I(\mathbf{y}_i; \mathbf{z}_i) = -\frac{1}{n\sigma^4} \text{Tr}(\mathbf{Q}_i). \quad (4.42)$$

Note that the MMSE matrix is a function of N_i which can vary across patches within a cluster, where the \mathbf{z}_i patches are considered to be realizations of the random

variable \mathbf{z} . We write the above relations in terms of the MI of the random variables \mathbf{y} and \mathbf{z} and the MMSE matrix \mathbf{Q}_i as

$$\frac{d}{d\mathbf{C}_z} I(\mathbf{y}; \mathbf{z}) \mathbf{C}_z = \frac{1}{M_k \sigma^2} \sum_{i=1}^{M_k} N_i \mathbf{Q}_i, \quad \text{and} \quad (4.43)$$

$$\frac{d}{d\sigma^2} I(\mathbf{y}; \mathbf{z}) = - \frac{1}{n M_k \sigma^4} \sum_{i=1}^{M_k} \text{Tr}(\mathbf{Q}_i) \quad (4.44)$$

by taking the average over all patches within the cluster Ω_k .

4C Derivation of Overall Entropy

In this section we derive an expression for the overall entropy from the cluster-wise entropy. Our choice of features lead to patches in any given cluster being geometrically similar, thus allowing us to assume that such patches are realizations of some random variable \mathbf{z} sampled from some unknown pdf $p_k(\mathbf{z})$ in each cluster. To estimate the entropy of the entire image, we thus need to derive an expression relating the entropy of the clusters with that of the entire image. For this, without loss of generality, we assume that the image consists of $K = 2$ disjoint clusters Ω_1 and Ω_2 with corresponding pdfs p_1 and p_2 . The overall pdf of \mathbf{z} can then be written as

$$p(\mathbf{z}) = \omega_1 p_1 + \omega_2 p_2, \quad (4.45)$$

where ω_k is the probability of \mathbf{z}_i being sampled from p_k , and $\omega_1 + \omega_2 = 1$. The overall entropy can be derived as

$$\begin{aligned} H(p) &= - \int_{\Omega} p(\mathbf{z}) \ln p(\mathbf{z}) d\mathbf{z} \\ &= - \left[\int_{\Omega_1} p(\mathbf{z}) \ln p(\mathbf{z}) d\mathbf{z} + \int_{\Omega_2} p(\mathbf{z}) \ln p(\mathbf{z}) d\mathbf{z} \right] \end{aligned} \quad (4.46)$$

In our case, any given patch is assumed to belong to either Ω_1 or Ω_2 , resulting in

$$p_k(\mathbf{z}_i) = \begin{cases} p_k, & \text{if } \mathbf{z}_i \in \Omega_k \\ 0, & \text{otherwise,} \end{cases} \quad (4.47)$$

for $k = 1, 2$. This allows us to write Eq. 4.46 as

$$H(p) = - \left[\int_{\Omega_1} \omega_1 p_1 \ln(\omega_1 p_1) d\mathbf{z} + \int_{\Omega_2} \omega_2 p_2 \ln(\omega_2 p_2) d\mathbf{z} \right]. \quad (4.48)$$

The above expression can be further simplified by writing

$$\begin{aligned} - \int_{\Omega_1} \omega_1 p_1 \ln(\omega_1 p_1) d\mathbf{z} &= \omega_1 \left[- \int_{\Omega_1} p_1 \ln p_1 d\mathbf{z} - \int_{\Omega_1} p_1 \ln \omega_1 d\mathbf{z} \right] \\ &= \omega_1 H(p_1) - \omega_1 \ln \omega_1. \end{aligned} \quad (4.49)$$

Thus, the overall entropy can be derived as

$$\begin{aligned} H(p) &= \omega_1 H(p_1) + \omega_2 H(p_2) - [\omega_1 \ln \omega_1 + \omega_2 \ln \omega_2] \\ &= \sum_{k=1}^2 \omega_k H(p_k) - \sum_{k=1}^2 \omega_k \ln \omega_k, \end{aligned} \quad (4.50)$$

where $\sum_k \omega_k = 1$. In our derivation, the only assumption we have made is that of the clusters being disjoint, which is true in our case where the clustering is based on geometric similarity of patches. The above expression in Eq. 4.50, derived with $K = 2$ clusters, can then be generalized to an arbitrary number (K) of disjointed clusters as

$$H(p) = \sum_{k=1}^K \omega_k H(p_k) - \sum_{k=1}^K \omega_k \ln \omega_k.$$

This provides us with an expression to calculate the overall entropy of an image from its K cluster-wise entropy estimates.

Chapter 5

Patch-based Locally Optimal Wiener (PLOW) Denoising

Abstract – This chapter deals with the practical application of the bounds framework to image denoising. We show that the formulation of the bound in Chapter 2 implies that, for Gaussian noise, a cluster-wise LMMSE estimator is optimal. However, such a patch-based estimator needs to account for the presence of photometric redundancies. Here we derive such an estimator, the parameters of which are estimated from the given noisy image. Experimentally we show that our proposed method achieves performance that is comparable or exceeding the current state-of-the-art methods.

5.1 Introduction

In Chapter 2 we analyzed the performance bounds for the problem of image denoising. In our study, we specifically considered patch-based methods, where the

observation model was posed as (Eq. 2.1)

$$\mathbf{y}_i = \mathbf{z}_i + \boldsymbol{\eta}_i, \quad (5.1)$$

with \mathbf{y}_i representing the vectorized patch centered at i . Using a Bayesian Cramér-Rao bound [82–84] analysis, we showed that the MSE of denoising (estimating) any given patch in the image is bounded from below by Eq. 2.17 as¹

$$E [\|\mathbf{z}_i - \hat{\mathbf{z}}_i\|^2] \geq \text{Tr} \left[(\mathbf{J}_i + \mathbf{C}_z^{-1})^{-1} \right], \quad (5.2)$$

where $\hat{\mathbf{z}}_i$ is the estimate of \mathbf{z}_i , \mathbf{J}_i is the Fisher information matrix (FIM), and \mathbf{C}_z is the patch covariance matrix. This covariance matrix captures the complexity of the patches.

The FIM, on the other hand, is influenced by the noise characteristics. When additive WGN is considered, the FIM takes the form (Eq. 2.23)

$$\mathbf{J}_i = N_i \frac{\mathbf{I}}{\sigma^2}, \quad (5.3)$$

where \mathbf{I} is the identity matrix, σ is the noise standard deviation, and N_i is the level of photometric redundancy for patch \mathbf{z}_i . Considering noisy observations, such similarity was defined between noisy patches in Eq. 3.12 as

$$\mathbf{y}_j = \mathbf{y}_i + \tilde{\boldsymbol{\epsilon}}_{ij} \quad \text{such that} \quad \|\tilde{\boldsymbol{\epsilon}}_{ij}\|^2 \leq \gamma_n^2 = \gamma^2 + 2\sigma^2 n, \quad (5.4)$$

where γ is a small threshold. This allowed us to identify similar patches and estimate N_i directly from the noisy image in Sec. 3.3.2.

The bounds expression of (5.2) takes into account the complexity of the image patches as well as the redundancy level and the noise variance corrupting the image.

¹We re-iterate here that, in practice, the bounds are calculated using Eq. 2.16. We use Eq. 2.17 for its simplicity and the insights that it presents.

In Chapter 2, the bound was shown to characterize the performance of the optimal affine-biased denoising method. In particular, for WGN, the right-hand side of (5.2) is the performance achieved by the optimal LMMSE estimator, with \mathbf{J}_i and \mathbf{C}_z being the parameters of the estimator. The Wiener filter is, in fact, the LMMSE estimator that achieves this lower bound [71]. Thus, a patch-based Wiener filter, where the parameters are estimated accurately, can lead to near-optimal denoising. This forms the motivation behind our approach. However, such an optimal filter also needs to exploit photometric redundancies, as will be made apparent in the next section.

5.2 Patch-based Wiener filter

Irrespective of the noise characteristics, the expression in (5.2) leads to the lowest MSE theoretically achievable by any patch-based denoising method. This expression was derived in Chapter 2 assuming that the underlying unknown image patches \mathbf{z}_i are (independent) realizations of a random variable \mathbf{z} . Further, image patches that are *geometrically* similar were considered to be sampled from the same (unknown) probability density function (pdf) $p_k(\mathbf{z})$. When the corrupting noise is WGN, the LMMSE estimate of \mathbf{z}_i obtained by the patch-based Wiener filter from its noisy observation \mathbf{y}_i takes the form [71]

$$\hat{\mathbf{z}}_i = \bar{\mathbf{z}} + \mathbf{C}_z \mathbf{C}_y^{-1} (\mathbf{y}_i - \bar{\mathbf{z}}), \quad (5.5)$$

where $\bar{\mathbf{z}}$ and \mathbf{C}_z are the first and second moments of the pdf $p_k(\mathbf{z})$ from which all patches geometrically similar to $\mathbf{z}_i \in \Omega_k$ are assumed to be independently sampled. The covariance of the (geometrically similar) noisy image patches can be expressed as

$$\mathbf{C}_y = \mathbf{C}_z + \sigma^2 \mathbf{I}. \quad (5.6)$$

Thus, the parameters of the LMMSE filter remain the same for all patches that are considered to be similar in structure. A similar approach was applied to the problem of super-resolution and demosaicing by Shao *et al.* [129].

In Chapter 2, we showed that the optimal denoiser is (conditionally) biased and the bias in estimating a particular \mathbf{z}_i patch can be modeled as an affine function of \mathbf{z}_i (Eq. 2.3). It is easy to see that the (conditional) bias of the estimator in Eq. 5.5 is in keeping with that observation:

$$\begin{aligned} E[\hat{\mathbf{z}}_i | \mathbf{z}_i] &= \bar{\mathbf{z}} + \mathbf{C}_z \mathbf{C}_y^{-1} (E[\mathbf{y}_i | \mathbf{z}_i] - \bar{\mathbf{z}}) \\ &= \bar{\mathbf{z}} + \mathbf{C}_z \mathbf{C}_y^{-1} (\mathbf{z}_i - \bar{\mathbf{z}}) \quad (\because E[\mathbf{y}_i | \mathbf{z}_i] = \mathbf{z}_i \text{ when } E[\boldsymbol{\eta}_i | \mathbf{z}_i] = \mathbf{0}) \\ &= \mathbf{C}_z \mathbf{C}_y^{-1} \mathbf{z}_i + (\mathbf{I} - \mathbf{C}_z \mathbf{C}_y^{-1}) \bar{\mathbf{z}} \end{aligned} \quad (5.7)$$

$$\Rightarrow \mathbf{b}(\mathbf{z}_i) = E[\hat{\mathbf{z}}_i - \mathbf{z}_i | \mathbf{z}_i] = \underbrace{(\mathbf{C}_z \mathbf{C}_y^{-1} - \mathbf{I})}_{\mathbf{F}} \mathbf{z}_i + \underbrace{(\mathbf{I} - \mathbf{C}_z \mathbf{C}_y^{-1})}_{\mathbf{u}} \bar{\mathbf{z}}. \quad (5.8)$$

Further, as $\bar{\mathbf{z}}$, \mathbf{C}_z and \mathbf{C}_y are the same for all $\mathbf{z}_i \in \Omega_k$, the parameters of the affine model (\mathbf{F} and \mathbf{u}) remain the same for all geometrically similar patches. From Eq. 5.8, it also follows that the LMMSE estimate has zero expected error (that is, $E[E[\mathbf{z}_i - \hat{\mathbf{z}}_i | \mathbf{z}_i]] = \mathbf{0}$). The MSE of the estimate for all $\mathbf{z}_i \in \Omega_k$ can then be expressed in terms of the error covariance as [71]

$$E[(\mathbf{z}_i - \hat{\mathbf{z}}_i)(\mathbf{z}_i - \hat{\mathbf{z}}_i)^T] = \left(\mathbf{C}_z^{-1} + \frac{\mathbf{I}}{\sigma^2} \right)^{-1} \quad (5.9)$$

$$\Rightarrow E[\|\mathbf{z}_i - \hat{\mathbf{z}}_i\|^2] = \text{Tr} \left(E[(\mathbf{z}_i - \hat{\mathbf{z}}_i)(\mathbf{z}_i - \hat{\mathbf{z}}_i)^T] \right) = \text{Tr} \left[\left(\mathbf{C}_z^{-1} + \frac{\mathbf{I}}{\sigma^2} \right)^{-1} \right]. \quad (5.10)$$

Comparing the MSE above to the expression in (5.2), it is clear that the (cluster-wise) LMMSE estimator achieves the bounds for $N_i = 1$, which is the case when *photometric* similarities are not observed in the input image.

In general, natural images exhibit some level of photometric redundancies.

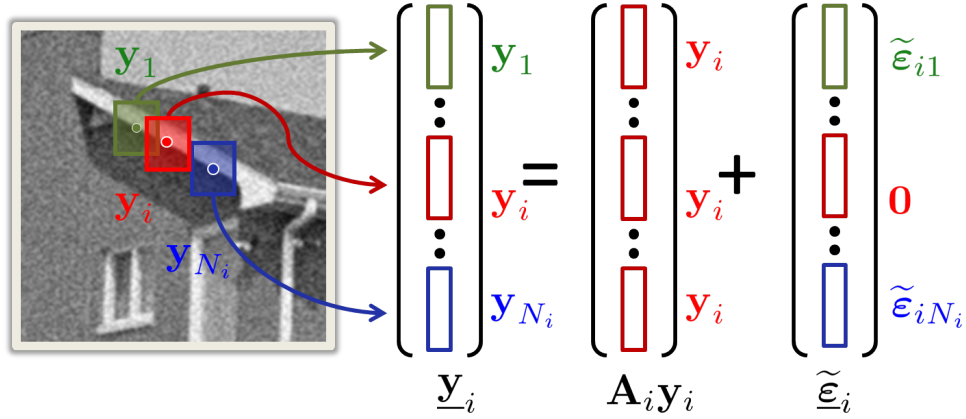


Figure 5.1: Illustration of the data model formed by expressing all photometrically similar patches. Here \mathbf{y}_i is the reference patch, and $\mathbf{y}_1 \dots \mathbf{y}_N$ are patches that satisfy the similarity condition of Eq. 5.4. All these patches are then accumulated to form the data model of Eq. 5.11.

Exploiting such repetitions forms the core of many denoising methods [12,19,20,43,49] where photometrically similar patches are considered to be multiple observations of a single latent patch, with the differences arising (ideally) due to noise only. Consequently, most similarity-based methods identify *photometrically similar* patches within the noisy image to perform denoising, with the most similar patches exerting the greatest influence in the denoising process. Our framework can also be generalized to exploit such *photometric* redundancies within any given noisy image. These patches need not necessarily be spatially proximal (in keeping with [12]), thereby giving rise to a so-called *non-local* patch-based Wiener filter for denoising, as we describe next.

5.3 Patch-based Locally Optimal Wiener Filter (PLOW)

As mentioned earlier, photometric similarity among patches, as required to exploit redundancy, is a stricter condition than the geometric similarity property used for clustering. We, therefore, require an additional step of identifying the \mathbf{y}_j patches

that are photometrically similar to any given patch y_i . These y_j patches all satisfy the condition of Eq. 5.4 and can be expressed as follows (See Fig. 5.1)

$$\begin{aligned}
\underline{\mathbf{y}}_i &= \mathbf{A}_i \mathbf{y}_i + \tilde{\underline{\boldsymbol{\varepsilon}}}_i \\
&= \mathbf{A}_i (\mathbf{z}_i + \boldsymbol{\eta}_i) + (\underline{\boldsymbol{\varepsilon}}_i + \underline{\boldsymbol{\eta}}_i - \mathbf{A}_i \boldsymbol{\eta}_i) \\
&= \mathbf{A}_i \mathbf{z}_i + \underbrace{\underline{\boldsymbol{\varepsilon}}_i + \underline{\boldsymbol{\eta}}_i}_{\underline{\boldsymbol{\zeta}}_i}
\end{aligned} \tag{5.11}$$

where $\underline{\mathbf{y}}_i$ is a vector formed by concatenating all the y_j patches that are photometrically similar to y_i , $\underline{\boldsymbol{\eta}}_i$ is the corresponding noise patches stacked together, $\tilde{\underline{\boldsymbol{\varepsilon}}}_i$ and $\underline{\boldsymbol{\varepsilon}}_i$ are vectors consisting of concatenated difference vectors $\tilde{\boldsymbol{\varepsilon}}_{ij}$ and $\boldsymbol{\varepsilon}_{ij}$ respectively, and \mathbf{A}_i is the matrix formed by vertically stacking N_i identity matrices, each of size $n \times n$. Letting $\mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}$ denote the covariance matrix for the error vector $\underline{\boldsymbol{\zeta}}_i = \underline{\boldsymbol{\varepsilon}}_i + \underline{\boldsymbol{\eta}}_i$, we can write the corresponding LMMSE (Wiener) filter as [71]

$$\begin{aligned}
\hat{\mathbf{z}}_i &= \bar{\mathbf{z}} + \mathbf{C}_z \mathbf{A}_i^T \left(\mathbf{A}_i \mathbf{C}_z \mathbf{A}_i^T + \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i} \right)^{-1} \left(\underline{\mathbf{y}}_i - \mathbf{A}_i \bar{\mathbf{z}} \right) \\
&= \bar{\mathbf{z}} + \left(\mathbf{C}_z^{-1} + \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} \mathbf{A}_i \right)^{-1} \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} \left(\underline{\mathbf{y}}_i - \mathbf{A}_i \bar{\mathbf{z}} \right).
\end{aligned} \tag{5.12}$$

As before, the parameters $\bar{\mathbf{z}}$ and \mathbf{C}_z are the moments obtained from the geometrically similar patches within each cluster. The above expression leads to the optimal estimator for the non-local data model of Eq. 5.11.

Under sufficiently strong WGN, $\boldsymbol{\zeta}$ is approximately Gaussian since, compared to the noise $\boldsymbol{\eta}_i$, the $\boldsymbol{\varepsilon}_{ij}$ vectors are small by definition (Eq. 3.4). If we assume the components of $\boldsymbol{\varepsilon}_{ij}$ vectors to be uncorrelated, the expression in Eq. 5.12 can be further simplified. Note that the $\boldsymbol{\varepsilon}_{ij}$ vectors are independent of the $\boldsymbol{\eta}_i$ noise vectors, which

results in $\mathbf{C}_{\underline{\zeta}_i}$ exhibiting the diagonal form (see derivation in Appendix 5A)

$$\mathbf{C}_{\underline{\zeta}_i} = \mathbf{C}_{\underline{\varepsilon}_i} + \mathbf{C}_{\underline{\eta}_i} = \begin{bmatrix} \ddots & & \mathbf{0} \\ & \delta_{ij}^2 \mathbf{I} & \\ \mathbf{0} & & \ddots \end{bmatrix}, \quad (5.13)$$

$$\text{where } \delta_{ij}^2 = \frac{1}{n} E [\|\mathbf{z}_i - \mathbf{z}_j\|^2] + \sigma^2 \quad (5.14)$$

$$\begin{aligned} &= \frac{1}{n} (E [\|\mathbf{y}_i - \mathbf{y}_j\|^2] - 2\sigma^2 n) + \sigma^2 \\ &= \frac{1}{n} E [\|\mathbf{y}_i - \mathbf{y}_j\|^2] - \sigma^2. \end{aligned} \quad (5.15)$$

Denoting $w_{ij} = \delta_{ij}^{-2}$, the LMMSE estimate of Eq. 5.12 can be alternately expressed as (see Appendix 5B)

$$\hat{\mathbf{z}}_i = \bar{\mathbf{z}} + \left(\mathbf{C}_{\mathbf{z}}^{-1} + \sum_{j=1}^{N_i} w_{ij} \mathbf{I} \right)^{-1} \sum_{j=1}^{N_i} w_{ij} (\mathbf{y}_j - \bar{\mathbf{z}}) \quad (5.16)$$

$$= \bar{\mathbf{z}} + \left(\frac{\mathbf{C}_{\mathbf{z}}^{-1}}{\sum_{j=1}^{N_i} w_{ij}} + \mathbf{I} \right)^{-1} \sum_{j=1}^{N_i} \frac{w_{ij}}{\sum_{j=1}^{N_i} w_{ij}} (\mathbf{y}_j - \bar{\mathbf{z}}). \quad (5.17)$$

From Eq. 5.16, we see that a weighted contribution of each similar patch is used to come up with a denoised estimate for each \mathbf{z}_i where the contributing factor of any \mathbf{y}_j gets larger with increasing similarity (δ_{ij}^{-2}). The error covariance matrix for such an estimator is approximately [71]

$$\mathbf{C}_e \approx \left(\mathbf{C}_{\mathbf{z}}^{-1} + \sum_{j=1}^{N_i} w_{ij} \mathbf{I} \right)^{-1}. \quad (5.18)$$

Comparing the above expression to the bounds formulation of (5.2), we see that when $\sum_j w_{ij} = \frac{N_i}{\sigma^2}$, the estimator achieves the theoretic bounds for denoising².

²For the sake of clarity, we assume here that $\bar{\mathbf{z}}$ and $\mathbf{C}_{\mathbf{z}}$ in each cluster are known and, w_{ij} is essentially deterministic [21]. Hence, the expected estimator error remains zero and Eq. 5.18 approximates the MSE of the estimator in Eq. 5.16. In practice, these parameters are estimated from a limited number of noisy patches resulting in higher MSE than that predicted by the lower bound (Eq. 5.18 or (5.2)).

Such a scenario arises when the underlying noise-free image contains N_i exact replicas for a patch \mathbf{z}_i , that is, when $E[\|\mathbf{z}_i - \mathbf{z}_j\|^2] = 0$. In practice, such levels of redundancy are rare, and even if very similar patches exist, identifying such patches can be challenging under noise contamination, resulting in higher MSE.

Although Eq. 5.17 provides a nice formulation for the estimator, it can lead to mathematical instabilities in denoising as the covariance matrix \mathbf{C}_z can be rank deficient or ill-conditioned. To circumvent the possibility of errors due to inversion, we make use of the matrix inversion lemma [96] to state an alternate form of the PLOW filter (see Appendix 5B for entire derivation):

$$\hat{\mathbf{z}}_i = \left[\sum_{j=1}^{N_i} \frac{w_{ij} \mathbf{y}_j}{\sum_{j=1}^{N_i} w_{ij}} \right] + \left[\sum_{j=1}^{N_i} \frac{w_{ij}}{\sum_{j=1}^{N_i} w_{ij}} \left(\sum_{j=1}^{N_i} w_{ij} \mathbf{C}_z + \mathbf{I} \right)^{-1} (\bar{\mathbf{z}} - \mathbf{y}_j) \right]. \quad (5.19)$$

This is an interesting formulation where the first part of Eq. 5.19 is exactly the expression for the NLM [12] filter. In the second part, since $\bar{\mathbf{z}}$ is obtained from all *geometrically* similar patches in a cluster, it can be considered as a naïve denoised estimate which is over-smoothened. This latter part of the expression in Eq. 5.19 filters the residuals between the noisy similar patches and this naïve estimate. These filtered residuals are then added to the weighted mean of photometrically similar patches. The second part, thus, forms a “correction term” that improves the NLM estimate by a directional filtering of the residuals based on their shared geometric structure. This suppresses the noise further, while restoring more of the finer details in the image patches. When structural information of image patches are ignored (that is, all structures are equally probable, implying a large determinant of \mathbf{C}_z), we obtain the NLM filter as a sub-optimal approximation (in terms of MSE) of our formulation in Eq. 5.19.

Until now, we have presented and analyzed the theoretical basis for our proposed approach. In the next section, we provide a practical outline for our algorithm

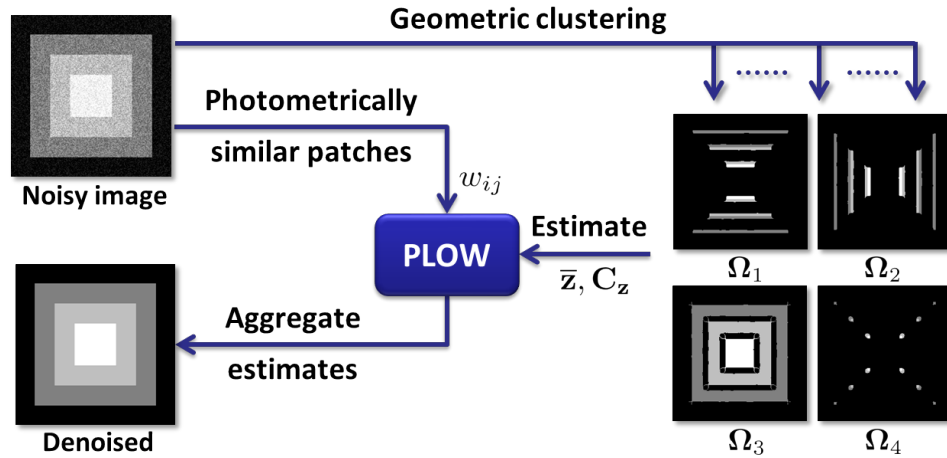


Figure 5.2: Outline of our patch-based locally optimal Wiener (PLOW) filtering method.

that details the estimation of each parameter of the proposed filter from a given noisy image.

5.4 Parameter Estimation for Denoising

Eq. 5.19 presents a mathematical formulation for our patch-based locally optimal filtering (PLOW) scheme. Two of the filter parameters can be identified as the first and second moments (\bar{z} & C_z) of the unknown pdf from which the geometrically similar patches are assumed to be sampled. These parameters need to be estimated from the input noisy image patches. In Chapter 3, we described a way of clustering geometrically similar patches from a noisy image and estimating the cluster-wise moments to estimate the denoising bounds. A similar methodology can be employed here as well. As with the bounds estimation process, we also identify the photometrically similar patches using each noisy patch as the reference. However, for the purpose of denoising, we need to additionally quantify the similarity between the noisy patches

as w_{ij} . Since the parameters to estimate are quite similar, the PLOW denoising framework (Fig. 5.2) shares strong similarity with that of the bounds estimation process of Chapter 3. We discuss the parameter estimation process next.

5.4.1 Geometric Clustering and Moment Estimation

The first step of our denoising method is grouping together patches of similar geometric structure. Presence of noise makes identifying of such structure difficult. Moreover, such grouping needs to be independent of the patch intensities. In Sections 3.2.1 and 3.3.1, we showed that K-Means clustering using LARK features provides a reasonably accurate clustering, even when the input image is contaminated by considerable noise. This is illustrated in Fig. 3.6 where the clusters of the noisy and noise-free Barbara images are quite similar. For our denoising purposes, we employ the same clustering mechanism as that outlined in Sec. 3.2.1

Once the image is segmented into regions of structural similarity, we estimate the moments, namely mean and covariance, from the noisy member patches of each cluster. Since the η_i noise patches are assumed to be zero mean iid, the mean of the underlying noise-free image can be approximated by the expectation of the noisy patches within each cluster as

$$\hat{\mathbf{z}} = E[\mathbf{y}_i \in \Omega_k] \approx \frac{1}{M_k} \sum_{\mathbf{y}_i \in \Omega_k} \mathbf{y}_i \quad (5.20)$$

where Ω_k denotes the k -th cluster with cardinality M_k . Note that the accuracy of this estimate is dependent on M_k . If too few patches are present, the mean vector will remain noisy.

The covariance matrix \mathbf{C}_z is also estimated from the noisy patches within the cluster. For this we make use of the relation between the covariance of the noisy (\mathbf{C}_y)

and noise-free patches (\mathbf{C}_z) from Eq. 5.6. We, thus, need to first estimate \mathbf{C}_y . Covariance estimation is an active research area with a wide variety of applications [106, 107, 114]. The simplest of such estimators, the sample covariance, is the maximum likelihood estimate of the covariance of a pdf estimated from its observed samples. Although other estimators, for example, bootstrapping [107], shrinkage-based [106] methods, etc. exist, we found no discernible improvement in denoising performance when such more complex estimators were used. When the number of samples (patches in a cluster) are few compared to the dimension (number of pixels in each patch) of the data, the sample covariance can be inaccurate. For such cases, robust estimators proposed by Kritchman *et al.* [114] may also be used.

Working with the sample covariance estimate $\hat{\mathbf{C}}_y$, we estimate the covariance of the underlying noise-free patches as,

$$\hat{\mathbf{C}}_z = \left[\hat{\mathbf{C}}_y - \sigma^2 \mathbf{I} \right]_+, \quad (5.21)$$

where σ^2 is the noise covariance and $[\mathbf{X}]_+$ denotes the matrix \mathbf{X} with its negative eigenvalues replaced by zero (or a very small positive value), as done before in Sec. 3.3.1 (also [130]). For this, we need to accurately estimate the noise standard deviation first. Here we use a gradient-based estimator as [100]

$$\hat{\sigma} = 1.4826 \text{ median} (|\nabla \mathbf{Y} - \text{median}(\nabla \mathbf{Y})|), \quad (5.22)$$

where $\nabla \mathbf{Y}$ is the vectorized form of the gradient of the input image \mathbf{Y} . The gradient image $\nabla \mathbf{Y}$ is calculated as

$$\nabla \mathbf{Y} = \frac{1}{\sqrt{6}} \text{vec} \left(\mathbf{Y} \star \begin{bmatrix} 2 & -1 \\ -1 & 0 \end{bmatrix} \right). \quad (5.23)$$

Here $\text{vec}(\cdot)$ denotes the vectorization operation (column-wise or row-wise) and the convolution (\star) operation simply implies addition of the forward gradients in the horizontal and vertical directions. In Chapter 3 (and [130]), we showed that the shrinkage estimator of Eq. 5.21 is accurate enough to compute the bounds directly from the noisy image. In the present case of denoising a similar observation holds.

5.4.2 Calculating Weights for Similar Patches

In our work, we first identify patches within the noisy image that are *photometrically similar* to a given reference patch. Once the similar patches are identified, we perform denoising with the more similar patches exerting greater influence in the denoising process. This is ensured by the analytically derived weight w_{ij} which determines the contributing factor for a patch \mathbf{y}_j in denoising the reference patch \mathbf{y}_i . The weight w_{ij} is related to the inverse of the expected squared ℓ_2 distance between the underlying noise-free patches and a noise term (Eq. 5.14):

$$\delta_{ij}^2 = \frac{1}{n} E [\|\mathbf{z}_i - \mathbf{z}_j\|^2] + \sigma^2 = \frac{1}{n} E [\|\mathbf{y}_i - \mathbf{y}_j\|^2] - \sigma^2, \text{ with } w_{ij} = \delta_{ij}^{-2}. \quad (5.24)$$

Although the weight calculation formulation in Eq. 5.14 is statistically well-motivated, in practice it is difficult to estimate as we need to approximate $E [\|\mathbf{y}_i - \mathbf{y}_j\|^2]$ from a single \mathbf{y}_i and \mathbf{y}_j pair. Here, we approximate this similarity measure (see Appendix 5C for derivation) by³

$$w_{ij} \approx \frac{1}{\sigma^2} \exp \left\{ -\frac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{h^2} \right\}, \quad (5.25)$$

where the scalar multiplier $\frac{1}{\sigma^2}$ also ensures that the denoiser defaults to that of Eq. 5.5 when no photometrically similar patches are observed; that is, when $\mathbf{y}_j = \{\mathbf{y}_i\}$. The

³Note that the expression in Eq. 5.25 is similar to that introduced in [12]. In Appendix 5C, we motivate this formulation statistically and derive it as an approximation to the distance metric in Eq. 5.24.

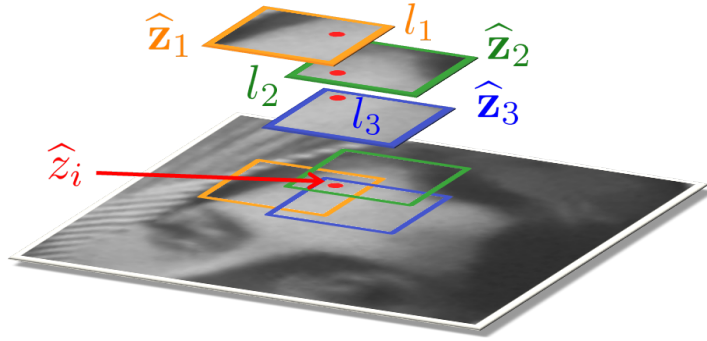


Figure 5.3: An illustration of how a pixel is estimated multiple times due to overlapping patches. Here we show 3 such overlapping patches. In each estimated patch \hat{z}_r (here $r = 1, 2, 3$), the same pixel is estimated as its l_r -th pixel which we denote as $\hat{z}_{r,l}$. These estimates are finally combined to form the final estimate \hat{z}_i .

smoothing parameter h^2 is a positive parameter that controls the rate at which the contributing factor is driven to zero as the patches become less similar. Although tunable in general, for our denoising purposes this parameter is kept fixed at $h^2 = 1.75\sigma^2n$. This was empirically found to be close to the optimal h^2 value for a wide range of images, across different noise levels.

Note that photometrically similar patches are necessarily geometrically similar too, and, hence, we could limit our search to within the cluster of the reference patch. However, errors in clustering (see Fig. 3.6) can limit the number of similar patches identified. On the other hand, scanning the entire image can be time consuming. Consequently, we restrict ourselves to a relatively small search window (30×30 pixels) in our search for photometrically similar patches. Apart from speed considerations, as the motivation was in [12], this also leads to better denoising performance [18].

5.4.3 Aggregating Multiple Pixel Estimates

Until now, we have estimated all the parameters needed to perform the filtering of Eq. 5.19. The filter is run on a per-patch basis (although parameters are estimated from multiple patches), yielding denoised estimates for each patch of the noisy input. To avoid block artifacts at the patch boundaries, the patches are overlapping. As a result, we obtain multiple estimates for the pixels lying on the overlapping regions. This is shown in Fig. 5.3 where z_i is estimated multiple times as a part of different patches. These multiple estimates need to be aggregated to form the final denoised image.

The simplest method of aggregating such multiple estimates is to average the estimates for each pixel. However, such naïve averaging will lead to an over-smoothed image. Alternatively, in keeping with our overall approach, we can combine the multiple estimates in an LMMSE scheme that takes into account the relative confidence in each estimate as measured by the inverse of the estimator error variance. The error covariance of our proposed estimator is approximated by (Eq. 5.18)

$$\mathbf{C}_e \approx \left(\hat{\mathbf{C}}_{\mathbf{z}}^{-1} + \sum_{j=1}^{N_i} w_{ij} \mathbf{I} \right)^{-1}. \quad (5.26)$$

Let us denote \hat{z}_{rl} as the denoised estimate for the l -th pixel in the r -th patch (see Fig. 5.3). Then, the variance v_{rl} of the error associated with the l -th pixel estimate is given by the l -th diagonal element of \mathbf{C}_e . Concatenating the multiple estimates \hat{z}_{rl} in a vector $\hat{\mathbf{z}}_{ir}$, we can write

$$\hat{\mathbf{z}}_{ir} = \mathbf{1} z_i + \boldsymbol{\tau}_{ir}, \quad (5.27)$$

where $\mathbf{1}$ is a vector of all ones, and $\boldsymbol{\tau}$ is the error vector assumed to be zero mean with covariance $\mathbf{C}_{\boldsymbol{\tau}} = \text{diag}[\dots v_{rl} \dots]$. The LMMSE estimate for the i -th pixel of the image

is then

$$\begin{aligned}\hat{z}_i &= (\sigma_z^{-2} + \mathbf{1}^T \mathbf{C}_\tau^{-1} \mathbf{1})^{-1} \mathbf{1}^T \mathbf{C}_\tau^{-1} \hat{\mathbf{z}}_{ir} \\ &= \frac{\sum_{r=1}^R v_{rl}^{-1} \hat{z}_{rl}}{\sum_r v_{rl}^{-1} + \sigma_z^{-2}},\end{aligned}\tag{5.28}$$

where σ_z^2 is the variance of z_i which forms the prior information.

Note that although we estimate the covariance ($\hat{\mathbf{C}}_z$) of the image patches, this does not provide us with a pixel-wise variance estimate σ_z^2 . This is a result of considering overlapping patches where any given pixel z_i can lie in different locations in different patches (see Fig. 5.3). Moreover, the overlapping patches may also be structurally different and, hence, lie in different clusters with different corresponding $\hat{\mathbf{C}}_z$ matrices. In the absence of a particular σ_z , we consider all possible z_i values (within the intensity range [0-255]) to be equally likely, leading to the variance of the discrete uniform distribution $\sigma_z^2 = (256^2 - 1)/12 \Rightarrow \sigma_z^{-2} = 0$. This reduces Eq. 5.28 to the weighted least squared error estimate

$$\hat{z}_i = \frac{\sum_{r=1}^R v_{rl}^{-1} \hat{z}_{rl}}{\sum_r v_{rl}^{-1}},\tag{5.29}$$

where the number of estimates (R) of the i -th pixel depends on the size of the patch (n), the amount of overlap⁴ and the position of the pixel in the the patches (pixels towards the edge of a patch are more likely to lie in overlapping regions). Note that r indexes only those R patches that include the i -th pixel of the image and the position l of the i -th pixel is dependent upon the patch r being considered.

⁴Note that a larger overlap implies more patches for clustering, moment estimation and higher levels of redundancy among image patches. This makes the estimation process robust and allows for improved performance. However, this performance comes at the cost of speed. A reasonable approach is to use all patches (at one pixel shifts) for parameter estimation, and denoise only every few (overlapping) patches. The aggregation step can then be used to reconstruct the entire image. Performance of such a scheme, visually and MSE-wise, is reasonably close to that obtained by denoising densely.

Algorithm 1: PLOW denoising (see Fig. 5.2)

Input: Noisy image: \mathbf{Y}

Output: Denoised image: $\hat{\mathbf{Z}}$

- 1 Set parameters: patch size $n = 11 \times 11$, number of clusters $K = 15$;
 - 2 Estimate noise standard deviation $\hat{\sigma}$ (Eq. 5.22);
 - 3 Set parameter: $h^2 = 1.75\hat{\sigma}^2n$;
 - 4 $\mathbf{y}_i \leftarrow$ extract overlapping patches of size n from \mathbf{Y} ;
 - 5 $\mathbf{L} \leftarrow$ compute LARK features for each \mathbf{y}_i ;
 - 6 $\Omega_k \leftarrow$ geometric clustering with K-Means(\mathbf{L}, K);
 - 7 **foreach** Cluster Ω_k **do**
 - 8 Estimate mean patch $\hat{\mathbf{z}}$ (Eq. 5.20);
 - 9 Estimate cluster covariance $\hat{\mathbf{C}}_{\mathbf{z}}$ (Eq. 5.21);
 - 10 **foreach** Patch $\mathbf{y}_i \in \Omega_k$ **do**
 - 11 $\mathbf{y}_j \leftarrow$ identify photometrically similar patches (Eq. 5.4);
 - 12 $w_{ij} \leftarrow$ compute weights for all \mathbf{y}_j (Eq. 5.25);
 - 13 $\hat{\mathbf{z}}_i \leftarrow$ estimate denoised patch (Eq. 5.19);
 - 14 $\mathbf{C}_{e_i} \leftarrow$ calculate estimate error covariance (Eq. 5.18);
 - 15 **end**
 - 16 **end**
 - 17 $\hat{\mathbf{Z}} \leftarrow$ aggregate multiple estimates from all $\{\hat{\mathbf{z}}_i\}$ and $\{\mathbf{C}_{e_i}\}$ (Eq. 5.29);
-

The entire process of performing denoising with our PLOW framework is algorithmically presented in Algo. 1. As can be expected, the accuracy of estimating the parameters is dependent on the strength of the noise corrupting the image. Noise af-

fects different parameter estimation steps differently. The moment estimation steps are dependent on the ability of the clustering step to classify structurally similar patches. Although the LARK features are quite robust, errors in clustering due to noise cannot be fully avoided. This is demonstrated in Fig. 3.6 where differences in clustering the noisy and noise-free images are apparent.

Although outliers do influence the moment estimates, the process that is most sensitive to noise is the weight calculation of Eq. 5.25. Identifying photometrically similar patches becomes challenging in the presence of strong noise [110,130], which in turn influences the similarity measure calculation of Eq. 5.25. To alleviate such detrimental effects of noise, we pre-filter the image once before the parameters of the proposed framework are learned. Note that such pre-filtering is quite typical of competing approaches [49], and is necessary only for strong noise. For the pre-processing step, we apply our algorithm once on the input noisy image with a reduced noise variance estimate to ensure that finer details are not lost. The necessary filter parameters are then learned from the resultant noise-suppressed image. These parameters are then applied to the original noisy image for denoising. In our experiments such an approach betters the denoising performance considerably. We demonstrate the denoising capacity of our method in the next section.

5.5 Experimental Results

In this section we evaluate the proposed denoising method through experiments on various images at different noise levels. Since our method is motivated by our bounds formulation [70], we first compare the ideal denoising performance of our method (using “oracle” parameters) with the MSE predicted by the bounds. Later, we

estimate the parameters directly from the noisy images, as outlined in Sec. 5.4 and compare those results to various popular denoising methods. We also apply our method, with a minor modification, to color images. Finally, we address the practical case of denoising real noisy images where the noise characteristics are unknown and not necessarily Gaussian, or uncorrelated. In each case, we will show that our results are comparable, in terms of MSE (PSNR⁵), SSIM [66], and the recently introduced no-reference quality metric Q [67] (wherever applicable), to those obtained by state-of-the-art denoising methods, and in many cases visually superior.

Since our method was designed specifically with the aim of achieving the theoretical limits of performance, we first compare our results to the predicted performance bounds [70]. For this first experiment, we compute the “oracle” denoising parameters from the noise-free images. To be precise, we compute the structure-capturing LARK features from the noise-free image and perform clustering. These “oracle” clusters are then used to estimate the moments $\bar{\mathbf{z}}, \mathbf{C}_{\mathbf{z}}$ from the latent image. We also use the ground-truth image to identify the photometrically similar patches and compute the weights w_{ij} for each noise-free reference patch. The final denoising using the “oracle” parameters is, of course, applied to the noisy image.

Figures 5.4(c) & 5.5(c) show the optimal performance of our method when considering WGN of standard deviation 25. Not surprisingly, the results are quite impressive in terms of denoising achieved with finer details being retained at the same time. The MSE obtained are 49.42 and 27.97 for the Barbara and house images respectively. For the Barbara image, the lowest MSE predicted by the denoising bound (MSE

⁵Peak signal-to-noise ratio (PSNR) is measured in decibels (dB) and calculated as $10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right)$ for images with intensity range $[0 - 255]$. An improvement of 1dB reduces the MSE by approximately 20%.

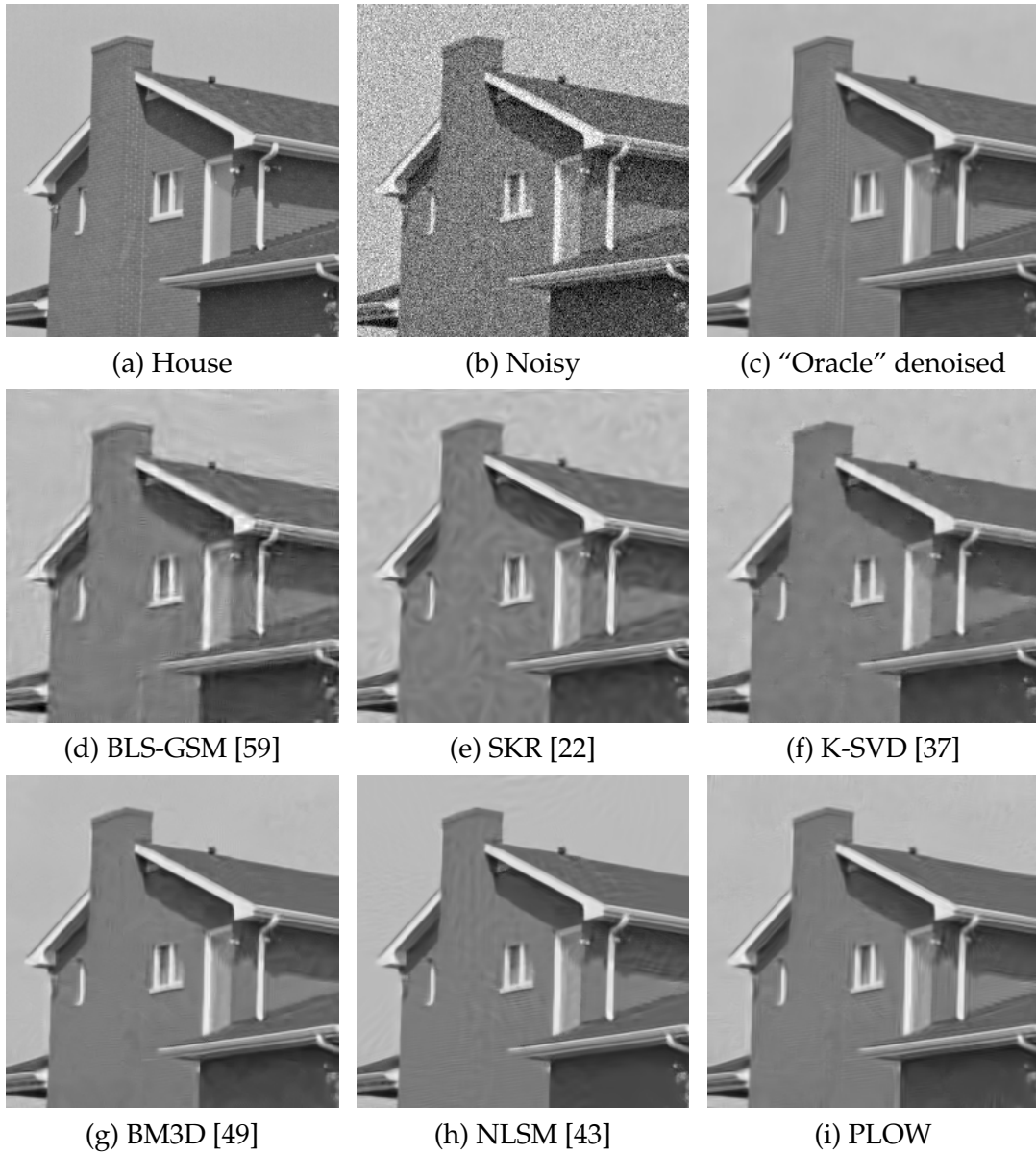


Figure 5.4: Comparison of denoising results for the house image corrupted by WGN of $\sigma = 25$. (a) Original image, (b) noisy input, (c) PLOW with "oracle" parameters (MSE 27.97), (d) BLS-GSM [59] (MSE 49.16), (e) SKR [22] (MSE 47.69), (f) K-SVD [37] (MSE 40.99), (g) BM3D [49] (MSE 33.36), (h) NLSM [43] (MSE 32.31), and (i) PLOW (MSE 34.51). High resolution images can be viewed at <http://users.soe.ucsc.edu/~priyam/PLow/>.



Figure 5.5: Comparison of denoising results for the Barbara image corrupted by WGN of $\sigma = 25$. (a) Original image, (b) noisy input, (c) PLOW with “oracle” parameters (MSE 49.42), (d) BM3D [49] (MSE 56.17), (e) NLSM [43] (MSE 61.00), and (f) PLOW (MSE 62.64). High resolution images can be viewed at <http://users.soe.ucsc.edu/~priyam/PLow/>.

50.24) is achieved⁶, while for the house image the bound (MSE 14.82) is still lower. One reason for such discrepancy between the theoretical prediction and what we obtain in practice, even with oracle parameters, is that the theory in the bounds (Chapter 2) is built on the assumption of *exact* replicas of patches being observed. However, in real-

⁶It may seem here that the lower bounds are breached, albeit marginally, for the Barbara image. However, in Chapter 3, the bounds were calculated with 5 clusters, whereas we use 15 clusters here. It was shown in Fig. 3.8 that using more clusters reduces the bounds further, although the reduction is nominal.



(a) Original (b) Noisy (c) BM3D [49] (d) NLSM [43] (e) PLOW

Figure 5.6: Comparison of denoising performance with some leading denoising methods for Lena, man and stream images (Fig. 1.7) corrupted by $\sigma = 25$. Cropped regions are shown here. Full images can be compared at <http://users.soe.ucsc.edu/~priyam/PLow/>.

ity, such replicas are rare, even in “noise-free” images⁷. It is encouraging to note that in comparison to other methods, the optimal MSE is well below the state-of-the-art for the house image for which the bounds predict the possibility of improved performance (see Chapter 6). On the other hand, the optimal performance for the more complicated Barbara image is comparable to that of BM3D (Fig. 5.5), in keeping with the bounds predicting little improvement to be gained (see Fig. 3.11).

Having established that our method performs near-optimal denoising with “oracle” knowledge of parameters, we experiment with the more practical case when

⁷The term “noise-free” here is an idealization used to imply the original image before noise is added. In general, images captured are invariably noisy due to the imaging process [1]. That images considered to be ground-truth also contain noise, albeit in low strengths, has been illustrated in [101].

the parameters are estimated directly from the noisy image, as outlined in Sec. 5.4. In Figures 5.4 & 5.5, we compare our results to various high fidelity methods for image denoising. In Table 5.1 we quantify the performances for a variety of benchmark images, across different noise levels. The results there show that, in terms of PSNR, SSIM [66] and the no-reference quality metric Q [67], our method is quite comparable to BM3D [49] and NLSM [43]. While BM3D is quite fast, the algorithm’s high performance has not been well-justified on theoretical grounds as of yet. On the other hand, NLSM can be quite complicated in terms of the steps involved. In contrast, our method is well-motivated, and provides a statistical explanation for its performance. Moreover, when “oracle” filter parameters are used, our method generally improves on the state-of-the-art performance, especially for strong noise cases. This shows the true potential of our denoising approach, given improved estimates of the parameters.

In terms of visual quality, our method is comparable to NLSM and BM3D, even outperforming them in many cases where images exhibit higher levels of redundancy. This can be observed in Fig. 5.5 where our result is more visually pleasing when compared to NLSM and BM3D, both of which produce more artifacts in the smoother floor and face regions. As with the quantitative measures, generally the visual quality is greatly improved when “oracle” parameters are used, though this is of course not practical. This improvement is more pronounced for the strong noise cases and for images containing finer details where parameter estimation is more error prone. This is illustrated in Fig. 5.4, where the “oracle” denoised image retains most of the brick pattern in the house facade. Later, in Chapter 6, we will see that this agrees with the conclusions of our performance bounds that expects greater improvement in performance for the class of smoother images. Images containing more semi-stochastic

Table 5.1: Denoising performance of some popular methods (NLSM [43], BM3D [49]) under WGN corruption, compared to PLOW, with and without oracle information. Results noted are average PSNR (top), SSIM [66] (middle) & Q -measure [67] (bottom) over 5 independent noise realizations for each σ .

σ	House (256×256)				Lena (512×512)				Barbara (512×512)			
	NLSM	BM3D	PLOW	Oracle	NLSM	BM3D	PLOW	Oracle	NLSM	BM3D	PLOW	Oracle
5	39.91	39.80	39.52	40.00	38.72	38.73	38.66	38.83	38.46	38.30	37.98	38.18
	0.958	0.957	0.954	0.960	0.945	0.945	0.946	0.947	0.965	0.965	0.946	0.966
	42.35	42.58	42.20	43.21	35.18	35.37	34.75	36.13	69.80	69.60	69.14	70.27
15	35.27	34.95	34.72	35.67	34.17	34.26	33.90	34.43	32.98	33.09	32.17	33.17
	0.902	0.890	0.893	0.923	0.893	0.895	0.890	0.902	0.920	0.923	0.913	0.926
	36.10	36.37	36.98	36.35	21.08	21.07	21.59	21.55	55.41	55.25	55.28	55.04
25	33.14	32.89	32.70	33.68	31.84	32.07	31.92	32.47	30.34	30.67	30.20	31.21
	0.866	0.859	0.859	0.898	0.855	0.861	0.859	0.874	0.876	0.886	0.879	0.899
	20.07	20.11	20.39	20.14	11.43	11.45	11.69	11.64	37.87	37.80	37.72	37.48
50	28.99	29.25	29.08	30.68	27.55	28.58	28.32	29.44	25.68	26.75	26.19	28.01
	0.814	0.802	0.780	0.856	0.774	0.788	0.759	0.825	0.748	0.778	0.755	0.841
	-	-	-	-	-	-	-	-	-	-	-	-
σ	Peppers (256×256)				Boat (512×512)				Stream (512×512)			
	NLSM	BM3D	PLOW	Oracle	NLSM	BM3D	PLOW	Oracle	NLSM	BM3D	PLOW	Oracle
5	38.14	38.06	37.69	37.89	37.36	37.28	37.24	37.43	35.75	35.75	35.59	35.66
	0.955	0.956	0.954	0.956	0.941	0.939	0.941	0.943	0.964	0.964	0.962	0.964
	76.37	76.17	75.60	76.67	37.21	37.38	36.95	37.73	31.12	30.94	30.58	30.96
15	32.76	32.65	31.82	32.45	32.17	32.11	31.53	32.23	28.88	28.74	28.71	28.87
	0.905	0.906	0.899	0.905	0.855	0.854	0.840	0.868	0.852	0.845	0.849	0.852
	64.00	64.02	64.99	64.78	27.16	27.47	28.38	27.99	21.51	22.21	19.74	22.32
25	30.06	30.07	29.53	30.06	29.73	29.83	29.59	30.11	26.27	26.14	26.20	26.45
	0.864	0.868	0.859	0.869	0.794	0.800	0.794	0.823	0.745	0.735	0.747	0.761
	49.55	49.87	50.13	50.16	14.38	14.49	14.19	14.64	12.18	12.54	12.14	12.39
50	25.16	25.85	26.32	26.84	25.46	26.20	26.13	27.12	22.43	23.08	23.38	24.01
	0.766	0.775	0.752	0.802	0.656	0.685	0.674	0.742	0.489	0.535	0.571	0.641
	-	-	-	-	-	-	-	-	-	-	-	-

Noisy images are clipped to lie within the [0-255] intensity range.

Reliance on detecting anisotropic regions in noisy images makes Q -measure inapplicable for $\sigma = 50$ [67].

texture typically exhibit lower levels of patch redundancy. For such images, BM3D typically does a better job of denoising. However, even in such cases, our denoising results are visually comparable to the state-of-the-art, as shown in Fig. 5.6 where we compare our (cropped) results to NLSM [43] and BM3D [49] for some fairly textured regions of different images.

As a next step, we apply our method to the problem of denoising color images. In [131] it was pointed out that optical wavelengths at which the human eye perceives each of the red, green and blue colors have considerable overlap. Consequently,

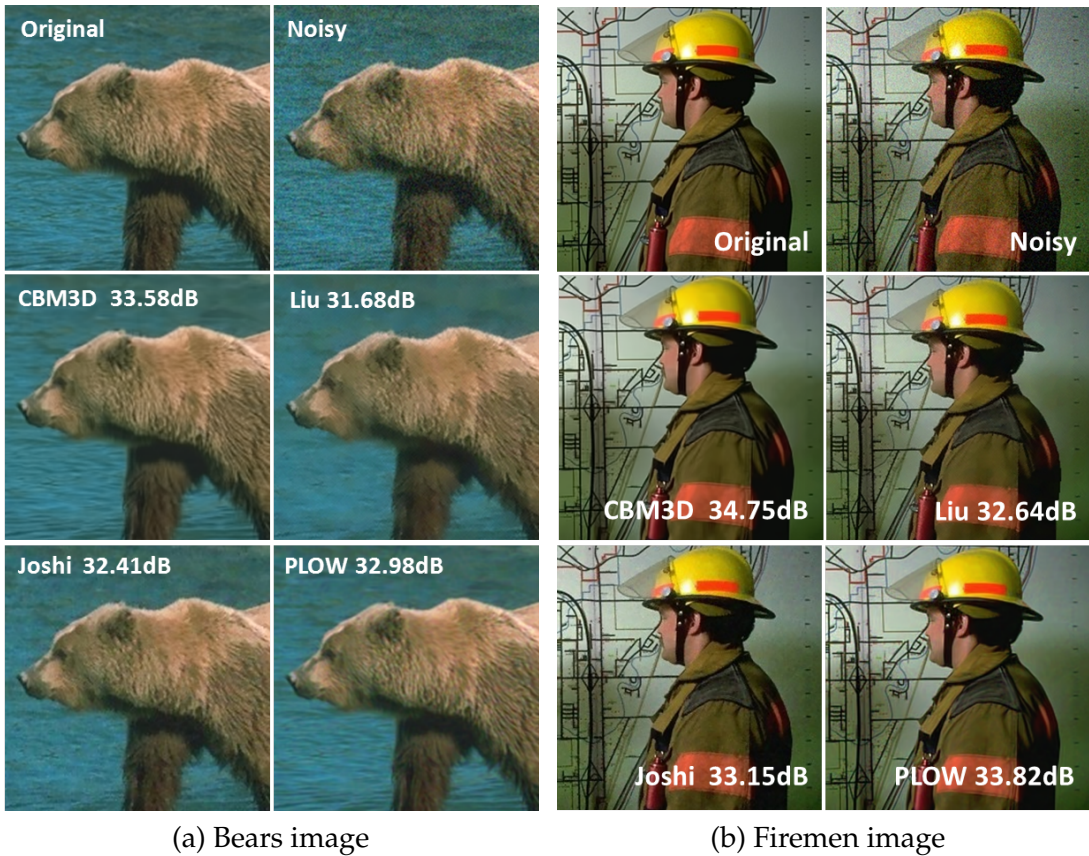


Figure 5.7: Comparison of (cropped) denoising results for color images corrupted by 5% WGN. The methods compared to are CBM3D [49], segmentation-based denoising proposed by Liu *et al.* [44], and denoising with two-color prior by Joshi *et al.* [45]. Full images at native resolutions can be viewed at <http://users.soe.ucsc.edu/~priyam/PLOW/>.

many color denoising methods take into account such dependencies, either implicitly or explicitly. Mairal *et al.* [132] illustrated the usefulness of enforcing constraints across color channels to reduce color washing effects. Other methods, such as [45], account for such correlation implicitly by modeling the color information at each pixel.

A different approach to treating such correlated color information is through color-space conversion where the information between color spaces can be largely decorrelated. Such an approach was employed by Dabov *et al.* [49] in extending the

BM3D algorithm for color images (CBM3D). There the authors identify similar patches using the luminance channel, to which the human visual system is more sensitive. Denoising is, however, performed on all channels simultaneously. In general, any gray-scale denoising method can be applied to denoising color images through such transformations. However, such color-space conversions alter the statistical characteristics of the noise. Consequently, we perform denoising in the RGB color space, but only the noisy image luminance is used to perform geometric clustering. The parameters for denoising are, however, learned individually in each color channel.

Fig. 5.7 illustrates the results obtained by our method with its naïve extension to color images. The noisy images are formed by adding simulated 5% WGN in each channel⁸. In terms of PSNR, the best performing method overall is CBM3D [49]. However, our method is visually quite comparable to it, and significantly better than Liu *et al.* [44] where there is considerable loss of finer details, and Joshi *et al.* [45] where the denoised images still retain some noise. These results are encouraging considering that CBM3D and [45] are specifically designed to handle color images.

Until now, our experiments involved images corrupted by simulated WGN. Although the Gaussian pdf makes a good noise model, real noise is signal dependent [1, 44]. To demonstrate our performance in such situations, we apply our method to denoising some real noisy images with unknown noise characteristics⁹. For these experiments, an estimate of the noise variance was used as an input to our method. The best results optimized using the Q -metric [67] are shown in Fig. 5.8 where we com-

⁸The original images form a part of the Berkeley segmentation dataset [133]. The noisy images, along with results for methods by Liu *et al.* [44] and Joshi *et al.* [45], were obtained from http://research.microsoft.com/en-us/um/redmond/groups/ivm/twocolordeconvolution/supplemental_results/denoising.html. 5% noise corresponds to $\sigma \approx 12$ in each color channel.

⁹The optimality of the LMMSE, and, hence, PLOW filter holds even for non-Gaussian noise, as long as it is not data-dependent [71]. However, even for signal-dependent noise, PLOW performs quite well.

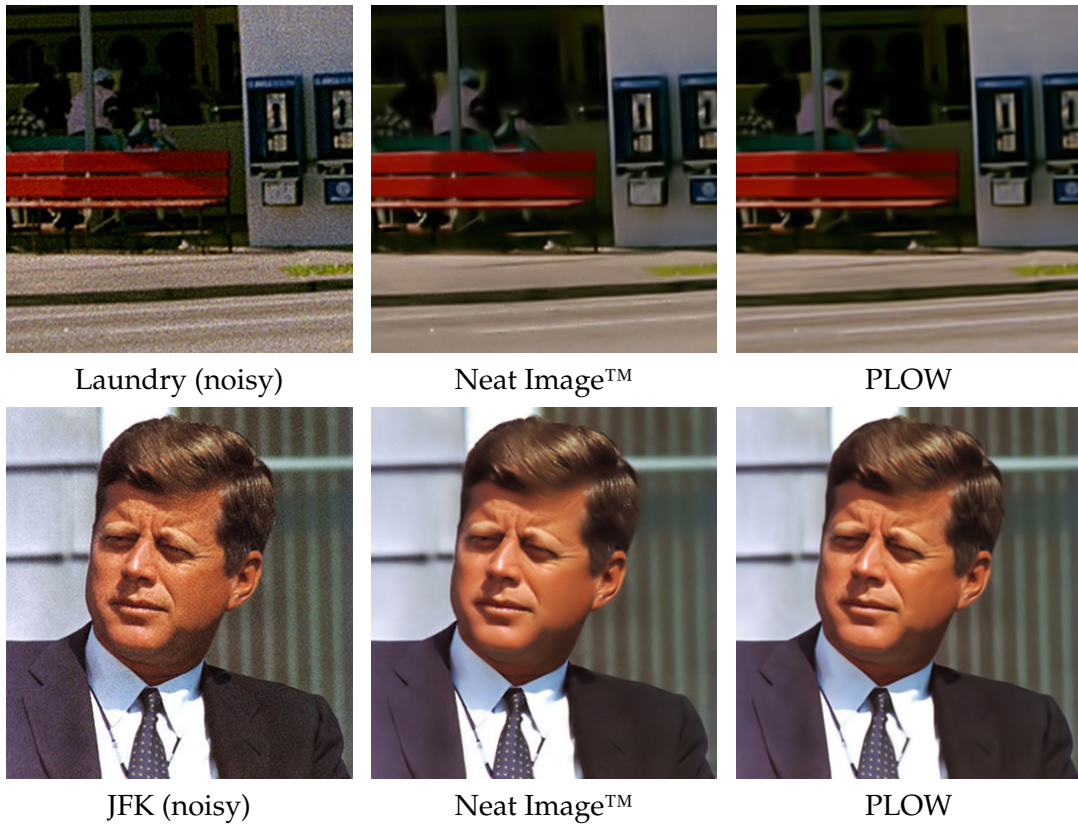
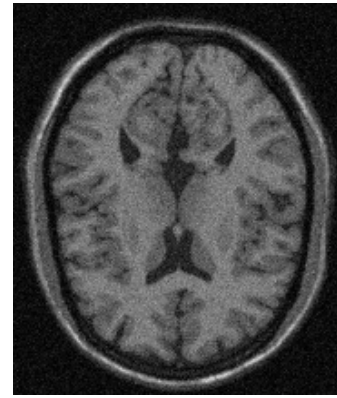


Figure 5.8: Denoising of some real noisy color images. Closer inspection shows that our proposed method produces results with less residual noise and more textural detail than the commercial Neat Image method (<http://www.neatimage.com>). High resolution versions of images shown above can be compared at <http://users.soe.ucsc.edu/~priyam/PLOW/>.

pare our results to the commercial Neat Image™ denoising method that specifically handles intensity-dependent noise profiles. Even though the noise is correlated with the underlying image, our method suppresses the noise effectively, while retaining the finer details. Such performance encourages us to apply PLOW to denoising images captured from different sources, under different conditions with widely different noise characteristics. For example, in magnetic resonance imaging (MRI) and low-light images the noise statistics are known to be Rician and Poisson distributed respectively. Moreover, noise in old photographs can also be patterned, as is visible in the flat sky



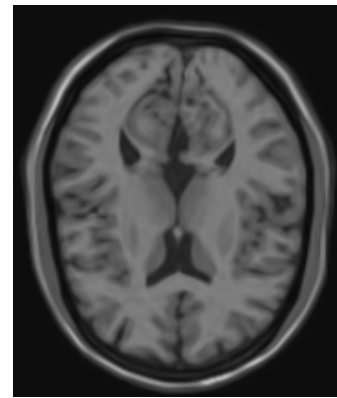
(a) Old photograph



(b) Brain MRI



(c) Denoised



(d) Denoised



(e) Low-light image



(f) Denoised

Figure 5.9: Restoration of images with non-Gaussian noise profiles: (a) an old photograph, (b) MRI of human brain, and (e) a low-light image. (c), (d) and (f) are images denoised by PLOW.

region of Fig. 5.9(a). Although our method is developed based on an optimality criterion that holds for signal-independent noise, it performs quite well in restoring images with such diverse noise profiles, as demonstrated in Fig. 5.9.

We point out here that the parameters used for our method are kept fixed across all noise levels and images. For all our experiments, we use a patch size of $n = 11 \times 11$, with the number of clusters K set to 15. The smoothing parameter, which controls the amount of denoising, is also kept fixed at $h^2 = 1.75\sigma^2n$. In general, these parameters can be tuned on a per image basis, manually or using some no-reference image quality measure [67]. In our opinion, such tunable parameters make a method less practical. Results presented in here, thus, use the fixed parameter settings mentioned above. However, for highly textured images (*e.g.* boat and stream images), the noise variance tends to be over-estimated by Eq. 5.22 when considering strong noise ($\sigma = 25$). This results in slightly over-smoothened denoised images. For such cases, we provided our algorithm with a lower noise variance.

Summary – In this chapter, we proposed PLOW - a patch-based locally optimal Wiener filter for image denoising. The proposed method was motivated from the formulation of the denoising bound introduced in Chapter 2. By design, the PLOW filter makes use of both geometrically and photometrically similar patches to estimate different filter parameters. We showed that the parameters can be estimated quite accurately from the given noisy image in a framework which is similar to the bounds estimation process of Chapter 3. The resulting denoised images were shown to be comparable or improving upon the state-of-the-art in terms of visual quality and different quan-

titative measures. We also showed that our PLOW method is capable of effectively suppressing noise for a wide variety of applications, even when the noise profile is not intensity-independent iid Gaussian for which PLOW was derived.

5A Derivation of Noise Covariance ($\mathbf{C}_{\underline{\zeta}_i}$) for Similarity Model

Here we derive an expression for the covariance matrix $\mathbf{C}_{\underline{\zeta}_i}$ based on the data model (Eq. 5.11)

$$\underline{\mathbf{y}}_i = \mathbf{A}_i \mathbf{z}_i + \underbrace{\underline{\boldsymbol{\varepsilon}}_i + \underline{\boldsymbol{\eta}}_i}_{\underline{\zeta}_i}, \quad (5.30)$$

where $\underline{\boldsymbol{\varepsilon}}_i = [\dots \boldsymbol{\varepsilon}_{ij}^T \dots]^T$ and $\underline{\boldsymbol{\eta}}_i = [\dots \boldsymbol{\eta}_j^T \dots]^T$ obtained from all patches \mathbf{y}_j similar to a given \mathbf{y}_i . As per definition of $\boldsymbol{\varepsilon}_{ij}$ (Eq. 3.4), $\underline{\boldsymbol{\varepsilon}}_i$ and $\underline{\boldsymbol{\eta}}_i$ are independent of each other, which leads to the covariance matrix being

$$\mathbf{C}_{\underline{\zeta}_i} = \mathbf{C}_{\underline{\boldsymbol{\eta}}_i} + \mathbf{C}_{\underline{\boldsymbol{\varepsilon}}_i}. \quad (5.31)$$

Assuming the noise $\boldsymbol{\eta}_i$ to be iid, the covariance matrix $\mathbf{C}_{\underline{\boldsymbol{\eta}}_i}$ takes the form

$$\mathbf{C}_{\underline{\boldsymbol{\eta}}_i} = \sigma^2 \mathbf{I}_q, \quad (5.32)$$

where \mathbf{I}_q is the $q \times q$ identity matrix with dimension $q = nN_i$ dependent on the level of redundancy exhibited by the \mathbf{y}_i patch. Further, assuming that the pixels within a (noise-free) patch \mathbf{z}_i and, as a result, $\boldsymbol{\varepsilon}_{ij}$ are iid, we obtain a diagonal form for $\mathbf{C}_{\underline{\boldsymbol{\varepsilon}}_i}$. The diagonal elements for this matrix can be derived from the definition of Eq. 3.4 as

$$\begin{aligned} \boldsymbol{\varepsilon}_{ij} &= \mathbf{z}_j - \mathbf{z}_i = (\mathbf{y}_j - \mathbf{y}_i) - (\boldsymbol{\eta}_j - \boldsymbol{\eta}_i) \\ \Rightarrow E[\|\boldsymbol{\varepsilon}_{ij}\|^2] &= E[\|(\mathbf{y}_j - \mathbf{y}_i) - (\boldsymbol{\eta}_j - \boldsymbol{\eta}_i)\|^2] \end{aligned}$$

$$\begin{aligned}
&= E[\|\mathbf{y}_j - \mathbf{y}_i\|^2] + E[\|\boldsymbol{\eta}_j - \boldsymbol{\eta}_i\|^2] - 2E[(\mathbf{y}_j - \mathbf{y}_i)^T(\boldsymbol{\eta}_j - \boldsymbol{\eta}_i)] \\
&= E[\|\mathbf{y}_j - \mathbf{y}_i\|^2] + 2\sigma^2n \\
&\quad - 2(E[(\mathbf{z}_j - \mathbf{z}_i)^T(\boldsymbol{\eta}_j - \boldsymbol{\eta}_i)] + E[(\boldsymbol{\eta}_j - \boldsymbol{\eta}_i)^T(\boldsymbol{\eta}_j - \boldsymbol{\eta}_i)]) \\
&= E[\|\mathbf{y}_j - \mathbf{y}_i\|^2] + 2\sigma^2n - 2E[\|\boldsymbol{\eta}_j - \boldsymbol{\eta}_i\|^2] \\
&= E[\|\mathbf{y}_j - \mathbf{y}_i\|^2] - 2\sigma^2n, \tag{5.33}
\end{aligned}$$

where the second-to-last step assumes the noise to be independent of \mathbf{z} . As mentioned before, assuming the $\boldsymbol{\varepsilon}_{ij}$ vectors to be iid, we can write

$$\mathbf{C}_{\boldsymbol{\varepsilon}_{ij}} = \left(\frac{1}{n} E[\|\mathbf{y}_j - \mathbf{y}_i\|^2] - 2\sigma^2 \right) \mathbf{I} \tag{5.34}$$

$$\Rightarrow \mathbf{C}_{\boldsymbol{\varepsilon}_i} = \begin{bmatrix} \ddots & & \mathbf{0} \\ & \mathbf{C}_{\boldsymbol{\varepsilon}_{ij}} & \\ \mathbf{0} & & \ddots \end{bmatrix}, \tag{5.35}$$

from which we obtain the covariance matrix $\mathbf{C}_{\boldsymbol{\zeta}_i}$ as

$$\mathbf{C}_{\boldsymbol{\zeta}_i} = \mathbf{C}_{\boldsymbol{\varepsilon}_i} + \mathbf{C}_{\boldsymbol{\eta}_i} = \begin{bmatrix} \ddots & & \mathbf{0} \\ & \delta_{ij}^2 \mathbf{I} & \\ \mathbf{0} & & \ddots \end{bmatrix}, \tag{5.36}$$

where $\delta_{ij}^2 = \frac{1}{n} E[\|\boldsymbol{\varepsilon}_{ij}\|^2] + \sigma^2 = \frac{1}{n} E[\|\mathbf{y}_j - \mathbf{y}_i\|^2] - \sigma^2$. Note that in the above expression, the covariance for the $\boldsymbol{\varepsilon}_i$ and, hence, $\boldsymbol{\zeta}_i$ are estimated patch-wise, whereas the covariance related to the (homogeneous) noise $\boldsymbol{\eta}_i$ only varies in dimensionality depending on the redundancy level of the patch under consideration.

5B Derivation of Redundancy Exploiting Wiener Filter

Here we derive the LMMSE estimator for the data model in Eq. 5.11. As shown in Eq. 5.12, the LMMSE estimator for each patch can be obtained using its N_i nearest neighbors as

$$\hat{\mathbf{z}}_i = \bar{\mathbf{z}} + \left(\mathbf{C}_{\mathbf{z}}^{-1} + \mathbf{A}_i^T \mathbf{C}_{\underline{\zeta}_i}^{-1} \mathbf{A}_i \right)^{-1} \mathbf{A}_i^T \mathbf{C}_{\underline{\zeta}_i}^{-1} \left(\underline{\mathbf{y}}_i - \mathbf{A}_i \bar{\mathbf{z}} \right), \quad (5.37)$$

where $\mathbf{A}_i = [\mathbf{I} \dots \mathbf{I}]^T$ is formed by stacking N_i identity matrices, each of size $n \times n$. With $\mathbf{C}_{\underline{\zeta}_i}$ having a diagonal form (Eq. 5.13), we can simplify Eq. 5.37 by noting that

$$\begin{aligned} \mathbf{A}_i^T \mathbf{C}_{\underline{\zeta}_i}^{-1} \left(\underline{\mathbf{y}}_i - \mathbf{A}_i \bar{\mathbf{z}} \right) &= [\mathbf{I} \dots \mathbf{I}] \begin{bmatrix} \ddots & & \mathbf{0} \\ & [\delta_{ij}^{-2} \mathbf{I}] & \\ \mathbf{0} & & \ddots \end{bmatrix} \left(\begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_{N_i} \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{z}} \\ \vdots \\ \bar{\mathbf{z}} \end{bmatrix} \right) \\ &= \begin{bmatrix} \dots & \delta_{ij}^{-2} \mathbf{I} & \dots \end{bmatrix} \begin{bmatrix} \vdots \\ (\mathbf{y}_j - \bar{\mathbf{z}}) \\ \vdots \end{bmatrix} \\ &= \sum_{j=1}^{N_i} \delta_{ij}^{-2} (\mathbf{y}_j - \bar{\mathbf{z}}), \quad \text{and,} \end{aligned} \quad (5.38)$$

$$\mathbf{A}_i^T \mathbf{C}_{\underline{\zeta}_i}^{-1} \mathbf{A}_i = \begin{bmatrix} \dots & \delta_{ij}^{-2} \mathbf{I} & \dots \end{bmatrix} \begin{bmatrix} \mathbf{I} \\ \vdots \\ \mathbf{I} \end{bmatrix} = \sum_{j=1}^{N_i} \delta_{ij}^{-2} \mathbf{I}. \quad (5.39)$$

This gives rise to a simplified LMMSE estimator expression having the form

$$\begin{aligned} \hat{\mathbf{z}}_i &= \bar{\mathbf{z}} + \left(\mathbf{C}_{\mathbf{z}}^{-1} + \sum_{j=1}^{N_i} \delta_{ij}^{-2} \mathbf{I} \right)^{-1} \sum_{j=1}^{N_i} \delta_{ij}^{-2} (\mathbf{y}_j - \bar{\mathbf{z}}) \\ &= \bar{\mathbf{z}} + \left[\sum_{j=1}^{N_i} \delta_{ij}^{-2} \left(\frac{\mathbf{C}_{\mathbf{z}}^{-1}}{\sum_{j=1}^{N_i} \delta_{ij}^{-2}} + \mathbf{I} \right) \right]^{-1} \sum_{j=1}^{N_i} \delta_{ij}^{-2} (\mathbf{y}_j - \bar{\mathbf{z}}) \end{aligned} \quad (5.40)$$

$$= \bar{\mathbf{z}} + \left(\frac{\mathbf{C}_{\mathbf{z}}^{-1}}{\sum_{j=1}^{N_i} \delta_{ij}^{-2}} + \mathbf{I} \right)^{-1} \sum_{j=1}^{N_i} \frac{\delta_{ij}^{-2}}{\sum_{j=1}^{N_i} \delta_{ij}^{-2}} (\mathbf{y}_j - \bar{\mathbf{z}}). \quad (5.41)$$

Implementing this estimator requires inverting $\mathbf{C}_{\mathbf{z}}$. However, $\mathbf{C}_{\mathbf{z}}$ can be ill-conditioned and even rank deficient, leading to inaccurate estimation of \mathbf{z}_i . To alleviate this problem, we make use of the matrix inversion lemma [96] to obtain a form that does not require inversion of $\mathbf{C}_{\mathbf{z}}$:

$$\left(\frac{\mathbf{C}_{\mathbf{z}}^{-1}}{\sum_j \delta_{ij}^{-2}} + \mathbf{I} \right)^{-1} = \mathbf{I} - \left(\sum_j \delta_{ij}^{-2} \mathbf{C}_{\mathbf{z}} + \mathbf{I} \right)^{-1}. \quad (5.42)$$

This leads to an alternative expression for the LMMSE estimator as

$$\begin{aligned} \hat{\mathbf{z}}_i &= \bar{\mathbf{z}} + \left[\mathbf{I} - \left(\sum_j \delta_{ij}^{-2} \mathbf{C}_{\mathbf{z}} + \mathbf{I} \right)^{-1} \right] \sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} (\mathbf{y}_j - \bar{\mathbf{z}}) \\ &= \bar{\mathbf{z}} + \sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} (\mathbf{y}_j - \bar{\mathbf{z}}) - \left(\sum_j \delta_{ij}^{-2} \mathbf{C}_{\mathbf{z}} + \mathbf{I} \right)^{-1} \sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} (\mathbf{y}_j - \bar{\mathbf{z}}) \\ &= \sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} \mathbf{y}_j - \left(\sum_j \delta_{ij}^{-2} \mathbf{C}_{\mathbf{z}} + \mathbf{I} \right)^{-1} \sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} (\mathbf{y}_j - \bar{\mathbf{z}}) \\ &= \sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} \left[\mathbf{y}_j - \left(\sum_j \delta_{ij}^{-2} \mathbf{C}_{\mathbf{z}} + \mathbf{I} \right)^{-1} (\mathbf{y}_j - \bar{\mathbf{z}}) \right] \\ &= \left[\sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} \mathbf{y}_j \right] - \left[\sum_j \frac{\delta_{ij}^{-2}}{\sum_j \delta_{ij}^{-2}} \left(\sum_j \delta_{ij}^{-2} \mathbf{C}_{\mathbf{z}} + \mathbf{I} \right)^{-1} (\mathbf{y}_j - \bar{\mathbf{z}}) \right]. \quad (5.43) \end{aligned}$$

Note that the first part of the above formulation is closely related to NLM [12] (using weights $w_{ij} = \delta_{ij}^{-2}$) with an added term that processes the residuals between the noisy patches and the estimated mean patch.

5C Derivation of Approximate Similarity Measure

In Sec. 5.3, we derived an extension for the Wiener filter where photometrically similar patches contribute in denoising a given reference patch. This was ana-

lyzed through a modified per-patch data model. In the final filter formulation, the contributing weights of photometrically similar patches were determined to be $w_{ij} = \delta_{ij}^{-2}$ where (Eq. 5.14)

$$\delta_{ij}^2 = \frac{1}{n} E [\|\mathbf{z}_i - \mathbf{z}_j\|^2] + \sigma^2. \quad (5.44)$$

However, computing δ_{ij} for a pair of \mathbf{z}_i and \mathbf{z}_j random vectors is not practical given only the two observations. Further, this would require access to the noise-free image. In practice, δ_{ij} and, hence, w_{ij} need to be estimated from the corresponding noisy observations \mathbf{y}_i and \mathbf{y}_j respectively. However, let us first assume that the noise-free patches are made available to us. Next, we show that the weight formulation employed in Eq. 5.25 is simply an approximation that can be derived from Eq. 5.44.

Let us re-write Eq. 5.25 in terms of the original noise-free image patches as

$$w_{ij} = \frac{1}{\sigma^2} \exp \left\{ -\frac{\|\mathbf{z}_i - \mathbf{z}_j\|^2}{h^2} \right\} \quad (5.45)$$

$$\Rightarrow \delta_{ij}^2 = w_{ij}^{-1} = \sigma^2 \exp \left\{ \frac{\|\mathbf{z}_i - \mathbf{z}_j\|^2}{h^2} \right\}. \quad (5.46)$$

Note that the above equation would be the ideal weights that we *estimate* using the noisy observations \mathbf{y}_i and \mathbf{y}_j in Eq. 5.25. Let us define $\lambda = \frac{\|\mathbf{z}_i - \mathbf{z}_j\|^2}{h^2}$. Since we consider only photometrically similar patches that satisfy the condition in Eq. 3.4, we know that $\|\mathbf{z}_i - \mathbf{z}_j\|^2 \leq \gamma^2 \ll \sigma^2 n$. Thus, by choosing $h^2 \geq \sigma^2 n$, we can guarantee that $\lambda < 1$. Consequently, as h^2 increases, λ approaches 0. We can then write the Taylor expansion of the exponential function around $\lambda = 0$ as

$$\begin{aligned} e^\lambda &= 1 + \lambda + O(\lambda^2) \approx 1 + \lambda && \text{[since } \lambda < 1\text{]} \\ &= 1 + \frac{\|\mathbf{z}_i - \mathbf{z}_j\|^2}{\sigma^2 n} \\ \Rightarrow \delta_{ij}^2 &= \sigma^2 e^\lambda \approx \sigma^2 + \frac{1}{n} \|\mathbf{z}_i - \mathbf{z}_j\|^2. \end{aligned} \quad (5.47)$$

Comparing Eq. 5.44 with the above expression, it is easy to observe their similarities. As mentioned earlier, the expected value of Eq. 5.44 cannot be calculated accurately from a single pair of \mathbf{z}_i and \mathbf{z}_j observations. As a result, it is ignored when computing the weights.

The above derivation assumed knowledge of the distance between the noise-free similar patches. As such, these are the “oracle” weights that we would ideally want to use for denoising. However, in practice only the noisy patches are observed. Consequently, the actual weight function is approximated by replacing the noise-free $\mathbf{z}_i, \mathbf{z}_j$ patches with their corresponding noisy $\mathbf{y}_i, \mathbf{y}_j$ observations, giving rise to the expression in Eq. 5.25 as

$$w_{ij} = \frac{1}{\sigma^2} \exp \left\{ -\frac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{h^2} \right\}. \quad (5.48)$$

Note that the distance between noisy patches can be much higher than those between the underlying noise-free patches. As a result, a larger smoothing term is needed for denoising. In our work, we set $h^2 = 1.75\sigma^2n$ for all noise levels and images.

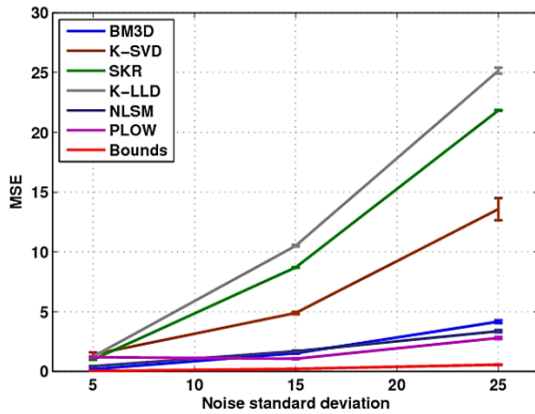
Chapter 6

Conclusions and Future Work

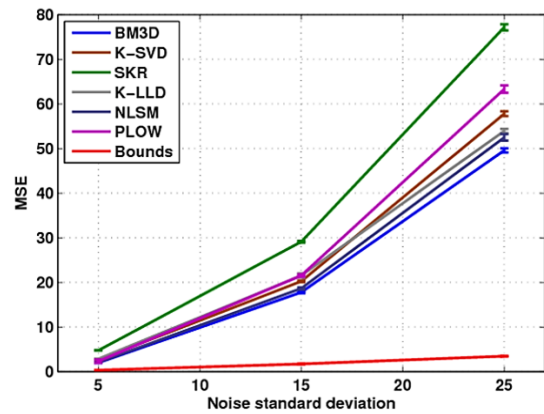
6.1 Conclusions

In this thesis, we analyzed the problem of image denoising in detail¹. We derived an expression that quantifies the performance limit of any affine-biased denoising method in Chapter 2. We showed that such a limit depends on the image content as well as the statistical characteristics of the corrupting noise. The dependence on image content is accounted for in the form of patch complexity and variability, as well as the photometric redundancies for the case of WGN. The formulation, thus, provides a quantification of the advantage in exploiting patch redundancies that has found widespread use since it was first proposed in [12, 13]. The obvious advantage of non-local frameworks is demonstrated in Fig. 6.1(a) where we compare the denoising performance for various methods for the synthetically generated stripes image containing multiple exact replicas of each patch. For strong noise, the non-local

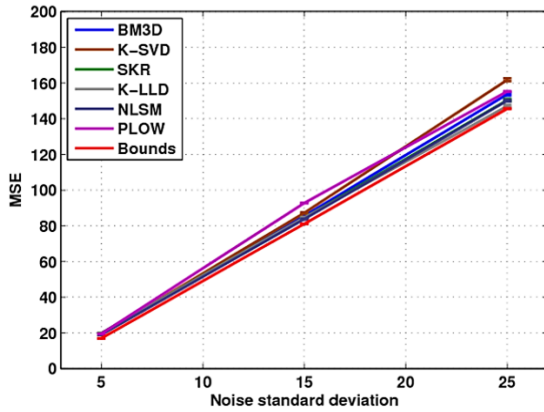
¹Parts of this thesis have been published in various refereed journals and conferences which appear in the bibliography as [24, 29, 68, 70, 110, 130, 134–137]. Work presented in Chapter 5 is currently under review [69].



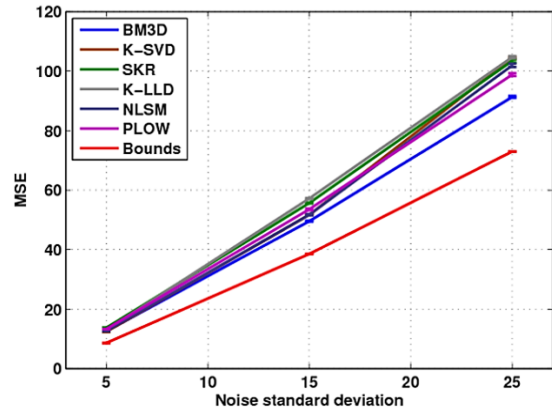
(a) Stripes image



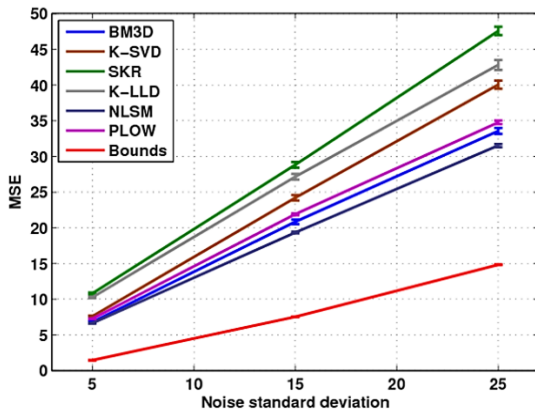
(b) Box image



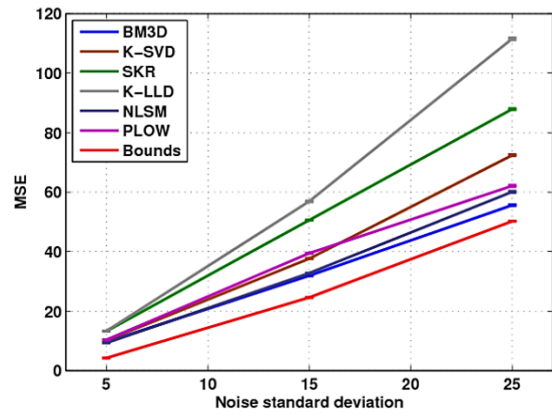
(c) Grass image



(d) Cloth image



(e) House image



(f) Barbara image

Figure 6.1: Comparison of denoising results with MSE bounds for some benchmark images corrupted by varying levels of additive WGN.

methods, namely BM3D [49], NLSM [43] and PLOW [68, 69], clearly outperform the local approaches. High levels of redundancy as well as low patch complexity result in our bounds predicting a very small lower bound even for quite strong noise levels ($\sigma = 25$).

On the other hand, images that contain mostly semi-stochastic texture (e.g., grass image of Fig. 2.2(d)) demonstrate high variability among its patches, with each patch itself being structurally quite complex. The bound for this particular image is much higher in comparison to the simpler stripes image. Low redundancy levels also translate to indistinguishable performance difference between the local and non-local methods, as shown in Fig. 6.1(c). Interestingly, for this image, the predicted bound is very close to the MSE obtained by the methods, implying almost no room for performance improvement. A similar observation is made for the somewhat less random cloth image as well (Fig. 6.1(d)), although in this case some modest improvement is still possible.

Most natural images consist of a mix of simple and complex patches. In Chapter 4, we showed that patch complexity (captured by the covariance matrix \mathbf{C}_z) is related to patch redundancy. Intuitively, we can expect lower redundancy levels for more complex patches (e.g., patches in corner and texture regions), whereas more similar patches can be expected for smoother patches. This leads to better denoising for images lacking much texture. Our bound formulation, being data dependent, is in keeping with this intuition, as illustrated in Fig. 6.1(e) & (f). As expected, the bounds for the house image is quite lower than those for the Barbara image. This relative denoising difficulty is also seen in the performance of the methods to which we have compared.

Table 6.1: Some images ranked according to improvement in denoising yet to be achieved, as predicted by our bounds. The noise standard deviation is 25 and the bounds are calculated using 11×11 patches.

Image	K-SVD [37]	SKR [22]	K-LLD [24]	BM3D [49]	NLSM [43]	PLOW	Bound	Relative Efficiency ¹ (RE)	(in dB)
Box	57.78	77.17	53.93	49.56	52.49	63.31	3.42	0.069	11.61
Stripes	13.56	21.83	25.15	4.16	3.36	2.79	0.55	0.197	7.05
House	40.05	47.57	42.82	33.57	31.56	34.77	14.82	0.470	3.28
Lena	48.09	44.09	46.02	40.46	42.57	41.79	19.66	0.486	3.13
Boat	78.39	78.44	77.45	67.17	69.20	71.46	38.70	0.576	2.40
Cloth	104.36	103.42	104.68	91.33	101.97	98.82	72.98	0.799	0.97
Barbara	72.39	87.91	111.58	55.62	60.13	62.09	50.24	0.903	0.44
Grass	161.74	150.39	147.13	153.64	150.16	155.26	145.58	0.990	0.05
Mandrill	185.60	196.20	195.75	188.84	178.94	192.66	181.61	–	–

$$^1 \text{Relative Efficiency (RE)} = \frac{\text{MSE Bound}}{\text{MSE of best performing method}}$$

dB figures are: $-10 \log_{10}(\text{RE})$, which indicate room for improvement.

Of the natural images used as benchmarks in this thesis, the Barbara image is particularly interesting. Although rich in texture, a majority of the complex patches follow definite patterns, thus providing higher redundancy levels than the texture regions of most other images. The presence of more similar patches is advantageous to the non-local methods. Not surprisingly, these methods achieve much better performance than the local denoising filters, especially when the corrupting noise is strong (see Fig. 6.1(f)). That the non-local methods exploit such redundancies efficiently is demonstrated by the fact that the performance of such methods is comparable to the bounds. Observe that the room for improvement is much smaller than the smoother house image indicating the possibility of better redundancy exploitation, but more than the highly textured grass image where very few similar patches are observed.

Comparing performances of the state-of-the-art methods to the bounds allows us to gauge the room for improvement in denoising performance of any given

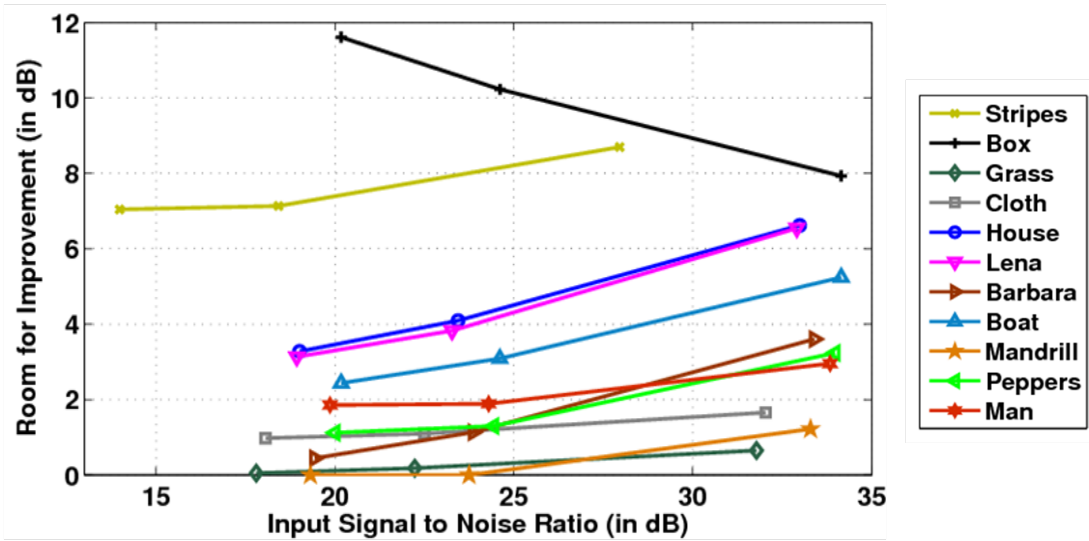


Figure 6.2: Comparison of denoising results with MSE bounds for some benchmark images corrupted by varying levels of additive WGN.

image. In Table 6.1 we rank various benchmark images based on their relative efficiency (RE) which is calculated as the ratio between the MSE bound and the MSE of the best denoising method for each image. A graphical representation of the room for improvement for various images as a function of input signal-to-noise ratios (SNR) is presented in Fig. 6.2. The plots for various images there indicate possible improvement for almost all images at higher SNRs which usually corresponds to weak noise. However, at the lower SNRs (strong noise) the prediction is quite different. Interestingly, the graph can be split roughly into three regions based on the amount of improvement that can be expected at the lower SNRs. At one end of the spectrum lie the synthetic images where many exact replicas exist for each image patch. Although denoising performance for such images are impressive (Fig. 6.1(a) & (b)), the bounds predict large possible improvements (in terms of dB).

Although the synthetic images are useful to study the effects of patch redun-

dancies, large numbers of exact replicas present in those images rarely occur in natural images. As a result, the bounds for natural images are usually much higher. The room for improvement for natural images can also be seen to be much lower than those for the synthetic images used in our study. Even for these natural images, the plots at low SNRs can be segregated into two regions. At one end lie the predominantly textured images where high patch complexity and variability (and, hence, lower redundancies) result in high MSE bounds. When compared to the best denoising methods, little to no room for improvement is predicted for this particular class of images that are rich in semi-stochastic texture. On the other hand, denoising performance for smoother images that lack such widespread complex structures (e.g., house image) can still be considerably improved. Note that such images also typically exhibit higher levels of patch (photometric) redundancies. Not surprisingly, the *non-local* approaches enjoy a clear performance advantage for such images.

From the above discussion, it becomes apparent that image denoising as a problem is not dead – yet. This is particularly true for the class of smoother images containing sufficiently large number of repeating patterns. On the surface, this may appear to be in direct contradiction to the observations in [76] where Levin and Nadler compared the bounds to the best denoising methods and concluded that the performance of current non-parametric approaches cannot be improved upon, unless considerably larger patches are used. Note that the authors there study the bounds as a *pixel-wise* estimation problem, where prior information is learned using a vast database of image patches. Larger patch sizes then allow information from spatially farther pixels to be used in denoising a pixel of interest. In a sense, this translates to better performance

being possible for methods capable of exploiting not necessarily local redundancies efficiently. This is directly in keeping with our conclusions.

Using this as the motivation, we designed a non-local LMMSE filter (PLOW) in Chapter 5 where, apart from geometrically similar patches, photometric similarity was used to achieve impressive denoising performance. The challenge there, as with any other non-local denoising method, was to accurately identify the similar patches from their noisy observations. This task becomes non-trivial for patches where the local SNR is low; that is, when the corrupting noise overwhelms the underlying patch structure. For such extreme conditions, inaccuracies in clustering and estimating the moment parameters (\bar{z}, C_z) contributed to further loss of performance. However, our experiments there show that the PLOW method is quite robust to minor inaccuracies and performs considerable image restoration even for strong noise cases that one may expect to encounter in general imaging applications. Moreover, our experiments also indicate that our method is capable of effectively addressing images corrupted by many different noise profiles.

As mentioned earlier, our PLOW filter was designed explicitly to achieve the performance bounds. Yet, for the smoother natural images, it fails to achieve the theoretic limits, even when the various filter parameters are estimated from the noise-free ground truth. This naturally raises the question of achievability of the bounds. To answer this, we must point out that our bounds were derived assuming that parameters of the bound such as the clusters and their moments are known a priori. However, as these parameters themselves are *estimated*, the practically reachable optimal MSEs may be somewhat higher than what the bounds predict. One can then expect to obtain tighter denoising bounds by taking into consideration the limits of estimating these

parameters as well. However, comparing the bounds for some textured images to the performance of the denoising methods (Figures 6.1(a), (c), (d) & (f)), it is clear that the bounds developed here are not overly optimistic.

However, for smoother natural images (example, house image), even the optimal denoising method may not achieve the bounds which were derived assuming N_i photometrically *identical* patches. These N_i patches would then contribute equally in the denoising process, i.e. $w_{ij} = 1/\sigma^2$. In practice, however, the patches that appear similar still contain minor dissimilarities. As a result, the w_{ij} weights deviate from their ideal $1/\sigma^2$ values even when computed from the (noise-free) ground truth images. Using a large number of similar patches increases the contribution of such errors, forcing us to consider only a few (say $N_i \leq 10$) similar patches in denoising any reference patch using PLOW. Moreover, the weights w_{ij} used to compute the contribution of any given similar patch is only an approximation of the optimal weights (see Sec. 5C). The combination of these practical limitations in dealing with patch redundancies handicaps even the “oracle” PLOW estimator, resulting in higher MSEs than those predicted by the bounds. This is especially true for the class of smoother images. Note that these different practical limitations affect the performance of all non-local estimators, making our bounds formulation an effective lower bound. This highlights the need for combining similar patches through intelligent handling of their minor dissimilarities to improve denoising performance for smoother images. This should be the focus in designing the next generation of denoising filter.



Figure 6.3: Image formation model showing the different degradation steps that the image goes through due to camera hardware limitations. The captured (raw) signal undergoes further in-camera processing before it is finally stored.

6.2 Future Works and Extensions

In this thesis, we analyzed the problem of image denoising. This study was focused on non-local patch-based methods, particularly for grayscale images. Although this in itself is important and interesting, it is restrictive and not completely indicative of real world scenarios. In this section, we discuss possible extensions to make our work applicable to different practical applications.

6.2.1 Extending PLOW to Different Degradation Models

The problem of denoising requires us to estimate the underlying image patches given their noisy observations. In formulating the problem as such, we neglect the effect of other factors that degrade the quality of captured images. The input signal can be thought to be some (unknown) ideal image that represents the scene being captured. As illustrated in Fig. 6.3, each patch from the actual or ideal image undergoes various degradations before it is stored in the camera. Since images are stored in digital form with a fixed sampling frequency, it is important to ensure that the input signal is band-limited so as to avoid unwanted aliasing effects. The captured image can therefore be considered to be a blurred version of the ideal sharp image whose patches are z_i . This

blurring process can be modeled by a matrix operator \mathbf{H} which is dependent on the point spread function (PSF) of the camera.

The blurred image is then downsampled according to the finite grid of sensor elements in the camera. This downsampling process can be thought of as selecting every few pixels of the blurred image and can be modeled by a decimation operator \mathbf{D} . For color images, each sensor (or pixel) captures information pertaining to a particular range of wavelength (color) forming a mosaiced image sampled according to some color filter array of which the Bayer pattern [138] is the most popular. Denoting this sampling process as another matrix operation (\mathbf{B}), we can mathematically express each observed image patch as

$$\mathbf{y}_i = \mathbf{BDH}\mathbf{z}_i + \boldsymbol{\eta}_i, \quad (6.1)$$

where $\boldsymbol{\eta}_i$ is the noise that arises from various sources within the camera. The degraded image that is formed at this stage is often referred to as the raw image. The final full-color image that is finally seen by the user undergoes many different in-camera processes that include demosaicing, color mapping, gamma correction, and compression among others.

Depending upon the application, it is often necessary to invert the effects of one or more of the degradations shown in Fig. 6.3. Below we discuss how the PLOW method and our bounds analysis can be extended for such problems.

Denoising color images – It is easy to see that for the problem of denoising grayscale images, all the degradations other than noise are neglected and the target image to be recovered is considered to be grayscale. In that case, the patch-wise observation model

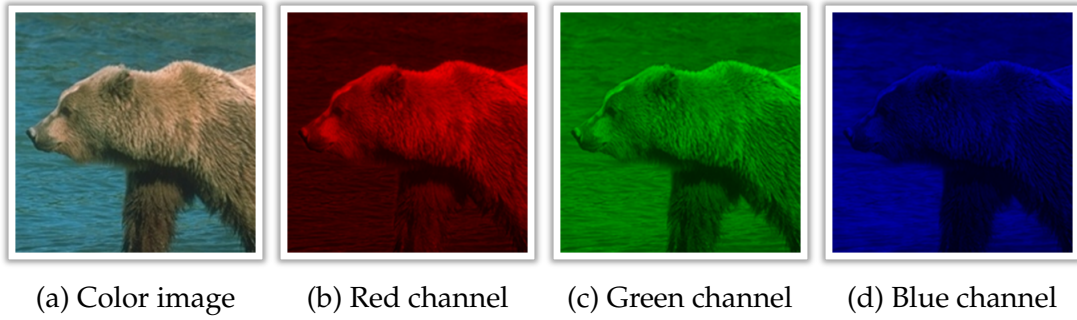


Figure 6.4: Illustration of correlation among the red, green and blue color channels. Observe that texture and edges in any one channel corresponds to those in other channels.

is written as (Eq. 2.1)

$$\mathbf{y}_i = \tilde{\mathbf{z}}_i + \boldsymbol{\eta}_i, \quad (6.2)$$

where the objective is to recover the $\tilde{\mathbf{z}}_i = \mathbf{D}\mathbf{H}\mathbf{z}_i$ patches. When considering color images, it is common practice to assume full-color (or demosaiced) images. The $\tilde{\mathbf{z}}_i$ patches then contain information about red, green and blue color channels. In [131], Hunt showed that these color channels are correlated. A simple illustration showing such correlation for the bears image is shown in Fig. 6.4. There we see that edges and texture regions exist in the same location in all color channels, although the strength of such edges and intensity of the pixels may differ considerably.

A popular approach of handling such correlations in denoising color images is to perform some color transformation that reduces the correlation among the different channels. Transformation to luminance-chrominance channels, such as Y-Cb-Cr, is widely used. Since the human vision system is more sensitive to intensity variations than to changes in chroma, aggressive denoising is usually performed in the Cb-Cr channels, while the intensity (Y) channel is treated more carefully. This results in reduced denoising complexity in comparison to treating each channel equally. However, one disadvantage of such an approach is that color transforms tend to alter noise char-

acteristics. Many denoising methods rely on the fact that the corrupting noise is uncorrelated, spatially and across channels. Unfortunately, the latter property no longer holds in transformed color spaces. It is, thus, useful to address color image denoising in the RGB color space, taking into account the dependencies across color channels.

Although developed for denoising grayscale images, our PLOW method can be easily adapted for denoising color images, as shown in Chapter 5. There the input noisy color image was first transformed to Y-Cb-Cr color space, and the luminance channel was used to perform geometric clustering of image patches. The motivation for this was that the structure of the patches remain roughly the same across all color channels (Fig. 6.4) and, hence, it is not necessary to cluster each color channel individually. Once clustering information was obtained, each of the red, green and blue color channels were denoised independently. This is tantamount to processing each color channel as a separate grayscale image. Although considerable denoising performance was achieved, this is clearly sub-optimal considering that it disregards the inter-dependence of structure across color channels. Lukin *et al.* [139] showed that denoising color images taking into account such cross-color information can lead to lower MSE than what our bounds predict separately for each color channel.

We consider improving denoising performance for color images using cross-color dependencies as a possible extension to our PLOW method. For color images, each observed color patch \mathbf{y}_i^c can be written in the form of Eq. 6.2 as

$$\underbrace{\begin{bmatrix} \mathbf{y}_i^R \\ \mathbf{y}_i^G \\ \mathbf{y}_i^B \end{bmatrix}}_{\mathbf{y}_i^c} = \underbrace{\begin{bmatrix} \tilde{\mathbf{z}}_i^R \\ \tilde{\mathbf{z}}_i^G \\ \tilde{\mathbf{z}}_i^B \end{bmatrix}}_{\mathbf{z}_i^c} + \underbrace{\begin{bmatrix} \boldsymbol{\eta}_i^R \\ \boldsymbol{\eta}_i^G \\ \boldsymbol{\eta}_i^B \end{bmatrix}}_{\boldsymbol{\eta}_i^c}, \quad (6.3)$$

where \mathbf{z}_i^c is the underlying color patch formed by concatenating corresponding patches from the red, green and blue color channels. The PLOW filter can be developed for color images in exactly the same way as done for grayscale images in Chapter 5. To account for photometric redundancy, we need to identify patches that have similar intensity as well as color. An analogous LMMSE filter can be formulated for the color image patches as

$$\hat{\mathbf{z}}_i^c = \bar{\mathbf{z}}^c + \left(\mathbf{C}_{\mathbf{z}^c}^{-1} + \sum_{j=1}^{N_i} w_{ij} \mathbf{I} \right)^{-1} \sum_{j=1}^{N_i} w_{ij} (\mathbf{y}_j^c - \bar{\mathbf{z}}^c). \quad (6.4)$$

Note that $\mathbf{C}_{\mathbf{z}^c}$ being the covariance matrix estimated for the color patch, it automatically captures the correlation between pixels across the different color components.

In adapting the PLOW filter for color patches, the weighting term w_{ij} that measures similarity between patches \mathbf{y}_i^c and \mathbf{y}_j^c also needs careful consideration. The formulation for the weight can be similar to that used for grayscale images as

$$w_{ij} = \frac{1}{\sigma^2} \exp \left(-\frac{\|\mathbf{y}_i^c - \mathbf{y}_j^c\|^2}{h^2} \right), \quad (6.5)$$

with $h^2 = 1.75\sigma^2n$ being used for our experiments in Chapter 5. When the strength of the noise affecting the different color channels is similar, the above method for calculating the weights work well. However, this is not always the case in practice [44, 140]. The weight measure, as well as the covariance estimation step, then needs to be suitably modified to account for the different noise variances to prevent over-smoothing or under-smoothing in any color channel.

Denoising raw images – Denoising of color images, as explained above, involves estimating the color patches that have already been demosaiced from raw images. Such demosaicing is usually performed in the camera itself and the process does not usually

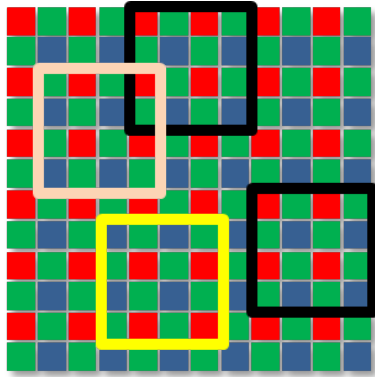


Figure 6.5: Illustration of patch formation in Bayer patterned raw images. The patches with black borders can be compared as they have the same pattern while those bordered in other colors should not be considered.

account for the presence of noise in the raw data. Since the demosaicing step involves interpolation of pixel values in each color channel by taking into account information from other color channels, it introduces spatial and cross-color correlation between the noise samples as well. Consequently, it is often advantageous to perform denoising before any color interpolation is performed [140–142].

Our PLOW method can be used for denoising raw images as well. Note that the raw image patches can be modeled exactly as that of color patches in Eq. 6.3, and the corresponding PLOW filter can be derived (Eq. 6.4). As with the color case, the covariance matrix implicitly takes into account the correlation among color channels in each patch for denoising. However, care must be taken to ensure that all patches considered have the same color sampling patterns and only such patches are compared in our non-local PLOW denoising framework. That is to say that, referring to Fig. 6.5, we should only make use of patches with black borders to learn the filter parameters and to compare to a reference patch, while those with different colored borders should

not be considered in the denoising process. However, in doing so we are forced to ignore useful information.

Considering only patches of a particular pattern restricts the amount of overlap between patches, thus limiting the number of patches within the image. As filter parameters for PLOW need to be estimated from fewer patches, and the number of similar patches observed are also reduced, it may well be that the denoised image remains somewhat noisy, especially when the input image is corrupted by strong noise. A possible way of improving denoising in such situations is to denoise the image multiple times, each time considering different patterned patches (say, patches with patterns matching the yellow bordered patch in Fig. 6.5). For the Bayer pattern, at most 4 different denoised versions of the raw image need to be obtained, which can then be combined to further suppress the residual noise. Once the raw image is denoised, demosaicing and other in-camera operations can be applied to produce a full color image.

PLOW for demosaicing, interpolation and deblurring – The image formation model of Eq. 6.1 can be written in a simpler form by combining different degradation operations into a single matrix operation as

$$\mathbf{y}_i = \mathbf{\Upsilon} \mathbf{z}_i + \boldsymbol{\eta}_i. \quad (6.6)$$

Depending on what operation is assimilated into $\mathbf{\Upsilon}$ and what is ignored, different problems in image processing can be formulated with the above patch model. For example, when color information is to be estimated from a (noisy) raw image, the above problem reduces to that of demosaicing – estimating the full color (RGB) patches \mathbf{z}_i , given their partial noisy observations. As with grayscale images, the local patch model

can be extended to a non-local variant accounting for all N_i patches that are photometrically similar to a reference patch \mathbf{y}_i as

$$\underline{\mathbf{y}}_i = \mathbf{A}_i \Upsilon \mathbf{z}_i + \underline{\boldsymbol{\zeta}}_i, \quad (6.7)$$

where \mathbf{A}_i is formed by vertical concatenation of N_i identity matrices, each of size $3n \times 3n$. Note that, since we are dealing with raw images, all patches in the image must have similar color sampling patterns. The LMMSE filter for the above data model can be written from Eq. 5.12 as

$$\hat{\mathbf{z}}_i = \bar{\mathbf{z}} + \left(\mathbf{C}_z^{-1} + \Upsilon^T \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} \mathbf{A}_i \Upsilon \right)^{-1} \Upsilon^T \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} \left(\underline{\mathbf{y}}_i - \mathbf{A}_i \Upsilon \bar{\mathbf{z}} \right). \quad (6.8)$$

One interesting observation here is that there is no constraint on invertibility of Υ . The estimator in Eq. 6.8 is, thus, capable of handling many different degradation models. Following the derivation of the PLOW filter in Chapter 5, we note that:

$$\begin{aligned} \Upsilon^T \mathbf{A}_i^T &= \Upsilon^T [\mathbf{I} \dots \mathbf{I}] = [\Upsilon^T \dots \Upsilon^T], \\ \Rightarrow \Upsilon^T \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} &= [\Upsilon^T \dots \Upsilon^T] \begin{bmatrix} \ddots & & \mathbf{0} \\ & [w_{ij} \mathbf{I}] & \\ \mathbf{0} & & \ddots \end{bmatrix} = [\dots [w_{ij} \Upsilon^T] \dots] \quad (6.9) \\ \Rightarrow \Upsilon^T \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} \left(\underline{\mathbf{y}}_i - \mathbf{A}_i \Upsilon \bar{\mathbf{z}} \right) &= [\dots [w_{ij} \Upsilon^T] \dots] \begin{bmatrix} \vdots \\ \mathbf{y}_j - \Upsilon \bar{\mathbf{z}} \\ \vdots \end{bmatrix} \\ &= \sum_{j=1}^{N_i} w_{ij} \Upsilon^T (\mathbf{y}_j - \Upsilon \bar{\mathbf{z}}), \quad \text{and} \\ \Upsilon^T \mathbf{A}_i^T \mathbf{C}_{\underline{\boldsymbol{\zeta}}_i}^{-1} \mathbf{A}_i \Upsilon &= [\dots [w_{ij} \Upsilon^T] \dots] \begin{bmatrix} \vdots \\ \Upsilon \\ \vdots \end{bmatrix} = \sum_{j=1}^{N_i} w_{ij} \Upsilon^T \Upsilon. \end{aligned}$$

This allows us to rewrite the expression for the estimate in Eq. 6.8 as

$$\Rightarrow \hat{\mathbf{z}}_i = \bar{\mathbf{z}} + \left(\mathbf{C}_z^{-1} + \sum_{j=1}^{N_i} w_{ij} \mathbf{\Upsilon}^T \mathbf{\Upsilon} \right)^{-1} \sum_{j=1}^{N_i} w_{ij} \mathbf{\Upsilon}^T (\mathbf{y}_j - \mathbf{\Upsilon} \bar{\mathbf{z}}), \quad (6.10)$$

where w_{ij} measures the similarity between the degraded \mathbf{y}_i and \mathbf{y}_j image patches.

Going back to the case of demosaicing, when $\mathbf{\Upsilon} = \mathbf{B}$, we note that the expression in Eq. 6.10 allows us to estimate the color patches from the raw observations by reversing the mosaicing process. In many existing demosaicing approaches found in the literature the raw image is considered to be noise-free. This implicitly assumes that the data has been denoised by some method prior to demosaicing. However, in the image formation pipeline, demosaicing is often applied before denoising. Thus, a demosaicing method that accounts for the presence of noise is highly desirable. Our generalized PLOW filter can be applied to demosaic noisy raw images.

The generalized PLOW filter of Eq. 6.10 provides a locally optimal solution for inversion of other degradations as well. When $\mathbf{\Upsilon} = \mathbf{H}$, the problem takes the form of deblurring the input image. For the case of $\mathbf{\Upsilon} = \mathbf{HD}$, we obtain a solution for the image interpolation problem. However, the challenge in all the cases lies in estimating the parameters of the filter, namely $\bar{\mathbf{z}}$ and \mathbf{C}_z . We consider the extension of our PLOW method to different degradation models as a possible future work.

6.2.2 Guided Filtering with Image Pairs

In image denoising, as well as many other applications discussed in the previous section, estimating the filter parameters from the input degraded image can be challenging. In Chapter 3, we presented ways of estimating the $\bar{\mathbf{z}}$ and \mathbf{C}_z parameters from the noisy image itself. However, when the corrupting noise is strong, such estimates can be erroneous. One approach of circumventing such errors is through the

use of image databases from which the parameters can be learned. This approach was advocated in [110], and also in [76] where the entire prior $p(\mathbf{z})$ was calculated from multiple noise-free images to compute denoising bounds. The motivation here is to ensure a rich enough collection of patches such that, for any given patch in the input (noisy) image, multiple similar noise-free patches are available in the database. Unfortunately, this cannot be guaranteed, given the vast variability in local structures of natural images. This is a limitation of any method employing databases for estimation. However, in applications where the span of patch structures in the input image is considerably limited, such approaches can be quite successful. That is to say that, if we were interested in denoising face images only, then a database of ground truth face images will be likely to be rich enough for the input patches.

A variant of this problem is when a set of images, each differently degraded, is used to estimate a single ideal image. One such interesting image restoration problem was framed by Petschnigg *et al.* [143] where a pair of images of a dark scene, one captured with flash on and the other with no flash, was used to estimate a single sharp noise-free image. In general, an image captured under low-light without flash is blurry and noisy, while the corresponding sharper flash image contains artificial color and shadows introduced by the flash. Following this work, many authors have proposed solutions to image restoration using pairs of images captured under different degradation models [144–146].

Our PLOW method can be adapted to be used in a guided filtering mechanism to restore a pair of flash and no-flash images. An LMMSE-based framework for such guided filtering was used to considerably improve upon the state-of-the-art results by Seo and Milanfar [147]. A patch-based non-local extension to that approach

could be employed within the PLOW framework. This can be an interesting possible application of the PLOW filter.

6.2.3 Accounting for Intensity Dependent Noise

In deriving the expression for the MSE bounds in Chapter 2 we assumed the noise to be iid and signal independent. The formulation of the bound is still quite general in the sense that the pdf of the noise is not limited to any specific form, as long as the regularity condition of Eq. 2.5

$$E \left[\frac{\partial \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}} \right] = \mathbf{0}, \quad \forall \mathbf{z} \quad (6.11)$$

is satisfied. The effect of noise characteristics on denoising performance is captured by the FIM. In Chapter 3, we presented ways of estimating the bounds assuming the corrupting noise to be WGN.

Although WGN forms a popular noise model for many denoising methods, a recent trend of increasing interest in addressing signal dependent noise can be seen [3, 44, 140, 142, 148, 149]. Such noise profiles are observed in practice for many imaging applications such as MRI and low-light imaging. The exact form of the bound, as shown in (2.24), does not hold in such cases. However, a similar derivation as that presented in Sec. 2.4.1 can be carried out to derive bounds for intensity dependent noise, as long as the condition in Eq. 6.11 is satisfied. As with the Gaussian noise case, the expression for the bounds for signal-dependent noise may also be indicative of the form of the filter that can achieve optimal performance for such noise profiles. We consider this to be a very interesting direction where the work in this thesis can be extended.

Bibliography

- [1] G. E. Healey and R. Kondepudy, "Radiometric CCD camera calibration and noise estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 267–276, March 1994.
- [2] Y. Tsin, V. Ramesh, and T. Kanade, "Statistical calibration of CCD imaging process," in *Proceedings of IEEE International Conference on Computer Vision*, Vancouver, B.C., July 2001, pp. 480–487.
- [3] H. Faraji and W. J. MacLean, "CCD noise removal in digital images," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2676–2685, September 2006.
- [4] F. Alter, Y. Matsushita, and X. Tang, "An intensity similarity measure in low-light conditions," in *Proceedings of European Conference on Computer Vision*, vol. 4, Graz, Austria, May 2006, pp. 267–280.
- [5] F. J. Anscombe, "The transformation of Poisson, binomial and negative-binomial data," *Biometrika*, vol. 35, no. 3/4, pp. 246–254, December 1948.
- [6] M. Mäkitalo and A. Foi, "Optimal inversion of the Anscombe transformation in low-count Poisson image denoising," *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 99–109, January 2011.
- [7] A. Foi, "Noise estimation and removal in MR imaging: the variance-stabilization approach," in *Proceedings of IEEE International Symposium on Biomedical Imaging*, Chicago, IL, April 2011, to appear.

- [8] L. P. Yaroslavsky, *Digital Picture Processing*. Secaucus, NJ: Springer-Verlag New York, Inc., 1985.
- [9] S. M. Smith and J. M. Brady, "SUSAN - A new approach to low level image processing," *International Journal of Computer Vision*, vol. 23, no. 1, pp. 45–78, 1997.
- [10] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of International Conference on Computer Vision*, Washington, DC, USA, January 1998, pp. 839–846.
- [11] H. Takeda, "Locally adaptive kernel regression methods for multi-dimensional signal processing," Ph.D. dissertation, University of California, Santa Cruz, September 2010.
- [12] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising methods, with a new one," *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.
- [13] S. P. Awate and R. T. Whitaker, "Unsupervised, information-theoretic, adaptive image filtering for image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 364–376, 2006.
- [14] M. Mahmoudi and G. Sapiro, "Fast image and video denoising via nonlocal means of similar neighborhoods," *IEEE Signal Processing Letters*, vol. 12, no. 12, pp. 839–842, December 2005.
- [15] R. C. Bilcu and M. Vehvilainen, "Fast nonlocal means for image denoising," in *Proceedings of the SPIE Conference on Digital Photography III*, vol. 6502, March 2007, p. 65020R.
- [16] Y.-L. Liu, J. Wang, X. Chen, Y.-W. Guo, and Q.-S. Peng, "A robust and fast non-local means algorithm for image denoising," *Journal of Computer Science and Technology*, vol. 23, no. 2, pp. 270–279, March 2008.

- [17] N. Dowson and O. Salvado, "Hashed nonlocal means for rapid image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 485–499, March 2011.
- [18] V. Duval, J.-F. Aujol, and Y. Gousseau, "On the parameter choice for the non-local means," March 2010, CMLA Preprint. [Online]. Available: <http://hal.archives-ouvertes.fr/docs/00/46/88/56/PDF/nlmeans2.pdf>
- [19] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Transactions on Image Processing*, vol. 15, no. 10, pp. 2866–2878, October 2006.
- [20] —, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 45–69, August 2008.
- [21] P. Milanfar, "A tour of modern image filtering," *IEEE Signal Processing Magazine*, 2011, submitted.
- [22] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, February 2007.
- [23] H. Knutsson and C.-F. Westin, "Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, New York, NY, June 1993, pp. 515–523.
- [24] P. Chatterjee and P. Milanfar, "Clustering-based denoising with locally learned dictionaries," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1438–1451, July 2009.

- [25] H. J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1688–1704, September 2010.
- [26] M. P. Wand and M. C. Jones, *Kernel Smoothing*, ser. Monographs on Statistics and Applied Probability. London; New York: Chapman and Hall, 1995.
- [27] H. Takeda, S. Farsiu, and P. Milanfar, "Higher order bilateral filters and their properties," in *Proceedings of the SPIE Conference on Computational Imaging V*, vol. 6498, San Jose, CA, February 2007, p. 64980S.
- [28] A. Buades, B. Coll, and J.-M. Morel, "The staircasing effect in neighborhood filters and its solution," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1499–1505, July 2006.
- [29] P. Chatterjee and P. Milanfar, "A generalization of non-local means via kernel regression," in *Proceedings of SPIE Conference on Computational Imaging VI*, vol. 6814, San Jose, CA, January 2008, p. 68140P.
- [30] H. J. Seo and P. Milanfar, "Video denoising using higher order optimal space-time adaptation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, NV, April 2008, pp. 1249–1252.
- [31] T. Brox, O. Kleinschmidt, and D. Cremers, "Efficient nonlocal means for denoising of textural patterns," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1083–1092, July 2008.
- [32] G. Gilboa and S. Osher, "Nonlocal linear image regularization and supervised segmentation," *SIAM Multiscale Modeling and Simulation*, vol. 6, no. 2, pp. 595–630, July 2007.
- [33] D. Barash, "A fundamental relationship between bilateral filtering, adaptive

- smoothing, and the nonlinear diffusion equation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 844–847, 2002.
- [34] P. Perona and J. Mallik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Learning*, vol. 12, no. 7, pp. 629–639, July 1990.
- [35] D. Tschumperlé and R. Deriche, "Vector-valued image regularization with PDE's: A common framework for different applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 506–517, April 2005.
- [36] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1141–1151, October 2002.
- [37] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, December 2006.
- [38] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, November 2006.
- [39] W. Hong, J. Wright, K. Huang, and Y. Ma, "Multiscale hybrid linear models for lossy image representation," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3655–3671, December 2006.
- [40] G. Yu, G. Sapiro, and S. Mallat, "Image modeling and enhancement via structured sparse model selection," in *Proceedings of IEEE International Conference on Image Processing*, Hong Kong, September 2010, pp. 1641–1644.
- [41] D. D. Muresan and T. W. Parks, "Adaptive principal components and image denoising," in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, Barcelona, Spain, September 2003, pp. 101–104.

- [42] L. Zhanga, W. Donga, D. Zhanga, and G. Shib, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, no. 4, pp. 1531–1549, April 2010.
- [43] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proceedings of IEEE International Conference on Computer Vision*, Kyoto, Japan, September-October 2009, pp. 2272–2279.
- [44] C. Liu, R. Szeliski, S. B. Kang, C. L. Zitnick, and W. T. Freeman, "Automatic estimation and removal of noise from a single image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 299–314, February 2008.
- [45] N. Joshi, C. L. Zitnick, R. Szeliski, and D. Kriegman, "Image deblurring and denoising using color priors," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami, Florida, June 2009, pp. 1550–1557.
- [46] T. S. Cho, N. Joshi, C. L. Zitnick, S. B. Kang, R. Szeliski, and W. T. Freeman, "A content-aware image prior," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 2010, pp. 169–176.
- [47] S. Roth and M. J. Black, "Fields of experts," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, April 2009.
- [48] M. Elad, "Why simple shrinkage is still relevant for redundant representations?" *IEEE Transactions on Image Processing*, vol. 52, no. 12, pp. 5559–5569, December 2006.
- [49] K. Dabov, A. Foi, V. Katkovnik, and K. O. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, August 2007.
- [50] J.-L. Starck, E. J. Candes, and D. L. Donoho, "The curvelet transform for image

- denoising," *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 670–684, June 2002.
- [51] M. N. Do and M. Vetterli, "The finite ridgelet transform for image representation," *IEEE Transactions on Image Processing*, vol. 12, no. 1, pp. 16–28, January 2003.
- [52] R. Eslami and H. Radha, "Translation-invariant contourlet transform and its application to image denoising," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3362–3374, November 2006.
- [53] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.
- [54] E. P. Simoncelli and E. H. Adelson, "Noise removal via Bayesian wavelet coring," in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, Lausanne, Switzerland, September 1996, pp. 379–382.
- [55] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1522–1531, September 2000.
- [56] M. Elad, B. Matalon, and M. Zibulevsky, "Image denoising with shrinkage and redundant representations," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, New York, NY, 2006, pp. 1924–1931.
- [57] C. R. Jung and J. Scharcanski, "Adaptive image denoising and edge enhancement in scale-space using the wavelet transform," *Pattern Recognition Letters*, vol. 24, no. 7, pp. 965–971, April 2003.
- [58] F. Luisier, T. Blu, and M. Unser, "A new SURE approach to image denoising: Interscale orthonormal wavelet thresholding," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 593–606, March 2007.

- [59] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using a scale mixture of Gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, November 2003.
- [60] M. J. Wainwright and E. P. Simoncelli, "Scale mixtures of Gaussians and the statistics of natural images," in *Advances in Neural Information Processing Systems*, S. A. Solla, T. K. Leen, and K.-R. Müller, Eds., vol. 12. Denver, CO: MIT Press, May 2000, pp. 855–861.
- [61] D. Andrews and C. Mallows, "Scale mixtures of normal distributions," *Journal of Royal Statistical Society, Series B (Methodological)*, vol. 36, pp. 99–102, 1974.
- [62] S. Lyu and E. P. Simoncelli, "Modeling multiscale subbands of photographic images with fields of Gaussian scale mixtures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 693–706, April 2009.
- [63] Y. Hou, C. Zhao, D. Yang, and Y. Cheng, "Comments on "Image denoising by sparse 3-D transform-domain collaborative filtering"," *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 268–270, January 2011.
- [64] M. Jansen, *Noise Reduction by Wavelet Thresholding*. New York: Springer-Verlag, 2001.
- [65] D. L. Donoho and I. M. Johnstone, "Minimax estimation via wavelet shrinkage," *The Annals of Statistics*, vol. 26, no. 3, pp. 879–921, June 1998.
- [66] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.
- [67] X. Zhu and P. Milanfar, "Automatic parameter selection for denoising algorithms using a no-reference measure of image content," *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3116–3132, December 2010.

- [68] P. Chatterjee and P. Milanfar, "Patch-based locally optimal denoising," in *Proceedings of IEEE International Conference on Image Processing*, Brussels, Belgium, September 2011, to appear.
- [69] —, "Patch-based near-optimal image denoising," *IEEE Transactions on Image Processing*, 2011, submitted.
- [70] —, "Is denoising dead?" *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 895–911, April 2010.
- [71] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, ser. Signal Processing. Upper Saddle River, N.J.: Prentice-Hall, Inc., 1993, vol. 1.
- [72] M. Unser, B. L. Trus, and A. C. Steven, "A new resolution criterion based on spectral signal-to-noise ratios," *Ultramicroscopy*, vol. 23, no. 1, pp. 39–51, 1987.
- [73] S. Voloshynovskiy, O. Koval, and T. Pun, "Image denoising based on the edge-process model," *Signal Processing*, vol. 85, no. 10, pp. 1950–1969, October 2005.
- [74] J. Polzehl and V. Spokoiny, "Image denoising: Pointwise adaptive approach," *Annals of Statistics*, vol. 31, no. 1, p. 3057, 2003.
- [75] T. Treibitz and Y. Y. Schechner, "Recovery limits in pointwise degradation," in *Proceedings of IEEE International Conference on Computational Photography*, San Francisco, CA, April 2009, pp. 1–8.
- [76] A. Levin and B. Nadler, "Natural image denoising: Optimality and inherent bounds," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011, to appear.
- [77] S. Kay and Y. C. Eldar, "Rethinking biased estimation," *Signal Processing Magazine*, vol. 25, no. 3, pp. 133–136, May 2008.

- [78] Y. C. Eldar, "Rethinking biased estimation: Improving maximum likelihood and the Cramér-Rao bound," *Foundations and Trends in Signal Processing*, vol. 1, no. 4, pp. 305–449, 2008.
- [79] —, "MSE bound with affine bias dominating the Cramér-Rao bound," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3824–3836, August 2008.
- [80] N. Draper and H. Smith, *Applied Regression Analysis*, 3rd ed., ser. Probability and Statistics. Hoboken, N.J.: Wiley-Interscience, April 1998.
- [81] L. P. Seidman, "Performance limitations and error calculations for parameter estimation," *Proceedings of the IEEE*, vol. 58, pp. 644–652, May 1970.
- [82] H. Cramér, *Mathematical Methods of Statistics*. Princeton, N.J.: Princeton University Press, 1946.
- [83] C. R. Rao, "Information and the accuracy attainable in the estimation of statistical parameters," *Bulletin of Calcutta Mathematical Society*, vol. 37, pp. 81–89, 1945.
- [84] —, "Minimum variance and the estimation of several parameters," in *Proceedings of Cambridge Philosophical Society*, vol. 43, 1946, pp. 280–283.
- [85] J. Ziv and M. Zakai, "Some lower bounds on signal parameter estimation," *IEEE Transactions on Information Theory*, vol. IT-15, no. 3, pp. 386–391, May 1969.
- [86] H. L. van Trees and K. L. Bell, Eds., *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering / Tracking*, 1st ed. Piscataway, NJ: Wiley-IEEE Press, August 2007.
- [87] H. L. van Trees, *Detection, Estimation, and Modulation Theory*. New York: Wiley, 1968.
- [88] J. A. Fessler and A. O. Hero, "Cramér-Rao lower bounds for biased image reconstruction," in *Proceedings of 36th Midwest Symposium on Circuits and Systems*, vol. 1, August 1993, pp. 253–256.

- [89] T. Y. Young and R. A. Westerberg, "Error bounds for stochastic estimation of signal parameters," *IEEE Transactions on Information Theory*, vol. IT-17, no. 5, September 1971.
- [90] Z. Ben-Haim and Y. C. Eldar, "A lower bound on the Bayesian MSE based on the optimal bias function," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5179–5196, November 2009.
- [91] J. S. D. Bonet and P. Viola, "A non-parametric multi-scale statistical model for natural images," in *Advances in Neural Information Processing*. MIT Press, 1997, pp. 773–779.
- [92] S.-C. Zhu, "Statistical modeling and conceptualization of visual patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 691–712, June 2003.
- [93] E. P. Simoncelli and B. A. Olshausen, "Natural image statistics and neural representation," *Annual Review of Neuroscience*, vol. 24, pp. 1193–1216, May 2001.
- [94] A. Srivastava, A. B. Lee, E. P. Simoncelli, and S.-C. Zhu, "On advances in statistical modeling of natural images," *Journal of Mathematical Imaging and Vision*, vol. 18, pp. 17–33, 2003.
- [95] Y. Weiss and W. Freeman, "What makes a good model of natural images?" in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, June 2007, pp. 1–8.
- [96] M. A. Woodbury, "Inverting modified matrices," Statistical Research Group, Princeton University, Princeton, NJ, Memorandum Report 42, 1950.
- [97] A. Hyvärinen, J. Hurri, and P. O. Hoyer, *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision*, ser. Computational Imaging and Vision. Springer London, 2009, vol. 39.

- [98] I. Motoyoshi, S. Nishida, L. Sharan, and E. H. Adelson, "Image statistics and the perception of surface qualities," *Nature*, vol. 447, pp. 206–209, May 2007.
- [99] L. Sharan, Y. Li, I. Motoyoshi, S. Nishida, and E. Adelson, "Image statistics for surface reflectance perception," *Journal of Optical Society of America A: Optics, Image Science, and Vision*, vol. 25, no. 4, pp. 846–865, April 2008.
- [100] F. R. Hampel, "The influence curve and its role in robust estimation," *Journal of the American Statistical Association*, vol. 69, pp. 383–393, 1974.
- [101] D. Zoran and Y. Weiss, "Scale invariance and noise in natural images," in *Proceedings of IEEE International Conference on Computer Vision*, Kyoto, Japan, September–October 2009, pp. 2209–2216.
- [102] S. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, March 1982.
- [103] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, August 1995.
- [104] R. Xu and D. Wunsch II, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, May 2005.
- [105] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
- [106] Y. Chen, A. Wiesel, Y. C. Eldar, and A. O. Hero III, "Shrinkage algorithms for MMSE covariance estimation," *IEEE Transactions on Signal Processing*, pp. 5016–5029, October 2010.
- [107] B. Efron, "Bootstrap methods: Another look at the jackknife," *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979.

- [108] M. R. Chernick, *Bootstrap Methods: A Guide for Practitioners and Researchers*, 2nd ed., ser. Probability and Statistics. Hoboken, N.J.: Wiley-Interscience, November 2007.
- [109] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*, 1st ed., ser. Springer Texts in Statistics. Springer, 2004.
- [110] P. Chatterjee and P. Milanfar, "Learning denoising bounds for noisy images," in *Proceedings of IEEE International Conference on Image Processing*, Hong Kong, September 2010, pp. 1157–1160.
- [111] Y. C. Eldar and N. Merhav, "Minimax MSE-ratio estimation with signal covariance uncertainties," *IEEE Transactions on Signal Processing*, vol. 53, no. 4, pp. 1335–1347, April 2005.
- [112] D. D. Muresan and T. W. Parks, "Adaptive principal components and image denoising," in *Proceedings of IEEE International Conference on Image Processing*, Barcelona, Spain, September 2003, pp. 101–104.
- [113] S. Kritchman and B. Nadler, "Determining the number of components in a factor model from limited noisy data," *Chemometrics and Intelligent Laboratory Systems*, vol. 94, no. 1, pp. 19–32, November 2008.
- [114] —, "Non-parametric detection of the number of signals, hypothesis testing and random matrix theory," *IEEE Transactions on Signal Processing*, vol. 57, no. 10, pp. 3930–3941, October 2009.
- [115] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed., ser. Wiley Series in Telecommunications and Signal Processing. Hoboken, N.J.: John Wiley & Sons Inc., 2006.
- [116] D. P. Palomar and S. Verdú, "Gradient of mutual information in linear vector

- Gaussian channels," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 141–154, January 2006.
- [117] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, ser. Monographs on Statistics and Applied Probability. London: Chapman and Hall, 1986.
- [118] A. Rényi, "On measures of information and entropy," in *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, vol. 1, Berkeley, CA, 1961, pp. 547–561.
- [119] D. Evans, "A law of large numbers for nearest neighbor statistics," *Proceedings of the Royal Society A: Mathematical, Physical & Engineering Sciences*, vol. 464, no. 2100, pp. 3175–3192, December 2008.
- [120] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, July and October 1948.
- [121] K. Zografos and S. Nadarajah, "Expressions for Rényi and Shannon entropies for multivariate distributions," *Statistics and Probability Letters*, vol. 71, no. 1, pp. 71–84, January 2005.
- [122] L. F. Kozachenko and N. N. Leonenko, "On statistical estimation of entropy of random vector," *Problems of Information Transmission*, vol. 23, no. 2, pp. 95–101, 1987.
- [123] H. Singh, N. Misra, V. Hnizdo, A. Fedorowicz, and E. Demchuk, "Nearest neighbor estimates of entropy," *American Journal of Mathematical and Management Sciences*, vol. 23, no. 3 & 4, pp. 301–321, 2003.
- [124] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical Review E*, vol. 69, no. 6, p. 066138, June 2004.

- [125] M. N. Gorla, N. N. Leonenko, V. Mergel, and P. L. Novi Inverardi, "A new class of random vector entropy estimators and its applications in testing statistical hypotheses," *Journal of Nonparametric Statistics*, vol. 17, no. 3, pp. 277–297, April–May 2005.
- [126] N. N. Leonenko, L. Pronzato, and V. Savani, "A class of Rényi information estimators for multidimensional densities," *Annals of Statistics*, vol. 36, no. 5, pp. 2153–2182, 2008.
- [127] A. Kaltchenko and N. Timofeeva, "Bias reduction via linear combination of nearest neighbor entropy estimators," *International Journal on Information and Coding Theory*, vol. 1, no. 1, pp. 39–56, 2009.
- [128] E. Liitiainen, A. Lendasse, and F. Corona, "On the statistical estimation of Rényi entropies," in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing*, Grenoble, France, September 2009, pp. 1–6.
- [129] M. Shao, K. E. Barner, and R. C. Hardie, "Partition-based interpolation for color filter array demosaicking and super-resolution reconstruction," *Optical Engineering*, vol. 44, p. 107003, October 2005.
- [130] P. Chatterjee and P. Milanfar, "Practical bounds on image denoising: From estimation to information," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1221–1233, May 2011.
- [131] R. W. G. Hunt, *The Reproduction of Colour*. John Wiley & Sons, Inc, November 2005.
- [132] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, January 2008.

- [133] 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 *Proceedings of 8th International Conference on Computer Vision*, vol. 2, Vancouver, B.C., July 2001, pp. 416–423.
- [134] H. J. Seo, P. Chatterjee, H. Takeda, and P. Milanfar, "A Comparison of Some State of the Art Image Denoising Methods," in *Proceedings of the 41st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, November 2007, pp. 518–522.
- [135] P. Chatterjee and P. Milanfar, "Image denoising using locally learned dictionaries," in *Proceedings of IS&T/SPIE Electronic Imaging Conference on Computational Imaging*, vol. 7246, San Jose, CA, January 2009, p. 72460V.
- [136] —, "Bias modeling for image denoising," in *Proceedings of 43rd Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, November 2009.
- [137] —, "Fundamental limits of image denoising: Are we there yet?" in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Dallas, TX, March 2010, pp. 1358 – 1361.
- [138] B. E. Bayer, "Color imaging array," *US Patent 3971065*, July 1976.
- [139] V. Lukin, S. Abramov, N. Ponomarenko, K. Egiazarian, and J. Astola, "Image filtering: Potential efficiency and current problems," in *Proceedings of IEEE Conference on Acoustics, Speech and Signal Processing*, Prague, Czech Republic, May 2011, to appear.
- [140] P. Chatterjee, N. Joshi, S. B. Kang, and Y. Matsushita, "Noise suppression in low-light images through joint denoising and demosaicing," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

- [141] S. H. Park, H. S. Kim, S. Linsel, M. Parmar, and B. A. Wandell, "A case for denoising before demosaicking color filter array data," in *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, November 2009.
- [142] L. Zhang, R. Lukac, X. Wu, and D. Zhang, "PCA-based spatially adaptive denoising of CFA images for single-sensor digital cameras," *IEEE Transactions on Image Processing*, vol. 18, no. 4, pp. 797–812, April 2009.
- [143] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," in *ACM Transactions on Graphics*, vol. 23, no. 3, 2004, pp. 664–672.
- [144] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM Transactions on Graphics*, vol. 26, no. 3, p. 96, July 2007.
- [145] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, "Image deblurring with blurred/noisy image pairs," in *ACM Transactions on Graphics*, vol. 26, no. 3, July 2007.
- [146] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proceedings of European Conference on Computer Vision*, vol. 1, Crete, Greece, September 2010, pp. 1–14.
- [147] H. J. Seo and P. Milanfar, "Robust flash denoising/deblurring by iterative guided filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, submitted.
- [148] A. Foi, "Clipped noisy images: Heteroskedastic modeling and practical denoising," *Signal Processing*, vol. 89, no. 12, pp. 2609–2629, December 2009.
- [149] C.-A. Deledalle, F. Tupin, and L. Denis, "Poisson NL means: Unsupervised non local means for Poisson noise," in *Proceedings of IEEE International Conference on Image Processing*, Hong Kong, September 2010, pp. 801–804.