MDL Denoising Revisited

Teemu Roos, Petri Myllymäki, and Jorma Rissanen

***Abstract*— We refine and extend an earlier minimum description length (MDL) denoising criterion for wavelet-based denoising. We start by showing that the denoising problem can be reformulated as a clustering problem, where the goal is to obtain separate clusters for informative and non-informative wavelet coefficients, respectively. This suggests two refinements, adding a code-length for the model index, and extending the model in order to account for subband-dependent coefficient distributions. A third refinement is the derivation of soft thresholding inspired by predictive universal coding with weighted mixtures. We propose a practical method incorporating all three refinements, which is shown to achieve good performance and robustness in denoising both artificial and natural signals.**

***Index Terms*— Minimum description length (MDL) principle, wavelets, denoising.**

I. Introduction

WAVELETS are widely applied in many areas of signal processing [1], where their popularity owes largely to efficient algorithms on the one hand and advantages of sparse wavelet representations on the other. The sparseness property means that while the distribution of the original signal values may be very diffuse, the distribution of the corresponding wavelet coefficients is often highly concentrated, having a small number of very large values and a large majority of very small values [2]. It is easy to appreciate the importance of sparseness in signal compression, [3], [4]. The task of removing noise from signals, or *denoising*, has an intimate link to data compression, and many denoising methods are explicitly designed to take advantage of sparseness and compressibility in the wavelet domain, see e.g., [5]–[7].

Among the various wavelet-based denoising methods, those suggested by Donoho and Johnstone [8], [9] are the best known. They follow the frequentist minimax approach, where the objective is to asymptotically minimize the worst-case L^2 risk simultaneously for signals, for instance, in the entire scale of Hölder, Sobolev, or Besov classes, characterized by certain smoothness conditions. By contrast, Bayesian denoising methods minimize the *expected* (Bayes) risk, where the expectation is taken over a given prior distribution supposed to govern the unknown true signal [10], [11]. Appropriate prior models with very good performance in typical benchmark tests, especially for images, include the class of generalized Gaussian densities [6], [12], [13], and scale-mixtures of Gaussians [14], [15] (both of which include the Gaussian and double exponential densities as special cases).

Authors are with the Complex Systems Computation Group, Helsinki Institute for Information Technology HIIT. e-mails: teemu.roos@cs.helsinki.fi, petri.myllymaki@cs.helsinki.fi, jorma.rissanen@mdl-research.org. This work was supported in part by the Academy of Finland under projects MINOS and MODEST, the Finnish Technology Agency under projects PMMA and KUKOT, and the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

A third approach to denoising is based on the minimum description length (MDL) principle [16]–[20]. Several different MDL denoising methods have been suggested [6], [12], [21]–[25]. We focus on what we consider as the most pure MDL approach, namely that of Rissanen [24]. Our motivation is two-fold: First, as an immediate result of refining and extending the earlier MDL denoising method, we obtain a new practical method with greatly improved performance and robustness. Secondly, the denoising problem turns out to illustrate theoretical issues related to the MDL principle, involving the problem of unbounded parametric complexity and the necessity of encoding the model class. The study of denoising gives new insight to these issues.

Formally, the denoising problem is the following. Let $y^n = (y_1, \ldots, y_n)^T$ be a signal represented by a real-valued column vector of length n. The signal can be, for instance, a time-series or an image with its pixels read in a row-by-row order. Let $\mathcal{W}$ be an $n \times m$ regressor matrix whose columns are basis vectors. We model the signal y^n as a linear combination of the basis vectors, weighted by coefficient vector $\beta^n = (\beta_1, \ldots, \beta_m)^T$, plus Gaussian i.i.d. noise:

$$y^n = \mathcal{W}\beta^m + \epsilon^n, \quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_N^2), \tag{1}$$

where σ_N^2 is the noise variance. Given an observed signal y^n, the ideal is to obtain a coefficient vector $\tilde{\beta}^m$ such that the signal given by the transform $\tilde{y}^n = \mathcal{W}\tilde{\beta}^m$ contains the informative part of the observed signal, and the difference $y^n - \tilde{y}^n$ is noise.

For technical convenience, we adopt the common restriction on $\mathcal{W}$ that the basis vectors form a *complete orthonormal* basis. This implies that the number of basis vectors is equal to the length of the signal, $m = n$, and that all the basis vectors are orthogonal unit vectors. There are a number of wavelet transforms that conform to this restriction, for instance, the Haar transform and the family of Daubechies transforms [1], [26]. Formally, the matrix $\mathcal{W}$ is of size $n \times n$ and orthogonal with its inverse equal to its transpose. Also the mapping $\beta^n \mapsto \mathcal{W}\beta^n$ preserves the Euclidean norm, and we have Parseval's equality:

$$||\beta^n|| = \sqrt{\langle \beta^n, \beta^n \rangle} = \sqrt{\langle \mathcal{W}\beta^n, \mathcal{W}\beta^n \rangle} = ||\mathcal{W}\beta^n||. \tag{2}$$

Geometrically this means that the mapping $\beta^n \mapsto \mathcal{W}\beta^n$ is a rotation and/or a reflection. From a statistical point of view, this implies that any spherically symmetric density, such as Gaussian, is invariant under this mapping. All these properties are shared by the mapping $y^n \mapsto \mathcal{W}^T y^n$. We call $\beta^n \mapsto \mathcal{W}\beta^n$ the inverse wavelet transform, and $y^n \mapsto \mathcal{W}^T y^n$ the forward wavelet transform. Note that in practice the transforms are not implemented as matrix multiplications but by a fast wavelet transform similar to the fast Fourier transform (see [1]), and in fact not even the matrices need be written down.

For complete bases, the conventional maximum likelihood (least squares) method obviously fails to provide denoising unless the coefficients are somehow restricted since the solution $\tilde{\beta}^n = \mathcal{W}^T y^n$ gives the reconstruction $\tilde{y}^n = \mathcal{W}\mathcal{W}^T y^n = y^n$ equal to the original signal, including noise. The solution proposed by Rissanen [24] is to consider each subset of the basis vectors separately and to choose the subset that allows the shortest description of the data at hand. The length of the description is determined by the negative logarithm of the so called normalized maximum likelihood (NML) model.

In the linear–quadratic case, the NML model involves an integral, which is undefined unless the range of integration (the support) is restricted. This, in turn, implies hyper parameters, which have received increasing attention in various contexts involving, e.g., Gaussian, Poisson and geometric models [17], [20], [27]–[30]. Rissanen used renormalization to remove them and to obtain a second-level NML model. Although the range of integration has to be restricted also in the second-level NML model, the range for ordinary regression problems does not affect the resulting criterion and can be ignored. Roos et al. [31] give an interpretation of the method which avoids the renormalization procedure and at the same time gives a simplified view of the denoising process in terms of two Gaussian distributions fitted to informative and non-informative coefficients, respectively. In this paper we carry this interpretation further and show that viewing the denoising problem as a clustering problem suggests several refinements and extensions to the original method.

The rest of this paper is organized as follows. A brief introduction to model selection by the MDL principle, and the NML criterion in particular, is given in Sec. II. In Sec. III we reformulate the denoising problem as a task of clustering the wavelet coefficients in two or more sets with different distributions. In Sec. IV we propose three different modifications of Rissanen's method, suggested by the clustering interpretation. In Sec. V the modifications are shown to significantly improve the performance of the method in denoising both artificial and natural signals. The conclusions are summarized in Sec. VI.

II. Model Selection by MDL

The minimum description length (MDL) principle states that we should choose the model that yields the shortest description of the data together with the description of the model itself [16]–[20]. When probabilistic models are used, the description lengths are given by negative logarithms of probability or density[1] values; this can be justified by the Kraft-McMillan theorem, see [18], [20]. In the following, we use natural logarithms, which gives the code lengths in nats (one nat is equal to $1/\ln 2 \approx 1.443$ bits).

A. Stochastic Complexity

The stochastic complexity of a sequence under a given model class is a central concept in the MDL principle. Its interpretation as the length of the shortest achievable description of the data given a model class (a set of distributions) makes it a yardstick for the comparison of different model classes. In recent formulations of MDL, stochastic complexity is defined using the so called normalized maximum likelihood (NML) model, originally introduced by Shtarkov [32] for data compression; for the role of NML in MDL model selection, see [17], [18], [20], [33], [34].

Since the introduction of the NML universal model in the context of MDL, there has been significant interest in the evaluation of NML stochastic complexity for different practically relevant model classes, both exactly and asymptotically. For discrete models, exact evaluation is often computationally infeasible since it involves a normalizing coefficient which is a sum over all possible data-sets. For continuous cases, the normalizing coefficient is an integral which can be solved in only a few cases. Under certain conditions on the model class, different versions of stochastic complexity (which include two-part, mixture, and NML forms) have the same asymptotic form — the so called Fisher information approximation, see e.g. [17], [20], [33]. However, for small data-sets and for model classes that do not satisfy the necessary conditions, the asymptotic form is not accurate [35].

We will now define the NML model, and discuss its basic properties.

B. Normalized Maximum Likelihood (NML)

Let again $y^n = (y_1, \ldots, y_n) \in \mathbb{R}^n$, $n \in \mathbb{N}$ be a sequence. We consider a model class $\mathcal{M} = \{f(\cdot\,;\,\theta)\ :\ \theta \in \Theta\}$, i.e., a set of density functions over sequences in $\mathbb{R}^n$. We denote the maximum likelihood parameters by $\hat{\theta}(y^n)$. The ML parameters do not have to be unique — in fact the model does not even have to be parametric — since we will only use the maximized likelihood $f(y^n\,;\,\hat{\theta}(y^n))$.

The *normalized maximum likelihood* (NML) universal model is given by

$$f_{nml}(y^n) = \frac{f(y^n\,;\,\hat{\theta}(y^n))}{C_n}\ ,\ \ C_n = \int_{\mathcal{A}} f(z^n\,;\,\hat{\theta}(z^n))\,dz^n\ ,$$

where the range of integration $\mathcal{A}$ can be either the set of all possible sequences of length n, or only a subset, and C_n is a normalizing constant ensuring that the result is indeed a probability density function (over the set $\mathcal{A}$). In the discrete-data case, the integral is replaced by the corresponding sum.

As shown by Shtarkov [32], the NML model is the unique minimax optimal universal model in the sense that it minimizes the worst-case *regret*

$$\max_{y^n \in \mathcal{A}} \ln \frac{f(y^n\,;\,\hat{\theta}(y^n))}{f_{nml}(y^n)} = \min_{g} \max_{y^n \in \mathcal{A}} \ln \frac{f(y^n\,;\,\hat{\theta}(y^n))}{g(y^n)},$$

where g can be any density function. The above log-likelihood ratio, or the regret, can be interpreted as the excess number of bits used to encode y^n using model g relative to the minimum achieved by the ML parameters.

For some model classes, the normalizing coefficient is finite only if the range of the data is restricted, see e.g. [17], [24], [30]. The logarithm of the normalizing coefficient, $\ln C_n$, is called the *parametric complexity*. It is equal to both the minimax and maximin regret under log-loss, see e.g. [24], [36], which makes the quantity interesting in its own right.

[1]For continuous data, a constant depending on quantization level is usually omitted.

On the other hand, we have the usual Fisher information approximation [17]

$$\ln C_n = \frac{d}{2}\ln\frac{n}{2\pi} + \ln\int_\Theta \sqrt{\det I(\theta)}\, d\theta + o(1) \ , \qquad (3)$$

where d is the dimension of the parameter space. It is also non-trivial to apply due to the integral involving the Fisher information $I(\theta)$. Using only the leading term (without 2π), i.e., the BIC criterion [37], gives a rough approximation. Even if the Fisher information formula can be used, there are practical circumstances where it gives a very poor approximation [35]. As expected, rough approximations tend to perform worse in model selection tasks than more refined approximations, or ideally, the exact solution, see e.g. [18, Chap. 9].

III. Denoising and Clustering

A. Extended Model

We modify the basic model (1) in such a way that there is no need for renormalization. This is achieved by inclusion of the coefficient vector β in the model as a variable and by selection of a (prior) density for β. While the resulting NML model will be equivalent to Rissanen's renormalized solution, the new formulation is easier to interpret and directly suggests several refinements and extensions.

Consider a fixed subset $\gamma \subseteq \{1,\ldots,n\}$ of the coefficient indices. In the resulting model class, $\mathcal{M}_\gamma$, the coefficients β_i with $i \in \gamma$ are modelled as independent outcomes from a zero-mean Gaussian distribution with variance τ^2. In the basic "hard threshold" version, all β_i with $i \notin \gamma$ are forced to be equal to zero. Thus the extended model is given by

$$y^n = \mathcal{W}\beta^n + \epsilon^n, \quad \begin{cases} \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0,\sigma_N^2), & \\ \beta_i \overset{i.i.d.}{\sim} \mathcal{N}(0,\tau^2), & \text{if } i \in \gamma, \\ \beta_i = 0, & \text{otherwise.} \end{cases} \qquad (4)$$

This way of modeling the coefficients is akin to the so called *spike and slab* model often used in Bayesian variable selection [38], [39] and applications to wavelet-based denoising [40], [41] (and references therein). In relation to the sparseness property mentioned in the introduction, the 'spike' consists of coefficients with $i \notin \gamma$ that are equal to zero, while the 'slab' consists of coefficients with $i \in \gamma$ described by a Gaussian density with mean zero. This is a simple form of a scale-mixture of Gaussians with two components. In Sec. IV-B we will consider a model with more than two components.

Let $c^n = \beta^n + \mathcal{W}^T\epsilon^n$, where $\mathcal{W}^T\epsilon^n$ gives the representation of the noise in the wavelet domain. The vector c^n is the wavelet representation of the signal y^n, and we have

$$y^n = \mathcal{W}\beta^n + \mathcal{W}\mathcal{W}^T\epsilon^n = \mathcal{W}c^n.$$

It is easy to see that the maximum likelihood parameters are obtained directly from

$$\hat\beta_i = \begin{cases} c_i, & \text{if } i \in \gamma, \\ 0, & \text{otherwise.} \end{cases} \qquad (5)$$

The i.i.d. Gaussian distribution for ϵ^n in (4) implies that the distribution of $\mathcal{W}^T\epsilon^n$ is also i.i.d. and Gaussian with the same variance, σ_N^2. As a sum of two independent random variates, each c_i has a distribution given by the convolution of the densities of the summands, β_i and the ith component of $\mathcal{W}^T\epsilon^n$. In the case $i \notin \gamma$ this is simply $\mathcal{N}(0,\sigma_N^2)$. In the case $i \in \gamma$ the density of the sum is also Gaussian, with variance given by the sum of the variances, $\tau^2+\sigma_N^2$. All told, we have the following simplified representation of the extended model where the parameters β^n are implicit:

$$y^n = \mathcal{W}c^n, \quad c_i \overset{i.i.d.}{\sim} \begin{cases} \mathcal{N}(0,\sigma_I^2), & \text{if } i \in \gamma, \\ \mathcal{N}(0,\sigma_N^2), & \text{otherwise,} \end{cases} \qquad (6)$$

where $\sigma_I^2 := \tau^2 + \sigma_N^2$ denotes the variance of the informative coefficients, and we have the important restriction $\sigma_I^2 \geq \sigma_N^2$ which we will discuss more below.

Due to orthogonality of the transform $\mathcal{W}$, the density of signal y^n under the extended model (6) is equal to the density of the wavelet representation $c^n = \mathcal{W}^T y^n$ under a Gaussian mixture:

$$f(y^n\,;\,\sigma_I^2,\sigma_N^2) = \prod_{i\in\gamma}\phi(c_i\,;\,0,\sigma_I^2)\prod_{i\notin\gamma}\phi(c_i\,;\,0,\sigma_N^2) \ , \qquad (7)$$

where $\phi(\cdot\,;\,\mu,\sigma^2)$ is the Gaussian density function.

B. Denoising Criterion

The task of choosing a subset γ can now be seen as a clustering problem: each wavelet coefficient belongs either to the set of the informative coefficients with variance σ_I^2, or the set of non-informative coefficients with variance σ_N^2. The MDL principle gives a natural clustering criterion by minimization of the code-length achieved for the observed signal (see [42]). Once the optimal subset is identified, the denoised signal is obtained by setting the wavelet coefficients to their maximum likelihood values (5); i.e., retaining the coefficients in γ and discarding the rest, and doing the inverse transformation. It is well known that this amounts to an orthogonal projection of the signal to the subspace spanned by the wavelet basis vectors in γ.

The code length under the model (6) depends on the values of the two parameters, σ_I^2 and σ_N^2. The NML density under the extended model (6) for a given coefficient subset γ is defined as

$$f_{nml}(y^n\,;\,\gamma) := \frac{f(y^n\,;\,\hat\sigma_I^2,\hat\sigma_N^2)}{C_\gamma},$$

where the numerator is given by (7), and

$$\hat\sigma_I^2 = \frac{1}{k(\gamma)}\sum_{i\in\gamma} c_i^2 \ , \qquad \hat\sigma_N^2 = \frac{1}{n-k(\gamma)}\sum_{i\notin\gamma} c_i^2, \qquad (8)$$

where $k = k(\gamma)$ is the number of coefficients in subset γ, are the maximum likelihood parameters for data y^n (for details, see the appendix). The important normalizing coefficient C_γ depends on both the model class $\mathcal{M}_\gamma$ and the sample size n.

Restricting the data such that the maximum likelihood parameters satisfy

$$\sigma_{\min}^2 \leq \hat\sigma_N^2, \hat\sigma_I^2 \leq \sigma_{\max}^2,$$

and ignoring the constraint $\sigma_N^2 \leq \sigma_I^2$, the code length under the extended model (6) is approximated by

$$\frac{n-k}{2}\ln\frac{S(y^n)-S_\gamma(y^n)}{n-k}+\frac{k}{2}\ln\frac{S_\gamma(y^n)}{k}+\frac{1}{2}\ln k(n-k), \tag{9}$$

plus a constant independent of γ; here $S(y^n)$ and $S_\gamma(y^n)$ denote the sum of the squares of all the wavelet coefficients and the coefficients for which $i \in \gamma$, respectively (see the appendix for a proof). The code length formula is very accurate even for small n since it involves only the Stirling approximation of the Gamma function.

Remark 1: The set of sequences satisfying the restriction $\sigma_{\min}^2 \leq \hat{\sigma}_N^2, \hat{\sigma}_I^2 \leq \sigma_{\max}^2$ depends on γ. For instance, consider the case $n = 2$. In a model with $k = 1$, the restriction corresponds to a union of four squares: $c_1, c_2 \in [-\sigma_{\max}, -\sigma_{\min}] \cup [\sigma_{\min}, \sigma_{\max}]$. On the other hand, in a model with either $k = 0$ or $k = 2$, the relevant area is an annulus (two-dimensional spherical shell): $c_1^2 + c_2^2 = 2\hat{\sigma}^2 \in [2\sigma_{\min}^2, 2\sigma_{\max}^2]$. However, the restriction can be understood as a definition of the support of the corresponding NML model, not a rigid restriction on the data, and hence models with varying γ are still comparable as long as the maximum likelihood parameters for the observed sequence satisfy the restriction. In practice, the restriction is of no consequence, since we can choose as wide a range as to guarantee that the data falls within the support, and the range doesn't appear in the final criterion.

The code length obtained is identical to that derived by Rissanen with renormalization [24] (note the correction to the third term of (9) in [43]). The formula has a concise and suggestive form that originally lead to the interpretation in terms of two Gaussian densities [31]. It is also the form that has been used in subsequent experimental work with somewhat mixed conclusions [31], [44]: While for Gaussian low variance noise it gives better results than a universal threshold of Donoho and Johnstone [8] (VisuShrink), over-fitting occurs in noisy cases [31] (see also Sec. V below).

Remark 2: It was proved in [24] that the criterion (9) is minimized by a subset γ which consists of some number k of the largest or smallest wavelet coefficients in absolute value. It was also felt that in denoising applications the data are such that the largest coefficients will minimize the criterion. The above alternative formulation gives a natural solution to this question: by the inequality $\sigma_I^2 \geq \sigma_N^2$, which implies the inequality $\hat{\sigma}_I^2 \geq \hat{\sigma}_N^2$ in expectation (although not with certainty), the set of coefficients with larger variance, i.e., the one with larger absolute values, should be retained, rather than *vice versa*.

Remark 3: The NML model corresponding to the extended model (6) is identical to Rissanen's renormalized model only if the inequality $\sigma_I^2 \geq \sigma_N^2$ is ignored in the calculations (see the appendix). However, the following proposition (proved in the appendix) shows that the effect of doing so is independent of k, and hence irrelevant.

Proposition 1: The effect of ignoring the constraint $\sigma_N^2 \leq \sigma_I^2$ is exactly one bit.

We can safely ignore the constraint and use the model without the constraint as a starting point for further developments for the sake of mathematical convenience.

IV. Refined MDL Denoising

A. Encoding the Model Class

It is customary to ignore encoding of the index of the model class in MDL model selection; i.e., encoding the number of parameters when the class is in one-to-one correspondence with the number of parameters. One simply picks the class that enables the shortest description of the data without considering the number of bits needed to encode the class itself. Note that here we do not refer to encoding the parameter values as in two-part codes, which are done implicitly in the so-called 'one-part codes' such as the NML and mixture codes. In most cases there are not too many classes and hence omitting the code length of the model index has no practical consequence. When the number of model classes is large, however, this issue does become of importance. In the case of denoising, the number of different model classes is as large as 2^n (with n as large as $512 \times 512 = 262,144$) and, as we show, encoding of the class index is crucial.

The encoding method we adopt for the class index is simple. We first encode k, the number of retained coefficients with a uniform code, which is possible since the maximal number n is fixed. This part of the code can be ignored since it only adds a constant to all code lengths. Secondly, for each k there are a number of different model classes depending on which k coefficients are retained. Note that while the retained coefficients are always the *largest* k coefficients, this information is not available to the decoder at this point and the index set to be retained has to be encoded. There are $\binom{n}{k}$ sets of size k, and we use a uniform code yielding a code length $\ln\binom{n}{k}$ nats, corresponding to a prior probability

$$\pi(\gamma) = \binom{n}{k}^{-1} = \frac{k!(n-k)!}{n!}. \tag{10}$$

Applying Stirling's approximation to the factorials and ignoring all constants w.r.t. γ gives the final code length formula

$$\frac{n-k}{2}\ln\frac{S(y^n)-S_\gamma(y^n)}{(n-k)^3}+\frac{k}{2}\ln\frac{S_\gamma(y^n)}{k^3}. \tag{11}$$

The proof can be found in the appendix.

This way of encoding the class index is by no means the only possibility but it will be seen to work sufficiently well, except for one curious limitation: As a consequence of modeling both the informative coefficients and the noise by densities from the same Gaussian model, the code length formula approaches the same value as k approaches either zero or n, which actually are disallowed. Hence, it may be that in cases where there is little information to recover, the random fluctuations in the data may yield a minimizing solution near $k = n$ instead of a correct solution near $k = 0$. A similar phenomenon has been demonstrated for "saturated" Bernoulli models with one parameter for each observation [28], and resembles the inconsistency problem of MDL in Markov chain order selection [45]: In all these cases pure random noise is incorrectly identified as maximally regular data. In order to

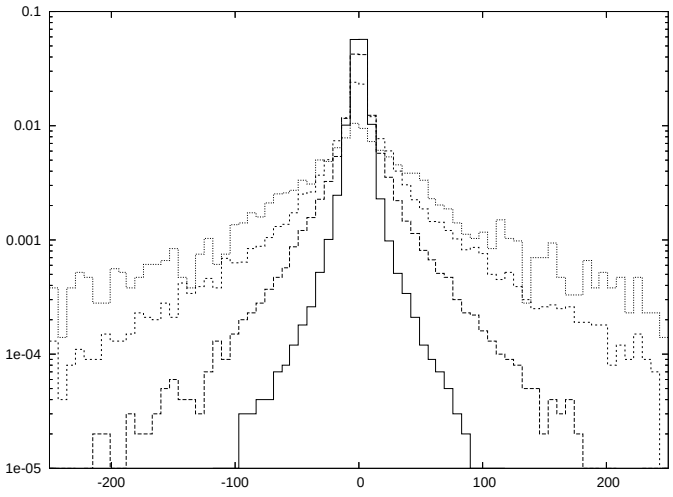

Fig. 1. Log-scale representation of the empirical histograms of the wavelet coefficients on dyadic levels 6–9 for the Boat image (see Sec. V below). Finer levels have narrower (smaller variance) distributions than coarser levels; the finest level (9) is drawn with solid line.

prevent this we simply restrict $k \leq .95n$, which seems to avoid such problems. A general explanation and solution for these phenomena would be of interest[2].

B. Subband Adaptation

It is an empirical fact that for most natural signals the coefficients on different subbands corresponding to different frequencies (and orientations in 2D data) have different characteristics. Basically, the finer the level, the smaller the variance of the coefficients, see Fig. 1. (This is not the case for pure Gaussian noise or, more interestingly, signals with fractal structure [2].) Within the levels, the histograms of the subbands for different orientations of 2D transforms typically differ somewhat, but the differences between orientations are not as significant as between levels.

In order to take the subband structure of wavelet transforms into account, we let each subband $b \in \{1, \ldots, B\}$ have its own variance, τ_b. We choose the set of the retained coefficients separately on each subband, and let γ_b denote the set of the retained coefficients on subband b, with $k_b := |\gamma_b|$. For convenience, let γ_0 be the set of all the coefficients that are not retained. Note that this way we have $k_0 + \ldots + k_b = n$. In order to encode the retained and the discarded coefficients on each subband, we use a similar code as in the 'flat' case (Sec. IV-A). For each subband $1, \ldots, B$, the number of nats needed is $\ln \binom{n_b}{k_b}$.

Ignoring again the constraint $\sigma_I^2 \geq \sigma_N^2$ for the sake of mathematical convenience, the levels can be treated as separate sets of coefficients with their own Gaussian densities just as in the previous subsection, where we had only two such sets.

[2]Perhaps a solution could be found in algorithmic information theory (Kolmogorov complexity) and the concept of Kolmogorov *minimal* sufficient statistic [46] which is the simplest one of many equally efficient descriptions. However, for practical purposes, a modification of the concept is needed in order to account for the fluctuations near the extremes; these are easily succumbed by $\mathcal{O}(1)$ terms usually ignored in algorithmic information theory.

ALGORITHM 1.
Input: signal y^n

0. set $c^n \leftarrow \mathcal{W}^T y^n$
1. initialize $k_b = n_b$ for all $b \in \{1, \ldots, B\}$
2. do until convergence
3. for each $b \in \{B_0 + 1, \ldots, B\}$
4. optimize k_b wrt. criterion (12)
5. end
6. end
7. for each $i \in \{1, \ldots, n\}$
8. if $i \notin \gamma$ then set $c_i \leftarrow 0$
9. end
10. output $\mathcal{W}c^n$

Fig. 2. Outline of an algorithm for subband-adaptive MDL denoising. The coarsest B_0 subbands are not processed in the loop of Steps 3–5. In Step 8, the final model γ is defined by the largest k_b coefficients on each subband b. A soft thresholding variation to Step 8 is described in Sec. IV-C.

The code length function, including the code length for γ, becomes after Stirling's approximation to the Gamma function and ignoring constants as follows:

$$\sum_{b=0}^{B} \left(\frac{k_b}{2} \ln \frac{S_{\gamma_b}(y^n)}{k_b} + \frac{1}{2} \ln k_b \right) + \sum_{b=1}^{B} \ln \binom{n_b}{k_b}. \qquad (12)$$

The proof is omitted since it is entirely analogous to the proof of Eq. (9) (see the appendix), the only difference being that now we have $B + 1$ Gaussian densities instead of only two. Notwithstanding the added code-length for the retained indices, for the case $B = 1$ this coincides with the original setting, where the subband structure is ignored, Eq. (9), since we then have $k_0 = n - k_1$. This code can be extended to allow $k_b = 0$ for some subbands simply by ignoring such subbands, which formally corresponds to reducing B in such cases[3].

Finding the index sets γ_b that minimize the NML code length simultaneously for all subbands b is computationally demanding. While on each subband the best choice always includes some k_b largest coefficients, the optimal choice of k_b on subband b depends on the choices made on the $B-1$ other subbands. A reasonable approximate solution to the search problem is obtained by iteration through the subbands and, on each iteration, finding the locally optimal coefficient set on each subband, given the current solution on the other subbands. Since the total code length achieved by the current solution never increases, the algorithm eventually converges, typically after not more than five iterations. Algorithm 1 in Fig. 2 implements the above described method. Following established practice [9], [12], all coefficients are retained on the smallest (coarsest) subbands[4].

[3]In fact, when reducing B the constants ignored also get reduced. This effect is very small compared to terms in (12), and can be safely ignored since codes with positive constants added to the code lengths are always decodable.

[4]We retain all subbands below level 4, i.e., all subbands with 16 or less coefficients. This has little effect to the present method, but since it is important for other methods to which we compare, especially SureShrink, we adopted the practice in order to facilitate comparison.

C. Soft Thresholding by Mixtures

The methods described above can be used to determine the MDL model, defined by a subset γ of the wavelet coefficients, that gives the shortest description to the observed data. However, in many cases there are several models that achieve nearly as good a compression as the best one. Intuitively, it seems then too strict to choose the single best model and discard all the others. A modification of the procedure is to consider a *mixture*, where all models indexed by γ are weighted by Eq. (10):

$$f_{mix}(y^n) := \sum_\gamma f_{nml}(y^n \,;\, \gamma)\, \pi(\gamma). \tag{13}$$

Such a mixture model is universal (see e.g. [19], [20]) in the sense that with increasing sample size the per sample average of the code length $-n^{-1} \ln f_{mix}(y^n)$ approaches that of the best γ for all y^n. Consequently, predictions obtained by conditioning on past observations converge to the optimal ones achievable with the chosen model class. A similar approach with mixtures of trees has been applied in the context of compression [47].

For denoising purposes we need a slightly different setting since we cannot let n grow. Instead, given an observed signal y^n, consider another image z^n from the same source (with the same γ and β^n but different ϵ^n). We denote the joint likelihood of signals y^n and z^n under the mixture density (13) by $f_{mix}(y^n, z^n)$. Denoising is now equivalent to estimating the expected value of z^n, which is given by $\mathcal{W}\beta^n$. Obtaining predictions for z^n given y^n from the mixture is in principle easy: one only needs to evaluate a conditional mixture

$$\begin{aligned} f_{mix}(z^n \mid y^n) &= \frac{f_{mix}(y^n, z^n)}{f_{mix}(y^n)} \\ &= \sum_\gamma f_{nml}(z^n \mid y^n \,;\, \gamma)\, \pi(\gamma \mid y^n). \end{aligned}$$

with new updated 'posterior' weights for the models, obtained by multiplying the NML density by the prior weights and normalizing wrt. γ:

$$\pi(\gamma \mid y^n) := \frac{f_{nml}(y^n \,;\, \gamma)\pi(\gamma)}{\sum_{\gamma'} f_{nml}(y^n \,;\, \gamma')\pi(\gamma')}. \tag{14}$$

Since in the denoising problem we only need the mean value instead of a full predictive distribution for the coefficients, we can obtain the predicted mean as a weighted average of the predicted means corresponding to each γ by replacing the density $f_{nml}(z^n \mid y^n \,;\, \gamma)$ by the coefficient value $c_i = c_i(y^n)$ obtained from y^n for $i \in \gamma$ and zero otherwise, which gives the denoised coefficients

$$\sum_\gamma c_i\, \mathbb{I}_{i\in\gamma}\, \pi(\gamma \mid y^n) = c_i \sum_{\gamma \ni i} \pi(\gamma \mid y^n), \tag{15}$$

where the indicator function $\mathbb{I}_{i\in\gamma}$ takes value one if $i \in \gamma$ and zero otherwise. Thus the mixture prediction of the coefficient value is simply c_i times the sum of the weights of the models where $i \in \gamma$ with the weights given by Eq. (14).

The practical problem that arises in such a mixture model is that summing over all the 2^n models is intractable. Since this sum appears as the denominator of (14), we cannot evaluate the required weights. We now derive a tractable approximation. To this end, given a fixed model (index set) γ, let $\gamma_1 \ldots 1_i \ldots \gamma_n$ denote a model which is obtained from γ by forcing the ith coefficient into the model, i.e., setting $i \in \gamma$. Similarly, let $\gamma_1 \ldots 0_i \ldots \gamma_n$ denote a model which is obtained from γ by setting $i \notin \gamma$. The weight with which each individual coefficient contributes to the mixture prediction can be obtained from

$$\begin{aligned} r_i := \frac{\sum_{\gamma\ni i} \pi(\gamma \mid y^n)}{\sum_{\gamma\not\ni i} \pi(\gamma \mid y^n)} &= \frac{\sum_{\gamma\ni i} \pi(\gamma \mid y^n)}{1 - \sum_{\gamma\ni i} \pi(\gamma \mid y^n)} \\ \iff \sum_{\gamma\ni i} \pi(\gamma \mid y^n) &= \frac{r_i}{1+r_i}, \end{aligned} \tag{16}$$

where the sums are over all models γ that include or exclude the ith coefficient. Note that ratio r_i is equal to

$$r_i = \frac{\sum_\gamma \pi(\gamma_1 \ldots 1_i \ldots \gamma_n \mid y^n)}{\sum_{\gamma'} \pi(\gamma'_1 \ldots 0_i \ldots \gamma'_n \mid y^n)}.$$

This can be approximated by

$$\frac{\sum_\gamma \pi(\gamma_1 \ldots 1_i \ldots \gamma_n \mid y^n)}{\sum_{\gamma'} \pi(\gamma'_1 \ldots 0_i \ldots \gamma'_n \mid y^n)} \approx \frac{\pi(\hat\gamma_1 \ldots 1_i \ldots \hat\gamma_n \mid y^n)}{\pi(\hat\gamma_1 \ldots 0_i \ldots \hat\gamma_n \mid y^n)} := \tilde r_i,$$

where $\hat\gamma = \hat\gamma_1 \ldots \hat\gamma_n$ is the model with maximal NML posterior weight (14). The approximation amounts to replacing the exponential sums in the numerator and the denominator by their largest terms, assuming that forcing γ_i to be one or zero has no effect on the other components of $\hat\gamma$. The ratio of two weights can be evaluated without knowing their common denominator, and hence this gives an efficient recipe for approximating the weights needed in Eq. (15).

Intuitively, if fixing $\gamma_i = 0$ decreases the posterior weight significantly compared to $\gamma_i = 1$, the approximated value of r_i becomes large and the i'th coefficient is retained near its maximum likelihood value c_i. Conversely, coefficients that increase the code length when included in the model are shrunk towards zero. Thus, the mixing procedure implements a general form of 'soft' thresholding, of which a restricted piecewise linear form has been found in many cases superior to hard thresholding in earlier work [8], [12]. Such soft thresholding rules have been justified in earlier works by their improved theoretical and empirical properties, while here they arise naturally from a universal mixture code. The whole procedure for mixing different coefficient subsets can be implemented by replacing Step 8 of Algorithm 1 in Fig. 2 by the instruction

$$\text{set } c_i \leftarrow c_i\, \frac{\tilde r_i}{1+\tilde r_i}$$

where $\tilde r_i$ denotes the approximated value of r_i. The behavior of the resulting soft threshold is illustrated in Fig. 3.

V. Experimental Results

A. Data and Setting

The effect of the three refinements of the MDL denoising method was assessed separately and together on a set of artificial 1D signals [9] and natural images[5] commonly used for

[5]The images were the same as used in many earlier papers, available at `http://decsai.ugr.es/~javier/denoise/`.

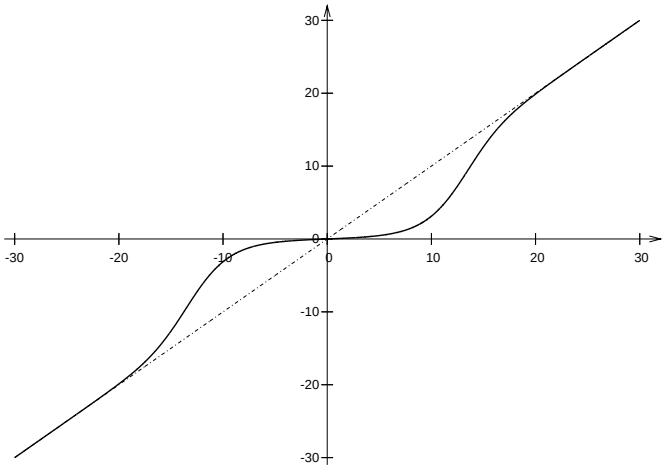

Fig. 3. The behavior of the soft thresholding method implemented by Algorithm 2 for one of the subbands of the Boat image with no added noise (see Sec. V): the original wavelet coefficient value c_i on the x-axis, and the thresholded value $c_i\,\tilde{r}_i/(1+\tilde{r}_i)$ on the y-axis. For coefficients with large absolute value, the curve approaches the diagonal (dotted line). The general shape of the curve is always the same but the scale depends on the data: the more noise, the wider the non-linear part.

benchmarking. The signals were contaminated with Gaussian pseudo-random noise of known variance σ^2, and the denoised signal was compared with the original signal. The Daubechies D6 wavelet basis was used in all experiments, both in the 1D and 2D cases. The error was measured by the peak-signal-to-noise ratio (PSNR), defined as

$$\mathit{PSNR} := 10 \cdot \log_{10}\left(\frac{\mathit{Range}^2}{\mathit{MSE}}\right),$$

where *Range* is the difference between the maximum and minimum values of the signal (for images $\mathit{Range} = 255$); and *MSE* is the mean squared error. The experiment was repeated 15 times for each value of σ^2, and the mean value and standard deviation was recorded.

The compared denoising methods were the original MDL method [24] without modifications; MDL with the modification of Sec. IV-A; MDL with the modifications of Secs. IV-A and IV-B; and MDL with the modifications of Secs. IV-A, IV-B and IV-C. For comparison, we also give results for three general denoising methods applicable to both 1D and 2D signals, namely VisuShrink [8], SureShrink [9], and BayesShrink [12][6].

B. Results

Figure 4 illustrates the denoising results for the *Blocks* signal [9] with signal length $n = 2048$. The original signal, shown in the top-left display, is piece-wise constant. The standard deviation of the noise is $\sigma = 0.5$. The best method, having the highest *PSNR* (and equivalently, the smallest *MSE*) is the MDL method with all the modifications proposed in the present work, labeled MDL (A-B-C) in the figure. Another case, the *Peppers* image with noise standard deviation $\sigma = 30$, is shown in Fig. 5, where the best method is BayesShrink. Visually, SureShrink and BayesShrink give a similar result with some remainder noise left, while MDL (A-B-C) has removed almost all noise but suffers from some blurring.

The relative performance of the methods depends strongly on the noise level. Figure 6 illustrates this dependency in terms of the relative PSNR compared to the MDL (A-B-C) method[7]. It can be seen that the MDL (A-B-C) is uniformly the best among the four MDL methods except for a range of small noise levels in the *Peppers* case, where the original method [24] is slightly better. Moreover, it can be seen that the modifications of Secs. IV-B and IV-C improve the performance on all noise levels for both signals.

In [31], the poor performance of the original MDL method in the high-noise regime was attributed to splitting what is essentially a single Gaussian density into a mixture of two Gaussians. It can be seen that the problem is remedied already by the inclusion of the encoding of the model index (Sec. IV-A). The right panels of Fig. 6 show that the overall best method is BayesShrink, except for small noise levels in *Blocks*, where the MDL (A-B-C) method is the best. This is explained by the fact that the generalized Gaussian model used in BayesShrink is especially apt for natural images but less so for 1D signals of the kind used in the experiments.

The above observations generalize to other 1D signals and images as well, as shown by Tables I and II. For some 1D signals (*Heavisine*, *Doppler*) the SureShrink method is best for some noise levels. In images, BayesShrink is consistently superior for low noise cases, although it can be debated whether the test setting where the denoised image is compared to the original (natural) image, which in itself already contains some noise (as all natural images do), gives meaningful results in the low noise regime. For moderate to high noise levels, BayesShrink, MDL (A-B-C) and SureShrink typically give similar PSNR output.

VI. Conclusions

We have revisited an earlier MDL method for wavelet-based denoising for signals with additive Gaussian white noise. In doing so we gave an alternative interpretation of Rissanen's renormalization technique for avoiding the problem of unbounded parametric complexity in normalized maximum likelihood (NML) codes. This new interpretation suggested three refinements to the basic MDL method which were shown to significantly improve empirical performance.

The most significant contributions are: i) an approach involving what we called the *extended model*, to the problem of unbounded parametric complexity which may be useful not only in the Gaussian model but, for instance, in the Poisson and geometric families of distributions with suitable prior densities for the parameters; ii) a demonstration of the importance of encoding the model index when the number

[6]All the compared methods are available as a free package, downloadable at `http://www.cs.helsinki.fi/teemu.roos/denoise/`. The package includes the source code in C, using wavelet transforms from the Gnu Scientific Library (GSL). All the experiments of Sec. V can be reproduced using the package.

[7]Figure 6 and Tables I–II give the sample standard deviation (SD) for each entry over 15 repetitions; the standard error of the mean, $\mathrm{SE}_{\bar{x}}$ is obtained as $\mathrm{SE}_{\bar{x}} = \mathrm{SD}/\sqrt{15}$.

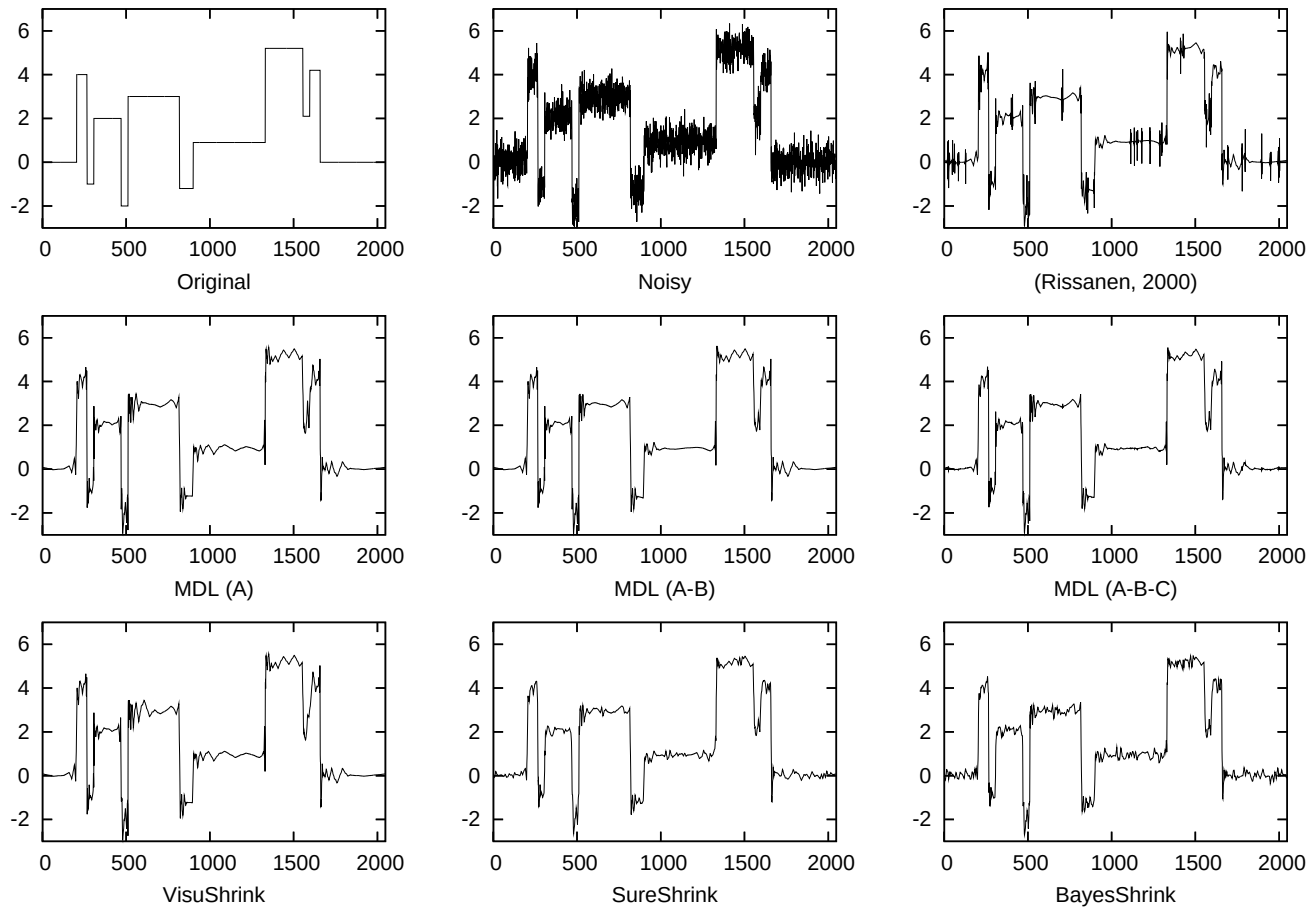

Fig. 4. Simulation Results. Panels from top to bottom, left to right: Blocks signal [9], sample size $n = 2048$; noisy signal, noise standard deviation $\sigma = 0.5$, PSNR=23.2; original MDL method [24], PSNR=28.5; MDL with modification of Sec. IV-A, PSNR=29.0; MDL with modifications of Secs. IV-A and IV-B, PSNR=29.6; MDL with modifications of Secs. IV-A, IV-B and IV-C, PSNR=30.1; VisuShrink [8], PSNR=28.6; SureShrink [9], PSNR=28.9; BayesShrink [12], PSNR=29.8. (Higher PSNR is better).

of potential models is large; iii) a combination of universal models of the mixture and NML types, and a related predictive technique which should also be useful in MDL denoising methods (e.g. [21], [22], [25]) that are based on finding a single best model, and other predictive tasks.

The extended model gives an interpretation of the MDL denoising method as one where the noiseless wavelet coefficients are modeled by a Gaussian density, with an additional point mass at the origin. In practice, it is likely that in certain domains, such as natural images, other density models will fit the actual data significantly better; the BayesShrink method, for instance, is based on a generalized Gaussian density model. The fact that the density model is explicit in the extended model also makes it possible to consider variations obtained by using more elaborate density models, which potentially improves the performance still more.

Appendix I
Postponed Proofs

Proof of Eq. (9)*:* The proof of Eq. (9) is technically similar to the derivation of the *renormalized* NML model in [24], which goes back to [48].

First note that due to orthonormality, the density of y^n under the extended model is always equal to the density of c^n evaluated at $\mathcal{W}^T y^n$. Thus, for instance, the maximum likelihood parameters for data y^n are easily obtained by maximizing the density of c^n at $\mathcal{W}^T y^n$. We repeat the density of c^n here from (7) for convenience:

$$f(c^n \,;\, \sigma_I^2, \sigma_N^2) = \prod_{i \in \gamma} \phi(c_i \,;\, 0, \sigma_I^2) \prod_{i \notin \gamma} \phi(c_i \,;\, 0, \sigma_N^2), \quad (17)$$

$\phi(\cdot \,;\, \mu, \sigma^2)$ denotes a Gaussian density function with mean μ and variance σ^2.

Let $S_\gamma(y^n)$ be the sum of squares of the wavelet coefficients with $i \in \gamma$:

$$S_\gamma(y^n) := \sum_{i \in \gamma} c_i^2.$$

and let $S(y^n)$ denote the sum of all wavelet coefficients. With slight abuse of notation, we also denote these two by $S_\gamma(c^n)$ and $S(c^n)$, respectively. Let k be the size of the set γ.

Using this notation, the maximum likelihood parameters (8) can be written as

$$\hat{\sigma}_I^2 = \frac{S_\gamma(y^n)}{k}, \quad \hat{\sigma}_N^2 = \frac{S(y^n) - S_\gamma(y^n)}{n - k}. \quad (18)$$

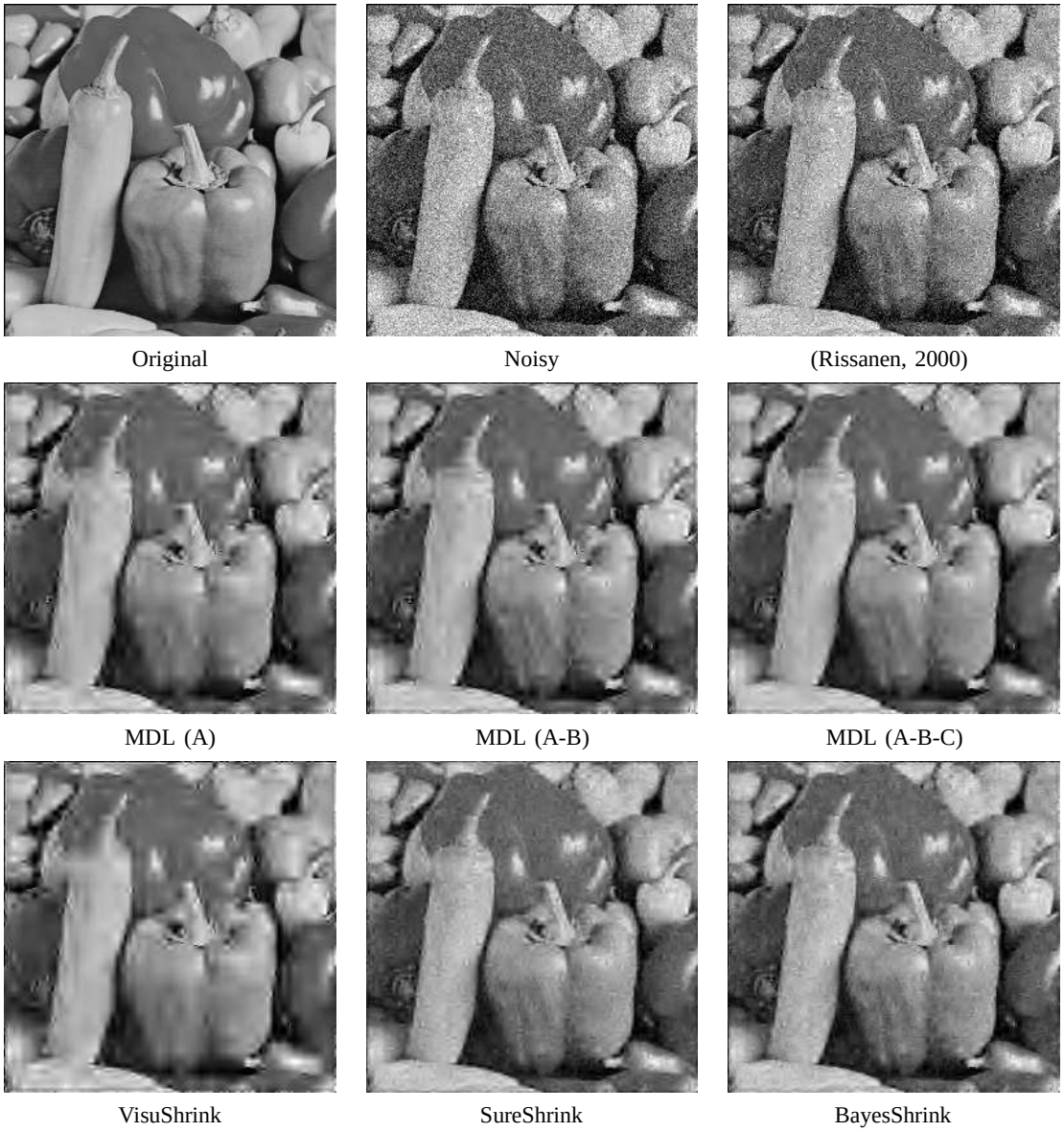

Fig. 5. Simulation Results. Panels from top to bottom, left to right: Peppers image, $n = 256 \times 256$; noisy image, noise standard deviation $\sigma = 30$, PSNR=18.6; original MDL method [24], PSNR=19.9; MDL with modification of Sec. IV-A, PSNR=23.9; MDL with modifications of Secs. IV-A and IV-B, PSNR=24.9; MDL with modifications of Secs. IV-A, IV-B and IV-C, PSNR=25.5; VisuShrink [8], PSNR=23.2; SureShrink [9], PSNR=24.6; BayesShrink [12], PSNR=25.9. (Higher PSNR is better).

With the maximum likelihood parameters (18) the likelihood (17) becomes

$$f(c^n\ ;\ \hat{\sigma}_I^2, \hat{\sigma}_N^2) = (2\pi\hat{\sigma}_I^2)^{-k/2} \exp\left(-\frac{S_\gamma(y^n)}{2\hat{\sigma}_I^2}\right) \times (2\pi\hat{\sigma}_N^2)^{-(n-k)/2} \exp\left(-\frac{S(y^n) - S_\gamma(y^n)}{2\hat{\sigma}_N^2}\right),$$

which further simplifies to

$$(2\pi e)^{-n/2} \left(\frac{S_\gamma(y^n)}{k}\right)^{-\frac{k}{2}} \left(\frac{S(y^n) - S_\gamma(y^n)}{n-k}\right)^{-\frac{n-k}{2}}. \quad (19)$$

The normalization constant C_γ is also easier to evaluate by integrating the likelihood in terms of c^n:

$$C_\gamma = A \int \left(S_\gamma(c^n)\right)^{-k/2} \left(S(c^n) - S_\gamma(c^n)\right)^{-\frac{n-k}{2}}\, dc^n, \quad (20)$$

where A is given by

$$A = (2\pi e)^{-n/2} k^{k/2} (n-k)^{\frac{n-k}{2}},$$

and the range of integration R is defined by requiring that the maximum likelihood estimators (18) are both within the interval $[\sigma^2_{\min}, \sigma^2_{\max}]$. It will be seen that the integral diverges without these bounds. The integral can be written in in two parts, the first one involving only the coefficients with $i \in \gamma$, and the second one involving only the coefficients with $i \notin \gamma$. Furthermore, the resulting two integrals depend on the coefficients only through the values $S_\gamma(c^n)$ and $S(c^n) - S_\gamma(c^n)$, and thus, they can be expressed in terms of these two quantities as the integration variables — we denote them respectively by s_1 and s_2. The associated Riemannian volume elements are infinitesimally thin spherical shells (surfaces of balls); the first one with dimension k and radius $s_1^{1/2}$, the

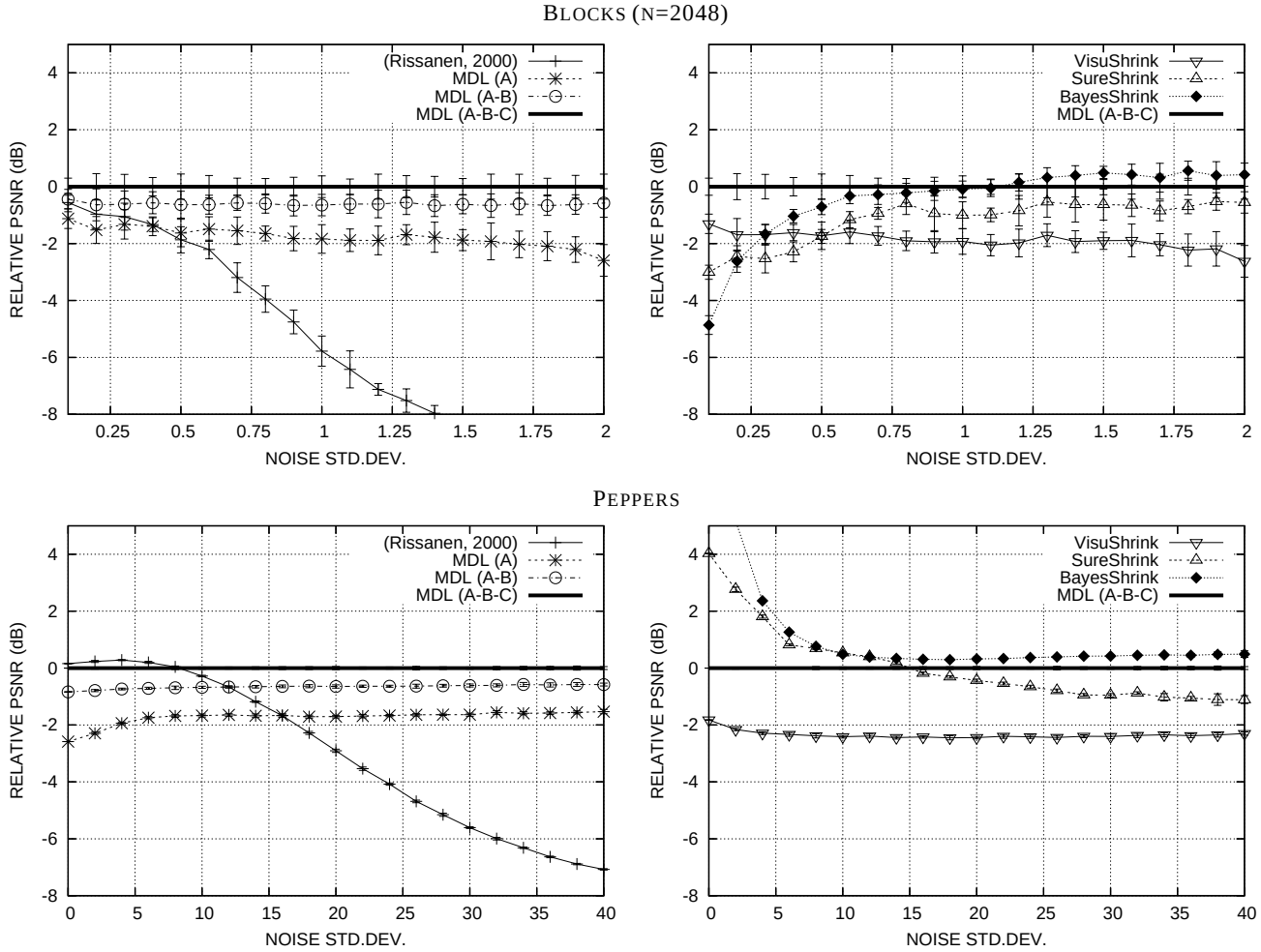

Fig. 6. Simulation Results. PSNR difference compared to the proposed method (MDL with modifications of Secs. IV-A, IV-B and IV-C), see Figs. 4 and 5. Errorbars show standard deviation of PSNR over 15 repetitions (too small to be visible in bottom row). Top row: Blocks signal [9], sample size $n = 2048$. Bottom row: Peppers image, $n = 256 \times 256$. Left panels show the effect of each of the three modifications in Sec. IV; right panels show comparison to VisuShrink [8], SureShrink [9], and BayesShrink [12].

second one with dimension $n-k$ and radius $s_2^{1/2}$, given by

$$\frac{\pi^{k/2} s_1^{k/2-1}}{\Gamma(k/2)}\, ds_1, \quad \frac{\pi^{(n-k)/2} s_2^{(n-k)/2-1}}{\Gamma((n-k)/2)}\, ds_2.$$

Thus the integral in (20) is equivalent to

$$\int_{k\sigma_{\min}^2}^{k\sigma_{\max}^2} \frac{\pi^{k/2} s_1^{k/2-1}}{\Gamma(k/2)} s_1^{-k/2}\, ds_1 \times \int_{(n-k)\sigma_{\min}^2}^{(n-k)\sigma_{\max}^2} \frac{\pi^{(n-k)/2} s_2^{(n-k)/2-1}}{\Gamma((n-k)/2)} s_2^{-(n-k)/2}\, ds_2.$$

Both integrands become simply of the form $1/x$ and hence, the value of the integral is given by

$$\frac{\pi^{n/2}}{\Gamma(k/2)\Gamma((n-k)/2)} \left(\ln \frac{\sigma_{\max}^2}{\sigma_{\min}^2}\right)^2, \tag{21}$$

Plugging (21) into (20) gives the value of the normalization constant

$$C_\gamma = \frac{k^{k/2}(n-k)^{(n-k)/2}}{(2e)^{n/2}\Gamma(k/2)\Gamma((n-k)/2)} \left(\ln \frac{\sigma_{\max}^2}{\sigma_{\min}^2}\right)^2.$$

Normalizing the numerator (19) by C_γ, and canceling like terms finally gives the NML density:

$$f_{nml}(y^n) = \frac{\Gamma(k/2)\Gamma((n-k)/2)}{\pi^{n/2}(S_\gamma(y^n))^{k/2}(S(y^n) - S_\gamma(y^n))^{(n-k)/2}} \times \left(\ln \frac{\sigma_{\max}^2}{\sigma_{\min}^2}\right)^{-2}, \tag{22}$$

and the corresponding code length becomes

$$-\ln f_{nml}(y^n) = \frac{k}{2}\ln S_\gamma(y^n) + \frac{n-k}{2}\ln(S(y^n) - S_\gamma(y^n)) - \ln\Gamma\left(\frac{k}{2}\right) - \ln\Gamma\left(\frac{n-k}{2}\right) + \frac{n}{2}\ln\pi + 2\ln\ln\frac{\sigma_{\max}^2}{\sigma_{\min}^2}.$$

Applying Stirling's approximation

$$\ln\Gamma(z) \approx \left(z - \frac{1}{2}\right)\ln z - z + \frac{1}{2}\ln 2\pi,$$

TABLE I

NUMERICAL RESULTS. THE PEAK-SIGNAL-TO-NOISE RATIO FOR VARIOUS 1D SIGNALS, DENOISING METHODS, AND NOISE LEVELS. COLUMNS: NOISE STANDARD DEVIATION σ; PSNR FOR DIFFERENT METHODS $\pm$ STANDARD DEVIATION OF EACH ENTRY IN 15 REPETITIONS (SEE ALSO FIG. 6), BEST VALUE(S) IN BOLDFACE.

	(Rissanen, 2000)	MDL (A)	MDL (A-B)	MDL (A-B-C)	VisuShrink	SureShrink	BayesShrink
Blocks ($n = 2048$)							
$\sigma = 0.1$	44.4 ±0.32	43.8 ±0.34	44.5 ±0.34	**44.9** ±0.30	43.6 ±0.34	41.9 ±0.24	40.1 ±0.33
0.5	28.9 ±0.46	29.1 ±0.48	30.1 ±0.47	**30.8** ±0.44	29.0 ±0.48	29.0 ±0.24	30.1 ±0.28
1.0	20.4 ±0.53	24.4 ±0.50	25.5 ±0.38	**26.2** ±0.38	24.3 ±0.45	25.2 ±0.47	26.1 ±0.28
1.5	15.0 ±0.21	21.6 ±0.37	22.8 ±0.33	23.4 ±0.29	21.5 ±0.30	22.8 ±0.55	**23.9** ±0.23
2.0	11.7 ±0.21	19.6 ±0.56	21.6 ±0.50	22.2 ±0.44	19.5 ±0.56	21.6 ±0.38	**22.6** ±0.41
Bumps ($n = 2048$)							
$\sigma = 0.1$	39.4 ±0.50	39.6 ±0.39	40.0 ±0.35	**40.7** ±0.36	39.2 ±0.41	38.8 ±0.26	38.3 ±0.31
0.5	20.6 ±0.17	26.8 ±0.48	27.8 ±0.55	**28.4** ±0.43	26.1 ±0.43	27.2 ±0.31	28.0 ±0.29
1.0	13.9 ±0.11	21.5 ±0.30	23.0 ±0.42	23.7 ±0.38	21.3 ±0.28	23.3 ±0.15	**24.0** ±0.29
1.5	10.3 ±0.14	18.6 ±0.50	20.6 ±0.39	21.3 ±0.46	18.9 ±0.42	20.5 ±0.40	**21.9** ±0.41
2.0	7.9 ±0.11	17.7 ±0.40	19.2 ±0.47	19.9 ±0.40	17.9 ±0.33	19.5 ±38	**20.3** ±0.46
Heavisine ($n = 2048$)							
$\sigma = 0.1$	51.3 ±0.77	50.4 ±0.67	51.3 ±0.50	**51.9** ±0.44	51.1 ±0.76	48.8 ±0.48	48.1 ±0.48
0.5	35.6 ±0.80	37.4 ±0.43	39.1 ±0.71	**39.5** ±0.55	37.7 ±0.71	38.3 ±0.48	38.9 ±0.52
1.0	27.0 ±1.16	32.9 ±0.65	34.1 ±0.62	34.6 ±0.57	33.2 ±0.67	**34.7** ±0.54	34.1 ±0.50
1.5	19.8 ±0.57	30.6 ±1.00	31.6 ±0.87	32.0 ±0.74	30.8 ±1.17	**32.3** ±0.77	**32.3** ±1.08
2.0	15.4 ±0.34	28.1 ±1.30	30.5 ±1.24	31.0 ±0.93	28.2 ±1.24	31.2 ±0.79	**31.3** ±0.93
Doppler ($n = 2048$)							
$\sigma = 0.1$	24.5 ±0.68	28.4 ±0.39	29.2 ±0.39	**29.8** ±0.39	28.3 ±0.41	28.6 ±0.40	29.5 ±0.50
0.5	6.2 ±0.17	17.8 ±0.69	19.3 ±0.86	19.9 ±0.84	17.7 ±0.67	19.6 ±0.83	**20.3** ±0.62
1.0	0.1 ±0.13	12.6 ±1.05	15.4 ±0.77	16.0 ±0.77	13.1 ±1.02	16.1 ±0.95	**16.2** ±0.77
1.5	-3.5 ±0.16	10.7 ±0.67	13.3 ±1.11	13.7 ±0.69	10.8 ±0.59	**14.0** ±0.58	13.9 ±1.30
2.0	-5.9 ±0.14	9.9 ±0.73	11.3 ±1.25	11.5 ±1.07	10.1 ±0.72	**12.2** ±0.76	11.8 ±1.11

to the Gamma functions yields now

$$\begin{aligned}-\ln f_{nml}(y^n) \approx \frac{k}{2}\ln S_\gamma(y^n) + \frac{n-k}{2}\ln(S(y^n) - S_\gamma(y^n)) \\ - \left(\frac{k-1}{2}\right)\ln\left(\frac{k}{2}\right) + \frac{k}{2} \\ - \left(\frac{n-k-1}{2}\right)\ln\left(\frac{n-k}{2}\right) + \frac{n-k}{2} \\ - \ln 2\pi + \frac{n}{2}\ln\pi + 2\ln\ln\frac{\sigma^2_{\max}}{\sigma^2_{\min}}.\end{aligned}$$

Rearranging the terms gives the formula

$$\begin{aligned}-\ln f_{nml}(y^n) \approx \frac{k}{2}\ln\frac{S_\gamma(y^n)}{k} + \frac{n-k}{2}\ln\frac{S(y^n) - S_\gamma(y^n)}{n-k} \\ + \frac{1}{2}\ln k(n-k) + const, \quad (23)\end{aligned}$$

where *const* is a constant wrt. γ, given by

$$const = \frac{n}{2}\ln 2\pi e - \ln 4\pi + 2\ln\ln\frac{\sigma^2_{\max}}{\sigma^2_{\min}}.$$

■

Proof of Proposition 1: The maximum likelihood parameters (18) may violate the restriction $\sigma_I^2 \geq \sigma_N^2$ that arises from the definition $\sigma_I^2 := \tau^2 + \sigma_N^2$. The restriction affects range of integration in Eq. (21) giving the non-constant terms as follows

$$\begin{aligned}\int_{k\sigma^2_{\min}}^{k\sigma^2_{\max}} \left(\int_{(n-k)\sigma^2_{\min}}^{((n-k)/k)s_1} s_1^{-1} s_2^{-1}\, ds_2 \right) ds_1 \\ = \int_{k\sigma^2_{\min}}^{k\sigma^2_{\max}} s_1^{-1}(\ln s_1 - \ln k\sigma^2_{\min})\, ds_1. \quad (24)\end{aligned}$$

Using the integral $\int s_1^{-1}\ln s_1\, ds_1 = \frac{1}{2}(\ln s_1)^2$ gives then

$$\frac{1}{2}(\ln k\sigma^2_{\max})^2 - \frac{1}{2}(\ln k\sigma^2_{\min})^2 - \ln k\sigma^2_{\min}\left(\ln\frac{\sigma^2_{\max}}{\sigma^2_{\min}}\right), \quad (25)$$

where the first two terms can be written as

$$\frac{1}{2}\left(\ln k\sigma^2_{\max} + \ln k\sigma^2_{\min}\right)\left(\ln\frac{\sigma^2_{\max}}{\sigma^2_{\min}}\right).$$

Combining with the third term of (25) changes the plus into a minus and gives finally

$$\frac{1}{2}\left(\ln\frac{\sigma^2_{\max}}{\sigma^2_{\min}}\right)\left(\ln\frac{\sigma^2_{\max}}{\sigma^2_{\min}}\right),$$

which is exactly half of the integral in Eq. (21), the constant terms being the same. Thus, the effect of the restriction on the code length where the *logarithm* of the integral is taken, is one bit, i.e., $\ln 2$ nats.

■

TABLE II

NUMERICAL RESULTS. THE PEAK-SIGNAL-TO-NOISE RATIO FOR VARIOUS IMAGES, DENOISING METHODS, AND NOISE LEVELS. COLUMNS: NOISE STANDARD DEVIATION σ; PSNR FOR DIFFERENT METHODS $\pm$ STANDARD DEVIATION OF EACH ENTRY IN 15 REPETITIONS (SEE ALSO FIG. 6), BEST VALUE(S) IN BOLDFACE.

	(Rissanen, 2000)	MDL (A)	MDL (A-B)	MDL (A-B-C)	VisuShrink	SureShrink	BayesShrink
Lena (512×512)							
$\sigma = 0$	39.1 $\pm$ –	36.6 $\pm$ –	38.5 $\pm$ –	39.3 $\pm$ –	37.3 $\pm$ –	43.2 $\pm$ –	**46.9** $\pm$ –
10	31.6 $\pm$0.02	30.8 $\pm$0.03	31.8 $\pm$0.02	32.4 $\pm$0.02	30.1 $\pm$0.02	32.8 $\pm$0.03	**33.1** $\pm$0.02
20	25.0 $\pm$0.04	27.8 $\pm$0.04	28.8 $\pm$0.03	29.4 $\pm$0.03	27.1 $\pm$0.02	29.5 $\pm$0.03	**29.9** $\pm$0.03
30	19.8 $\pm$0.03	26.0 $\pm$0.04	27.1 $\pm$0.03	27.6 $\pm$0.04	25.4 $\pm$0.03	27.8 $\pm$0.03	**28.2** $\pm$0.03
40	16.7 $\pm$0.01	24.9 $\pm$0.04	26.0 $\pm$0.03	26.5 $\pm$0.03	24.3 $\pm$0.03	26.4 $\pm$0.07	**27.0** $\pm$0.03
Boat (512×512)							
$\sigma = 0$	36.2 $\pm$ –	33.2 $\pm$ –	35.1 $\pm$ –	35.9 $\pm$ –	32.9 $\pm$ –	39.2 $\pm$ –	**40.3** $\pm$ –
10	30.2 $\pm$0.01	28.6 $\pm$0.02	29.8 $\pm$0.02	30.5 $\pm$0.02	28.0 $\pm$0.02	31.3 $\pm$0.02	**31.7** $\pm$0.01
20	24.2 $\pm$0.03	25.8 $\pm$0.03	26.8 $\pm$0.03	27.5 $\pm$0.03	25.2 $\pm$0.02	27.9 $\pm$0.03	**28.3** $\pm$0.02
30	19.6 $\pm$0.01	24.3 $\pm$0.03	25.2 $\pm$0.02	25.8 $\pm$0.02	23.7 $\pm$0.02	26.1 $\pm$0.02	**26.5** $\pm$0.02
40	16.6 $\pm$0.02	23.2 $\pm$0.03	24.2 $\pm$0.02	24.7 $\pm$0.03	22.8 $\pm$0.03	24.9 $\pm$0.06	**25.3** $\pm$0.02
House (256×256)							
$\sigma = 0$	41.4 $\pm$ –	36.7 $\pm$ –	42.5 $\pm$ –	43.5 $\pm$ –	41.0 $\pm$ –	47.4 $\pm$ –	**54.2** $\pm$ –
10	31.4 $\pm$0.05	30.7 $\pm$0.05	31.5 $\pm$0.04	32.1 $\pm$0.03	30.2 $\pm$0.07	32.5 $\pm$0.12	**32.8** $\pm$0.03
20	24.7 $\pm$0.06	27.3 $\pm$0.05	28.1 $\pm$0.05	28.7 $\pm$0.05	26.8 $\pm$0.06	28.7 $\pm$0.03	**29.2** $\pm$0.05
30	19.7 $\pm$0.04	25.4 $\pm$0.07	26.4 $\pm$0.05	27.0 $\pm$0.05	24.9 $\pm$0.07	26.9 $\pm$0.05	**27.4** $\pm$0.04
40	16.7 $\pm$0.04	24.2 $\pm$0.07	25.2 $\pm$0.07	25.7 $\pm$0.08	23.7 $\pm$0.08	25.4 $\pm$0.07	**26.2** $\pm$0.06
Peppers (256×256)							
$\sigma = 0$	38.9 $\pm$ –	36.1 $\pm$ –	37.9 $\pm$ –	38.7 $\pm$ –	36.9 $\pm$ –	42.7 $\pm$ –	**51.2** $\pm$ –
10	30.7 $\pm$0.03	29.3 $\pm$0.05	30.3 $\pm$0.04	31.0 $\pm$0.04	28.6 $\pm$0.05	**31.5** $\pm$0.03	**31.5** $\pm$0.04
20	24.7 $\pm$0.06	25.9 $\pm$0.05	26.9 $\pm$0.06	27.6 $\pm$0.05	25.1 $\pm$0.05	27.1 $\pm$0.05	**27.9** $\pm$0.06
30	19.9 $\pm$0.03	23.9 $\pm$0.06	24.9 $\pm$0.05	25.5 $\pm$0.04	23.1 $\pm$0.08	24.6 $\pm$0.06	**25.9** $\pm$0.05
40	16.8 $\pm$0.02	22.4 $\pm$0.08	23.3 $\pm$0.05	23.9 $\pm$0.05	21.6 $\pm$0.06	22.8 $\pm$0.14	**24.4** $\pm$0.12

Proof of Eq. (11): The relevant terms in the code length $\ln \binom{n}{k}$, i.e. those depending on k, for the index of the model class are

$$\begin{aligned} -\ln(k!(n-k)!) &= -\ln[k(k-1)!(n-k)(n-k)!] \\ &= -\ln(k(n-k)) - \ln\Gamma(k) - \ln\Gamma(n-k), \end{aligned}$$

which gives after Stirling's approximation (ignoring constant terms)

$$\begin{aligned} &-\ln(k(n-k)) - \left(k - \frac{1}{2}\right)\ln k + k \\ &\qquad - \left(n-k-\frac{1}{2}\right)\ln(n-k) + (n-k) \\ &= -\frac{k}{2}\ln k^2 - \frac{n-k}{2}\ln(n-k)^2 + \frac{1}{2}\ln k(n-k) + n. \end{aligned} \quad (26)$$

Adding this to Eq. (9) (without the constant n) gives Eq. (11). ∎

ACKNOWLEDGMENT

The authors thank Peter Grünwald, Steven de Rooij, Jukka Heikkonen, Vibhor Kumar, Hannes Wettig, and the anonymous referees for valuable comments.

REFERENCES

[1] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[2] ——, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, 1989.

[3] R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Trans. Information Theory*, vol. 38, no. 2, pp. 719–746, Mar. 1992.

[4] J. Villasenor, B. Belzer, and J. Liao, "Wavelet filter evaluation for image compression," *IEEE Trans. Image Processing*, vol. 4, no. 8, pp. 1053–1060, Aug. 1995.

[5] B. K. Natarajan, "Filtering random noise from deterministic signals via data compression," *IEEE Trans. Information Theory*, vol. 43, no. 11, pp. 2595–2605, Nov. 1995.

[6] M. Hansen and B. Yu, "Wavelet thresholding via MDL for natural images," *IEEE Trans. Information Theory (Special Issue on Information Theoretic Imaging)*, vol. 46, pp. 1778–1788, 2000.

[7] J. Liu and P. Moulin, "Complexity-regularized image denoising," *IEEE Trans. Image Processing*, vol. 10, no. 6, pp. 841–851, June 2001.

[8] D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[9] ——, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Statist. Assoc.*, vol. 90, no. 432, pp. 1200–1224, 1995.

[10] B. Vidakovic, "Nonlinear wavelet shrinkage with Bayes rules and Bayes factors," *J. Amer. Statist. Assoc.*, vol. 93, no. 441, pp. 173–179, 1998.

[11] F. Ruggeri and B. Vidakovic, "A Bayesian decision theoretic approach to the choice of thresholding parameter," *Statistica Sinica*, vol. 9, pp. 183–197, 1999.

[12] G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Proc.*, vol. 9, pp. 1532–1546, 2000.

[13] P. Moulin and J. Liu, "Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors," *IEEE Trans. Information Theory*, vol. 45, no. 3, pp. 909–919, Apr. 1999.

[14] M. Wainwright and E. Simoncelli, "Scale mixtures of Gaussians and the statistics of natural images," in *Advances in Neural Information Processing Systems*, S. Solla, T. Leen, and K.-R. Muller, Eds., vol. 12. MIT Press, May 2000, pp. 855–861.

[15] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Processing*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.

[16] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 445–471, 1978.

[17] ——, "Fisher information and stochastic complexity," *IEEE Trans. Information Theory*, vol. 42, no. 1, pp. 40–47, January 1996.

[18] ——, *Information and Complexity in Statistical Modeling*. Springer, 2007.

[19] P. Grünwald, "A Tutorial introduction to the minimum description length principle," in *Advances in MDL: Theory and Applications*, P. Grünwald, I. Myung, and M. Pitt, Eds. MIT Press, 2005.

[20] ——, *The Minimum Description Length Principle*. MIT Press, 2007.

[21] N. Saito, "Simultaneous noise suppression and signal compression using a library of orthonormal bases and the minimum description length criterion," in *Wavelets in Geophysics*. Academic Press, 1994, pp. 299–324.

[22] A. Antoniadis, I. Gijbels, and G. Gregoire, "Model selection using wavelet decomposition and applications," *Biometrika*, vol. 84, no. 4, pp. 751–763, 1997.

[23] H. Krim and I. Schick, "Minimax decription length for signal denoising and optimized representation," *IEEE Trans. Information Theory*, vol. 45, no. 3, pp. 898–908, 1999.

[24] J. Rissanen, "MDL denoising," *IEEE Trans. Information Theory*, vol. 46, no. 7, pp. 2537–2543, 2000.

[25] V. Kumar, J. Heikkonen, J. Rissanen, and K. Kaski, "Minimum description length denoising with histogram models," *IEEE Trans. Signal Processing*, vol. 54, no. 8, pp. 2922–2928, 2006.

[26] I. Daubechies, *Ten Lectures on Wavelets*. Society for Industrial & Applied Mathematics (SIAM), 1992.

[27] D. Foster and R. Stine, "The competitive complexity ratio," in *Proc. 2001 Conf. on Information Sciences and Systems*, 2001, pp. 1–6.

[28] ——, "The contribution of parameters to stochastic complexity," in *Advances in MDL: Theory and Applications*, P. Grünwald, I. Myung, and M. Pitt, Eds. MIT Press, 2005.

[29] F. Liang and A. Barron, "Exact minimax strategies for predictive density estimation, data compression, and model selection," *IEEE Trans. Information Theory*, vol. 50, no. 11, pp. 2708–2726, Nov. 2004.

[30] S. de Rooij and P. Grünwald, "An empirical study of MDL model selection with infinite parametric complexity," *J. Mathematical Psychology*, vol. 50, no. 2, pp. 149–166, 2006.

[31] T. Roos, P. Myllymäki, and H. Tirri, "On the behavior of MDL denoising," in *Proc. Tenth Int. Workshop on AI and Stat.*, R. G. Cowell and Z. Ghahramani, Eds. Society for AI and Statistics, 2005, pp. 309–316.

[32] Y. Shtarkov, "Universal sequential coding of single messages," *Problems of Information Transmission*, vol. 23, pp. 175–186, 1987.

[33] J. Rissanen, "Strong optimality of the normalized ML models as universal codes and information in data," *IEEE Trans. Information Theory*, vol. 47, no. 5, pp. 1712–1717, 2001.

[34] J. Myung, D. Navarro and M. Pitt, "Model selection by normalized maximum likelihood," *J. Mathematical Psychology*, vol. 50, pp. 167–179, 2006.

[35] D. Navarro, "A note on the applied use of MDL approximations," *Neural Computation*, vol. 16, no. 9, pp. 1763–1768, 2004.

[36] Q. Xie, and A. Barron, "Asymptotic minimax regret for data compression, gambling, and prediction," *IEEE Trans. Information Theory*, vol. 46, no. 2, pp. 431–445, 2000.

[37] G. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[38] T. J. Mitchell and J. J. Beauchamp, "Bayesian variable selection in linear regression (with discussion)," *J. Amer. Statist. Assoc.*, vol. 83, no. 404, pp. 1023–1032, Dec. 1988.

[39] E. I. George and R. E. McCulloch, "Approaches for Bayesian variable selection," *Statistica Sinica*, vol. 7, no. 2, pp. 339–374, 1997.

[40] H. A. Chipman, E. D. Kolaczyk, and R. E. McCulloch, "Adaptive Bayesian wavelet shrinkage," *J. Amer. Statist. Assoc.*, vol. 92, no. 440, pp. 1413–1421, 1997.

[41] F. Abramovich, T. Sapatinas, and B. Silverman, "Wavelet thresholding via a Bayesian approach," *J. Royal Statist. Soc. Series B*, vol. 60, no. 4, pp. 725–749, 1998.

[42] P. Kontkanen, P. Myllymäki, W. Buntine, J. Rissanen, and H. Tirri, "An MDL framework for data clustering," in *Advances in MDL: Theory and Applications*, P. Grünwald, I. Myung, and M. Pitt, Eds. MIT Press, 2005.

[43] J. Rissanen, "Lectures on statistical modeling theory," Aug. 2005, available at http://www.mdl-research.org/.

[44] J. Ojanen, T. Miettinen, J. Heikkonen, and J. Rissanen, "Robust denoising of electrophoresis and mass spectrometry signals with minimum description length principle," *FEBS Letters*, vol. 570, no. 1–3, pp. 107–113, 2004.

[45] I. Csiszár and P. Shields, "The consistency of the BIC Markov order estimator," *Annals of Statistics*, vol. 28, pp. 1601–1619, 2000.

[46] N. Vereshchagin and P. Vitányi, "Kolmogorov's structure functions and model selection," *IEEE Trans. Information Theory*, no. 12, pp. 3265–3290, Dec. 2004.

[47] F. Willems, Y. Shtarkov, and T. Tjalkens, "The context-tree weighting method: basic properties," *IEEE Trans. Information Theory*, vol. 41, no. 3, pp. 653–664, 1995.

[48] B. Dom, "MDL estimation for small sample sizes and its application to linear regression," IBM Research, Tech. Rep. RJ 10030, 1996.