Is Denoising Dead?

Priyam Chatterjee, *Student Member, IEEE*, and Peyman Milanfar, *Fellow, IEEE*

***Abstract*—Image denoising has been a well studied problem in the field of image processing. Yet researchers continue to focus attention on it to better the current state-of-the-art. Recently proposed methods take different approaches to the problem and yet their denoising performances are comparable. A pertinent question then to ask is whether there is a theoretical limit to denoising performance and, more importantly, are we there yet? As camera manufacturers continue to pack increasing numbers of pixels per unit area, an increase in noise sensitivity manifests itself in the form of a noisier image. We study the performance bounds for the image denoising problem. Our work in this paper estimates a lower bound on the mean squared error of the denoised result and compares the performance of current state-of-the-art denoising methods with this bound. We show that despite the phenomenal recent progress in the quality of denoising algorithms, some room for improvement still remains for a wide class of general images, and at certain signal-to-noise levels. Therefore, image denoising is not dead—yet.**

***Index Terms*—Bayesian Cramér–Rao lower bound (CRLB), bias, bootstrapping, image denoising, mean squared error.**

I. Introduction

IMAGE denoising has been a well-studied problem in the image processing community and continues to attract researchers with an aim to perform better restoration in the presence of noise. With the rise in the number of image sensors (or pixels) per unit area of a chip, modern image capturing devices are increasingly sensitive to noise. Camera manufacturers, therefore, depend on image denoising algorithms to reduce the effects of such noise artifacts in the resultant image. Recently proposed denoising methods use different approaches to address the problem. Of them, the best performing methods [1]–[6] can be shown to share a common framework in that they work by combining similar patches to effect denoising, although the parameters of said framework are estimated in rather different ways. These state-of-the-art algorithms produce very impressive, though quite comparable results; and this begs the question: Have we reached some limit of performance? Is denoising dead? Given the importance of this problem, our present work studies denoising in a statistical framework to derive a fundamental bound on the performance of denoising algorithms and compare the current state-of-the-art to it.[1] Literature on such performance limits exists for some of the more complex image processing problems such as image registration [7], [8] and super-resolution [9]–[12]. Performance limits to object or feature recovery in images in the presence of pointwise degradation has been studied by Treibitz *et al.* [13]. In their work, the authors study the effects of noise among other degradations and formulate expressions for the optimal filtering parameters that define the resolution limits to recovering any given feature in the image. While their study is practical, it does not define statistical performance limits to denoising general images. In [14], Voloshynovskiy *et al.* briefly analyze the performance of MAP estimators for the denoising problem. However, our bounds are developed in a much more general setting and, to the best of our knowledge, no comparable study currently exists for the problem of denoising. The present study will enable us to understand how well the state-of-the-art denoising algorithms perform as compared to these limits. From a practical perspective, it will also lead to understanding the fundamental limits of increasing the number of sensors in the imaging system with acceptable image quality being made possible by noise suppression algorithms.

Before we analyze image denoising statistically, we first define our image formation model as

$$y_i = z_i + \eta_i \tag{1}$$

where z_i is the actual pixel intensity at location $\mathbf{x}_i$ (indexed by i) and y_i is the observed pixel intensity. We assume that the corrupting noise η_i is independent identically distributed (IID) and sampled from a zero mean density of known variance σ^2. The aim of denoising algorithms is to recover the noise-free pixel intensity z_i. Most recent denoising algorithms [1]–[6] work on image patches[2], and, hence, we define the patch-based image model as

$$\mathbf{y}_i = \mathbf{z}_i + \boldsymbol{\eta}_i \tag{2}$$

where $\mathbf{z}_i$ is the actual patch intensity written in a (column-stacked or raster-scanned) vectorized form, $\mathbf{y}_i$ is the vectorized noisy image patch and $\boldsymbol{\eta}_i$ is a vectorized noise patch with a covariance $\sigma^2\mathbf{I}$. Note that the noise vectors $\boldsymbol{\eta}_i$ are uncorrelated only when the patches are nonoverlapping, something that we assume in developing our bounds formulation.

Although the measurement model of (2) is linear in the unknown image, the most successful methods [1]–[6] to date have taken a nonlinear estimation approach to this inverse problem, resulting in state-of-the-art performance. Of the various approaches, some of the recently proposed weighted patch-averaging filters prove to be quite effective for denoising. Of them, the recently proposed method of nonlocal means (NLM) [1] attempts to locate patches similar to a patch of

Manuscript received May 21, 2009; revised October 28, 2009. First published November 20, 2009; current version published March 17, 2010. This work was supported in part by the U.S. Air Force under Grant F49620-03-1-0387. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Michael Elad.

The authors are with the Department of Electrical Engineering, University of California, Santa Cruz, Santa Cruz, CA 95064 USA (e-mail: priyam@soe.ucsc.edu; milanfar@soe.ucsc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2009.2037087

[1]Supporting software to compute the bounds derived in this paper, and additional results are available at http://www.soe.ucsc.edu/~priyam/bounds/.

[2]Patches are defined as a neighborhood of pixels in a small fixed sized window. Such image patches can be overlapping or nonoverlapping. Most denoising algorithms work with overlapping patches.

interest, and denoise a particular pixel at the center of that patch by a weighted average of the center pixels of the similar patches. Since searching for similar patches in a large image can be computationally impractical, typically only a much smaller neighborhood of the patch under consideration is searched for possible matches. Kervrann *et al.* [2] improved on the performance of NLM by incorporating an adaptive search window along with a modified weight calculation formula. Takeda *et al.* [3] proposed a framework for denoising using kernel regression where the kernels are designed to be data adaptive. Another successful spatial domain denoising technique called K-SVD [4] assumes the image patches to be sparse-representable. It performs denoising by learning a large collection of patches (called a *dictionary*) from the noisy image such that each patch in the image can be expressed as a linear combination of only a few patches from the dictionary.[3] In [5], we presented a clustering-based framework (K-LLD) that takes advantage of the approaches in [3] and [4] to perform denoising. While all these methods perform denoising in the spatial domain, BM3D [6] takes a different approach to denoising. This method builds on the concept of NLM by identifying similar patches in an image and grouping them together. Denoising is, however, done in the transform domain. This method has proved to be very effective and can be considered to be the current state-of-the-art. Irrespective of the approaches taken by these denoising algorithms, the final aim is to estimate the $\mathbf{z}_i$ vectors at each spatial location from the noisy observations $\mathbf{y}_i$. We study the performance limits of denoising from the point of view of estimating these $\mathbf{z}_i$ vectors using the data model of (2).

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 given image can be denoised. In this paper, we consider the case of additive white Gaussian noise (AWGN), although other noise distributions are equally applicable. We take advantage of the geometric similarity approach advocated in [5] where the image is considered to be composed of a finite number of "clusters". Each such cluster contains patches of the image which have similar geometric structure. For instance, considering Fig. 1, there are four relevant clusters. Namely, these describe the horizontal edges, the vertical edges, the corners,[4] and the "flat" regions. It is worth noting that clusters are composed of regions which may have quite different actual gray values, but that are nonetheless visually similar (flat, edge, etc.). As we shall see later in this paper, the overall denoising performance bounds for a given image can then be derived by analyzing the respective performance bounds for each of the said clusters, and pooling these results together. For the sake of clarity of presentation, however, we first begin the analysis by assuming that the image of interest is composed of a single cluster of geometrically similar patches, as exemplified by Fig. 2(a). As a result of assuming geometric similarity across the entire image, we can think of each $\mathbf{z}_i$ vector as a realization of a vector random variable $\mathbf{z}$ sampled from some (unknown) probability density function $p(\mathbf{z})$. Our bounds on the MSE for denoising are then developed for estimating the random variable $\mathbf{z}$.

[3]Results for K-SVD reported in this paper are those obtained with the method outlined in [4]. Recently proposed variations of the method [15], [16] have shown promise in improving its performance even further.

[4]In this experiment, we have restricted the image patches to be grouped into 4 clusters, as a result of which the corners are grouped together, even though their directionalities are quite different. Ideally, they should be clustered differently by choosing a large enough number of clusters.

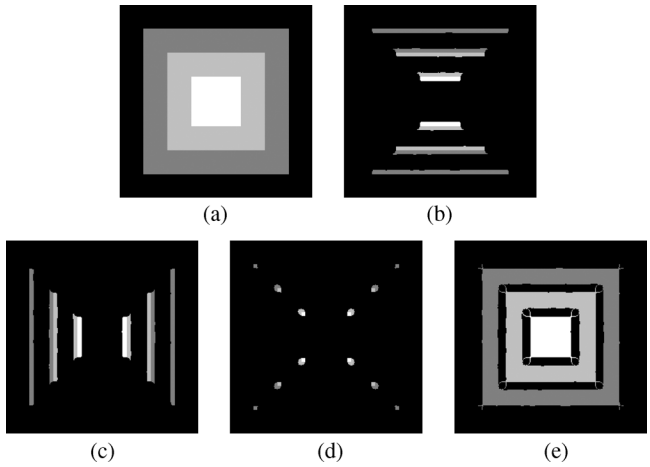


Fig. 1. Clustering of a simple image based on geometric similarity. Note how pixels in any particular cluster can have quite different intensities but similar geometric structure (edge, corner, flat regions, etc.). (a) Box image. (b) Cluster 1. (c) Cluster 2. (d) Cluster 3. (e) Cluster 4.

In the next section, 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 developed in Section IV through an Optimal Bias Bayesian Cramér–Rao Lower Bound (OB-CRLB) formulation for biased estimators that we explain in Section III. Until that point, the lower bound is developed assuming geometric homogeneity among patches in the image. Since patches in any given image can exhibit widely varying geometric structures, we extend our lower bound to general images in Section V. Using the method thus developed, we calculate performance bounds for denoising on various images and compare them to the performance (in terms of MSE) of some of the best existing denoising methods in Section VI. Finally, we conclude with some remarks on the future of denoising in Section VII.

II. Bias in Denoising

In this section, we study the bias in nonlinear estimators used to solve the denoising problem of (2). In estimation theory, it is well known that unbiased estimators do not always exist. Moreover, even when unbiased estimators do exist, it is sometimes advantageous to work with biased estimators as they may result in a lower MSE [17], [18]. Moreover, unbiased estimators for a difficult problem such as denoising will tend to have unacceptably large variance and, therefore, result in processed images that will not look very good to the viewer. 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 its bias as well as the covariance of the estimate. In order to study the bias of an estimator, we need a model. We claim that it is reasonable

to approximate the local[5] behavior of the bias function as affine. Namely

$$\mathbf{b}(\mathbf{z}) = E[\hat{\mathbf{z}}|\mathbf{z}] - \mathbf{z} = \mathbf{M}\mathbf{z} + \mathbf{u} \tag{3}$$

where $\mathbf{b}(\mathbf{z})$ is the bias of the estimator, and the matrix $\mathbf{M}$ 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 [19]. In Appendix A, we provide further mathematical justification for using such an affine model of the bias. Since we are dealing with patches that are geometrically similar, it is fair to assume here that $\mathbf{z}$ is a random vector that has a particular (as of yet unknown) pdf $p(\mathbf{z})$ and the model of (3) holds for every instance of $\mathbf{z}_i$ sampled from the (unknown) distribution. That is to say, for any particular patch within a cluster, the bias model $\mathbf{b}(\mathbf{z}_i) = \mathbf{M}\mathbf{z}_i + \mathbf{u}$ 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 we believe that it is a good starting point. In Appendix C, 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 ($\mathbf{M}$ 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 (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 $\mathbf{M}$ and $\mathbf{u}$). Fig. 2 shows the bias of the denoised intensity estimates obtained using ten runs of BM3D [6] and K-SVD [4] 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 additive white Gaussian noise of standard deviation 25. The noisy images are then denoised with each of the methods (using the default parameter settings in each case) and the mean denoised image is obtained for each method. From this, the bias vectors $\mathbf{b}(\mathbf{z}_i)$ are obtained for each method using nonoverlapping 11×11 patches. The bias vectors of all such patches are tiled to form the *method bias* images shown in Fig. 2. The bias for each method is then modeled by (3) and the model parameters ($\widehat{\mathbf{M}}$ and $\widehat{\mathbf{u}}$) are fit using least squares. The predicted bias patches $\widehat{\mathbf{b}}(\mathbf{z}_i)$ are then computed for each patch in each case. These vectors are tiled to form the predicted bias images in Fig. 2. 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 [20] which can be defined as

$$R^2 = 1 - \frac{\sum_i \left\| \mathbf{b}(\mathbf{z}_i) - \widehat{\mathbf{b}}(\mathbf{z}_i) \right\|_2^2}{\sum_i \left\| \mathbf{b}(\mathbf{z}_i) - \overline{\mathbf{b}}(\mathbf{z}) \right\|_2^2} \tag{4}$$

where i indexes all the patches in the image, $\mathbf{b}(\mathbf{z}_i)$ is the actual bias of the estimated intensity of the ith patch, $\overline{\mathbf{b}}(\mathbf{z})$ is the mean bias obtained by the denoising method across all patches in the image and $\widehat{\mathbf{b}}(\mathbf{z}_i) = \widehat{\mathbf{M}}\mathbf{z}_i + \widehat{\mathbf{u}}$ is the predicted bias obtained from the estimated parameters $\widehat{\mathbf{M}}$ and $\widehat{\mathbf{u}}$ of the affine model. We obtained high R^2 values[6] for the examples in Fig. 2 with various denoising methods [3]–[6], as can be seen from Table I. Our experiments with these denoising methods on other images[7] have yielded comparable results that confirm the goodness of the affine model (Table I).

TABLE I
R^2 VALUES FOR THE AFFINE MODEL FIT OF THE BIAS PRODUCED BY DIFFERENT METHODS FOR DIFFERENT IMAGES

Image	BM3D [6]	K-SVD [4]	SKR [3]	K-LLD [5]
Grass	0.863	0.801	0.833	0.810
House	0.916	0.955	0.959	0.963
Parrot	0.957	0.963	0.946	0.954

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 the parrot and house images (shown in Fig. 5), where we randomly selected patches from the image and tried to model the bias for such patches by estimating a single set of parameters ($\mathbf{M}$ and $\mathbf{u}$). For both the images, we obtained much lower values ($R^2 < 0.6$) for the goodness of fit. However, when only patches of similar geometry were considered for the same images, the R^2 values for the fit were considerably higher (Table I). These experiments indicate that the affine model is a good *local* fit, where locality is characterized by similarity in patch geometry. For the sake of completeness, we refer the interested reader to Appendix C 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 the paper, we will assume an affine model for the bias to derive the theoretical performance limits of denoising.

[5]Here, *local* refers to the members of a cluster of geometrically similar patches across the image. As a result, for general images made up of geometrically nonhomogeneous patches, we have to use a different $\mathbf{M}$ and $\mathbf{u}$ for each cluster. That is to say, the bias is modeled as a different affine function for each cluster.

[6]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.

[7]For general images such as the house and parrot images (Fig. 5) 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 Section V where we discuss the bound calculation process for general images. The mean R^2 values across 5 clusters are reported in Table I.

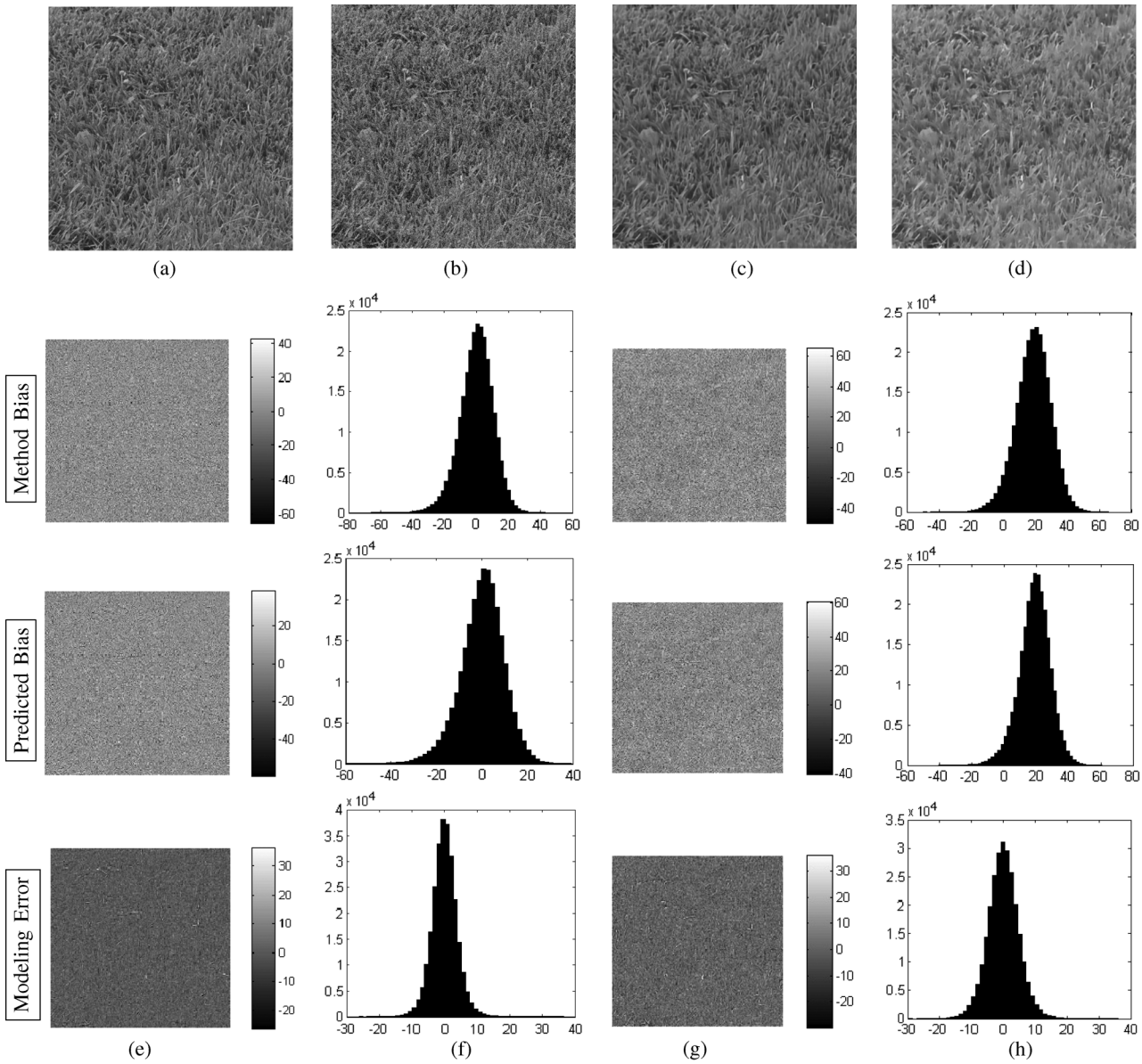


Fig. 2. Visual comparison of the actual bias obtained from BM3D [6] and K-SVD [4] and reconstructed bias using affine model fit for the 512 × 512 grass image. The patch size chosen was 11 × 11. We can see that the histograms for the modeling errors in both the cases are centered around zero and have short tails. (a) Original image; (b) noisy image; (c) denoised (BM3D); (d) denoised (K-SVD); (e) BM3D; (f) histogram of (e); (g) K-SVD; (h) histogram of (g).

III. Optimal Bias Bayesian Cramér–Rao Lower Bound (OB-CRLB)

In the statistics and signal processing literature, a number of bounds exist to evaluate performance limits of estimation. While some bounds were developed for the estimation of a deterministic parameter (for instance, those proposed by Seidman [21], Cramér [22] and Rao [23], [24]), others, such as those developed by Ziv *et al.* [25], 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 [18] ($\mathbf{z}$ in our case). As a result, Bayesian versions have been derived for many of the bounds developed for the deterministic case [26]. In our work, we build on a Bayesian version of the classical CRLB [27]. In its simplest form, the CRLB is a lower bound on the variance of an unbiased estimator of $\mathbf{z}$, subject to the regularity condition

$$E\left[\frac{\partial \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}}\right] = \mathbf{0}, \quad \forall\, \mathbf{z} \tag{5}$$

on the conditional probability density function $p(\mathbf{y}|\mathbf{z})$, assuming 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 [26], [28] where the joint pdf $p(\mathbf{y}, \mathbf{z})$ is directly used. The two pdf's are related by

$$p(\mathbf{y}, \mathbf{z}) = p(\mathbf{y}|\mathbf{z})p(\mathbf{z}) \tag{6}$$

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 (12). Assuming for now that an unbiased estimator of $\widehat{\mathbf{z}}$ exists, the bound on the (conditional) covariance of the estimate is given by the CRLB as

$$\mathrm{cov}(\widehat{\mathbf{z}}|\mathbf{z}) = E\left[(\widehat{\mathbf{z}} - E[\widehat{\mathbf{z}}|\mathbf{z}])\,(\widehat{\mathbf{z}} - E[\widehat{\mathbf{z}}|\mathbf{z}])^T\,|\mathbf{z}\right] \geqslant \mathbf{J}^{-1} \quad (7)$$

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

$$\mathbf{J} = -E\left[\frac{\partial^2 \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}\;\partial \mathbf{z}^T}\right]. \quad (8)$$

The estimator which achieves this lower bound is said to be efficient. While this provides us with a simple method of evaluation of 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 $\widehat{\mathbf{z}}$ is given by

$$\mathrm{cov}(\widehat{\mathbf{z}}|\mathbf{z}) \geqslant \left(\frac{\partial E[\widehat{\mathbf{z}}|\mathbf{z}]}{\partial \mathbf{z}}\right)\mathbf{J}^{-1}\left(\frac{\partial E[\widehat{\mathbf{z}}|\mathbf{z}]}{\partial \mathbf{z}}\right)^T \quad (9)$$

$$= (\mathbf{I}+\mathbf{M})\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M})^T \quad (10)$$

where $\mathbf{I}$ denotes the identity matrix and (10) comes from making use of our bias model of (3). It is useful to note here that the estimator covariance for the affine model is only influenced by the parameter $\mathbf{M}$ (which can also be termed 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 (7). Performance limits for image reconstruction problems based on the biased CRLB have been studied by Fessler *et al.* [29] using a constraint on the bias gradient. Using the relation in (10), we can calculate a lower bound on the conditional MSE in estimating $\mathbf{z}$ as

$$\begin{aligned} E\left[||\mathbf{z}-\widehat{\mathbf{z}}||^2|\mathbf{z}\right] &= \mathrm{Tr}\left\{\mathrm{cov}(\widehat{\mathbf{z}}|\mathbf{z})\right\} + ||\mathbf{b}(\mathbf{z})||^2 \\ &\geqslant \mathrm{Tr}\left\{(\mathbf{I}+\mathbf{M})\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M})^T\right\} \\ &\quad + ||\mathbf{b}(\mathbf{z})||^2 \end{aligned} \quad (11)$$

where $\mathrm{Tr}\{.\}$ denotes the trace of a matrix. Now, by the law of total expectation, the overall Bayesian MSE can be expressed as (12), shown at the bottom of the page. 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. Knowledge of $p(\mathbf{z})$ as a prior has been used by Young *et al.* [30] to derive a Bayesian MSE bound for the estimation of a scalar parameter that is known to lie within a fixed interval. Recently, their results have been generalized for an unconstrained vector case by Ben-Haim *et al.* [31]. 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 [30], [31]. This is related to the subject of statistical modeling of images, which has seen much activity [32]–[38] and is still the subject of some controversy. Happily, as described in Section IV-A 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. The bound formulation of (12) used in our work is related to those used in [30] and [31] but differs from the Bayesian CRLB (B-CRLB) of van Trees[8] [26], [28], as alluded to earlier. 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.

IV. Lower Bound on the MSE

In this section, we derive the bound using expressions for the bias model parameters ($\mathbf{M}$ and $\mathbf{u}$) that minimize the right hand side of (12). We also derive an analytical expression for the FIM and discuss how we derive the covariance of image patches that is needed to derive the MSE bound.

A. 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 $\mathbf{M}$ and $\mathbf{u}$ that minimize (12). This is in line with the approach advocated

[8]The FIM used in the B-CRLB formulation of van Trees [26], [28] is calculated from the joint pdf $p(\mathbf{y},\mathbf{z})$ whereas in our case (and also [30], [31]) it is calculated from the conditional pdf $p(\mathbf{y}|\mathbf{z})$. Hence, the B-CRLB of [28] 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.

$$\begin{aligned} E\left[||\mathbf{z}-\widehat{\mathbf{z}}||^2\right] &= \int_{\mathbf{z}} E\left[||\mathbf{z}-\widehat{\mathbf{z}}||^2|\mathbf{z}\right] p(\mathbf{z})d\mathbf{z} \\ &\geqslant \underbrace{\int_{\mathbf{z}} \left[\mathrm{Tr}\left\{(\mathbf{I}+\mathbf{M})\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M})^T\right\} + (\mathbf{M}\mathbf{z}+\mathbf{u})^T(\mathbf{M}\mathbf{z}+\mathbf{u})\right] p(\mathbf{z})d\mathbf{z}}_{Q} \end{aligned} \quad (12)$$

in [31]. We can thus obtain the optimal $\mathbf{M}$ and $\mathbf{u}$ (denoted as $\mathbf{M}^*$ and $\mathbf{u}^*$, respectively) by solving the optimization problem

$$\{\mathbf{M}^*, \mathbf{u}^*\} = \arg \min_{\{\mathbf{M},\mathbf{u}\}} \int_{\mathbf{z}} \left[\mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M})\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M})^T \right\} + (\mathbf{M}\mathbf{z}+\mathbf{u})^T(\mathbf{M}\mathbf{z}+\mathbf{u}) \right] p(\mathbf{z}) d\mathbf{z}. \quad (13)$$

The optimum $\mathbf{M}^*$ and $\mathbf{u}^*$ can be obtained by differentiating Q (defined in (12)) with respect to $\mathbf{M}$ and $\mathbf{u}$ and solving the simultaneous system of equations

$$\frac{\partial Q}{\partial \mathbf{u}} = \mathbf{0}, \quad \frac{\partial Q}{\partial \mathbf{M}} = \mathbf{0}. \quad (14)$$

Solving these simultaneous equations results in expressions for the optimum bias model parameters

$$\mathbf{M}^* = -\mathbf{J}^{-1}\left[\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z})\right]^{-1} \quad (15)$$

$$\mathbf{u}^* = -\mathbf{M}^* E[\mathbf{z}] = \mathbf{J}^{-1}\left[\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z})\right]^{-1} E[\mathbf{z}]. \quad (16)$$

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 $\mathbf{z}$. The derivations are detailed in Appendix B. Thus, we are able to obtain expressions for $\mathbf{M}^*$ and $\mathbf{u}^*$ that result in the theoretical lower bound on the MSE for any affine-biased denoiser.[9] 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 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 $\mathbf{M}^*$ and $\mathbf{u}^*$ in the expression for Q [given in (13)].

Once we have obtained 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 (12) by plugging in the obtained expressions of the parameters from (15) and (16) as

$$\begin{aligned}
Q_{\min} &= \int_{\mathbf{z}} \left[\mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M}^*)\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M}^*)^T \right\} + (\mathbf{M}^*\mathbf{z}+\mathbf{u}^*)^T(\mathbf{M}^*\mathbf{z}+\mathbf{u}^*) \right] p(\mathbf{z}) d\mathbf{z} \\
&= \mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M}^*)\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M}^*)^T \right\} + \int_{\mathbf{z}} (\mathbf{z}-E[\mathbf{z}])^T \mathbf{M}^{*T}\mathbf{M}^* (\mathbf{z}-E[\mathbf{z}]) p(\mathbf{z}) d\mathbf{z} \\
&= \mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M}^*)\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M}^*)^T \right\} + E\left[(\mathbf{z}-E[\mathbf{z}])^T \mathbf{M}^{*T}\mathbf{M}^* (\mathbf{z}-E[\mathbf{z}]) \right] \\
&= \mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M}^*)\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M}^*)^T \right\} + E\left[\mathrm{Tr}\left\{ \mathbf{M}^* (\mathbf{z}-E[\mathbf{z}])(\mathbf{z}-E[\mathbf{z}])^T \mathbf{M}^{*T} \right\} \right] \\
&= \mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M}^*)\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M}^*)^T \right\} + \mathrm{Tr}\left\{ E\left[\mathbf{M}^* (\mathbf{z}-E[\mathbf{z}])(\mathbf{z}-E[\mathbf{z}])^T \mathbf{M}^{*T} \right] \right\} \\
&= \mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M}^*)\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M}^*)^T \right\} + \mathrm{Tr}\left\{ \mathbf{M}^* E\left[(\mathbf{z}-E[\mathbf{z}])(\mathbf{z}-E[\mathbf{z}])^T \right] \mathbf{M}^{*T} \right\} \\
&= \mathrm{Tr}\left\{ \mathbf{M}^*\mathbf{J}^{-1}\mathbf{M}^{*T} + 2\mathbf{M}^*\mathbf{J}^{-1} + \mathbf{J}^{-1} + \mathbf{M}^*\mathrm{cov}(\mathbf{z})\mathbf{M}^{*T} \right\} \\
&= \mathrm{Tr}\left\{ \mathbf{M}^* \left(\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z})\right) \mathbf{M}^{*T} + 2\mathbf{M}^*\mathbf{J}^{-1} + \mathbf{J}^{-1} \right\} \\
&= \mathrm{Tr}\left\{ \mathbf{J}^{-1}\left(\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z})\right)^{-1}\mathbf{J}^{-1} - 2\mathbf{J}^{-1}\left(\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z})\right)^{-1}\mathbf{J}^{-1} + \mathbf{J}^{-1} \right\} \\
&= \mathrm{Tr}\left\{ \mathbf{J}^{-1} - \mathbf{J}^{-1}\left(\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z})\right)^{-1}\mathbf{J}^{-1} \right\} \\
&= \mathrm{Tr}\left\{ \left(\mathbf{J} + \mathrm{cov}^{-1}(\mathbf{z})\right)^{-1} \right\} \quad (17)
\end{aligned}$$

where the last equality is derived from the matrix inversion lemma [39]. Equation (17) thus allows us to get a neat expression for the lower bound on the MSE for the denoising problem, that is

$$E\left[\|\mathbf{z}-\hat{\mathbf{z}}\|^2\right] \geqslant \mathrm{Tr}\left[\left(\mathbf{J} + \mathrm{cov}^{-1}(\mathbf{z})\right)^{-1} \right]. \quad (18)$$

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 (or within a cluster). 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 $\mathrm{cov}(\mathbf{z})$. 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 (Table II).

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 [19]. Moreover, the expression for the lower bound is precisely that of the Linear Minimum Mean Square Error (LMMSE) estimate for the problem [27]. In theory, this bound is achievable by an affine estimator with exact knowledge of the first and second order moments of $p(\mathbf{z})$. In practice, however, the moments can only be *estimated* from the given noisy image, leading to a sub-optimal performance. Also, 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 [27]. 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.* [31]. 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. Extending our approach to the case where the bias is higher order will incorporate correspondingly higher order moments of the distribution of $\mathbf{z}$ (see Appendix C). For practical computation of the bound, we use the noise-free image to estimate the covariance of $\mathbf{z}$. In Section IV-C, we explain this process in detail. But first, we derive an analytical expression for the FIM, assuming Gaussian white noise.

[9] It is interesting to note that this optimization indeed yields a negative definite $\mathbf{M}^*$ as can be seen in (15).

TABLE II
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	BM3D [6]	K-SVD [4]	SKR [3]	K-LLD [5]	Bound	Relative Efficiency[12] (R.E.)	(in dB)
Box	49.56	57.78	77.17	53.93	3.42	0.069	(11.61)
Stripes	4.16	13.56	21.83	25.15	0.55	0.132	(8.79)
House	33.57	40.05	47.57	42.82	14.82	0.441	(3.56)
Lena	40.46	48.09	44.09	46.02	19.66	0.486	(3.13)
Boats	67.17	78.39	78.44	77.45	38.70	0.576	(2.40)
Cloth	91.33	104.36	103.42	104.68	72.98	0.799	(0.97)
Barbara	55.62	72.39	87.91	111.58	50.24	0.903	(0.44)
Parrot	86.98	100.29	98.54	93.64	83.82	0.964	(0.16)
Mandrill	185.60	196.20	195.75	188.84	181.61	0.979	(0.09)
Grass	153.64	161.74	150.39	147.13	145.58	0.989	(0.05)

[12]Relative Efficiency (RE) =MSE Bound/MSE of best performing method
decibel figures are: $-10\ \log_{10}(\mathrm{RE})$, which indicate room for improvement.

B. Fisher Information Matrix

The expression for the MSE bound in (18) holds true for any noise distribution, which in turn needs to be taken into account in deriving an analytical expression for the FIM $\mathbf{J}$. Hence, our framework can be used to derive bounds for any noise distribution. In this paper, however, we only consider the case of additive white Gaussian noise (AWGN). 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 will be duplicated in neighboring patches. To make our derivation of the FIM simple, we will assume the image patches to be nonoverlapping. This allows us to assert that the 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

$$p(\mathbf{y}|\mathbf{z}) = \frac{1}{(\sqrt{2\pi}\sigma)^{n|\Omega|}} \exp\left\{ \sum_j \frac{-\|\mathbf{y}_j - \mathbf{z}_j\|^2}{2\sigma^2} \right\} \qquad (19)$$

where $|\Omega|$ is the total number of (nonoverlapping) 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 [1], [2], [6] infer information about a single patch by taking into account multiple similar[10] patches. Such algorithms in essence estimate the $\mathbf{z}_i$ vector from multiple similar noisy $\mathbf{y}_j$ vectors. In such a scenario, we obtain an expression for the FIM $(\mathbf{J})$ as

$$\frac{\partial \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}_i} = \sum_j \frac{(\mathbf{y}_j - \mathbf{z}_j)}{\sigma^2} \quad \text{where} \quad \mathbf{z}_j \approx \mathbf{z}_i \qquad (20)$$

$$\Rightarrow \mathbf{J} = -E\left[\frac{\partial^2 \ln p(\mathbf{y}|\mathbf{z})}{\partial \mathbf{z}_i \partial \mathbf{z}_i^T} \right] = N \frac{\mathbf{I}}{\sigma^2} \qquad (21)$$

[10]Similarity in those cases means, similar in terms of patch intensities. We denote the similarity between the two patches $\mathbf{z}_i$ and $\mathbf{z}_j$ as $\mathbf{z}_i \approx \mathbf{z}_j$. In (22), we define similarity more precisely and describe how to identify similar patches to compute denoising bounds.

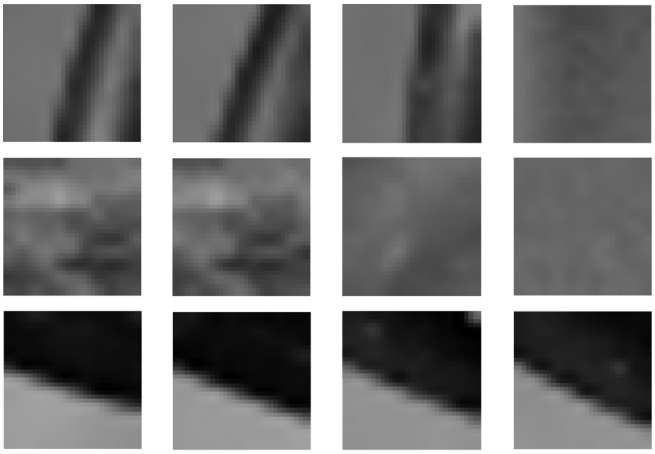
assuming that N similar patches are taken into account in denoising any given patch. Note that (21) is only an approximate expression for the FIM. The FIM takes this exact form only when N *identical* patches are considered. It is also important to reiterate that (21) holds only when we assume that the patches are nonoverlapping. 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 nonoverlapping case where the noise patches can be considered to be IID.

The expression for the FIM [and, hence, the bound in (18)] thus takes into account the strength of the noise, as well as the number of radiometrically similar patches that are considered in denoising any given patch. In (21), we obtain an expression for the FIM considering N radiometrically similar patches being available in denoising any given patch. However, the number of such similar patches 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. 1(a)] have fewer matching patches than the smoother regions. Thus, using a fixed value of N for the entire image is unwise. In this section, we describe how a value of N is chosen adaptively in a given image. But first, we 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

$$\mathbf{z}_j = \mathbf{z}_i + \boldsymbol{\epsilon}_j \quad \text{such that } \|\boldsymbol{\epsilon}_j\|^2 \leqslant \gamma, \qquad (22)$$

where γ is a small threshold value. It can be seen that one needs to choose a proper threshold to ensure few false positives and negatives in the similarity detection process. 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

$$\gamma = \left(\frac{p \times 255}{100} \right)^2 \times n \qquad (23)$$



(a) (b) (c) (d)

Fig. 3. Some query patches and their respective least similar neighbors as defined by (22) with various values of p found from a dictionary of approximately 450,000 patches from four different images. (a) Query; (b) $p = 5$; (c) $p = 7$; (d) $p = 10$.

where n is the number of pixels in each image patch. The value of p is empirically chosen such that radiometric similarity of patches that satisfy (22) can be guaranteed for all patches. For this, we devised an experiment where 11×11 patches from four different images were used to form a database of approximately 450,000 radiometrically (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 shows some reference patches with interesting structure and the corresponding *least* similar patches that satisfied (22) for different values of p. It can be seen from Fig. 3 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 the paper.

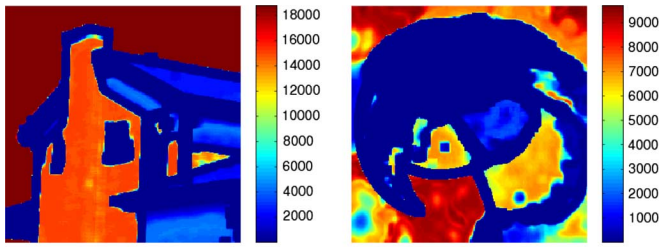
From (22), it is obvious that the number of similar patches will vary from patch to patch and this we denote as N_i. As such, we calculate the FIM on a per patch basis as

$$\mathbf{J}_i = N_i \frac{\mathbf{I}}{\sigma^2} \tag{24}$$

where N_i similar patches are taken into account in denoising a 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 image can be calculated as the aggregate of the patch-wise MSE bounds as

$$E\left[\|\mathbf{z} - \widehat{\mathbf{z}}\|^2\right] \geqslant \frac{1}{|\Omega|} \sum_{i=1}^{|\Omega|} \mathrm{Tr}\left[\left(\mathbf{J}_i + \mathrm{cov}^{-1}(\mathbf{z})\right)^{-1}\right]. \tag{25}$$

Although the FIM is derived for nonoverlapping 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. 4 shows the spatial distribution of N_i values for the house and parrot images calculated with 11×11 patches. As can be expected, N_i takes much larger values for the smoother regions than the edge and textured regions. While the FIM is calculated on a per patch basis, the covariance in (25) is estimated from all the patches in the image, as we describe in the next section.



(a) (b)

Fig. 4. Spatial distribution of N_i values for a patch size of 11×11 on (a) house image, and (b) parrot image, shown in Fig. 5.

C. Estimating the Covariance Matrix

The expression for the lower bound on the MSE given in (25) relies on the computation of the covariance of $\mathbf{z}$. In order to estimate this second moment, we use a bootstrapping method [40] which is robust with respect to possibly small number of samples. 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 parameter of interest (in our case, the second order moment) is 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 resampling is performed sufficiently many times [41]. 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 $\mathrm{cov}(\mathbf{z})$, 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 nonoverlapping patches $\mathbf{z}_i$.
2) Generate $|\Omega|$ samples ($\mathbf{z}_{B,j}$) with replacement from the pool of available $\mathbf{z}_i$ samples (empirical distribution) to generate bootstrap sample set B.
3) Estimate $\mathrm{cov}(\mathbf{z})$ from the bootstrap sample set using the formula

$$\widehat{\mathrm{cov}}(\mathbf{z}) = \frac{1}{(|\Omega| - 1)} \sum_{j=1}^{|\Omega|} (\mathbf{z}_{B,j} - \bar{\mathbf{z}}_B)(\mathbf{z}_{B,j} - \bar{\mathbf{z}}_B)^T \tag{26}$$

   where $\bar{\mathbf{z}}_B$ is the mean of all the $\mathbf{z}_{B,j}$ vectors that make up the bootstrap sample set B.
4) Compute $Q_{\min}$ using the estimated $\mathrm{cov}(\mathbf{z})$ from (25).
5) Repeat steps 2 through 4, r times.

In each of the 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 $\widehat{Q}_{\min}$). The confidence interval of the MSE bound estimate can be readily calculated as the variance of the bootstrap estimates. The 95%

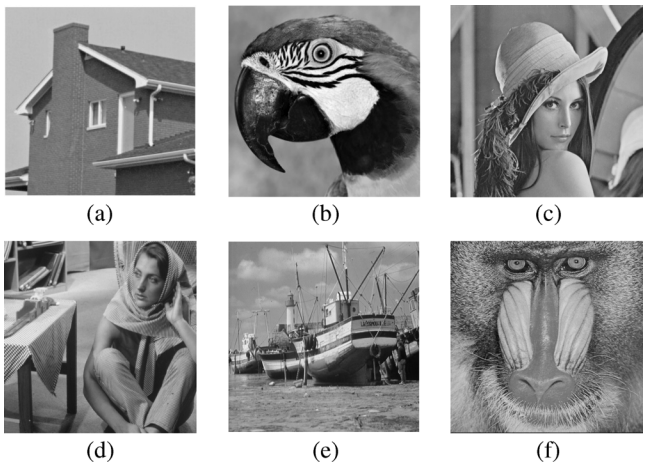


Fig. 5. Some examples of general images that we will use to calculate the bounds. (a) House image; (b) parrot image; (c) Lena image; (d) Barbara image; (e) boats image; (f) Mandrill image.

confidence interval is given by the Normal interval[11] formulation [42]

$$\widehat{Q}_{\min} \pm 2\sigma_Q \tag{27}$$

where σ_Q is the standard error of the bootstrapped estimate $\widehat{Q}_{\min}$. We now have a formulation for estimating the lower bound and its associated confidence interval for an image (or a cluster) where the patches are structurally similar. In the next section we extend this formulation to more general images.

V. Bounds for General Images

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. To make our formulation applicable to geometrically heterogeneous images (Fig. 5), we need to first cluster the image into geometrically similar regions. Analysis of denoising performance can be considered independently for each such region of similar structure. The performance limits on denoising for a particular image can then be calculated as an aggregate of the MSE bounds for each of the clusters. In our previous work [5], we introduced a scheme of clustering an image into geometrically similar regions for image denoising. Later in this section we give a brief description of our automatic "geometric" clustering. In the present scenario, however, we are chiefly interested in computing a performance bound. Therefore, we can 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.

A. Calculating the Bounds

Clustering the given image decomposes it into multiple segments such that patches of similar geometric structure are captured in the same cluster (see Fig. 7). 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 $\mathbf{M}_k^*$ and $\mathbf{u}_k^*$. Consequently, the bounds formulation of (25) holds separately for each cluster. However, before such a bound can be calculated for a particular cluster, we need to estimate the second moment of $p_k(\mathbf{z})$ for that cluster. This is done by forming a sample pool from the (noise-free) patches belonging to the cluster. An estimate of the actual covariance of $\mathbf{z}$ is then obtained by the bootstrapping mechanism outlined in Section IV-C. An MSE bound for each cluster is then obtained from (25). 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 from observing the sum of squared error (SSE) for the entire image, given by

$$\begin{aligned} \mathrm{SSE} &= \sum_{k=1}^{K} \mathrm{SSE}_k = \sum_{k=1}^{K} |\Omega_k| \widehat{Q}_k \\ \Rightarrow \widehat{Q} &= \frac{1}{|\Omega|}\mathrm{SSE} = \sum_{k=1}^{K} \frac{|\Omega_k|}{|\Omega|}\widehat{Q}_k = \sum_{k=1}^{K} \alpha_k \widehat{Q}_k \end{aligned} \tag{28}$$

where $\widehat{Q}$ is the estimate of the bound for the entire image, $\widehat{Q}_k$ and SSE_k are the estimates of the bounds on the MSE and the SSE respectively for the kth cluster, $|\Omega_k|$ denotes the cardinality of the set Ω_k, Ω denotes the set of all patches in the image, and α_k is the weight corresponding to the kth cluster in the averaging process. An expression for the 95% confidence interval can be obtained by calculating the standard deviation (σ_Q) of the $\widehat{Q}$ estimte

$$\sigma_Q = \left(\sum_{k=1}^{K} \alpha_k^2 \sigma_{Q_k}^2 \right)^{\frac{1}{2}} \tag{29}$$

where σ_{Q_k} is the standard deviation of the $\widehat{Q}_k$ estimate. The 95% confidence interval, as shown before in (27), is the Normal interval

$$\widehat{Q} \pm 2\sigma_Q. \tag{30}$$

We now have an expression for the MSE bound and its associated confidence interval computed from an independent analysis of each cluster. Referring to our discussions on the achievability of the bound in Section IV-A, we expect the bound to be theoretically achievable by a linear MMSE estimator in each cluster. 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. Due to all these nuances one can only hope to come up with an affine estimator with performance close to the bound, thus making our formulation a valid lower bound. In the next section, we briefly describe one particular method of performing clustering based on underlying patch geometry.

B. Practical ("Nonoracle") Clustering

Clustering is a much studied research problem that has led to a number of different methods [43], [44]. Image clustering is a subset of this huge field where researchers have devoted considerable attention to the choice of a clustering method as well

[11] This 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.

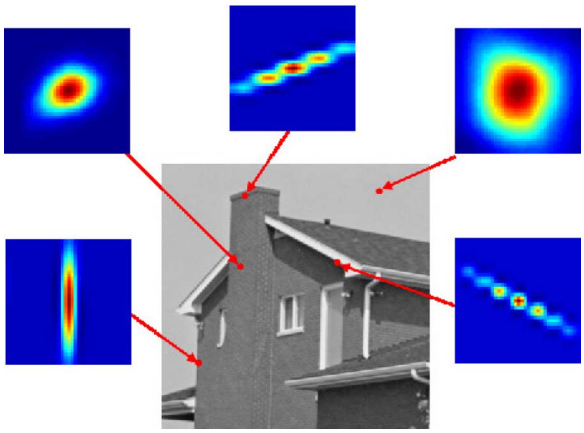


Fig. 6. Steering kernels at different locations of the house image. The patch size is chosen to be 11 × 11. Note how the kernels adapt to the underlying image structure.

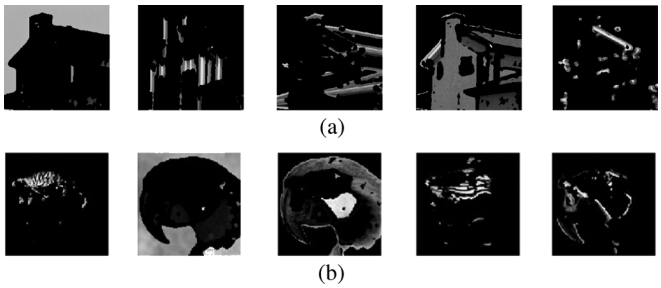


Fig. 7. Clustering results by K-Means algorithm on the house and parrot images. Notice how edges and patterns of a certain kind are clustered together even when they have different intensities. (a) Clustering of house image; (b) clustering of parrot image.

as the features to use to achieve the intended segmentation. 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 the average intensity of the patches. For this, we make use of the data adaptive steering kernels developed by Takeda *et al.* [3]. In that work on Steering Kernel Regression (SKR), robust estimates of the local gradients are taken into account in analyzing the similarity between two pixels in a patch. The gradients are then used to describe the shape and size of a canonical kernel (in particular, a Gaussian). The steering kernel weight at the jth pixel in the ith patch, which is a measure of similarity between the two pixels, is then given by

$$w_{ij} = \frac{\sqrt{\det(\mathbf{C}_j)}}{2\pi h^2} \exp\left\{-\frac{(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{C}_j (\mathbf{x}_i - \mathbf{x}_j)}{2h^2}\right\} \quad (31)$$

where h is a global smoothing parameter also known as the bandwidth of the kernel. The matrix $\mathbf{C}_j$ denotes the gradient covariance formed from the estimated vertical and horizontal gradient of the jth pixel that lies in the ith patch. This allows the Gaussian of (31) to align with the underlying image structure. The weight w_{ij} is calculated for each location in the ith patch to form the weight matrix (or kernel). It is interesting to see that the weight matrix thus formed is indicative of the underlying image geometry. This fact is illustrated in Fig. 6. Note that in each point of the weight matrix a different $\mathbf{C}_j$ is used to compute the weight, and, hence, the kernels do not have simple elliptical contours.

These kernels, normalized such that the weights in a particular patch sum to one, then form excellent descriptors of the underlying image geometry [5], [45]. We make use of these as feature vectors to perform clustering. We chose K-Means [5], [43] for clustering due to its simplicity and efficiency. K-Means requires as input the feature vectors (normalized steering kernels 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 varying geometric structures being grouped together, again resulting in an inaccurate MSE bound. The effect of the number of clusters on the predicted MSE bounds is discussed in Section VI.

Fig. 7 illustrates the effectiveness of using K-Means with our choice of normalized steering kernels as feature vectors. It can be seen that regions that are similar in geometry are clustered together, even though the pixel intensities may widely differ in any particular cluster. Note how even the finer texture like the facade of the house and the cheek of the parrot are correctly captured by the features and are hence differentiated from the largely smooth background. This shows that with our choice of features, a simple clustering method such as K-Means is able to perform geometric clustering accurately.

VI. Results

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. To start this section, we first show results obtained on simulated images of simple repeating patterns. We then show results obtained using uniform texture images and on more general images that consist of both smooth and texture regions. The bound in each case is influenced by the choice of certain parameters such as the size of the patches and the number of clusters used. We present results that illustrate how the bounds are influenced by each of these parameters.

Fig. 8(a) shows a simulated image (of size 220 × 220) that we generated to provide a proof of concept for our MSE bounds calculation. The image consists of simple repeating patterns (stripes), each 2 pixels wide. The image is made up of two gray levels (75 and 200). It is very easy to see that for a sufficiently large patch size 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 different noise strengths. We compare the lower bound to the MSE obtained using various state-of-the-art denoising methods ([3]–[6]). It can be seen from the plots in Fig. 8(c) 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 and predicts a lower bound on the MSE that can be seen to be rather

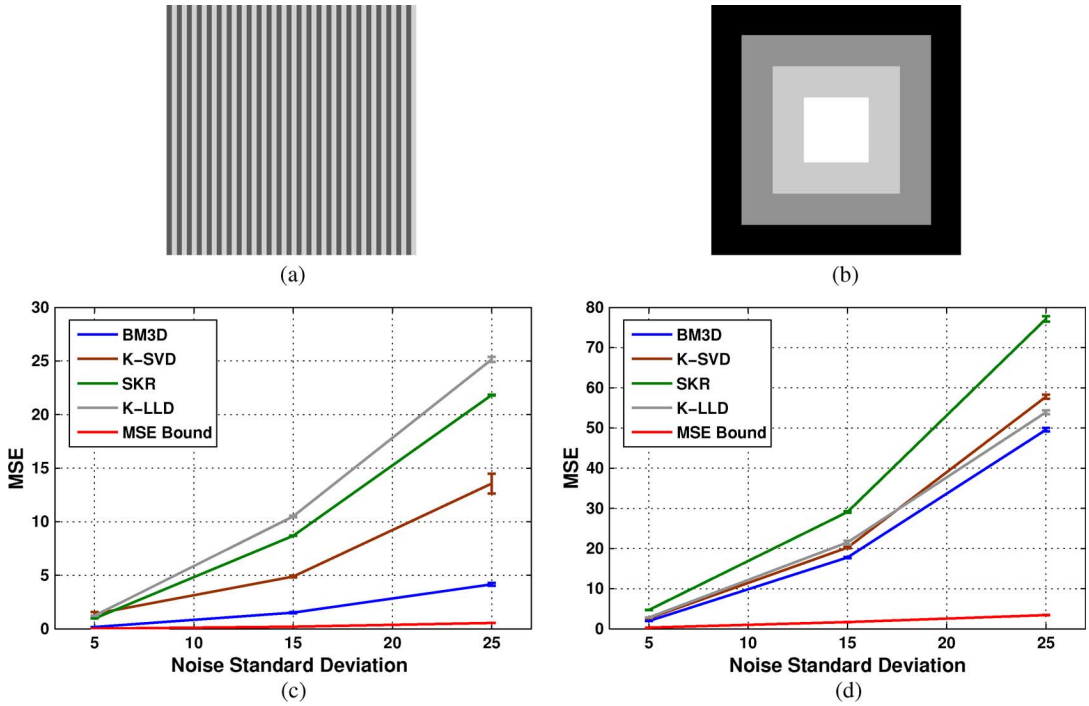


Fig. 8. MSE bounds computed on simulated images and compared with the performance of some state-of-the-art methods (BM3D [6], K-SVD [4], SKR [3], K-LLD [5]). The bounds are calculated using 11×11 patches. (a) Stripes image; (b) box image; (c) bounds for stripes image; (d) bounds for box image.

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. 8(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. 1, we make use of four clusters to capture the smooth, horizontal and vertical edges, and the corner regions. Fig. 8(d) shows the calculated MSE bounds for the box image for different noise standard deviations and compares them to the performance of denoising methods. It can be seen that this image is more difficult to denoise than the stripes image and the predicted MSE bound is 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 typically contain (stochastic) repetitive patterns. Patches from such images can be considered to be geometrically similar and hence can be grouped together in a single cluster. However, the patches typically contain more natural variability in their structure than the simulated stripes image. It can be seen in Fig. 9(c) that most of the methods perform quite comparably to the predicted bounds in the heavily textured grass image. The bound in Fig. 9(d) for another texture image (cloth) is lower than the best performing method (BM3D), but not significantly so. These seem to indicate that, theoretically, the performance can not 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 (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 show experimental results and bounds calculations for some general images (Fig. 5). Such images are typically made up of regions of both smooth and textured areas. Thus, clustering is needed to group together patches of similar geometric structure. In our experiments, we cluster each image into five clusters using the technique outlined in Section V-B. 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. 10 shows the MSE bounds obtained for these 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 idea of the relative difficulty and the amount of improvement that can be expected to be achieved in denoising the images. This can be seen in Table II where the images are ranked based on the relative efficiency which we define as the ratio of the predicted MSE bound and the MSE of the best denoising algorithm for each image. There it can also be seen that images containing a fair amount of texture are denoised quite well as compared to the predicted bounds and little room for improvement exists. However, at higher signal-to-noise levels, there is still room for improvement even for these images. This can be seen from Fig. 11 where we plot the relative efficiency (in decibels) across different noise levels. The plots there also indicate that better denoising can be expected for much smoother images, even though the absolute MSEs of denoising methods are comparatively lower than those for textured images (Fig. 10).

Until now, the bounds that we have shown were obtained with a fixed patch size (11×11). Not surprisingly, the patch size plays a role in calculation of the MSE bounds. Too large a patch size might capture regions of widely varying geometric structure

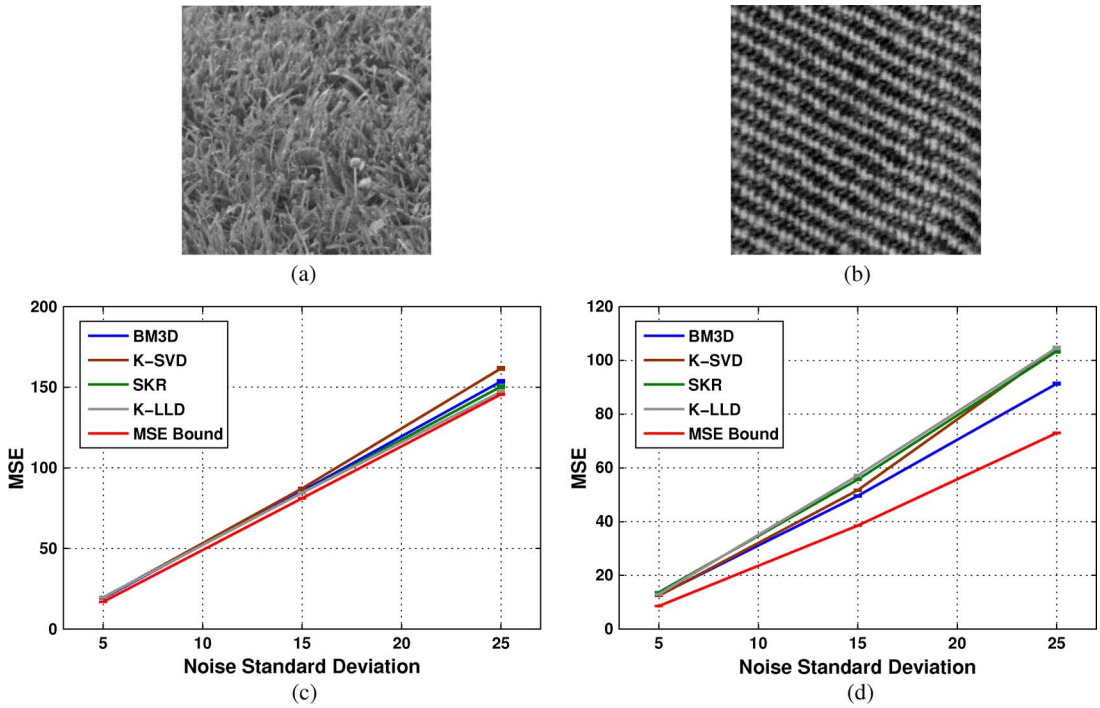


Fig. 9. Bounds for texture images compared to denoising performance of some state-of-the-art denoising methods. A single cluster is considered for this experiment with a patch size of 11 × 11. (a) Grass image; (b) cloth image; (c) bounds for grass image; (d) bounds for cloth image.

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 patches. But this effect is typically stabilized with patch sizes of 11 × 11 or beyond. Fig. 12(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 calculated 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.

Another 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. 12(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 ($K = O(|\Omega|)$). 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 $p(\mathbf{z})$. This can be seen for the general images of Fig. 5 where 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.

The experimental results in this section show that although the current state-of-the-art methods perform denoising quite well, there is still room for improvement in some cases. In that sense, given the tremendous variety of images, denoising is a problem that is still open. In practice, images containing more texture (e.g., parrot and Barbara images) are harder to denoise than smoother ones (e.g., house image) and this is shown in the denoising performance of all the state-of-the-art denoising methods to which we compared. However, our bounds seem to indicate that very little room for improvement exists for images rich in texture. One probable reason for this is that for naturally occurring textures, few similar patches may exist, leading to poor denoising. 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 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 can be seen from the various experiments shown in this section.

VII. Conclusion

In this paper, we studied performance bounds on denoising. We formulated a method of calculating the lower bounds on the MSE that accounts for the strength of the corrupting noise, the number of observations that are typically available to estimate a denoised patch, as well as the variability of the geometric structures in the image. It was derived taking into account the effect of prior information, without the knowledge of the entire prior

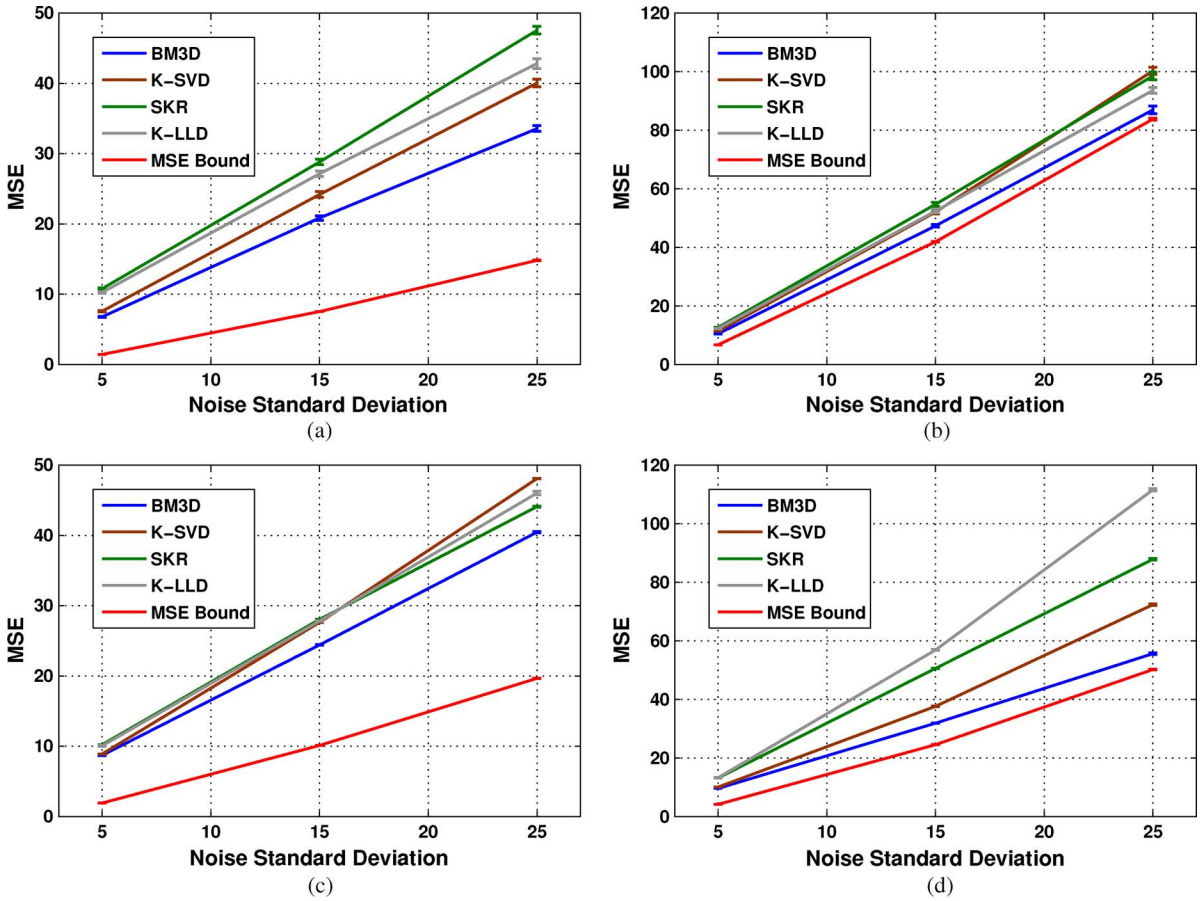


Fig. 10. Comparison of some state-of-the-art methods with our bounds formulation for some general images. The patch size is fixed at 11 × 11 and the number of clusters (K) used is 5 in all the cases. (a) House image; (b) parrot image; (c) Lena image; (d) Barbara image.

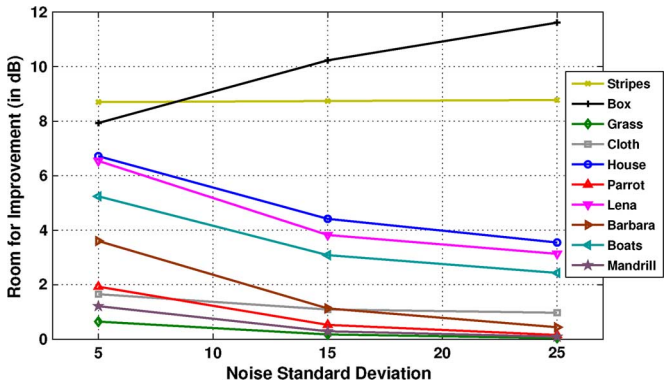


Fig. 11. Room for improvement (in decibels) for various images at different noise levels, as predicted by our bounds formulation.

pdf. The formulation is on a per cluster basis, where each cluster consists of patches of similar geometric structure. Our results showed that there is room for improvement in denoising a class of sufficiently simple images. On the other hand, images rich in (particularly repeating stochastic) texture have already been denoised close to the limit of performance, although some gain can yet be achieved at low noise levels where most methods exhibit bias and artifacts. Our bounds formulation can also be extended to study performance limits of denoising *videos*. For that, the entire video can be clustered into multiple clusters, each cluster being composed of $3-D$ space-time patches. The MSE bound can then be calculated from the bounds learned for each cluster, as has been done for the case of images.

In our present formulation, we assume that the noise-free image is available in order to predict how well the given image can be ideally denoised. As such, the bounds are also useful as a relative measure of denoising quality, which can allow us to rank images in terms of their denoising "complexity". Practically speaking, one may also want to know how well a particular *noisy* image can be denoised when no such ground truth is available. One way of obtaining such an estimate is through the use of Stein's unbiased risk estimator (SURE) [46] for a theoretically optimal estimator. Our method can be modified to study the performance bounds in such cases as well. One possible way of doing this is through a learning based method where a large database of a variety of noise-free images are available. These images can be clustered based on geometric similarity of patches and the covariance matrices can be calculated for each of the clusters. Then, given any noisy image, it can be clustered (as described in Section V-B) and N_i values for each patch can be estimated from either the noisy image patches or using patches from a naive denoised estimate of the image. The bound for each of the clusters can then be calculated using the covariance matrix looked up from the nearest cluster in the noise-free image database. Although the predicted bound in such a case will not

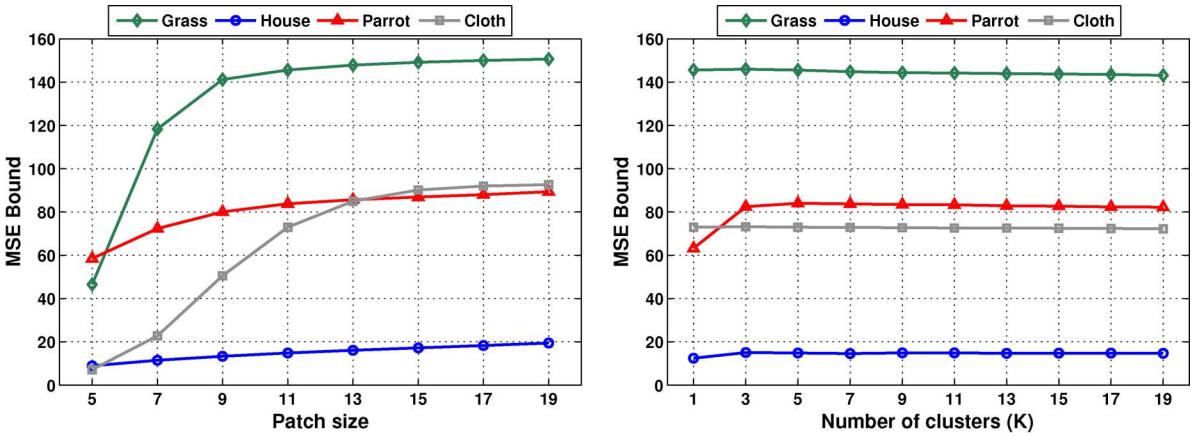


Fig. 12. MSE bounds for noise standard deviation 25 as a function of (a) varying patch size with $K = 1$ for the grass and cloth images, and $K = 5$ for house and parrot images and (b) varying number of clusters with patch size 11×11.

be as accurate, it can still serve as an indicator of the optimal parameters to use for denoising.

Appendix A
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 nonlinear 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

$$\mathbf{z}_j = \mathbf{z}_i + \boldsymbol{\epsilon}_j \quad \text{such that} \quad \|\boldsymbol{\epsilon}_j\|^2 \leqslant \gamma \tag{32}$$

where γ is some small threshold value and $\boldsymbol{\epsilon}_j$ is a vector. The denoised estimate $\widehat{\mathbf{z}}_i$ 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

$$\widehat{\mathbf{z}}_i = \sum_{j=1}^{N} \mathbf{W}_{ij}\mathbf{y}_j \tag{33}$$

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 (2), and (32) above, we can express (33) as

$$\begin{aligned}\widehat{\mathbf{z}}_i &= \sum_j \mathbf{W}_{ij}\mathbf{y}_j = \sum_j \mathbf{W}_{ij}(\mathbf{z}_j + \boldsymbol{\eta}_j) \\ &= \sum_j \mathbf{W}_{ij}(\mathbf{z}_i + \boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j).\end{aligned} \tag{34}$$

The expected value of this estimate can then be written as

$$\begin{aligned}E[\widehat{\mathbf{z}}_i|\mathbf{z}_i] &= E\left[\sum_j \mathbf{W}_{ij}(\mathbf{z}_i + \boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j)\right] \\ &= E\left[\sum_j \mathbf{W}_{ij}\right]\mathbf{z}_i + \sum_j E\left[\mathbf{W}_{ij}(\boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j)\right].\end{aligned} \tag{35}$$

This allows us to calculate the bias of such nonlinear weighted averaging methods as

$$\begin{aligned}\mathbf{b}(\mathbf{z}_i) &= E[\widehat{\mathbf{z}}_i|\mathbf{z}_i] - \mathbf{z}_i \\ &= \left(E\left[\sum_j \mathbf{W}_{ij}\right] - \mathbf{I}\right)\mathbf{z}_i + \sum_j E\left[\mathbf{W}_{ij}(\boldsymbol{\epsilon}_j + \boldsymbol{\eta}_j)\right] \\ &= \mathbf{M}_i\mathbf{z}_i + \mathbf{u}_i\end{aligned} \tag{36}$$

where $\mathbf{M}_i = (E[\sum_j \mathbf{W}_{ij}] - \mathbf{I})$ 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, $\mathbf{M}_i$ and $\mathbf{u}_i$) are different for each patch, we make the simplifying assumption that the same $\mathbf{M}$ and $\mathbf{u}$ provide an adequate approximation of the bias within members of the same cluster. This assumption is also statistically justified in Section II of the paper.

Appendix B
Optimal Parameters for Affine Bias Function

In this section we derive expressions for $\mathbf{M}$ and $\mathbf{u}$ that minimize the cost function of (13). This can be obtained by solving a system of simultaneous equations [shown in (14)]. To do this, we first solve for $\mathbf{u}$

$$\begin{aligned}\frac{\partial Q}{\partial \mathbf{u}} &= \frac{\partial}{\partial \mathbf{u}} \int_{\mathbf{z}} \Big[\mathrm{Tr}\left\{(\mathbf{I}+\mathbf{M})\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M})^T\right\} \\ &\qquad + (\mathbf{M}\mathbf{z}+\mathbf{u})^T(\mathbf{M}\mathbf{z}+\mathbf{u})\Big]\, p(\mathbf{z})d\mathbf{z} = 0 \\ \Rightarrow \int_{\mathbf{z}} &\left[\frac{\partial}{\partial \mathbf{u}}(\mathbf{M}\mathbf{z}+\mathbf{u})^T(\mathbf{M}\mathbf{z}+\mathbf{u})\right] p(\mathbf{z})d\mathbf{z} = 0 \\ \Rightarrow \mathbf{u} &= -\mathbf{M}\int_{\mathbf{z}} \mathbf{z}\, p(\mathbf{z})d\mathbf{z} = -\mathbf{M}E[\mathbf{z}].\end{aligned} \tag{37}$$

Taking the derivative with respect to $\mathbf{M}$, we get

$$\begin{aligned}\frac{\partial Q}{\partial \mathbf{M}} &= \frac{\partial}{\partial \mathbf{M}} \int_{\mathbf{z}} \left[\mathrm{Tr}\left\{ (\mathbf{I}+\mathbf{M})\mathbf{J}^{-1}(\mathbf{I}+\mathbf{M})^T \right\} \right. \\ &\qquad \left. +(\mathbf{M}\mathbf{z}+\mathbf{u})^T(\mathbf{M}\mathbf{z}+\mathbf{u}) \right] p(\mathbf{z})d\mathbf{z} = 0 \\ \Rightarrow & \int_{\mathbf{z}} \left[2(\mathbf{I}+\mathbf{M})\mathbf{J}^{-1} + 2(\mathbf{M}\mathbf{z}+\mathbf{u})\mathbf{z}^T \right] p(\mathbf{z})d\mathbf{z} = 0 \\ \Rightarrow & \mathbf{M}\left[\mathbf{J}^{-1} + \int_{\mathbf{z}} \mathbf{z}\mathbf{z}^T p(\mathbf{z})d\mathbf{z} \right] = -\left[\mathbf{J}^{-1} + \mathbf{u}\int_{\mathbf{z}} \mathbf{z}^T p(\mathbf{z})d\mathbf{z} \right] \\ \Rightarrow & \mathbf{M} = -\left[\mathbf{J}^{-1} + \mathbf{u}E[\mathbf{z}]^T \right] \left[\mathbf{J}^{-1} + E[\mathbf{z}\mathbf{z}^T] \right]^{-1}. \end{aligned} \tag{38}$$

Now, using (37) in (38), we get

$$\begin{aligned} & \mathbf{M}\left[\mathbf{J}^{-1} + E[\mathbf{z}\mathbf{z}^T] \right] = -\left[\mathbf{J}^{-1} - \mathbf{M}E[\mathbf{z}]E[\mathbf{z}]^T \right] \\ \Rightarrow & \mathbf{M}\left[\mathbf{J}^{-1} + E[\mathbf{z}\mathbf{z}^T] - E[\mathbf{z}]E[\mathbf{z}]^T \right] = -\mathbf{J}^{-1} \\ \Rightarrow & \mathbf{M}^* = -\mathbf{J}^{-1}\left[\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z}) \right]^{-1} \end{aligned} \tag{39}$$

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

$$\mathbf{M}^* = -\mathbf{J}^{-1}\left[\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z}) \right]^{-1} \tag{40}$$

$$\mathbf{u}^* = -\mathbf{M}^* E[\mathbf{z}] = \mathbf{J}^{-1}\left[\mathbf{J}^{-1} + \mathrm{cov}(\mathbf{z}) \right]^{-1} E[\mathbf{z}]. \tag{41}$$

Appendix C
Higher Order Bias Model

In Section II, 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 Section IV-A of the paper 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 given by

$$\begin{aligned} & \mathbf{b}(\mathbf{z}) = [\, b_1(\mathbf{z}) \ \ldots \ b_l(\mathbf{z}) \ \ldots \ b_n(\mathbf{z}) \,]^T \quad \text{and} \\ & b_l(\mathbf{z}) = a_l \mathbf{z}^T\mathbf{z} + \mathbf{m}_l^T\mathbf{z} + u_l \end{aligned} \tag{42}$$

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

$$\begin{aligned} Q &= \int_{\mathbf{z}} \left[\mathrm{Tr}\left\{ \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right) \mathbf{J}^{-1} \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right)^T \right\} \right. \\ &\qquad \left. +\mathbf{b}^T(\mathbf{z})\mathbf{b}(\mathbf{z}) \right] p(\mathbf{z})d\mathbf{z} \\ &= \int_{\mathbf{z}} \left[\frac{1}{\sigma^2}\mathrm{Tr}\left\{ \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right) \left(\frac{\partial E[\hat{\mathbf{z}}]}{\partial \mathbf{z}} \right)^T \right\} \right. \\ &\qquad \left. +\sum_{l=1}^{n} b_l^2(\mathbf{z}) \right] p(\mathbf{z})d\mathbf{z} \\ &= \int_{\mathbf{z}} \left[\frac{1}{\sigma^2}\mathrm{Tr}\left\{ \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right) \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right)^T \right\} \right. \\ &\qquad \left. +\sum_{l=1}^{n} b_l^2(\mathbf{z}) \right] p(\mathbf{z})d\mathbf{z} \end{aligned} \tag{43}$$

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

$$\begin{aligned} & \mathrm{Tr}\left\{ \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right) \left(\mathbf{I} + \frac{\partial \mathbf{b}(\mathbf{z})}{\partial \mathbf{z}} \right)^T \right\} \\ &= \mathrm{Tr}\left\{ \begin{bmatrix} \vdots \\ \left[\boldsymbol{c}_l + \frac{\partial b_l(\mathbf{z})}{\partial \mathbf{z}} \right]^T \\ \vdots \end{bmatrix} \begin{bmatrix} \vdots \\ \left[\boldsymbol{c}_l + \frac{\partial b_l(\mathbf{z})}{\partial \mathbf{z}} \right]^T \\ \vdots \end{bmatrix}^T \right\} \\ &= \sum_{l=1}^{n} (2a_l\mathbf{z} + \mathbf{m}_l + \mathbf{c}_l)^T (2a_l\mathbf{z} + \mathbf{m}_l + \mathbf{c}_l) \end{aligned} \tag{44}$$

where $\mathbf{c}_l$ is the lth column of the identity matrix containing all zeros except a one at the lth position and $\partial b_l(\mathbf{z})/\partial \mathbf{z} = (2a_l\mathbf{z} + \mathbf{m}_l)$. We can then write (43) as

$$\begin{aligned} Q = \int_{\mathbf{z}} \sum_{l=1}^{n} \left[\frac{1}{\sigma^2}(2a_l\mathbf{z} + \mathbf{m}_l + \mathbf{c}_l)^T(2a_l\mathbf{z} + \mathbf{m}_l + \mathbf{c}_l) \right. \\ \left. +\left(a_l\mathbf{z}^T\mathbf{z} + \mathbf{m}_l^T\mathbf{z} + u_l \right)^2 \right] p(\mathbf{z})d\mathbf{z}. \end{aligned} \tag{45}$$

As before, we take the derivatives of the right hand side of (45) with respect to the unknown parameters (a_l, $\mathbf{m}_l$ and u_l) and solve the equations to get expressions for the optimal parameters that minimize Q. Differentiating the right hand side of (45) with respect to a_l, $\mathbf{m}_l$, and u_l, we get three simultaneous equations

$$u_l = -a_l E[\mathbf{z}^T\mathbf{z}] - \mathbf{m}_l^T E[\mathbf{z}] \tag{46}$$

$$\begin{aligned} \mathbf{m}_l = & -\left(\frac{\mathbf{I}}{\sigma^2} + E[\mathbf{z}\mathbf{z}^T] \right)^{-1} \\ & \times \left(\frac{1}{\sigma^2}\mathbf{c}_l + 2\frac{a_l}{\sigma^2}E[\mathbf{z}] + a_l E[\mathbf{z}\mathbf{z}^T\mathbf{z}] + E[\mathbf{z}]u_l \right) \end{aligned} \tag{47}$$

$$\begin{aligned} a_l = & -\left(\frac{4}{\sigma^2}E[\mathbf{z}^T\mathbf{z}] + E[\mathbf{z}^T\mathbf{z}\mathbf{z}^T\mathbf{z}] \right)^{-1} \\ & \times \left(\frac{2\mathbf{c}_l^T}{\sigma^2} + \mathbf{m}_l^T\left(\frac{2}{\sigma^2} + E[\mathbf{z}\mathbf{z}^T\mathbf{z}] \right) + u_l E[\mathbf{z}^T\mathbf{z}] \right). \end{aligned} \tag{48}$$

Now, using the expression for u_l from (46) in (47) and (48), we get the system of equations in two variables

$$\begin{aligned} \mathbf{m}_l = & -\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z}) \right)^{-1} \\ & \times \left(\frac{\mathbf{c}_l}{\sigma^2} + a_l\left(\frac{2}{\sigma^2}E[\mathbf{z}] + E[\mathbf{z}\mathbf{z}^T\mathbf{z}] - E[\mathbf{z}]E[\mathbf{z}^T\mathbf{z}] \right) \right) \end{aligned} \tag{49}$$

$$a_l^* = \frac{\mathbf{c}_l^T \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z})}{\sigma^2 \left\{ \mathrm{Tr}\left[\mathbf{K}(\mathbf{z})\right] - \mathbf{S}^T(\mathbf{z}) \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z}) + \mathrm{Tr}^2\left[\mathrm{cov}(\mathbf{z})\right] \right\}} \tag{55}$$

$$a_l \left[\frac{4}{\sigma^2} E[\mathbf{z}^T\mathbf{z}] + E[\mathbf{z}^T\mathbf{z}\mathbf{z}^T\mathbf{z}] - \left(E[\mathbf{z}^T\mathbf{z}]\right)^2 \right] = -\mathbf{m}_l^T \left[2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right] - \frac{2\mathbf{c}_l^T}{\sigma^2} E[\mathbf{z}] \tag{50}$$

where we denote $\mathbf{S}(\mathbf{z}) = E[(\mathbf{z} - E[\mathbf{z}])(\mathbf{z} - E[\mathbf{z}])^T(\mathbf{z} - E[\mathbf{z}])]$ 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 $\mathbf{m}_l$ as given in (49) and plug it in (50) 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}] - \left(E[\mathbf{z}^T\mathbf{z}]\right)^2 \right] \\
&\quad = -\left[-\frac{\mathbf{c}_l^T}{\sigma^2} - a_l \left\{ 2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right\} \right] \\
&\qquad \times \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \left[2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right] \\
&\qquad - \frac{2\mathbf{c}_l^T}{\sigma^2} E[\mathbf{z}] \\
&\quad = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \\
&\qquad \times \left[2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right] - \frac{2\mathbf{c}_l^T}{\sigma^2} E[\mathbf{z}] \\
&\qquad + a_l \left[2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right]^T \\
&\qquad \times \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \left[2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right] \\
&\Rightarrow\ 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. \\
&\qquad - \left(2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right)^T \\
&\qquad \times \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \\
&\qquad \left. \times \left(2\left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] + \mathbf{S}(\mathbf{z}) \right) \right] \\
&\quad = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z}) \\
&\Rightarrow\ a_l \left[\frac{4}{\sigma^2} E[\mathbf{z}^T\mathbf{z}] + E\left[(\mathbf{z}^T\mathbf{z})^2\right] \right. \\
&\qquad -4E[\mathbf{z}^T] \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] \\
&\qquad \left. -4E[\mathbf{z}^T]\mathbf{S}(\mathbf{z}) - \mathbf{S}^T(\mathbf{z}) \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z}) \right] \\
&\quad = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z}).
\end{aligned} \tag{51}$$

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

$$\frac{4E[\mathbf{z}^T\mathbf{z}]}{\sigma^2} + E[\mathbf{z}^T\mathbf{z}\mathbf{z}^T\mathbf{z}] - 4E[\mathbf{z}^T] \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right) E[\mathbf{z}] - 4E[\mathbf{z}^T]\mathbf{S}(\mathbf{z}) = \mathrm{Tr}\left(\mathbf{K}(\mathbf{z})\right) + \mathrm{Tr}^2\left(\mathrm{cov}(\mathbf{z})\right) \tag{52}$$

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

$$\mathbf{K}(\mathbf{z}) = E\left[(\mathbf{z} - E[\mathbf{z}]) (\mathbf{z} - E[\mathbf{z}])^T (\mathbf{z} - E[\mathbf{z}]) (\mathbf{z} - E[\mathbf{z}])^T \right]. \tag{53}$$

This allows us to rewrite (51) as

$$a_l \left[\mathrm{Tr}\left(\mathbf{K}(\mathbf{z})\right) + \mathrm{Tr}^2\left(\mathrm{cov}(\mathbf{z})\right) - \mathbf{S}^T(\mathbf{z}) \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z}) \right] = \frac{\mathbf{c}_l^T}{\sigma^2} \left(\frac{\mathbf{I}}{\sigma^2} + \mathrm{cov}(\mathbf{z})\right)^{-1} \mathbf{S}(\mathbf{z}) \tag{54}$$

which leads to the expression for the optimal a_l parameter as (55), shown at the top of the page.

Skewness is very good indicator of reflectance properties of surfaces such as albedo and gloss [47], [48]. 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 [49]. 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 $\mathbf{S}(\mathbf{z})$ related to the skewness is close to zero and the optimal bias model then collapses to the affine model that we have used earlier.

REFERENCES

[1] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising methods, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, 2005.

[2] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2866–2878, Oct. 2006.

[3] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 349–366, Feb. 2007.

[4] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[5] P. Chatterjee and P. Milanfar, "Clustering-based denoising with locally learned dictionaries," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1438–1451, Jul. 2009.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. O. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[7] M. D. Robinson and P. Milanfar, "Fundamental performance limits in image registration," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1185–1199, Sep. 2004.

[8] M. Xu, H. Chen, and P. K. Varshney, "Ziv-Zakai bounds on image registration," *IEEE Trans. Signal Process.*, vol. 57, no. 5, pp. 1745–1755, May 2009.

[9] M. D. Robinson and P. Milanfar, "Statistical performance analysis of super-resolution," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1413–1428, Jun. 2006.

[10] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, pp. 1167–1183, Sep. 2002.
[11] Z. Lin and H.-Y. Shum, "Fundamental limits of reconstruction-based superresolution algorithms under local translation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 83–97, Jan. 2002.
[12] Z. Lin, J. He, X. Tang, and C.-K. Tang, "Limits of learning-based superresolution algorithms," *Int. J. Comput. Vis.*, vol. 80, no. 3, pp. 406–420, Aug. 2008.
[13] T. Treibitz and Y. Y. Schechner, "Recovery limits in pointwise degradation," presented at the IEEE Int. Conf. Computational Photography, San Francisco, CA, Apr. 2009.
[14] S. Voloshynovskiy, O. Koval, and T. Pun, "Image denoising based on the edge-process model," *Signal Process.*, vol. 85, no. 10, pp. 1950–1969, Oct. 2005.
[15] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *SIAM Multiscale Model. Simul.*, vol. 7, no. 1, pp. 214–241, Apr. 2008.
[16] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," presented at the IEEE Int. Conf. Computer Vision, Tokyo, Japan, Sep. 2009.
[17] S. Kay and Y. C. Eldar, "Rethinking biased estimation," *Signal Process. Mag.*, vol. 25, no. 3, pp. 133–136, May 2008.
[18] Y. C. Eldar, "Rethinking biased estimation: Improving maximum likelihood and the Cramér–Rao bound," *Found. Trends Signal Process.*, vol. 1, no. 4, pp. 305–449, 2008.
[19] Y. C. Eldar, "MSE bound with affine bias dominating the Cramér–Rao bound," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3824–3836, Aug. 2008.
[20] N. Draper and H. Smith, *Applied Regression Analysis*, ser. Probability and Statistics, 3rd ed. Hoboken, NJ: Wiley, Apr. 1998.
[21] L. P. Seidman, "Performance limitations and error calculations for parameter estimation," *Proc. IEEE*, vol. 58, pp. 644–652, May 1970.
[22] H. Cramér, *Mathematical Methods of Statistics*. Princeton, NJ: Princeton Univ. Press, 1946.
[23] C. R. Rao, "Information and the accuracy attainable in the estimation of statistical parameters," *Bull. Calcutta Math. Soc.*, vol. 37, pp. 81–89, 1945.
[24] C. R. Rao, "Minimum variance and the estimation of several parameters," in *Proc. Cambridge Philosophical Society*, 1946, vol. 43, pp. 280–283.
[25] J. Ziv and M. Zakai, "Some lower bounds on signal parameter estimation," *IEEE Trans. Inf. Theory*, vol. IT-15, no. 3, pp. 386–391, May 1969.
[26] *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*, H. L. van Trees and K. L. Bell, Eds., 1st ed. Piscataway, NJ: IEEE, Aug. 2007.
[27] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, ser. Signal Processing. Upper Saddle River, NJ: Prentice-Hall, 1993, vol. 1.
[28] H. L. van Trees, *Detection, Estimation, and Modulation Theory*. New York: Wiley, 1968.
[29] J. A. Fessler and A. O. Hero, "Cramér–Rao lower bounds for biased image reconstruction," in *Proc. 36th Midwest Symp. Circuits and Systems*, Aug. 1993, vol. 1, pp. 253–256.
[30] T. Y. Young and R. A. Westerberg, "Error bounds for stochastic estimation of signal parameters," *IEEE Trans. Inf. Theory*, vol. IT-17, no. 5, Sep. 1971.
[31] Z. Ben-Haim and Y. C. Eldar, "A lower bound on the Bayesian MSE based on the optimal bias function," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5179–5196, Nov. 2009.
[32] J. S. D. Bonet and P. Viola, "A non-parametric multi-scale statistical model for natural images," in *Advances in Neural Information Processing*. Cambridge, MA: MIT Press, 1997, pp. 773–779.
[33] W. Hong, J. Wright, K. Huang, and Y. Ma, "Multiscale hybrid linear models for lossy image representation," *IEEE Trans. Image Proces.*, vol. 15, no. 12, pp. 3655–3671, Dec. 2006.
[34] S.-C. Zhu, "Statistical modeling and conceptualization of visual patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 691–712, Jun. 2003.
[35] E. P. Simoncelli and B. A. Olshausen, "Natural image statistics and neural representation," *Annu. Rev. Neurosci.*, vol. 24, pp. 1193–1216, May 2001.
[36] A. Srivastava, A. B. Lee, E. P. Simoncelli, and S.-C. Zhu, "On advances in statistical modeling of natural images," *J. Math. Imag. Vis.*, vol. 18, pp. 17–33, 2003.
[37] S. Lyu and E. P. Simoncelli, "Modeling multiscale subbands of photographic images with fields of Gaussian scale mixtures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 693–706, Apr. 2009.
[38] Y. Weiss and W. Freeman, "What makes a good model of natural images?," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Minneapolis, MN, Jun. 2007, pp. 1–8.
[39] M. A. Woodbury, Inverting Modified Matrices, Statistical Research Group, Princeton Univ., Princeton, NJ, Memorandum Rep. 42, 1950.
[40] B. Efron, "Bootstrap methods: Another look at the Jacknife," *Ann. Statist.*, vol. 7, no. 1, pp. 1–26, 1979.
[41] M. R. Chernick, *Bootstrap Methods: A Guide for Practitioners and Researchers*, ser. Probability and Statistics, 2nd ed. Hoboken, NJ: Wiley, Nov. 2007.
[42] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*, ser. Springer Texts in Statistics, 1st ed. New York: Springer, 2004.
[43] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, March 1982.
[44] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.
[45] H. J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.
[46] C. M. Stein, "Estimation of the mean of a multivariate normal distribution," *Ann. Statist.*, vol. 9, no. 6, pp. 1135–1151, Nov. 1981.
[47] 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.
[48] L. Sharan, Y. Li, I. Motoyoshi, S. Nishida, and E. Adelson, "Image statistics for surface reflectance perception," *J. Opt. Soc. Amer. A*, vol. 25, no. 4, pp. 846–865, Apr. 2008.
[49] A. Hyvärinen, J. Hurri, and P. O. Hoyer, *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision*, ser. Computational Imaging and Vision. London, U.K.: Springer, 2009, vol. 39.

Priyam Chatterjee (S'07) received the B.Tech. degree in Information technology from the University of Kalyani, India, and the M.Tech. degree in electrical engineering from the Indian Institute of Technology (IIT), Bombay, India, in 2003 and 2006, respectively. He is currently pursuing the Ph.D. degree in electrical engineering at the University of California, Santa Cruz.

His research interests are in image and video processing (denoising, interpolation, deblurring, and super-resolution).

Peyman Milanfar (SM'98–F'10) received the B.S. degree in electrical engineering/mathematics from the University of California, Berkeley, and the Ph.D. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge.

He is a is Professor of electrical engineering at the University of California, Santa Cruz (UCSC). Prior to UCSC, he was at SRI (formerly Stanford Research Institute) and a Consulting Professor of computer science at Stanford University, Stanford, CA. In 2005, he founded MotionDSP, Inc. to bring state-of-art video enhancement technology to consumer and forensic markets. His technical interests are in statistical signal, image and video processing, and computational vision.

Dr. Milanfar won a National Science Foundation CAREER award in 2000. He is an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is a member of the Signal Processing Society's Image, Video, and Multidimensional Signal Processing (IVMSP) Technical Committee.