Image and Video Restoration with Nonlocal Transform-Domain Filtering

Kostadin Dabov

The Date

Abstract

Vision is the most important source of information for humans and the acquisition of images has numerous applications in science, engineering, medicine and also in consumer devices such as cameras, mobile phones, etc. Digital image acquisition, in particular, has become the most common means of producing images; it has almost fully replaced image acquisition on film. A typical digital acquisition device contains lens and a solid-state image sensor. The scene radiance is focused by the lens to the sensor, which integrates the irradiance at its surface during the exposure time and produces a 2-D array of digital values (i.e. a raw image). The lens and the sensor introduce a variety of degradations such as geometrical distortions, blur, and noise. In addition to inherent technological factors that cause these degradations, there are external factors such as scene illumination and a motion within the scene or between the scene and the lens. In order to obtain images that can be visualized, raw images typically undergo some image processing such as denoising, color-filter array interpolation (demosaicking), enhancement (sharpening), color correction, and white balancing.

This thesis deals with the problems of denoising, deblurring, and enhancement of images and video. We propose a particular filtering algorithm, called *BM3D*, as the basic tool for approaching these problems. Among the multitude of existing filtering algorithms, *BM3D* is the first to exploit simultaneously:

- *sparse transform-domain representations* (i.e. a transform spectrum characterized by few high-magnitude coefficients)
- *nonlocal image modeling* (manifested by self-similarity between image subsets).

We show that the *BM3D* filter not only outperforms works that are based on either one of the above two approaches but is currently one of the best image-denoising methods. The main goals of this thesis are to present and study the *BM3D* filter and to show that denoising is a fundamental tool in image processing, which can be applied to various image processing problems. We accomplish these goals by first studying the *BM3D* filter and some of its extensions and then presenting its application to:

- image deblurring,
- video denoising,

- image sharpening,
- raw-image denoising,
- RGB-image denoising.

Acknowledgements

This thesis is based upon studies conducted between years 2006 and 2010 at the Department of Signal Processing at Tampere University of Technology, Finland.

First, I would like to express my sincere gratitude to my supervisor Prof. Karen Egiazarian for his inspiring ideas, advices, and constant support. I am indebted to my co-supervisor Dr. Alessandro Foi for sharing his profound knowledge, giving example how to be constructively critical and how to achieve high-quality research. As a matter of fact, this thesis would have not been completed without the help of both of my supervisors.

I would like to thank Prof. Vladimir Katkovnik for his insightful advices and for clarifying the bigger picture in my research. I am grateful to Prof. Jaakko Astola, Prof. Moncef Gabbouj, and Prof. Ari Visa for leading the Department of Signal Processing to the right direction and creating a good atmosphere for doing research.

I am deeply thankful to my colleagues from the Transforms and Spectral Techniques group and to everybody else in the Department of Signal Processing for making it such a nice place to work. Special thanks go to Virve Larmilla, Ulla Siltaloppi, Elina Orava, Kirsi Järnström, Pirkko Ruotsalainen for being able to promptly help with practical and administrative matter.

Furthermore, I would like to express my appreciation from working on challenging topics with Leena Lepistö, Jarno Nikkanen, Martti Ilmoniemi, and Igor Curcio from Nokia.

I gratefully thank the Tampere Doctoral Programme in Information Science and Engineering (TISE) for funding my research and for arranging great courses and events; special thanks go to the TISE coordinator, Pertti Koivisto, who is always eager to provide important advices.

I thank all my friends and colleagues in Tampere and especially, Nikolay, Stanislav N. and Stanislav S., Vladislav, Anamaria, Aram, Atanas, Mihail, Dmytro, Dmitriy, Susanna, Robert, and many others.

Last but not least, I wish to thank Annika, Elli, and Leila for being my main inspiration.

Contents

List of Publications

The main contribution of this compound thesis is contained in the following publications.

[P1] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," *Proc. Electronic Imaging: Algorithms & Systems*, no. 6064A-30, San Jose, California, USA, January 2006.

[P2] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080-2095, August 2007.

[P3] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space," *Proc. IEEE Int. Conf. Image Process.*, San Antonio, TX, USA, September 2007.

[P4] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3D transform-domain collaborative filtering," *Proc. Euro. Signal Process. Conf.*, Poznan, Poland, September 2007.

[P5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Joint image sharpening and denoising by 3D transform-domain collaborative filtering," *Proc. Int. Workshop Spectral Meth. Multirate Signal Process.*, Moscow, Russia, September 2007.

[P6] K. Dabov, A. Foi, and K. Egiazarian, "Image restoration by sparse 3D transform-domain collaborative filtering," *Proc. Electronic Imaging: Algorithms & Systems*, no. 6812-07, San Jose, California, USA, January 2008.

[P7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "A nonlocal and shape-adaptive transform-domain collaborative filtering", *Proc. Int. Workshop on Local and Non-Local Approx. Image Process.*, Lausanne, Switzerland, August 2008.

[P8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "*BM3D* Image Denoising with Shape-Adaptive Principal Component Analysis", *Proc. Workshop Signal Process. Adaptive Sparse Structured Repres.*, Saint-Malo, France, April 2009.

In addition to the above publications, following are supplementary publications (which are related to the topic of the thesis) coauthored by the author of the thesis.

[S1] A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-Adaptive DCT for Denoising and Image Reconstruction," *Proc. Electronic Imaging: Algorithms & Systems*, no. 6064A-18, San Jose, California USA, January 2006.

[S2] D. Rusanovskyy, K. Dabov, and K. Egiazarian, "Moving-Window Varying Size 3D Transform-Based Video Denoising," *Proc. Int. Workshop Video Process. Quality Metrics*, Scottsdale, Arizona, USA, January 2006.

[S3] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Inverse halftoning by pointwise shape-adaptive DCT regularized deconvolution," *Proc. Int. Workshop Spectral Meth. Multirate Signal Process.*, Florence, Italy, September 2006.

[S4] V. Katkovnik, A. Foi, K. Dabov, and K. Egiazarian, "Spatially adaptive support as a leading model-selection tool for image filtering," *Proc. Workshop Inf. Theoretic Methods Science Engineering*, Tampere, Finland, August 2008.

The author of the thesis is the main contributor of the work presented in [P1-P8] and [S3]. For [S1], [S2], and [S4], the author has been responsible for the algorithm implementations. The contribution of all coauthors is nevertheless essential and without it, none of these publications would have been possible.

List of Abbreviations

1-D, 2-D, 3-D	One-, Two-, Three-Dimensional
AC	Alternating Current
AWGN	Additive White Gaussian Noise
BSNR	Blurred Signal-to-Noise Ratio
CCD	Charge-Coupled Device
CDF	Cumulative Distribution Function
CFA	Color-Filter Array
CMOS	Complementary Metal-Oxide Semiconductor
dB	Decibel
DC	Direct Current
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DST	Discrete Sine Transform
DWT	Discrete Wavelet Transform
$\mathbb{E}$	Expectation Operator
$\mathcal{F}$	Discrete Fourier transform Operator
FFT	Fast Fourier Transform
FPN	Fixed-Pattern Noise
GSM	Gaussian Scale Mixtures
HDR	High Dynamic Range
i.i.d.	Independently and Identically Distributed
ISNR	Improvement of Signal-to-Noise Ratio
JPEG	Joint Photographic Experts Group
LDR	Low Dynamic Range
LPA	Local Polynomial Approximation
MSE	Mean Squared Error
MPEG	Moving Picture Experts Group
MPix	Mega-Pixel
OMP	Orthogonal Matching Pursuit
ON	Orthonormal
PCA	Principal Component Analysis
PDE	Partial Differential Equation
PDF	Probability Density Function
PSNR	Peak
RGB	Red, Green, Blue color components
RV	Random Variable
SA-DCT	Shape-Adaptive Discrete Cosine Transform
SNR	Signal-to-Noise Ratio

SSIM	Structural Similarity Index Map
SVD	Singular Value Decomposition
var	Variance Operator
WGN	White Gaussian Noise

Chapter 1

Introduction

1.1 Background and motivation

Digital image acquisition is becoming increasingly widespread, evidenced by a constant growth in the production of image sensors. About 2.6 billion [Gro08] image sensors were sold in year 2008. In addition to traditional applications in digital cameras, image sensors find a wide range of applications in devices used in medicine, automation, astronomy, military and automotive industries, and increasingly in communication devices such as mobile phones [Pub09] which alone consumed the largest portion of all manufactured image sensors in year 2008.

The acquisition of a digital image (i.e. a 2-D array of numbers) from a physical scene is a process [The95, RSYD05, KWCL06] that typically involves lens and a solid-state image sensor. The scene radiance is focused by the lens to the sensor, which is a 2-D array of optoelectronic semiconductor elements (possibly covered by microlens array and a color-filter array). During the exposure, each element of the array absorbs photons, which (if their energy is high enough) generate electrons and these electrons (or electrical charge) are accumulated. Subsequently, the accumulated electrical charge is converted to voltage, amplified, and converted to a digital number. The digital image coming from a sensor is referred to as *raw image* throughout this thesis. Image acquisition is inevitably an imperfect process and raw images contain a plethora of degradations among which are various types of noise (illustrated in Fig. 1.1), blur, and geometrical distortions. In addition, a raw image typically cannot be directly visualized due to, e.g., application of CFA and uncalibrated color response. Therefore, digital postprocessing [RSYD05] is applied on a raw image in order to correct the degradations and to produce an image that corresponds to the physical scene and is suitable for visualization. Some typical postprocessing operations (illustrated in Figure 1.2) are: denoising, color-filter array interpolation, enhancement (sharpening, contrast improvement, etc.), color correction, white balancing, and correction of lens distortions.

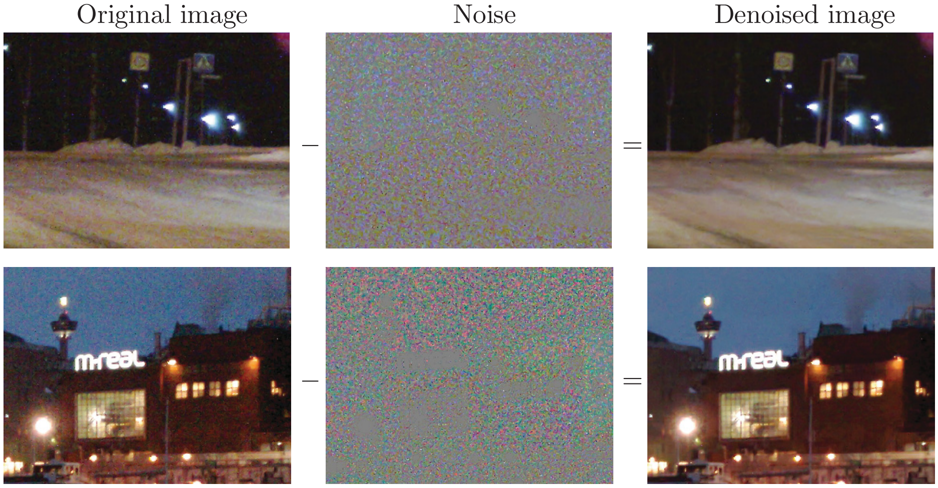


Figure 1.1: Illustration of images distored by noise. The filtered images are obtained after using the denoising algorithm that we discuss in Section 5.2. The original images are fragments of 5 MPix images taken with Nokia N95 mobile phone.

1.2 Objectives and scope of the research

The work done for this thesis is a contribution to image processing in general, and to image denoising and enhancement in particular. The structure of the presentation is briefly described below and is also illustrated in Fig. 1.3.

- First, we present and study a novel grayscale-image denoising filter, termed *BM3D* filter, and present the scientific background of works that it is related to.
- Second, we study various applications of the *BM3D* filter to other image-processing problems, including:
    - video denoising,
    - denoising of raw sensor images,
    - denoising RGB images
    - non-blind image deblurring
    - image sharpening

By including the above applications of the BM3D, we aim to show that image denoising is a fundamental tool in image processing, which can be applied to solve other (often more practical) image processing problems.

In order to represent image structures underlying the noise as sparsely as possible, the *BM3D* filter brings together two groups of existing filters — nonlocal filters and transform-domain filters. We use the term *sparse representation* throughout

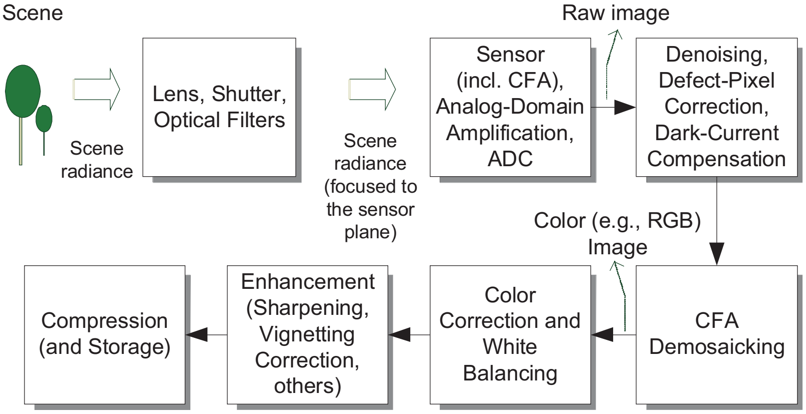


Figure 1.2: Simplified flowchart of a typical image-acquisition process and of some common postprocessing operations performed on raw images. Note: the particular order of most of the postprocessing operations may vary; e.g., denoising can be performed either on a raw image or on a RGB image (obtained after CFA demosaicking).

this work to denote a linear-transform (e.g., DCT, DFT, etc.) representation characterized by having few high-magnitude coefficients and plenty of low-magnitude coefficients. This term is a synonym for transform-domain *energy compaction* and transform-domain *decorrelation*, both of which are commonly used terms in image processing and in image coding, in particular. By exploiting sparse image representations, the *BM3D* filter is shown to be highly effective for denoising of additive white Gaussian noise (AWGN).

The second main goal of this work is to show how the *BM3D* filter can be applied to solve various image-processing problems which are more practical than the denoising of AWGN. In particular, we address some of the standard postprocessing operations performed in digital image-acquisition devices. These postprocessing applications include denoising of raw sensor data, sharpening, and deblurring. In addition, we consider the application of the *BM3D* filter to video denoising and to denoising of color RGB images. A peculiarity of all these applications is that they inherit the effectiveness of the *BM3D* filter and produce results that are competitive with the state-of-the-art.

1.2.1 Link to publications

It is important to give an explicit link between the eight publications assembled in this thesis and the particular parts where they are introduced. Publications [P1],[P2],[P7], and [P8] are considered in Chapter 4; they present the *BM3D* image-denoising filter and its extensions, i.e. the main contribution of this work. The very first work on image denoising based on collaborative 3-D filtering and grouping was done in [P1], which can be considered as a preliminary version of the *BM3D* filter. Subsequently, the *BM3D* filter was developed in the full-length article [P2].

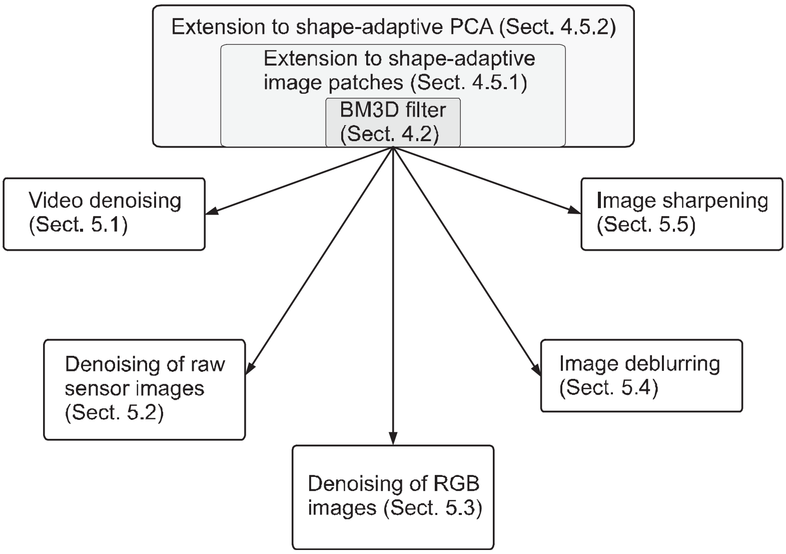


Figure 1.3: The main contributions of the thesis with links to their corresponding sections.

Extension of the *BM3D* filter to adaptive-shape neighborhoods was proposed in [P7], which is introduced in Section 4.5.1. Further extension of the *BM3D* to shape-adaptive PCA was developed in [P8], which is introduced in Section 4.5.2.

Publications [P6],[P5],[P4], and [P3] consider various applications of the *BM3D* filter and form the basis of Chapter 5. In particular, the image deblurring method from [P6] is considered in Section 5.4, the joint sharpening and denoising of images from [P5] is given in Section 5.5, the video denoising filter developed in [P4] is considered in Section 5.1, and the RGB-image denoising from [P3] is presented in Section 5.3.

1.3 Thesis outline

Chapter 2 contains preliminaries about statistics of natural images and about the image-noise models considered in this work. In particular, prior knowledge of natural-image statistics is of paramount importance for understanding how and why any image denoising method works. A somewhat brief introduction of the otherwise vast field of image denoising is given in Chapter 3. In particular, we consider a rough classification into spatial- and transform-domain filters. The main contribution of this thesis, the *BM3D* filter, is presented in Chapter 4, which comes as a logical continuation of the introduction. The application of the *BM3D* filter to various practical image processing problems is considered in Chapter 5. Illustration of how these contributions are structured is given in Fig. 1.3.

Chapter 2

Preliminaries: Natural-Image Statistics and Image Noise Models

Proper understanding of the image-restoration problem (i.e. properties of both image scenes and degradations) is an important step towards its solution. Therefore, in the sequel of this chapter we present the following relevant preliminary information:

- a brief overview of some important developments in the field of natural-image statistics (Section 2.1) and
- the considered image noise models and links to applications that they are used in (Section 2.2).

2.1 Natural-image statistics

Natural images contain redundant information. This claim is implicitly or explicitly assumed in any image processing application. Image coding and compression, in particular, almost entirely depend on image redundancies to reduce the bit size of image representation. Not surprisingly, all image restoration methods are also based on implicit or explicit assumptions of the true images that are underlying the degradations. Thus, knowledge of natural-image statistics is fundamental not only for understanding how the image denoising method proposed in this thesis works but also for understanding any other image restoration method.

The link between natural-image statistics and the human visual system has been established long ago [Bar61]. Vision, the most important source of information to humans, has evolved for millions of years to accommodate to the natural visual environment. It is agreed upon [Bar61, Bar72] that the visual processing in the brain has evolved to minimize the redundancy in representing natural images in order to achieve metabolic efficiency, as well as representational and learning efficiencies according to a more recent work [GF07]. Therefore, the modeling of natural images is traditionally studied in the context of image formation in the brain. However, to this date, there are no ground-truth models of either natural images or the human vision — due to their very high complexity. However, there is a vast amount of work on these topics and we shall follow some of the developments and draw some important conclusions, to which we shall refer later in the thesis in the context of image restoration. Before we proceed on this topic, we briefly introduce the basics of the visual formation in the brain. The light incident on the photoreceptors in the retina is converted to electrical impulses, processed, and then fed to the ganglion cells whose long axons (i.e. outputs) form the optic nerve which is capable of transmitting impulses over relatively long distances in the brain. The visual signals are next transmitted to the lateral geniculate nucleus of the thalamus (which serves as a relay for the sensory information) and subsequently they are transmitted to first layer (V1) of the visual cortex, which is the main processing unit for visual information in the brain. Simple cells in the V1 of the visual cortex, in particular, are among the very first processing units of visual information in the cortex. In this context, the receptive field of a cortical cell is the visual stimulus that triggers maximal response of the cell in terms of frequency of firings. In particular, the receptive fields of simple cells in the V1 were shown [HW62] to be edge-like patterns with various scales and orientations. Readers interested in in-depth coverage of visual neuroscience are referred to e.g. [CW03, HHH09].

Starting from the most basic properties of natural images, local image correlations have been among the first recorded statistics [Kre52] of natural images. In particular, the study [Kre52] has been done on television scenes and involved a derivation of single-pixel distributions and of correlations between neighboring pixels. A variety of later works [Fie87, BM87, TTC92] have found that the expectation of the Fourier spectrum magnitude of natural images follows a power-law curve, i.e. $\mathbb{E}\left[\left|\mathcal{F}\left\{y\right\}(f)\right|\right] \propto 1/f^{\alpha}$ where $\alpha \approx 1$ and f is the radial component of the spectrum when expressed in polar coordinates.

The Fourier spectrum alone does not offer good discrimination of spatially localized features such as oriented edges and singularities, which instead are scattered over a wide frequency range (i.e. they are not represented sparsely). Unfortunately, these features are of utmost importance in the formation of early vision since the receptive fields of the simple cells in the V1 are comprised of localized oscillations with various orientations and scales [HW62]. Following this observation, the time-frequency representations studied by Gabor [Gab46] have been found appropriate [Mar82, Wat83, SB82, Fie87] in the modeling of the receptive fields of cortical cells. With the later development of the multiscale decompositions, the time-frequency representation were generalized by them and these multiscale decompositions (e.g. wavelet decompositions) became an important tool in modeling both natural image scenes [Mal89b] and the receptive fields of simple cortical cells [Fie93]. The basis elements of the multiscale transforms are formed by translations, dilations (scalings), and rotations of some particular generating (e.g. wavelet) functions. In the context of natural image statistics, the following observations about the multiscale decompositions were confirmed in [Fie93]. First, the contrast in natural images is constant across scales; which means that under magnification, the amount of contrast is likely to remain the same. Second, it was shown that there exists correlation between structures across different scales. It was also shown [Fie93] that these properties are in agreement with the $1/f$ modeling of the amplitude spectrum of natural images.

Self-similarity and scale-invariance of natural images have been the foundations of fractal compression [Fis95] and of fractal image representations, in general. Expectably, natural images are never perfectly self-similar as they do not contain exact copies of themselves at different scales but as stated in [Fis95], they do contain a different, more relaxed form of self-similarity. That is, natural images contain regions which are similar to other regions at the same scale/orientation and across scales/orientations. This claim was supported by a recent study [AVT07] which showed that there exists significant affine self-similarity between square blocks of fixed size extracted from a single image. In particular, this work confirmed the existence of mutual information between translated blocks and, while not surprising, this result is fundamental for the development of the thesis as it is a manifestation of what we term *nonlocal image modeling*. In particular, by *nonlocal image modeling* we refer to modeling the structural similarities between image fragments at a fixed scale. This term is closely related to *self-similarity*, which typically has a more general interpretation that includes similarities across scales.

Following the observation that the human brain uses a sparse natural-image representation, Olshausen and Field performed [OF96] a search for a basis that maximizes the sparsity of representing square blocks extracted from natural images. The basis search was performed by minimization of a cost function with a penalty for the sparsity, which is represented by a particular nonlinearity. This approach produces bases whose elements are not necessarily orthogonal and in addition can be overcomplete. The most interesting result of this work however is that the resultant basis elements are comprised of localized and oriented oscillations — strongly resembling the receptive fields of simple cortical cells. This work on maximizing the sparseness was further developed in [Ols03b] where also space-time representations were considered. Assuming high-order non-Gaussian statistics of natural images blocks, Bell at al. [BS97a] proposed to use Independent Component Analysis (ICA) to represent local image neighborhoods. The basis elements obtained by ICA therein appear similar both to the receptive fields of simple cells in the visual cortex and also to the basis elements obtained by the sparseness maximization performed in [OF96]. In fact, the ICA algorithms also exploit maximization of sparsity in a particular form, either as a Laplacian prior [BS97a] or by maximizing the kurtosis when Fast-ICA [HO97] is used, where the kurtosis is used as a measure of sparsity. For a thorough study of natural-image statistics and ICA, the recent book [HHH09] is a recommended reference.

An interesting aspect of the representations that arise from maximization of sparsity or from ICA is that they are typically overcomplete [Ols03b]. The overcompleteness is also characteristic of the visual cortex where, e.g., in monkey V1 alone there are on the order of 50 times more output fibres than input ones. In addition, desirable basis properties (in terms of matching the receptive fields of simple cells) such as shift invariance [SFAH92b] and rotation invariance [GBP$^+$94] require overcompleteness. In fact, overcomplete linear representations have been found [RP00] to produce sparser representation of natural images than criticallysampled counterparts.

Considering the above mentioned results, below we list some of the observations about natural images, which we will make use of in the sequel of the thesis:

- there exists correlation between neighboring image pixels,
- images can be sparsely represented by multiscale transforms,
- there exists self-similarity between image subsets at a given scale (enabling *nonlocal image modeling*),
- overcomplete linear image representations are sparser than non-overcomplete ones.

We note that the above properties are in no way sufficient descriptors of natural images nor are they independent from each other. The emphasis on the term *nonlocal image modeling* is due to the fact that we shall often refer to it in the sequel.

2.2 Considered image noise models

Since the main contribution of this work is an image **denoising** method, the noise models are of particular importance. The random nature of noise is best reflected by a probabilistic modeling. In a very general scenario, each pixel of a noisy image is a realization of a RV with a PDF conditioned on realizations of other RVs and on the (unknown) noise-free image. However, for practical image denoising applications, parametric models (with few parameters) of the PDF are most commonly used. The following sections introduce the noise models considered in this thesis along with hints on the estimation of their parameters. These noise models are relevant to the practical imaging problems considered in Chapter 5.

2.2.1 Additive white Gaussian noise

Additive white Gaussian noise (AWGN) with zero mean and fixed variance is probably the simplest and most commonly used model in the image denoising literature (Chapter 5.2.2 in [GW06]). The AWGN observation model in the case of discrete images is given by

$$z(x) = y(x) + \eta(x), \tag{2.1}$$

where $z(\cdot)$ is a noisy pixel, $\eta(\cdot) \sim \mathcal{N}\left(0, \sigma^2\right)$ is independently distributed, y is the unknown noise-free signal and $x \in X$ is an index (i.e. pixel coordinate) from the finite d-dimensional domain $X \subset \mathbb{N}^d$; if not explicitly specified otherwise, in the sequel of the thesis we assume $d = 2$.

It is worth discussing why AWGN has a particularly important role in image denoising. Let us recall what the central limit theorem states: given independent RVs V_i, $i = 1, \ldots, n$, with $0 < \sigma_n^2 = var\left\{\sum_{i=1}^n V_i\right\} < \infty$ and under mild hypotheses (see [Sha99] for details), for large n, the distribution of the standardized average $\frac{1}{\sigma_n}\sum_{i=1}^n \left(V_i - \mathbb{E}\left\{V_i\right\}\right)$ approaches the standard normal distribution

$\mathcal{N}(0,1)$. This theorem has significant practical implication and is particularly applicable to devices that contain a lot of independent additive noise contributors, each having some arbitrary distribution that may well be unknown. In such a scenario, the sum of these noise contributions is best described as normally distributed. In particular, AWGN is a good model [Nak06] of the inevitable thermal noise, among other types of noise, in electronic devices.

The estimation of the standard deviation σ from a noisy signal has some well-established solutions such as the one proposed in [Don95] that uses the median absolute deviation of the finest-scale wavelet coefficients. This method however tends to overestimate σ, especially when the noise-free image spectrum contains significant amount of high-frequency components. Recent methods that are more robust in that regard can be seen in [DF09] and in the references therein.

Throughout the rest of the text, in the context of image denoising, unless another model is explicitly specified, we implicitly assume AWGN degradation. In addition, we utilize the notation from Eq. (2.1); i.e. z is a noisy signal, y is unknown noise-free signal and η is a white Gaussian noise realization.

2.2.2 Additive colored noise

Additive colored noise (Chapter 5.2.3 in [GW06]) model is considered in the thesis in relation with the problem of regularizing inverse problems in imaging (Section 5.4). The additive colored noise model in pixel domain is given by

$$z(x) = y(x) + (v * \eta)(x), \tag{2.2}$$

where $*$ denotes convolution, v is the impulse response that defines the noise correlation and $\eta(\cdot) \sim \mathcal{N}(0, \sigma^2)$ is independently distributed. We can express Eq. (2.2) in Fourier domain as

$$\begin{aligned} \mathcal{F}\{z\} &= \mathcal{F}\{y\} + \mathcal{F}\{v * \eta\} \\ &= \mathcal{F}\{y\} + \mathcal{F}\{v\}\mathcal{F}\{\eta\} \end{aligned}$$

where the convolution becomes elementwise multiplication in Fourier domain, and $|\mathcal{F}\{\eta\}| \sim \mathcal{N}(0, \sigma^2)$ is i.i.d. normally distributed just as η. We have omitted any normalizing factors from the above equation and we otherwise assume $\mathcal{F}$ is orthonormal. From the above equation, it can be seen that the noise can be fully defined by its spectrum $\mathcal{F}\{v\}$, which gives a meaningful interpretation to noise colors in analogy with the spectral content of pure colors. That is, red noise contains predominantly low-frequencies, blue noise contains predominantly high-frequencies, and white noise contains equal contribution from all frequency components. Naturally, white noise ($|\mathcal{F}\{v\}|^2 = const$) is equivalent to the AWGN model from Eq. (2.1).

The estimation of v (or, equivalently, $\mathcal{F}\{v\}$) involves estimation of a significant number of elements and thus does not have well-established universal solutions as in the case of variance estimation for AWGN. In the context of image deblurring, the authors of [FDKE06] proposed an estimation of v which relies on prior knowledge of the noise generation.

2.2.3 Signal-dependent noise modeling of raw (sensor) images

We start with a brief description of the image-acquisition process with solid-state image sensors. The radiance (reflected or originating) from a scene is focused by a lens to the sensor 2-D array of optoelectronic semiconductor elements. The irradiance at the surface of each element (pixel) of this 2-D array is integrated during the exposure time and converted to electrical charge; this is accomplished by an optoelectronic conversion of photons (whose energy is high enough) to electrons. The charge is subsequently transformed to voltage, which is amplified by a factor termed *analog gain* and converted via ADC to a digital number, which is quantized with precision in the range 8–16 bits. There exist two major image-sensor technologies — CCD and CMOS. In a CCD sensor, the accumulated charge is transported by shifting towards the end of each row (of the 2-D sensor array), where conversion to voltage and amplification are performed separately for each row. On the other hand, in a CMOS sensor, the conversion to voltage and amplification are performed at each pixel. A comparison of these two technologies can be found in, e.g., [Nak06].

Regardless of the sensor technology (CCD or CMOS), the image acquisition process is inevitably imperfect and raw images are degraded with various types of noise [The95, BLIA97, Nak06]. In particular, we consider the following rough classification of noise sources;

- temporal signal-dependent noise: photon shot noise;
- temporal signal-independent noise: reset noise, thermal noise, 1/f (flicker) noise, dark current shot noise;
- fixed-pattern noise (FPN), i.e. having no temporal variation: dark FPN, photo-response non uniformity (light FPN), defected pixels, hot spots.

For some of the above noise sources there exist effective countermeasures; e.g., the reset and the 1/f noise can be greatly reduced the correlated-double sampling technique [ET96]. In addition, the fixed-pattern noise can be reduced by subtracting a pre-computed noise pattern (that does not vary in time but can vary with the temperature, the input-signal, or the exposure time) from the raw image; also, the quantization noise can be reduced by increasing the number of bits of the ADC conversion. However, other noise types do not have effective countermeasures; e.g., the photon shot noise (due to the photon-counting nature of the optoelectronic semiconductor), the thermal noise (except if the temperature can be controlled), and the dark current photon shot noise.

We are interested in an overall approximate model of the above noise sources. The signal-dependent photon shot noise is manifested [The95, Nak06] by a Poissonian distribution of the pixel intensities, which in turn can be approximated by a normal distribution having variance proportional to its expectation. The rest of the noise sources can be approximated by AWGN [Nak06] (with variance dependent on the temperature and on the exposure time). According to these considerations, we adopt the following approximate Poissonian-Gaussian model of the noise in raw sensor images,

$$\begin{aligned} z(x) &= y(x) + \sigma(y(x))\,\xi(x) && (2.3)\\ &= y(x) + \eta_{\mathrm{p}}(y(x)) + \eta_{\mathrm{g}}(x) && (2.4)\end{aligned}$$

where $\xi(\cdot)$ is statistically independent with unit variance and zero mean, $\sigma(y(\cdot))$ is the standard deviation of the noise in z, which is a function of the unknown intensity $y(\cdot)$, η_{p} is a signal-dependent white noise contributed by the photon shot noise and η_{g} is WGN contributed by all of the white noise sources mentioned above. A detailed study of this noise model can be found in [FTKE08], which also considers the effect of the clipping (from above and from below); we shall mostly follow the derivations considered therein. To derive a parametric model of Eq. (2.4), we assume the following distributions,

$$\begin{aligned} \chi\left(y(x) + \eta_{\mathrm{p}}(y(x))\right) &\sim \mathcal{P}(\chi y(x))\\ \eta_{\mathrm{g}}(x) &\sim \mathcal{N}(0, b),\end{aligned}$$

where $\mathcal{P}$ stands for Poissonian distribution parametrized by $\chi \in \mathbb{R}^+$ and the parameter $b \in \mathbb{R}^+ \cup \{0\}$ is the variance of the AWGN component η_{g}. After simple calculations, the overall variance of the Poissonian-Gaussian noise model from Eq. (2.4) can be expressed as

$$\sigma^2(y(x)) = ay(x) + b, \tag{2.5}$$

where $a = \frac{1}{\chi}$. This model has only two parameters — a and b. An estimator of these two parameters has been proposed in [FTKE08]; Figure 2.1 presents estimated standard deviation by this method applied to a single raw image from four different cameras. In fact, the Poissonian-Gaussian noise model is well established [LSK+08, HDF10, TFG01, FAKE07, FTKE08] as a good model for the noise in raw sensor images.

According to the Poissonian-Gaussian noise model, a pixel's SNR grows approximately linearly with the increase of the intensity, i.e.,

$$\mathrm{SNR}(z(x)) = \frac{\mathbb{E}\{z(x)\}^2}{var\{z(x)\}} = \frac{y^2(x)}{ay(x) + b},$$

where at low intensities, the SNR is limited by the signal-independent component b and at higher intensities by the signal-dependent photon shot noise component $ay(x)$. The more photons absorbed in the semiconductor optoelectronic element of a given pixel, the higher the intensity and the higher the SNR of that pixel. The amount of absorbed photons depends mostly on the following factors

- Physical and technological factors:
  - pixel area (greater pixel area in general allows more sensed photons),
  - color-filter array (a filter that attenuates certain spectral components),
  - vignetting (gradual intensity decrease from the image center to the boundaries).

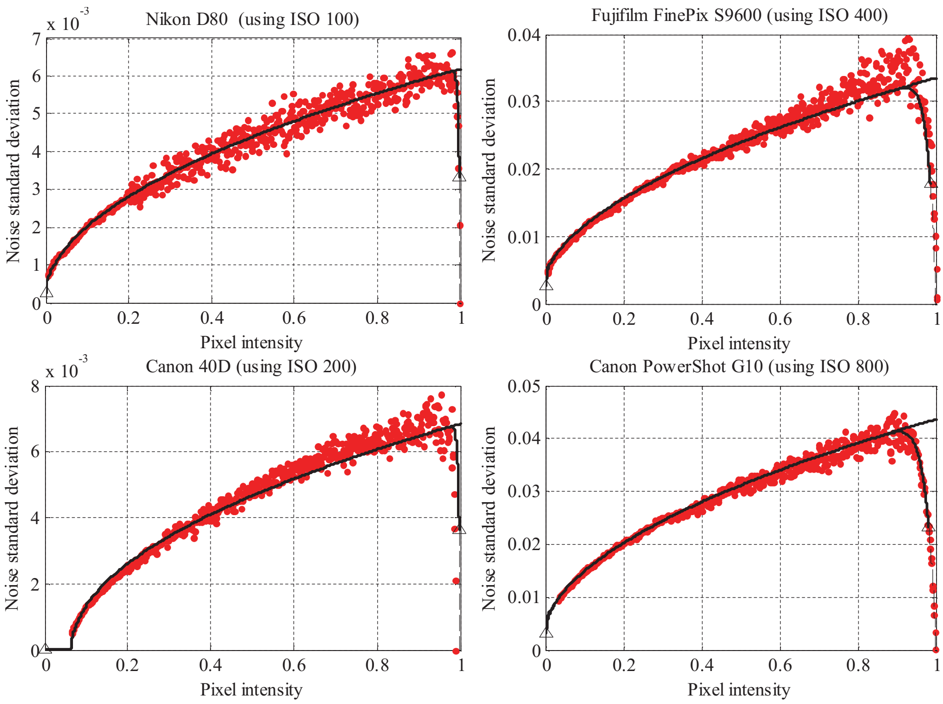


Figure 2.1: Illustration of the the noise standard deviation estimated by [FTKE08] as a function of the pixel intensity for four different cameras using different ISO values. The solid curves correspond to the Poissonian-Gaussian parametric model of the noise from Eq. (2.4). These curves offer a good fit to the indivdual estimates given by the red dots. The dashed lines represent the curves that take into account the clipping at the boundaries 0 and 1.

- Image-acquisition factors:
  - scene illumination (the illumination can greatly vary; e.g., from bright sun-light to dim indoor and dark-night conditions),
  - exposure time (the average amount of sensed photons grows linearly with the exposure time; short exposure times are preferred when there can be relative motion between camera and object).

It is important to mention at this point that an image with signal-dependent noise from Eq. (2.4) can undergo a variance-stabilizing transformation [Foi07, Foi09] that transforms the noise variance to be close to a constant. The noise in the resultant variance-stabilized image can then be treated as signal-independent with constant variance. Subsequently, a denoising method for additive white noise can be applied on such an image.

Chapter 3

Introduction to Image Denoising

This chapter gives a rather limited introduction to prior works in image denoising, a research topic that has attracted vast amount of contributions, particularly in the past two decades. We consider exclusively methods for attenuation of AWGN from images and we note that filters for signal-dependent noise and for colored noise can in many cases be obtained by extending an AWGN denoising method; this is also what we do in this thesis. We consider a rough classification into spatial- and transform-domain filters, based on whether these filters exploit linear-transform representations or not. Within each of these two major groups, we use finer classification which is by no means exhaustive nor the boundaries between the presented classes are well defined. In fact, there exist explicit links between many of the works that we present. The order in which the prior works are presented should logically lead the reader to the next chapters where we focus on the contribution of the thesis, the *BM3D* filter and the applications based on it. In that regard, of particular interest are the nonlocal spatial filters (Section 3.1.5), and the nonlocal transform-domain filters (Section 3.2.4) as the *BM3D* filter belongs to the latter class.

3.1 Spatial-domain filters

3.1.1 Local polynomial approximation

Among the earliest local estimators, the weighted average estimator was independently developed by Nadaraya [Nad64] and Watson [Wat64] in the context of nonparametric regression modeling. Assuming the AWGN model from Eq. (2.1), the Nadaraya-Watson estimator can be interpreted as a minimizer for the following weighted mean squares criterion

$$\hat{y}_{\text{N-W}}\left(x\right) = \arg\min_{C\in\mathbb{R}} \sum_{k\in X} w_h\left(x-k\right)\left[z\left(k\right)-C\right]^2 \tag{3.1}$$

where w_h is a predefined window function and h is a bandwidth parameter (also called scale or smoothing parameter) that controls the size of the local estimation neighborhood. The minimizer of the above equation and output of the estimator is given by

$$\hat{y}_{\text{N-W}}(x) = (g_h * z)(x), \qquad \forall x \in X \tag{3.2}$$

where $g_h = \dfrac{w_h}{\sum_k w_h(k)}$ and $*$ denotes convolution. The choice of w_h is based on desired properties of the estimator and, possibly, on a priori information of the true data y; the Gaussian window is a common choice. In the context of image denoising, the approximation with a constant in Eq. (3.1) corresponds to assuming uniform pixel intensities in a local neighborhood defined by w_h around the estimated pixel. This corresponds to the simplest image model considered in Section 2.1, for which the intensities of neighboring pixels are assumed locally correlated. To use a more complex model, the minimization in Eq. (3.1) can be done for a polynomial of degree $m \geq 0$, which results in a local polynomial approximation (*LPA*) [CD88, FG96, KEA06, Foi05]. The *LPA* estimate can be expressed by the following minimization problem

$$\hat{y}_{LPA}(x) = \arg\min_{p \in \mathcal{P}_m} \sum_{k \in X} w_h(x-k)\left[z(k) - p(k)\right]^2,$$

where $\mathcal{P}_m$ is the set of 2-D polynomials of degree m. It is noteworthy that the *LPA* estimator can be computed using convolution as in Eq. (3.2), where g_h is derived from the polynomial degree m and the window w_h; the derivation of g_h as well as a comprehensive study of the *LPA* estimators can be found in [KEA06]. From the image modeling point of view, the *LPA* estimator allows a more complex model of the signal within an image neighborhood as a polynomial. The idea of *LPA* is also known by different names such as moving least-squares, Savitzky-Golay filter, reproducing filter, moment filters, and kernel regression.

The fixed estimation kernel g_h (which depends only on the predefined parameters h, m, and window w_h) for the *LPA* estimators considered above, however, do not allow image adaptivity. One approach to achieve adaptive neighborhoods is the adaptive pointwise selection of the bandwidth h from the input noisy data, which has become a significant research topic [HSJM91, RSW95, JMS96, KEA06, Foi05]. One such adaptive bandwidth-selection procedure is the *ICI* rule [Kat99], which, combined with the *LPA* estimator, was shown [KEA06, Foi05] to achieve pointwise MSE that is close to the MSE when oracle bandwidth selection is used. In order to achieve directional adaptation, in addition to bandwidth adaptivity, the *LPA-ICI* has been extended by fusing individual directional *LPA-ICI* estimators [KFEA04, KEA06, Foi05]. The resultant anisotropic *LPA-ICI* was shown in the mentioned references to be very effective for various image processing applications allowing accurate reconstruction of edges and singularities with various orientations. Another estimator that exploits adaptive estimation kernel and *LPA* is the data-adapted kernel regression proposed [TFM07]. The kernel used for this estimator adapts to the local gradient orientation in order to avoid blurring of edges. The adaptivity involves three possible transformations of an otherwise circular kernel: elongation, scaling, and rotation, which are estimated from the local covariance.

3.1.2 Range filters

Another approach to make the estimation neighborhoods adaptive to the image data was employed by the *range filters*. In the class of range filters we include the bilateral filter [TM98], the sigma filter [Lee81], the Yaroslavski filter, the SUSAN filter, and other related works. The range filters use weighted averaging with adaptive weights that depend not only on the spatial distance to the estimated pixel but also on the distances between the pixel intensities (i.e. the image range). A general expression for the above definition of the range filters is

$$\hat{y}_{\mathrm{RF}}\left(x\right)=\sum_{k\in X}z\left(k\right)g_{\mathrm{sp}}\left(x-k\right)g_{\mathrm{rng}}\left[z\left(x\right)-z\left(k\right)\right], \tag{3.3}$$

where the kernel g_{sp} depends on spatial distance, just as in the *LPA* estimator, and g_{rng} contributes the range filtering by considering distances between intensities. It is the g_{rng} term that enables adaptivity to image data and allows to better preserve salient details such as edges without blurring; e.g., this can be achieved by using a smaller weight g_{rng} when the intensity difference is big. This corresponds to averaging over local neighborhoods with relatively homogeneous intensities and thus avoiding averaging across sharp edges. However, when the magnitude of the noise is comparable with or greater than the magnitude of edges, then the similarities between individual noisy pixels are not reliable to measure the similarities between their noise-free counterparts. One approach to overcome this drawback is to use similarities between image neighborhoods, as in the case of nonlocal filters, which will be discussed in Section 3.1.5.

3.1.3 Bayesian filters

Bayesian estimation [KR96, BS07] is a notable approach to image restoration. A very general formulation of the image-restoration problem in the Bayesian estimation framework is: given the observed noisy image z and prior information of the noise and of the noise-free image, find an estimate of the noise-free image. Let us consider the Bayes formula for the posterior probability

$$p_{\mathrm{post}}\left(\gamma\mid z\right)=\frac{p_{\mathrm{fit}}\left(z\mid\gamma\right)p_{\mathrm{prior}}\left(\gamma\right)}{p_{Z}\left(z\right)}, \tag{3.4}$$

where γ is an estimate of the noise-free image y, the *image prior* $p_{\mathrm{prior}}\left(\gamma\right)$ is the probability that γ belongs to the class of noise-free images, the *data fit* (or *likelihood*) $p_{\mathrm{fit}}\left(z\mid\gamma\right)$ is the conditional probability that z has been generated according to the assumed noise model from γ, and $p_{Z}\left(z\right)$ is the marginal probability of obtaining the particular noisy realization z; since p_{Z} does not depend on γ, it can be treated as a normalizing constant in most situations. Bayesian estimation deals with finding a solution γ using the posterior $p_{\mathrm{post}}\left(\gamma\mid z\right)$ from Eq. (3.4). A commonly used Bayesian estimator is the conditional mean of the posterior, i.e.,

$$\hat{y}_{\mathrm{CM}}=\int\gamma p_{\mathrm{post}}\left(\gamma\mid z\right)d\gamma,$$

which involves integration in a high dimensional space and for most useful image priors, does not have a closed-form solution. In practice, the problem is often discretized and solved by numerical integration. The Maximum A Posteriori (MAP) estimator is another important Bayesian estimator, which attempts to find a solution γ that maximizes the posterior, i.e.,

$$\hat{y}_{\mathrm{MAP}} = \arg\max_{\gamma} p_{\mathrm{post}}\left(\gamma \mid z\right). \tag{3.5}$$

In the case when the posterior $p_{\mathrm{post}}\left(\gamma \mid z\right)$ is characterized by one main lobe (e.g., a unimodal distribution), then the MAP estimator is a good choice.

By far, the biggest challenge [She03] in Bayesian estimation in general and in image processing in particular is the derivation of good (image) prior p_{prior}. This problem is equivalent to obtaining a good model of natural-image statistics, a topic discussed in Section 2.1, which is mostly unsolved problem due to the very high complexity of natural images. Various image priors have been proposed in the form of particular parametric models of images or of image patches. In this thesis, we mention various priors of the transform spectra of natural images in Sections 3.2.1 and 3.2.2; other image priors can be found, e.g., in Chapter 3 of [CS05], in Chapter 1.3 of [KR96], in [RB05, WG04], and in the recent [WF07].

3.1.4 Variational and PDE-based filters

The variational approach [CS05] to image restoration can be considered as a deterministic interpretation of the Bayesian MAP estimation from Eq. (3.5), where the restoration result is the argument of an iterative minimization of a particular functional, i.e.,

$$\hat{y}_{\mathrm{VAR}} = \arg\min_{\gamma} E\left(\gamma, z\right)$$

where the functional $E\left(\gamma, z\right)$ can typically be written as a sum of two terms,

$$E\left(\gamma, z\right) = E_{\mathrm{fit}}\left(\gamma, z\right) + E_{\mathrm{pen}}\left(\gamma\right), \tag{3.6}$$

where E_{fit} is a fidelity term that restricts the solution not to deviate from the input data and E_{pen} is a regularization penalty that imposes some useful prior model on the solution. If we make analogy to the Bayesian framework, the regularization term can be interpreted as the image prior and the fidelity term can be interpreted as the data fit. In fact, for particular parametric PDF models of the image prior and data fit (such as e.g. the Gaussian distribution), the MAP estimation given by Eq. (3.5) in the Bayes framework is exactly equivalent to solving a minimization of the form given in Eq. (3.6). Typically, $E_{\mathrm{fit}}\left(\gamma, z\right) = \left\|\gamma - z\right\|_{p}^{p'}$, i.e. the fidelity term is the $p-$norm of the difference between the input noisy image and the current solution γ. The standard Tikhonov regularization [Tik63] corresponds to $E_{\mathrm{fit}}\left(\gamma, z\right) = \left\|\gamma - z\right\|_{2}^{2}$ and $E_{\mathrm{pen}}\left(\gamma\right) = \left\|\gamma\right\|_{2}^{2}$, where the regularization term imposes image smoothness — i.e. the energy (2-norm) of the solution is minimized. A widely used penalty in image processing that avoids smoothing of salient details is the total variation (TV) penalty proposed in [ROF92], which is given as the 1-norm of the gradient of the solution. Regularization with the TV penalty results in smoothing of weakly varying details and preservation of salient (having strong variation) details such as edges.

Closely related to the variational framework discussed above, partial differential equations (PDEs) are yet another tool that have found a very wide application

[Sap06, AK06, CS05] in image processing and in image denoising in particular. Indeed, a PDE formulation of an image denoising problem can emerge from the Euler-Lagrange equations of a particular variational formulation or from applying the gradient descent method to minimize a functional of the form given in Eq. (3.6). Given a particular PDE model, the solution is given by an iterative (over time) process where the number of iterations (elapsed time) determines the degree of smoothness of the solution. The selection of optimal number of iterations (stopping time) is a key issue [CS05] in PDE based image processing. The anisotropic diffusion proposed in [PM90] is one of the widely adopted nonlinear PDE formulations in image processing, where the smoothing is guided by the spatial derivative, so that there is no smoothing across edges. Various applications of PDEs to image processing can be found in [GMM01, Sap06, AK06, CS05] and in the references therein. It is worth mentioning that a link between PDE methods, variational methods, and adaptive filters has been studied in [SKB01] and also the relation between anisotropic smoothing and the bilateral filter has been studied in [Bar02].

3.1.5 Nonlocal spatial filters

Nonlocal filtering is a relatively recent development in image processing. In this class we include filters that exploit the nonlocal modeling defined in Section 2.1. That is, a nonlocal filter exploits similarities between image neighborhoods (patches) from various spatial locations, hence the name *nonlocal*. To the author's knowledge, the NL-*means* proposed by Buades [BCM05b] used for the first time the term *nonlocal* in the context of image denoising. The NL-*means* estimates a pixel as the weighted average of pixels with weights that depend solely on the similarity between neighborhoods centered at these pixels and the neighborhood centered at the estimated pixel. Before we proceed, let us define of the NL-*means* estimator in case of discrete noisy observation (Eq. 2.1), which is given as a standard weighted averaging

$$\hat{y}_{\mathrm{NL}}(x) = \frac{\sum_{k\in X} g(k,x)\, z(k)}{\sum_{k\in X} g(k,x)},$$

with particular weight that manifests the nonlocal modeling, defined as

$$g(k,x) = \exp\left(-\frac{\sum_{\xi\in N} G_a(\xi)\left|z(k+\xi) - z(x+\xi)\right|^2}{h^2}\right), \tag{3.7}$$

where N is a predefined support of a square block, G_a is a 2-D Gaussian with standard deviation a, and h is a filtering parameter. A data-adaptive selection of the filtering parameter has been proposed in [VDVK09, DC09]. In the context of the NL-*means*, the notion of *similarity* between image neighborhoods can be interpreted as proportional to the weight given in Eq. (3.7). The key idea of the NL-*means* and the nonlocal filters in general is that they do not assume local intensity smoothness nor any particular local neighborhood model — unlike the local estimators considered in Section 3.1.1, which are essentially based on specific local models. Indeed, the NL-*means* estimator can be considered as a generalization of the range filtering concept (considered in Section 3.1.2) but whereas range filters exploit similarities between individual pixels, the NL-*means* exploits similarities between surrounding neighborhoods. Extensive study of the NL-*means* filter along with various extensions can be found in [BCM05b].

The optimal spatial adaptation (*OSA*) method [KB06] is a sophisticated nonlocal spatial estimator which is based on adaptive estimation neighborhoods. Thus, a pixel is estimated by a weighted sum of pixels within an adaptive neighborhood, where both the weights and the adaptive neighborhoods depend on similarities between image patches, i.e. exploiting nonlocal modeling. In contrast with the fixed estimation neighborhoods used for the NL-*means*, the estimation neighborhood of *OSA* filter is adaptively selected from a predefined set of local neighborhoods, using a change-point detection procedure similar to [Lep90] and to the *ICI* rules that we considered in Section 3.1. The overall *OSA* method delivers very good denoising performance [KB06] in terms of both PSNR and visual quality. It is worth discussing the fact that this otherwise nonlocal-based method exploits adaptive *local* estimation neighborhoods, where the largest possible neighborhood is not necessarily the optimal choice at every pixel location. This means that the non-local modeling of particular image details can be locally constrained.

Nonlocal variational formulation

The variational formulation of nonlocal priors has recently become and very active research topic [KOJ05, GO08, GO07, PBC08, ELB08]. These works have expressed the nonlocal image modeling as particular regularization functionals, which can be used as part of the overall variational minimization functional given in Eq. (3.6). This formulation is an important development since it allows to apply nonlocal filtering to variety of problems that exploit the variational framework, such as deblurring [KOJ05, Mig08], inpainting [PBC08], super-resolution [PBC08], compressive sensing [PBC08], image colorization [LTE08], and tomographic reconstruction [CMF$^+$08].

3.2 Transform-domain filters

In Section 2.1 we stated that linear transforms can be a very useful tool for modeling the redundancies in natural images. Therefore it is not surprising that they find a significant application in image processing. Indeed, transforms such as the DCT and the DFT have long [AR75] been used in signal and image processing to exploit redundancies in signals (audio, images, video) and represent them *sparsely*. For example, the ubiquitous MPEG 1, 2, and 4 video compression standards as well as the JPEG image compression standard all exploit the block DCT. The block DCT is realized by applying 2-D DCT on small disjoint image blocks, where local correlations are exploited to produce sparse image representation.

Following their introduction [BA83, Mey93, Mal89a, Dau88, Mal89b] to the signal-processing community, multiscale transforms (wavelets, pyramidal decompositions, etc.) have been found particularly effective in obtaining sparse image representations [Mal08], a very important property also discussed in Section 2.1 in the context of natural-image statistics. We give the following definition of multiscale transforms: linear transforms whose basis elements are obtained by dilations (scalings) and translations, and possibly by rotations, of one or more generating functions. A notable member of the multiscale transforms is the wavelet decomposition, which uses a wavelet generating function ψ of zero mean to compose a dictionary of basis elements, which can be expressed in the continuos domain as

$$\mathcal{D} = \left\{ \psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right) \right\}_{u \in \mathbb{R}, s>0},$$

where u is a shift (translation) parameter and s is a scale parameter. Dyadic wavelet decompositions where $u = 2^j$ are probably the most commonly used ones in image processing. The discrete dyadic wavelet decomposition can be efficiently computed using a cascaded two-band filterbank (see Chapter 7 in [Mal08]) in $O(N)$ time on a signal with N elements. In contrast with trigonometric transforms (such as the DCT, the DFT, etc.) and transforms with fixed spatial localization (such as the block DCT and the short-time Fourier transform), multiscale decompositions are more flexible as they allow for sparse representation of images structures with varying localization in time and in frequency and, possibly, with varying orientation.

The dictionary of a transform can in general be overcomplete, which means some strict subset of its elements forms a basis. Overcomplete multiscale representations, in particular, allow for much more flexibility and for sparser image representations than non-overcomplete ones and have become a very important tool in image restoration [Mal08, CD95, SFAH92a]. Translation-invariance of a multiscale decomposition is one property that requires overcompleteness [CD95]; this property is desirable because image structures are not necessarily aligned with the particular translations of the basis elements of a ON decomposition. Other commonly used overcomplete transforms are the dual-tree complex wavelets [Kin01], which are near translation invariant and near rotation invariant, and the steerable pyramid [SF95, GBP$^+$94], which allows for approximate rotation invariance. Not restricted only to dictionaries of multiscale transforms, overcomplete representations have been used in the extension of the block-DCT type of transforms to sliding-window transforms [ÖYE98, Gul03], where the overcompleteness is due to overlaps between successively transformed blocks.

The transforms considered above have fixed dictionaries. Thus, a question arises whether a transform can have dictionary of elements that are adapted to the input image. The answer is positive and the development such transforms that adapt to geometrical regularities in images is an active research topic. The ON decomposition into principal component (PCA) [Jol02] is one well established such method, where the basis elements are obtained as the eigenvectors of a covariance matrix (that is either a priori known or is empirically computed). The decomposition into independent components (ICA) [Hoy99] is another adaptive (overcomplete, in general) transform that has found applications in image processing. A different approach to adaptivity was sought by the shape-adaptive DCT (SA-DCT) [SM95], where the shapes of the neighborhoods on which this transform is applied are data-adapted; this transform has also been successfully applied to various image-processing problems [Foi07]. To name a few other adaptive transforms, we would like to mention some of the transforms that have adopted the "lets" ending: the brushlets [MC97], the ridgelets [Don98], the wedgelets [Don99], the curvelets [SCD02], the contourlets [DV05], the bandlets [LPM05], the shearlets [LLKW05], the surfacelets [LD07], the grouplets [Mal09]. Other adaptive transforms that are exploited very successfully in image denoising are given in Section 3.2.3.

3.2.1 Bayesian transform-domain filters

Bayesian estimation together with stochastic modeling of (multiscale) transform coefficients is a powerful tool in image processing [Mal08, Bov05]. The main problems in this area are the choice of transform and the derivation of prior models for its coefficients. A sparse image representation allows to use a relatively low-complexity prior with few parameters — whereas in pixel domain, low-complexity priors are not appropriate, in general, due to the huge diversity of natural images. Studies [WZ90, BS97b] on multiscale transforms of natural images have found that the transform coefficients have highly kurtoic marginal distribution characterized by a strong peak at zero and heavy tails, which can be modeled e.g. by the generalized Gaussian distribution [Mal08] (which also generalizes the Laplacian distribution). In addition to these findings, multiscale representations of natural images contain correlations both between coefficients from a given scale (intra-scale) and between coefficients from different scales (inter-scale). To model these correlations, multivariate priors must be considered. The Gaussian scale mixtures (GSM) model [RCB99, WS00] was found to be particularly effective for capturing intra-scale correlations by modeling neighborhoods of relatively small size (e.g., 3×3 or 5×5) of multiscale transform coefficients. This model was shown [PS03, PSWS03, GCP05, GCSP08, HS08] to be quite successful for image denoising. An extension of the GSM was recently proposed in [GPP09], where discrete mixture of linear projected GSMs were used either by projection onto fixed bases or onto data-adaptive PCA bases. Models of intra-scale correlations of neighboring coefficients by hidden Markov models have also been considered [CNB98, RCB99]. Without going into further details on this topic, we refer the reader to the recent developments [Sel08, RAS08, HS08, GPP09] and to the references therein. At this point we can conclude that multiscale transforms can be a very useful tool for deriving low-complexity image priors. The Bayesian estimation has had a significant implication in the derivation of shrinkage estimators — a topic discussed in the following section.

3.2.2 Shrinkage filters

Closely related to the Bayesian estimation and the variational formulation, *shrinkage* of a transform spectrum is one of the main image-restoration tools. A very simple general definition of shrinkage is: a procedure that attenuates the magnitudes of transform coefficients. A shrinkage estimator can be expressed in the following simple form

$$\hat{y}_{\text{shr}} = \mathcal{T}^{-1}\left\{\text{shrink}\left(\mathcal{T}\left\{z\right\}\right)\right\},$$

where $\mathcal{T}$ is a linear transform operator, $\mathcal{T}^{-1}$ is its inverse, and shrink is a shrinkage nonlinear operator. Early works [Yar85] of Yaroslavski showed good potential of empirical Wiener filtering (a form of shrinkage) being applied in block-DCT domain, which is defined (for AWGN) as

$$\mathrm{shink}_{\mathrm{WIE}}\left(\theta\right)=\theta\frac{\left|\hat{\theta}\right|^{2}}{\left|\hat{\theta}\right|^{2}+\sigma^{2}},\tag{3.8}$$

where $\left|\hat{\theta}\right|^{2}$ is an (empirical) estimate of the squared magnitude of θ. Notably, shrinkage of wavelet coefficients was established and extensively studied in milestone works of Donoho and Johnstone [DJ94, DJ95]. In particular, they studied soft- and hard-thresholding, defined respectively as

$$\begin{aligned}\mathrm{shrink}_{\mathrm{ST}}\left(\theta\right) &= \begin{cases}\mathrm{sign}\left(\theta\right)\left(\left|\theta\right|-\tau\right), & \text{if } \left|\theta\right|>\tau\\ 0, & \text{otherwise,}\end{cases}\\ \mathrm{shrink}_{\mathrm{HT}}\left(\theta\right) &= \begin{cases}\theta, & \text{if } \left|\theta\right|>\tau\\ 0, & \text{otherwise,}\end{cases}\end{aligned}\tag{3.9}$$

where τ is a threshold parameter, whose selection is an important problem. A well-adopted non-adaptive threshold is the so-called *universal threshold* $\sqrt{2\ln(n)}\sigma$, where n is the number of elements in the input signal (and σ^2 is the AWGN variance).

The development of data-adaptive thresholds and shrinkage functions has been and still is a very active research topic. The *SureShrink* proposed in [DJ95] uses an adaptive threshold that minimizes the Stein's unbiased risk estimate at each decomposition level. Bayesian estimation used with particular prior PDFs of multiscale transform coefficients has found a significant application in the derivation of shrinkage estimators. The Bayesian approaches considered in the previous section can be interpreted by some shrinkage rules which depend on the noise model and on the exploited priors. In what follows, we consider approaches that explicitly derive shrinkage operators from problems posed in the Bayesian framework. Among the first such approaches is the wavelet coring [SA96], which uses a generalized Laplacian prior for the coefficients of the steerable pyramid [SF95]. A later development [LL98] showed that wavelet shrinkage can be considered as the minimizer of some particular variational formulation; i.e. this work established an equivalence between these two denoising approaches. The bivariate shrinkage [SS02] exploits inter-scale correlations in wavelet transforms by a bivariate Laplacian prior of the parent-child pair of wavelet coefficients, where a parent coefficient has the same relative spatial location as its child but is at the next coarser scale. The *ProbShrink* estimator proposed in [PP06] exploits both inter- and intra-scale correlations. The recent SURE-LET [LB08] shrinkage estimator exploits inter-scale correlation and was later also extended [YCP08] to exploit intra-scale correlations. The trivariate shrinkage [YZW09] exploits a trivariate Gaussian prior to model both the inter- and intra-scale correlations.

Since shrinkage is the main denoising tool used throughout the thesis, it is worth considering the quadratic risks (i.e. MSE) of the hard- and soft-thresholding shrinkage operators defined above in the simplified scenario of non-adaptive threshold and orthonormal transform. The general formula of the quadratic risk for any

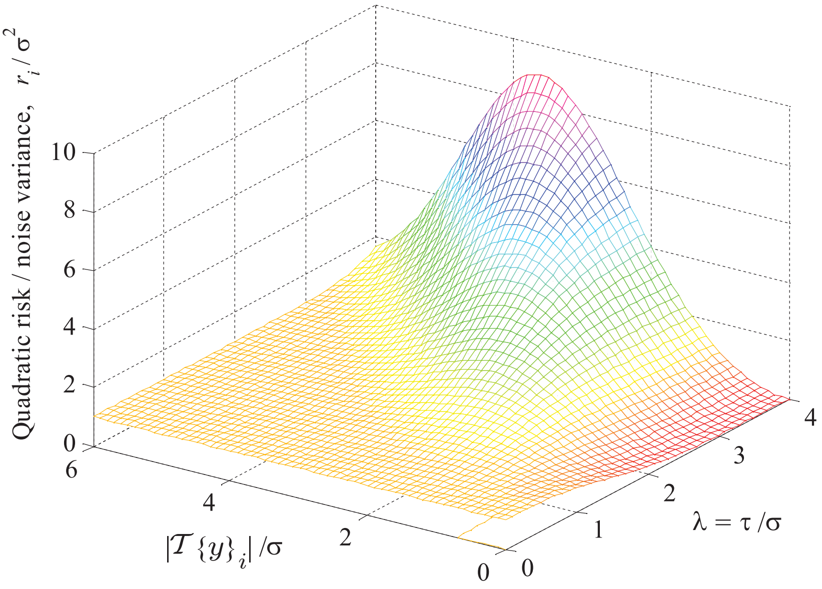


Figure 3.1: Plot of the quadratic risk (scaled by the noise variance σ^2), r_i/σ^2, of a single hard-thresholded transform coefficient $\mathcal{T}\{y\}_i$ (when using a fixed threshold $\tau = \lambda\sigma$) as a function of λ and the signal-to-noise amplitude ratio $|\mathcal{T}\{y\}_i| / \sigma$.

estimator $\hat{y}$ is

$$r_{\hat{y}} = \frac{1}{|X|} \sum_{x \in X} \mathbb{E}\left\{ \left(y(x) - \hat{y}(x)\right)^2 \right\}.$$

Given an orthonormal transform $\mathcal{T}$ and a fixed threshold $\tau = \lambda\sigma$, the quadratic risk of the hard-thresholding is derived as in [MAJ$^+$98],

$$r_{\hat{y}_{\mathrm{HT}}} = \frac{\sigma^2}{|X|} \sum_{i=1}^{|X|} r_{\mathrm{HT}} \left(\frac{|\mathcal{T}\{y\}_i|}{\sigma}, \lambda \right), \tag{3.10}$$

where $\mathcal{T}\{y\}_i$ denotes the i-th (noise-free) transform coefficient (for soft-thresholding, HT should be replaced by ST) and

$$\begin{aligned} r_{\mathrm{HT}}(\mu, \lambda) &= 1 + (\mu^2 - 1)\left[\Phi(\lambda - \mu) - \Phi(-\lambda - \mu)\right] + \\ &\quad + (\lambda - \mu)\,\phi(\lambda - \mu) + (\lambda + \mu)\,\phi(\lambda + \mu) \\ r_{\mathrm{ST}}(\mu, \lambda) &= \left(\mu^2 - \lambda^2 - 1\right)\left[\Phi(\lambda - \mu) - \Phi(-\lambda - \mu)\right] + \\ &\quad + 1 + \lambda^2 - (\lambda - \mu)\,\phi(\lambda + \mu) - (\lambda + \mu)\,\phi(\lambda - \mu), \end{aligned}$$

where $\mu = |\mathcal{T}\{y\}_i| / \sigma$ denotes the signal-to-noise amplitude ratio; ϕ and Φ are the standard normal PDF and CDF, respectively. In order to give an interpretation of the quadratic risk formula, in Fig. 3.1 we present the quadratic risk $r_i =$

$\sigma^2 r_{\mathrm{HT}}\left(\left|\mathcal{T}\left\{y\right\}_i\right|/\sigma, \lambda\right)$ when hard-thresholding a single transform coefficient $\mathcal{T}\left\{y\right\}_i$. We make the following observations based on this figure.

1. When $\mu \gg \lambda$ (i.e., $\left|\mathcal{T}\left\{y\right\}_i\right| \gg \lambda\sigma = \tau$, which means that it is very likely to retain the coefficient), then we observe $r_i \approx \sigma^2$, which means that the risk is the same as if we use the noisy coefficient (whose risk is also σ^2).

2. When λ is relatively large (e.g. $\lambda > 2$) and $\mu < 1$ (meaning that $\left|\mathcal{T}\left\{y\right\}_i\right| < \sigma$), then we observe that $r_i < \sigma^2$, which means that the hard-thresholded coefficient has smaller quadratic risk than the noisy one — that is the case when hard-thresholding benefits us.

3. When $\mu \approx \lambda$ (i.e., $\left|\mathcal{T}\left\{y\right\}_i\right| \approx \lambda\sigma = \tau$, which is the case when the coefficient has approximately the same magnitude as the threshold), then $r_i > \sigma^2$ and thus hard-thresholding is at disadvantage; i.e. its quadratic risk is greater than the risk of the noisy transform coefficient.

Based on the above observations, we can conclude that it is desirable to have as many as possible small-magnitude coefficients $\left|\mathcal{T}\left\{y\right\}_{\cdot}\right| \ll \sigma$ (in a ON-transform spectrum of the image y) and as few as possible high magnitude coefficients $\left|\mathcal{T}\left\{y\right\}_{\cdot}\right| \gg \lambda\sigma$. The ideal case corresponds to having only one non-zero transform coefficient — i.e. the case when one of the basis elements of the ON-transform coincides with the noise-free image y. The above observations exemplify why *sparsity* of an image representation is crucial when performing hard-thresholding, and shrinkage in general.

3.2.3 Adaptive-transform filters

Adaptive transforms, whose basis elements are adaptively computed from the input data (e.g., PCA, ICA, wavelet packets), can be a powerful tool for image denoising. It is reasonable to expect that the adaptivity of the basis elements only improves the sparsity of representing various image details as compared with transforms that have fixed basis elements.

An image denoising method that uses local Principal Component Analysis (PCA) was proposed in [MP03], where it was shown to be highly effective in reconstructing textures and oscillatory patterns. Independent Component Analysis (ICA) bases have also been shown [Hoy99] to be effective for sparse representation of natural-image patches and hence for image denoising. The pointwise shape-adaptive DCT (*P.SA-DCT*) image denoising [FKE07a, Foi07] exploits shrinkage of the SA-DCT spectra of adaptive-shape image neighborhoods. The adaptive-shape neighborhoods, obtained using the directional *LPA-ICI* estimator, capture highly homogeneous signal that can be sparsely represented by the SA-DCT. A recent extension [CM09a] of the steering kernel regression image denoising [TFM07] exploits adaptive transforms in addition to adaptive estimation neighborhoods (with fewer degrees of freedom as compared with the SA-DCT filter).

In the context of adaptive transforms, we wish to mention the *K-SVD* method [EA06], which can be regarded as a significant advance on this topic. The *K-SVD* is a framework for learning a dictionary of basis elements (atoms) that can represent image patches (of relatively small, fixed size) sparsely. Such a dictionary can be trained either from one or more representative noise-free images or alternatively from the input noisy image itself. The training of the dictionary is posed as an optimization problem which is solved by applying singular value decompositions and orthogonal matching pursuit (OMP) [MZ93]. The application of a learnt *K-SVD* dictionary in conjunction with OMP was shown [EA06] to achieve very good denoising results. In the recent work [MSE08], the *K-SVD* method was extended by employing a multiscale representation; i.e., dictionary elements at different scales of a quad-tree decomposition are utilized. This method, which we abbreviate *MS-K-SVD*, is among the state-of-the-art as we show later in Section 4.6.

3.2.4 Nonlocal transform-domain filters

The nonlocal spatial filters considered in Section 3.1.5 were shown (e.g., in [KB06]) to be competitive with the best transform-domain filters. Thus arises the question whether sparse transform-domain representations and nonlocal modeling can be combined so that the strengths of both techniques are preserved. We showed that the answer to this question is positive by proposing the image denoising method in [P1], where the following techniques were exploited:

- grouping: find and stack together similar blocks into 3-D arrays (exploiting nonlocal modeling)
- collaborative filtering: apply a 3-D transform in order to represent sparsely the 3-D arrays, attenuate noise by shrinkage, and invert the 3-D transform,
- aggregation: combine the estimates of the collaborative filtering in case of overlaps.

A generalization of this denoising method was proposed in [P2], which we called *BM3D*. The term *BM3D* stands for Block-Matching and 3-D filtering (block-matching is used to find similar blocks). The good denoising results of the *BM3D* filter [P2] not only confirmed that the joint application of linear transforms and nonlocal modeling is indeed effective, but they inspired applications of this denoising scheme to other image processing applications, which we consider in Chapter 5. In addition, we proposed extensions of the *BM3D* filter to anisotropic neighborhoods [P7] and to data-adaptive PCA representations [P8], both of which we consider in more details in Chapter 4.

The *BM3D* filter has been studied [KFEA10] in the context of nonparametric regression modeling in imaging. It is classified there as a multipoint nonlocal estimator, which uses a multiple-model of the grouped similar blocks. Following this model, the collaborative filtering is derived as a minimizer of a particular variational functional. Other important issues that are considered (separately from collaborative filtering) in this paper are overcomplete representations and the aggregation of individual estimates by weighted averaging. One of the first attempts to provide a theoretical justification of the combination of grouping, collaborative filtering, and aggregation was [KFE07], where mix-distribution observation modeling was considered. A variational formulation with a particular nonlocal image prior was proposed in [Kat09] and was shown to be minimized by an iterative application of the *BM3D* filter. A breakthrough in image deblurring occurred in the later work [KE08] which also exploits the *BM3D* filter in iterative variational minimization with a prior on sparsity.

In addition to the works done by the author of this thesis and by his collaborators, there exist other methods that belong to the class of nonlocal transform-domain filtering. One such development is the grouplets transform [Mal09] which is based on a modified Haar dyadic decomposition and, in analogy with the *BM3D* filter, on a specific grouping procedure. The grouping is realized as a multiscale association field that defines the structural similarities between neighborhoods centered at various spatial locations. It is quite obvious that this association field exploits nonlocal image modeling in a manner that is similar to the nonlocal filters. As suggested in [Mal09], the construction of the association field can be done by block-matching, yet another analogy to the *BM3D* filter. The overall grouplet spectrum is obtained by a weighted Haar dyadic decomposition modified by the multiscale association field. The modification is as follows; the difference (detail) and the averaging (approximation) coefficients are computed respectively by a difference and average between an pixel and its associated one — rather than between spatially neighboring pixels as in the ordinary Haar decomposition. The grouplets can be applied on an image or on a wavelet spectrum of an image, since the wavelet spectrum also contains structural regularities. In the latter case the overall transform was termed *grouping bandlets*. Whereas the few results given in [Mal09] for image denoising by shrinkage of grouping bandlets show good detail reconstruction, the MSE results are not competitive with the state-of-the-art. Application of the grouping bandlets to video denoising was proposed in [MY08] and to inpainting in [MCAFM08].

Another work that we wish to mention in this section is the image representation by nonlocal spectral bases [Pey08]. In particular, this work shows the superiority of a nonlocal manifold modeling of images patches over local one and shows how the models can be exploited for image denoising by thresholding of adapted orthogonal decomposition spectra.

Chapter 4

BM3D: a Nonlocal Transform-Domain Filter

In this chapter we study the *BM3D* filter [P2] and its generalizations to adaptive-shape neighborhoods [P7] and to data-adaptive PCA representations [P8]. In particular, we briefly explain the *BM3D* algorithm and present results that are mostly complementary to the results presented in [P1], [P2], [P7], [P8].

4.1 The basic *BM3D* algorithm

Let us first present the basic algorithm exploited by the *BM3D* filter.

Basic *BM3D* algorithm

- Partition the input image z into (overlapping) blocks $Z_{x \in X}$, where X is the set of these blocks' coordinates. For each block $Z_{x \in X}$ in the partition, do:
  - *grouping*: group blocks that are similar to Z_x into a 3-D array,
  $$\mathbf{Z}_x^{\mathrm{3D}} = \text{grouping}\left(Z_x\right), \tag{4.1}$$
  - *collaborative filtering*: filter the 3-D array $\mathbf{Z}_x^{\mathrm{3D}}$ by applying a 3-D transform $\mathcal{T}$, shrinkage, and the inverse 3-D transform,
  $$\hat{\mathbf{Y}}_x^{\mathrm{3D}} = \mathcal{T}^{-1}\left(\text{shrink}\left(\mathcal{T}\left(\mathbf{Z}_x^{\mathrm{3D}}\right)\right)\right). \tag{4.2}$$
- *Aggregation*: aggregate all filtered blocks from each group $\hat{\mathbf{Y}}_{x \in X}^{\mathrm{3D}}$ by weighted averaging at locations where they overlap.

The following paragraph and sub-sections give some additional details of the above algorithm, whereas a more detailed study can be found in [P2] or in [P1].

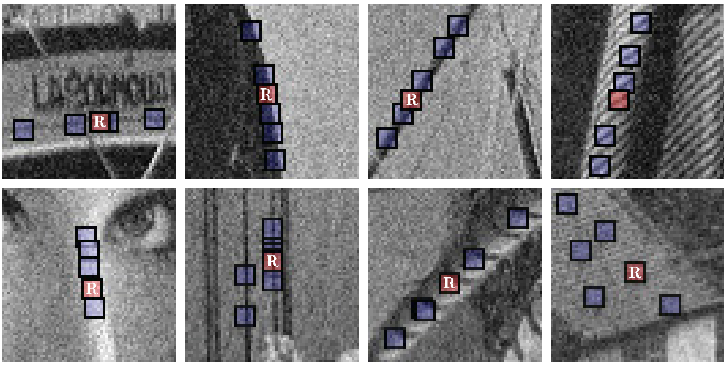


Figure 4.1: Illustration of grouping blocks from noisy natural images corrupted by AWGN with σ=15. Each fragment shows a reference block marked with R and a few of the blocks matched to it.

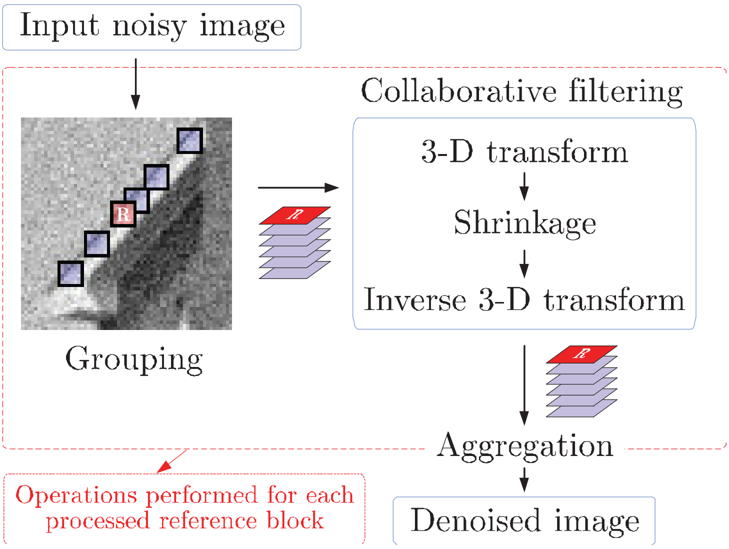


Figure 4.2: Flowchart of the basic *BM3D* algorithm. The block designated with R is the current reference block.

The partition (i.e., selection of the set X) of the input image into blocks can be performed as in [P2] where a raster scan is followed and each next block is taken with a fixed pixel shift from the previous one so that there is some mutual overlap. Such a selection enables two desirable properties; there exists at least one estimate for each image pixel and blocking artifacts are to some degree avoided. Below we discuss the two fundamental procedures of the *BM3D* algorithm, the grouping and the collaborative filtering. In the following sub-sections, the currently processed block is fixed as Z_x and we further denote it as *reference block*.

4.1.1 Grouping

The grouping procedure incorporates the nonlocal image modeling by finding blocks that are similar to the given reference block, Z_x, and stacking them together to form a 3-D array, $\mathbf{Z}_x^{\mathrm{3D}}$. Before we proceed, we would like to define *block similarity*, a term that is very frequently used not only in this section but in the rest of the thesis. We define the block similarity between two blocks as the inverse of a l_p-distance between these blocks, where $p = 2$ is assumed if p is not explicitly specified. When computed from (non-overlapping) blocks corrupted by AWGN, then the block-distance is also noisy with variance defined by Eq. (3) in [P2].

A natural question arises: what is the advantage of grouping similar blocks? The motivation for doing the grouping procedure is the induction of high correlation across the third dimension of the 3-D array $\mathbf{Z}_x^{\mathrm{3D}}$ (i.e. between the grouped blocks). This correlation can then be exploited to improve the estimation of the noise-free image underlying the AWGN. Furthermore, one may also contend that since the grouped blocks are already present in the image, a method such as a transform applied on the whole image can take advantage of their similarity without bothering to do any grouping. Whereas this is possible for highly structured images, for natural images in general, the diversity of details is so great that e.g. a fixed linear transform cannot achieve good sparsity for each of the extremely many possible configurations of the image details. The grouping procedure is a good solution for that problem — it exploits structural similarities of possibly nontrivial details to obtain a collection of similar image patches of relatively small size (e.g. 8×8 patches were used in [P2]). Its strength is in detecting structures in the image, where they are otherwise not easy to find. This is well illustrated in Fig. 4.1 where a few examples of grouping are given. In particular, a good example is the second image on the bottom row of this figure, which shows how blocks that belong to two different parallel edges are combined in a single group. The order in which the found similar blocks are stacked in $\mathbf{Z}_x^{\mathrm{3D}}$ was empirically found in [P2] to affect only marginally the denoising performance, and not in a consistent manner over the set of test images.

Another question is how to find the similar blocks. In [P1] we proposed to apply block-matching to compute block distances between the reference block Z_x and other blocks that belong to a search neighborhood of fixed size centered at x. The decision which blocks to group in $\mathbf{Z}_x^{\mathrm{3D}}$ is based on whether their distances are smaller than a predefined threshold and, in addition, on a restriction of the maximum number of grouped blocks, denoted by N_2 in [P1]. The latter means that at most N_2 blocks with the smallest block-distances to Z_x are grouped. An illustration of some particular groupings is given in Fig. 4.1.

4.1.2 Collaborative filtering

As we stated in the previous sub-section, the correlation induced in $\mathbf{Z}_x^{\mathrm{3D}}$ by the grouping can be exploited to improve the estimation. One possible approach to exploit this correlation, exemplified by the nonlocal spatial filters, is to perform a weighted averaging of the central pixels of all grouped blocks, where the weights depend on the blocks' similarity with the reference block. The result of this averaging is an estimate of the central pixel of the reference block. While this approach can work well for images with high degree of structural similarities (such as textures), it is outperformed in terms of both detail preservation and PSNR by transform-domain methods (such as the *BLS-GSM* [PSWS03]).

Naturally, arises the question: can sparsifying transforms be combined with nonlocal modeling so that both techniques complement each other?

A positive answer to the above question was first given in [P1] with the development of what we call collaborative filtering. The collaborative filtering used by the *BM3D* algorithm is a procedure that filters jointly a group of similar blocks and produces estimates for each one of them. The name *collaborative* stems from the collaboration of individual grouped blocks in the filtering of a whole group. This collaboration is essentially achieved by exploiting both similarities between grouped image blocks and similarities within each block. As discussed earlier in Section 3.2,

- linear transforms can be particularly effective in obtaining sparse representations when there exist correlations in the input signal and
- shrinkage is a powerful denoising tool that exploits sparse representations.

In line with these observations, we proposed as early as [P1] a collaborative filtering that comprises forward sparsifying 3-D transform $\mathcal{T}$, shrinkage (denoted shrink), and inverse 3-D transform $\mathcal{T}^{-1}$. A sparse representation after applying the 3-D transform $\mathcal{T}$ enables the shrinkage to effectively attenuate noise. An evaluation of various separable 3-D transforms was done in [P2], which showed that the MSE performance differences are not significant as long as the transform applied across the third (temporal) dimension of a group has a DC basis element. This result can be explained by the fact that the DC basis element (of the transform applied across the third dimension of a group) is the one that incorporates the similarity between the grouped blocks. In particular, the 3-D transforms that were used to generate the results in [P2] are separable compositions of 1-D Haar dyadic decomposition (across the third dimension of a group) and either a 2-D dyadic biorthogonal wavelet decomposition or the 2-D DCT. The shrinkage operators exploited in [P2] are either hard-thresholding or empirical Wiener filtering. The former uses the non-adaptive universal threshold and the latter requires preliminary estimates of the magnitudes of all 3-D transform coefficients, the computation of which is discussed in Section 4.2. The output of the collaborative filtering is essentially a 3-D array, $\hat{\mathbf{Y}}_x^{\mathrm{3D}}$, that contains an estimate of each pixel in each of the grouped blocks.

4.1.3 Aggregation

After performing grouping and collaborative filtering for all of the reference blocks $Z_{x\in X}$, we obtain the filtered groups $\hat{\mathbf{Y}}^{\mathrm{3D}}_{x\in X}$. Each group $\hat{\mathbf{Y}}^{\mathrm{3D}}_{x}$ contains an estimate of its corresponding reference block as well as estimates of each grouped block. Overlaps between these estimated blocks are practically inevitable, which means that $\hat{\mathbf{Y}}^{\mathrm{3D}}_{x\in X}$ is a redundant representation of the denoised image. Thus arises the question how should the obtained estimates be combined to form a single estimate of the noise-free image. A very simple solution is, e.g., to average all estimated blocks that overlap at a given pixel location. In [P1], we followed a more elaborate and well established (see [KFEA10]) weighted averaging with weights

$$w_x = \sigma^{-2} \left\| \frac{\mathrm{shrink}\left(\mathcal{T}\left(\mathbf{Z}^{\mathrm{3D}}_x\right)\right)}{\mathcal{T}\left(\mathbf{Z}^{\mathrm{3D}}_x\right)} \right\|_2^{-2}, \tag{4.3}$$

where divisions are performed elementwise and a division by zero is assumed zero; this equation is equivalent to Eqs. (10) and (11) from [P2], with shrink being respectively hard-thresholding from Eq. (3.9) or empirical Wiener filtering from Eq. (3.8). There are a few important properties of these weights.

- The weights are approximately[1] inversely proportional to the residual noise variance in each of the groups.
- The weighted average is the maximum likelihood estimate of the denoised pixel provided that the averaged pixel-estimates are unbiased, independent and normally distributed, and their variances are the inverse of their weights.
- Another interpretation of the proposed weights is that sparser 3-D transform-domain representations are given greater weights, i.e. manifested by a stronger attenuation of the spectrum by the shrinkage.

It is also worth noting that the very same weight w_x is assigned to each of the estimates contained in $\hat{\mathbf{Y}}^{\mathrm{3D}}_{x}$, including the reference block's estimated pixels. That means that there is no preference given for the reference block nor is there any preference based on the block-distance to the reference. In addition, in [P2] we proposed to further scale w_x across each block so that the pixels at the border of the block are given smaller weights and the central pixel is given the greatest weight; this was realized with a Kaiser window.

Weighted averaging (as well as nonlocal image modeling) is exploited by both the *BM3D* filter and the nonlocal spatial filters. Therefore, it is worth discussing what are the fundamental differences between these weighted averaging schemes. Indeed, the *BM3D* filter performs the actual noise attenuation by the shrinkage used in the collaborative filtering, which produces estimates for all pixels in each

---

[1] The word *approximately* is used since equality holds only if the grouped blocks are non-overlapping and the shrinkage does not depend on the input noisy image. If some of the grouped blocks overlap, which is very likely, then the noise in $\mathcal{T}\left(\mathbf{Z}^{\mathrm{3D}}_x\right)$ is correlated and taking into account this correlation involves computationally very expensive procedures of detecting the overlaps and computing the variance of each transform coefficient. In addition, both the hard-thresholding and the empirical Wiener filtering (used by the *BM3D* filter) do depend on the input data.

grouped block. Subsequently, weighted averaging is used only to combine estimates that overlap at a given pixel location, if there exist such overlaps at all. On the other hand, the nonlocal spatial filters solely rely on weighted averaging of noisy pixels to perform denoising, where the weights depend on similarities between neighborhoods centered at the averaged pixels and the neighborhood centered around the estimated pixel. That is, the weighted averaging is the primary denoising tool of the nonlocal spatial filters — which is not the case with the *BM3D* filter.

4.2 Two-step implementation of the *BM3D* filter

The overall *BM3D* filter proposed in [P2] exploits the basic *BM3D* algorithm described in Section 4.1 in two successive steps (illustrated in Fig. 3 of [P2]). In the first step, the grouping is done by block-matching in the input noisy image and the collaborative filtering uses hard-thresholding shrinkage. In the second step, the grouping is done by block-matching in the estimated image from the first step and the collaborative filtering exploits empirical Wiener filtering, which utilizes the estimated image from the first step to determine the shrinkage coefficients, given in Eq. (8) in [P2]. In this context, the first step of this algorithm can be viewed as an initialization step. The improvement contributed by the second step can be explained as follows. Performing the block-matching (used by the grouping procedure) in the initial estimate, rather than in the noisy image, means that the noise is not affecting the correctness of the block-matching. In addition, the empirical Wiener filtering is more effective than hard-thresholding when the estimate image from the second iteration is used for providing a reliable estimate of the power spectrum of the 3-D groups. The benefit of applying this two-step estimation has been empirically confirmed in every single experiment performed by the author. The particular selection of hard-thresholding and empirical Wiener filtering for the first and the second steps, respectively, has also been a result of an empirical optimization, where various other shrinkage schemes have been considered.

The complexity of the *BM3D* algorithm is linear with respect to the number of pixels in the image and the approximate number of arithmetic operations per pixel is given in Section IV.A in [P2]. From there it is also evident that the complexity can significantly vary depending on various parameters — allowing a trade-off with the denoising quality. In particular, in [P2] we proposed what we called “Fast Profile” with a set of algorithm parameters that reduce the complexity five times as compared with the default parameters used there, which were called “Normal Profile”. The performance (PSNR) difference between these two profiles is mostly marginal as shown in Fig. 9 in [P2].

4.3 Highly sparse 3-D transform representation

Sparse image representation is a central topic of this thesis. In Section 2.1, sparse representations were linked to the formation of vision in the brain and identified as very important concept in the modeling of natural-image statistics. In fact, the pursue of sparse image representations has long been established as one of the main research problems for the latter. In the introduction (Chapter 3), sparse representations were also identified as a very important tool in image restoration. In particular, the transform-domain filters from Section 3.2 explicitly rely on sparse transform-domain image representation to achieve denoising (e.g., by shrinkage).

In line with the above observations, the *BM3D* algorithm presented in Section 4.1 exploits grouping of similar blocks only in order to improve the sparsity of the representation in the 3-D transform domain. Given the importance of sparsity, it is interesting to see if the sparsity improvement of the *BM3D* can be measured in objective manner.

Various measures of sparsity exist [HR08] and to this date there is no single unanimously adopted such measure. Based on the study made in [HR08], we adopted the Hoyer measure of sparsity, defined originally in [Hoy04] for a discrete signal s with n elements as

$$\text{Hoyer}\left(s\right) = \frac{\sqrt{n} - \left\|s\right\|_1 / \left\|s\right\|_2}{\sqrt{n} - 1}$$

where $\left\|\cdot\right\|_p$ denotes the p-norm. The range of the this measure is $[0, 1]$, where unity (i.e. sparsest possible signal) is attained when there is only one nonzero element in s and zero (i.e. least sparse signal) is attained when all elements of s are equal. The values in between zero and unity are interpolating between these two boundary cases.

We performed experiment where we computed the average Hoyer sparsity of all noise-free groups' 3-D transform spectra. For this experiment we used the luminance of each of the 24 images from the Kodak dataset. For this experiment we restricted the original *BM3D* algorithm to group only blocks that are not mutually overlapping; this was done in order to have a fair comparison as otherwise the sparsity can be seen as a result of grouping the very same data. The main result of this experiment is given in Fig. 4.3, a comparison of the average sparsity of 3-D transform spectra with the average sparsity of 2-D transform spectra. The used 3-D transform was a separable composition of a 2-D DCT (applied on each block) and 1-D full dyadic Haar decomposition (applied across the third dimension). In the case of using only a 2-D transform, we applied only the 2-D DCT on each of the grouped blocks (thus skipping the 1-D transform across the third dimension). In order to take into account the effect of noise, the block-matching was performed on images blocks corrupted by AWGN with $\sigma = 15$ — whereas the grouped blocks were essentially noise-free. In addition, the experiments were performed with 8×8 blocks and maximum grouped $N_2 = 16$. A conclusion that can be made from the result given in Fig. 4.3 is that the 3-D transform domain representation is consistently sparser than the 2-D transform representation (by more than a 10 per-cent increase in the Hoyer measure) for each of the 24 considered test images.

4.4 Links to human visual perception

In this section we make some speculations about possible links between the grouping procedure used by the *BM3D* and the human vision. In Section 2.1 we pointed

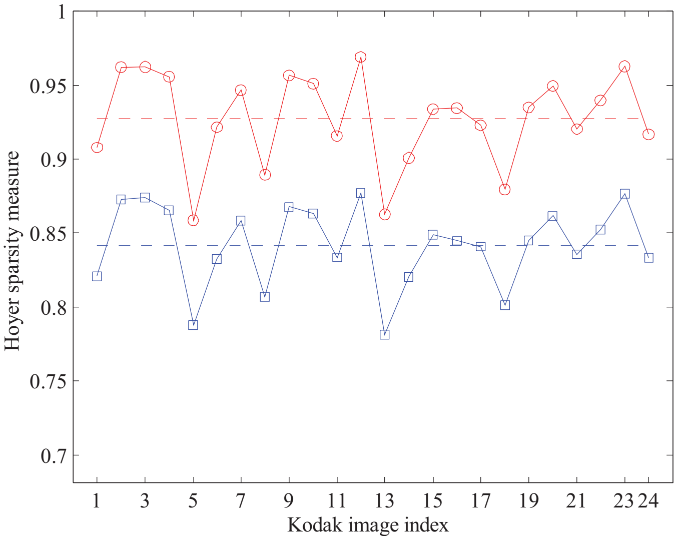


Figure 4.3: Comparison of the average sparsity of 3-D transform spectra (red) with the average sparsity of 2-D transform spectra (blue). The Hoyer sparsity measure is averaged over all noise-free groups' spectra.

out to evidence (prior work) that simple cells in the V1 of the visual cortex respond to simple local oscillations that vary in orientation and scale. An important question that arises is how are the individual responses combined for the purpose of subsequent processing such as recognition. Interestingly, the Gestalt theory [Kof35], which offers a non-constructive answer to this question, had been established in the beginning of the twentieth century — much earlier than the question above had arisen. In the context of vision, the Gestalt theory specifies that *perceptual grouping* occurs in the brain so that a whole (i.e. Gestalt) is perceived rather than a collection of the (various) individual objects that form it. That is, ”the whole is greater than the sum of its parts”. Various principles (laws) of perceptual grouping have been proposed. The principle of similarity says that similar objects tend to be perceived as a whole within a larger context of objects that are not similar. The principle of proximity says that objects that are mutually closer (as compared to other objects) are perceived as a whole. The principle of good continuity says that geometrically continuous (rather than discontinuous) objects are perceived as a whole. Other principles include the principles of symmetry, of closure, of common fate. Whereas the Gestalt theory does not specify how the perceptual grouping is realized in the brain, neuroscientists have filled that gap. Recent works [FH04, Lee03] suggest that complex feed-forward and feed-back connections between neurons in V1 of the visual cortex are responsible for some of the Gestalt groupings. This is a significant advance from the traditional understanding that neurons in V1 are fully characterized by the their individual response fields (i.e. ones that resemble the basis elements of multiscale decompositions).

In the context of Gestalt perceptual grouping, the grouping done by the *BM3D* filter can be seen as incorporating at least: the principle of similarity by grouping blocks that are similar to a reference block; the principle of proximity by grouping only blocks that are in a fixed search neighborhood around the reference block. That is, the grouping done by the BM3D filter exploits these principles in order to capture structural similarities in the image. This behavior is well illustrated in Fig. 4.1, where we show some grouped blocks which follow the structure of underlying objects (such as edges).

4.5 Extensions

In this section we consider extensions of the BM3D filter that attempt to further increase the sparsity of representing images in local 3-D transform domain. We consider extension to anisotropic neighborhoods and to data-adaptive PCA in the following sub-sections.

4.5.1 *SA-BM3D*: extension to anisotropic neighborhoods

The similarity between grouped blocks was shown (Section 4.3) to improve the sparsity in 3-D transform domain, which results in improved denoising results. A significant contribution to this sparsity comes from the intra-block similarity, or to put it more precisely, the ability of the transform to represent sparsely individual blocks. In particular, the *BM3D* filter uses square blocks of fixed size. It is thus interesting to know if employing neighborhoods that are not square can be beneficial for improving the sparsity. The exploitation of neighborhoods of adaptively varying shapes has already been identified as a solution to this problem by the *P.SA-DCT* filter [FKE07a]. This filter uses adaptive-shape neighborhoods, within which the noise-free signal is assumed to be highly homogeneous. This allows sparse representation of these neighborhoods in SA-DCT domain. The shapes of these neighborhoods are adaptively determined from the input image using the directional *LPA-ICI* estimator.

In pursue of improving the *BM3D* filter by increasing the sparsity in the 3-D transform domain, in [P7] we proposed a joint generalization of the *BM3D* and the *P.SA-DCT* filters. This method, termed *SA-BM3D*, exploits grouping of similar adaptive-shape neighborhoods into 3-D groups, which are generalized cylinders with adaptive-shape cross sections. In this manner, both the nonlocal image modeling and the local adaptivity to image details are exploited. The correlation (similarity) between the true-signal components within a group is increased as compared with the *BM3D* where the 3-D groups are parallelograms with fixed square cross-sections. The collaborative filtering of such groups uses a 3-D transform that is a separable composition of the (2-D) SA-DCT and the 1-D Haar dyadic decomposition. The overall algorithm in [P7] utilizes the same two-step procedure as the *BM3D*, where shrinkage by hard-thresholding is used in the first step and empirical Wiener filtering is used in the second step. We showed in [P7] that the developed *SA-BM3D* inherits the strengths of both the *P.SA-DCT* and the *BM3D* filters and outperforms both of them especially in terms of visual quality (i.e. lack or artifacts and preservation of edges).

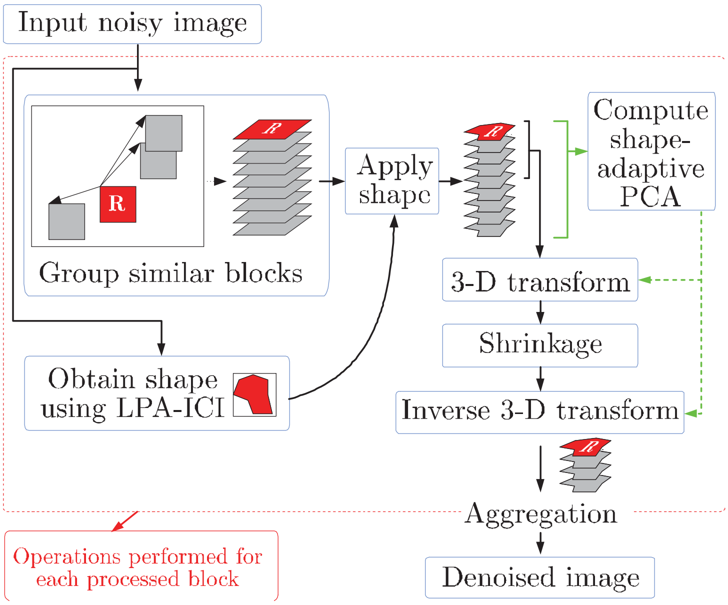


Figure 4.4: Flowchart of the *BM3D-SAPCA* algorithm.

4.5.2 *BM3D-SAPCA*: extension to data-adaptive PCA representations

In the previous section, we presented the *SA-BM3D* filter which uses adaptive-shape neighborhoods in order to increase the spatial correlation and consequently the sparsity as compared with the *BM3D*. However, even though the neighborhoods have adaptive shapes, the SA-DCT basis is still fixed for any given shape; i.e. the basis elements do not adapt to the signal within the grouped neighborhoods.

In order to enable data-adaptivity of the applied shape-adaptive transform, in [P8] we proposed the *BM3D-SAPCA* filter that uses principal component analysis (PCA) as part of the 3-D transform. Given a 3-D group of adaptive-shape image patches, the PCA basis is obtained by eigenvalue decomposition of an empirical second-moments matrix computed from these patches. Subsequently, the PCA basis is trimmed from those principal components whose corresponding eigenvalues are smaller than a threshold that is proportional to the noise variance. Hence, the overall 3-D transform is a separable composition of the PCA (applied on each adaptive-shape neighborhood) and the 1-D fixed Haar dyadic decomposition in the third dimension. In [P8] we showed that the *BM3D-SAPCA* is competitive and outperforms the current best denoising methods, including the *BM3D* filter that it generalizes, particularly in preserving image details and producing very few artifacts.

4.6 Results and discussion

Before we present the experiments with the *BM3D* filter and its extensions, we wish to mention some works that have already evaluated the *BM3D* filter and which are not authored by collaborators of the author of the thesis. A psycho-visual experiment that evaluated images filtered by various image denoising methods was performed by the authors of [VVdWPK06]. This experiment showed that the images denoised by a preliminary version of the *BM3D* filter have on average achieved the highest scores. In another more recent work [CM09b], the authors study the performance bounds of image denoising methods and in particular show that the *BM3D* is among the best performing methods considered therein.

In the sequel of this section we present experimental results that are mostly complementary to the ones already provided in [P2], [P7], and [P8]. All of these results are obtained by applying each of the compared methods on exactly the same noisy images, i.e. using exactly the same artificial AWGN realizations. These results were initially collected in [KFEA10] and in this thesis are used with the permission of the authors of this article. The same ten standard test images used therein are used in this thesis but we consider only the ten best-performing methods from these results; this is done mainly due to space constraints. Thus, the results of methods such as the NL-*means* and the *LPA-ICI* are left out of the comparisons considered in this work; for a more comprehensive comparison that includes these and other works, one should see [KFEA10]. We would like to note that the results of the *BM3D* algorithm can be reproduced by the publicly available Matlab script in [DF].

In Tables 4.1 and 4.2, we compare the current state-of-the-art in image denoising, including the *BM3D*, its generalizations, and also recent developments such as the *MS-K-SVD* [MSE08], the *P.SA-DCT* filter [FKE07a], the *OA-GSM* [HS08], the *SA-FIR* [KB08]. The methods presented in the tables are ordered from left to right by decreasing average PSNR/SSIM result. In addition the results listed in these tables, Fig. 4.7 contains plots with PSNR comparisons of the considered methods. In order to present objective results that are more closely related to the perceived image quality than the PSNR, in Tables 4.3 and 4.4, we present SSIM [WBSS04] results of the compared works. Evaluation of the visual quality can be done in Figs. 4.5 and 4.6, where we compare fragments of images filtered with all of the considered methods.

The *BM3D* filter and its extensions are shown in Tables 4.1, 4.2, 4.3, and 4.4 to outperform the other considered works in terms of PSNR and SSIM. Notably, these good results also correspond to superior detail preservation as can be seen in Figs. 4.5 and 4.6. It is noteworthy that we do not include the results of the recent *K-LLD* method [CM09a]; however, the referred article contains comparisons which show that while the *K-LLD* is competitive with the current best methods, it is outperformed in PSNR by the *BM3D* filter for the considered experiments.

It is noteworthy that the PSNR and the SSIM results given in Tables 4.1 and 4.2 are in mutual agreement in almost all of the cases; i.e., the ranking of the methods based on their PSNR and SSIM results are the same except for few cases.

Table 4.1: Comparison of the PSNR [dB] results of the current state-of-the-art image denoising methods. Methods are ordered from left to right by decreasing average PSNR result.

σ	*BM3D-SAPCA* [P8]	*SA-BM3D* [P7]	*BM3D* [P2]	*MS-K-SVD* [MSE08]	*SA-DCT* [FKE07a]	*K-SVD* [EA06]	*OA-GSM* [HS08]	*SAFIR* [KB08]	*BLS-GSM* [PSWS03]	*TLS* [HP06]
Montage										
5	**41.47**	41.33	41.14	40.96	40.97	40.09	39.14	39.40	39.09	36.18
15	**35.75**	35.47	35.15	34.77	34.91	33.86	33.03	33.27	32.93	32.16
20	**34.17**	33.93	33.61	33.29	33.36	32.34	31.43	31.75	31.32	30.50
25	**32.97**	32.74	32.37	32.02	32.11	31.13	30.17	30.55	30.07	29.66
35	**30.88**	30.80	30.46	30.12	30.12	29.30	28.27	28.73	28.21	28.00
Cameraman										
5	**38.57**	38.41	38.29	38.36	38.15	37.87	37.40	37.31	37.47	37.54
15	**32.37**	32.09	31.91	31.79	31.69	31.47	30.93	30.46	30.91	30.96
20	**30.88**	30.62	30.48	30.43	30.18	30.00	29.45	29.29	29.43	29.47
25	**29.81**	29.58	29.45	29.39	29.11	28.89	28.37	28.47	28.34	28.39
35	**28.17**	28.02	27.93	27.84	27.51	27.32	26.84	27.24	26.78	26.85
Boats										
5	**37.50**	37.30	37.28	37.36	37.14	37.24	37.03	36.25	36.99	37.09
15	**32.30**	32.07	32.14	32.14	31.79	31.77	31.72	31.52	31.72	31.59
20	**31.02**	30.81	30.88	30.86	30.49	30.39	30.41	30.32	30.40	30.25
25	**30.03**	29.84	29.91	29.85	29.48	29.32	29.40	29.36	29.39	29.21
35	**28.52**	28.37	28.43	28.37	27.93	27.71	27.91	27.91	27.92	27.69
Lena										
5	**38.86**	38.75	38.72	38.81	38.54	38.62	38.52	38.00	38.53	38.64
15	**34.43**	34.28	34.27	34.14	33.87	33.71	34.04	33.81	33.93	33.97
20	**33.20**	33.05	33.05	32.87	32.63	32.39	32.81	32.71	32.69	32.69
25	**32.23**	32.07	32.08	31.96	31.66	31.36	31.83	31.82	31.71	31.69
35	**30.72**	30.57	30.56	30.40	30.18	29.71	30.35	30.42	30.22	30.18
House										
5	**40.05**	39.90	39.83	39.91	39.38	39.34	38.93	38.14	38.67	39.15
15	**35.17**	35.06	34.94	34.96	34.14	34.25	33.73	33.90	33.60	33.82
20	**33.90**	33.85	33.77	33.67	32.92	33.10	32.52	32.94	32.35	32.58
25	**32.96**	32.91	32.86	32.71	31.93	32.07	31.54	32.14	31.35	31.60
35	31.38	**31.43**	31.38	31.10	30.39	30.29	29.98	30.75	29.81	30.07
Barbara										
5	**38.38**	38.14	38.31	38.34	37.49	38.11	37.97	37.02	37.81	38.19
15	**33.32**	32.96	33.11	33.00	31.39	32.41	32.25	32.00	31.90	32.55
20	**32.01**	31.67	31.78	31.59	30.00	30.84	30.76	30.60	30.35	31.06
25	**31.00**	30.65	30.72	30.34	28.95	29.58	29.58	29.39	29.15	29.89
35	**29.35**	28.95	28.98	28.63	27.35	27.70	27.81	27.45	27.35	28.12

Table 4.2: Comparison of the PSNR [dB] results of the current state-of-the-art image denoising methods. The last five rows contain the average PSNR result over all 10 test images considered in this table and in Table 4.1. The methods are ordered from left to right by decreasing average PSNR results.

σ	*BM3D-SAPCA* P8	*SA-BM3D* P7	*BM3D* P2	*MS-K-SVD* [MSE08]	*SA-DCT* [FKE07a]	*K-SVD* [EA06]	*OA-GSM* [HS08]	*SAFIR* [KB08]	*BLS-GSM* [PSWS03]	*TLS* [HP06]
Peppers										
5	**38.34**	38.21	38.12	38.22	37.99	37.79	37.31	37.22	37.31	37.22
15	**32.95**	32.68	32.70	32.45	32.45	32.20	31.82	32.09	31.82	32.09
20	**31.55**	31.22	31.29	31.07	31.04	30.80	30.40	30.81	30.40	30.81
25	**30.43**	30.08	30.16	30.05	29.92	29.67	29.29	29.79	29.29	29.79
35	**28.74**	28.35	28.52	28.37	28.27	28.04	27.64	28.24	27.64	28.24
Couple										
5	**37.63**	37.56	37.52	37.57	37.36	37.33	37.13	36.79	37.13	36.79
15	**32.24**	32.14	32.11	31.99	31.78	31.47	31.50	31.50	31.50	31.50
20	**30.88**	30.79	30.76	30.61	30.39	30.02	30.14	30.19	30.14	30.19
25	**29.82**	29.73	29.72	29.55	29.32	28.88	29.11	29.15	29.11	29.15
35	**28.23**	28.14	28.15	28.00	27.71	27.09	27.61	27.55	27.61	27.55
Hill										
5	**37.30**	37.16	37.14	37.18	37.03	37.03	36.99	36.54	36.99	36.54
15	**32.05**	31.86	31.86	31.90	31.60	31.47	31.48	31.34	31.48	31.34
20	**30.85**	30.69	30.72	30.70	30.40	30.19	30.28	30.20	30.28	30.20
25	**29.96**	29.82	29.85	29.80	29.50	29.23	29.39	29.35	29.39	29.35
35	**28.62**	28.53	28.56	28.45	28.22	27.79	28.13	28.11	28.13	28.11
Man										
5	**38.03**	37.84	37.82	37.88	37.63	37.53	37.53	37.03	37.53	37.03
15	**32.20**	31.94	31.93	31.86	31.70	31.52	31.66	31.53	31.66	31.53
20	**30.83**	30.59	30.59	30.52	30.32	30.16	30.29	30.25	30.29	30.25
25	**29.81**	29.61	29.62	29.59	29.32	29.08	29.29	29.30	29.29	29.30
35	**28.39**	28.23	28.22	28.17	27.91	27.62	27.83	27.97	27.83	27.97
Average (over 10 test images)										
5	**38.61**	38.46	38.42	38.46	38.17	38.10	37.79	37.37	37.75	37.56
15	**33.28**	33.05	33.01	32.90	32.53	32.41	32.22	32.14	32.14	32.10
20	**31.93**	31.72	31.69	31.56	31.17	31.02	30.85	30.91	30.75	30.71
25	**30.90**	30.70	30.67	30.52	30.13	29.92	29.80	29.93	29.70	29.69
35	**29.30**	29.14	29.12	28.95	28.56	28.26	28.24	28.44	28.14	28.14

Table 4.3: Comparison of the SSIM results (multiplied by 1000) of the current state-of-the-art image denoising methods. The methods are ordered from left to right by decreasing average SSIM results.

σ	*BM3D-SAPCA* P8	*SA-BM3D* P7	*BM3D* P2	*MS-K-SVD* [MSE08]	*SA-DCT* [FKE07a]	*K-SVD* [EA06]	*OA-GSM* [HS08]	*SAFIR* [KB08]	*BLS-GSM* [PSWS03]	*TLS* [HP06]
Montage										
5	982	**983**	982	980	981	971	977	967	974	967
15	**956**	955	954	950	951	927	939	922	932	917
20	**943**	941	940	938	936	910	923	903	916	894
25	**930**	927	926	924	922	895	907	886	901	873
35	**904**	899	896	892	893	869	875	853	872	834
Cameraman										
5	**963**	962	962	962	961	953	959	951	955	953
15	**910**	903	901	900	902	882	894	875	863	872
20	**886**	877	875	878	875	853	864	842	837	838
25	**864**	856	854	857	852	830	837	816	818	809
35	**828**	823	822	819	815	796	796	777	790	762
Boats										
5	**944**	940	939	941	940	937	941	935	921	934
15	**857**	853	854	856	848	841	842	845	835	846
20	**829**	824	826	825	816	809	804	813	803	814
25	**804**	799	801	798	789	780	772	786	775	786
35	**762**	757	759	753	740	732	720	740	728	738
Lena										
5	946	945	944	**947**	944	944	946	942	938	942
15	**898**	896	896	895	891	891	885	891	887	889
20	**881**	878	877	876	872	872	863	873	870	869
25	**865**	861	861	861	855	855	843	857	854	851
35	**837**	831	831	830	825	826	808	829	828	821
House										
5	**960**	958	957	958	955	950	954	947	932	942
15	**899**	897	891	897	882	872	877	869	866	866
20	**876**	**876**	873	871	862	853	860	851	854	846
25	**861**	**861**	859	856	847	838	845	837	844	829
35	**838**	**838**	837	832	822	814	814	811	825	798
Barbara										
5	**966**	965	965	965	963	963	964	962	957	961
15	**926**	923	923	921	910	912	910	909	904	901
20	**909**	906	905	901	886	889	881	883	876	871
25	**894**	888	887	879	862	866	850	856	844	842
35	**861**	850	848	836	811	820	795	807	785	787

Table 4.4: Comparison of the SSIM results (multiplied by 1000) of the current state-of-the-art image denoising methods. The last five rows contain the average SSIM results over all 10 test images considered in this table and in Table 4.3. The methods are ordered from left to right by decreasing average SSIM results.

σ	*BM3D-SAPCA* P8	*SA-BM3D* P7	*BM3D* P2	*MS-K-SVD* [MSE08]	*SA-DCT* [FKE07a]	*K-SVD* [EA06]	*OA-GSM* [HS08]	*SAFIR* [KB08]	*BLS-GSM* [PSWS03]	*TLS* [HP06]
Peppers										
5	**957**	956	956	**957**	955	948	954	948	949	948
15	**907**	**907**	**907**	900	902	883	898	888	893	884
20	**887**	**887**	**887**	882	881	860	876	865	873	858
25	**869**	868	868	863	862	839	855	843	854	834
35	**836**	835	834	826	827	804	819	804	822	792
Couple										
5	**953**	952	951	952	950	947	950	947	945	947
15	**878**	877	877	875	868	855	855	861	856	861
20	**849**	848	848	843	835	819	815	828	822	827
25	**821**	820	820	812	805	787	779	798	791	796
35	**773**	770	771	762	750	731	712	748	735	744
Hill										
5	**945**	943	943	944	943	941	943	941	937	940
15	**846**	839	839	844	832	829	823	827	814	827
20	**809**	803	804	804	792	788	777	788	775	788
25	**779**	774	775	773	759	755	740	755	744	756
35	**730**	726	728	721	709	704	683	706	697	706
Man										
5	**957**	954	954	955	952	949	951	951	948	951
15	**874**	867	867	867	861	855	855	859	853	859
20	**841**	832	833	832	823	819	815	823	816	822
25	**811**	803	805	804	791	789	779	792	785	790
35	**763**	756	758	754	741	740	725	740	739	737
Average (over 10 test images)										
5	**957**	956	955	956	954	950	954	949	946	949
15	**895**	892	891	891	885	875	878	875	870	872
20	**871**	867	867	865	858	847	848	847	844	843
25	**850**	846	846	843	834	823	821	823	821	817
35	**813**	809	808	802	793	784	775	781	782	772

Original

Noisy $\sigma = 25$
PSNR 20.15, SSIM 0.336

BLS-GSM [PSWS03]
PSNR 28.34, SSIM 0.808

TLS [HP06]
PSNR 28.40, SSIM 0.830

SAFIR [KB08]
PSNR 28.47, SSIM 0.818

OA-GSM [HS08]
PSNR 28.37, SSIM 0.815

K-SVD [EA06]
PSNR 28.89, SSIM 0.837

P.SA-DCT [FKE07a]
PSNR 29.11, SSIM 0.851

MS-K-SVD [MSE08]
PSNR 29.39, SSIM 0.857

BM3D [P2]
PSNR 29.45, SSIM 0.854

SA-BM3D [P7]
PSNR 29.58, SSIM 0.855

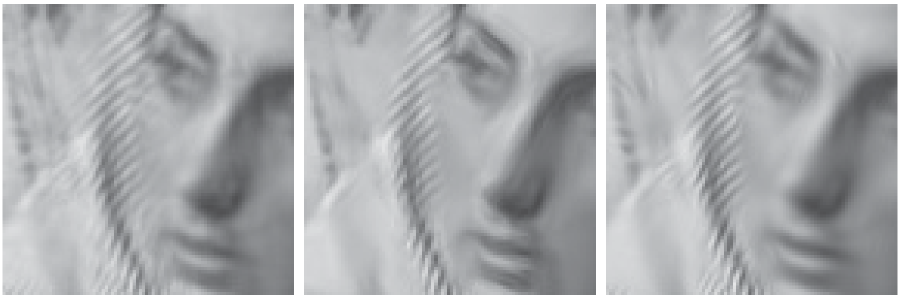
BM3D-SAPCA [P8]
PSNR 29.81, SSIM 0.864

Figure 4.5: Fragment of *Cameraman* filtered by the current state-of-the-art denoising methods; in addition, the noisy (σ=25) and the original fragments are also shown.

Original

Noisy $\sigma = 25$
PSNR 20.17, SSIM 0.401

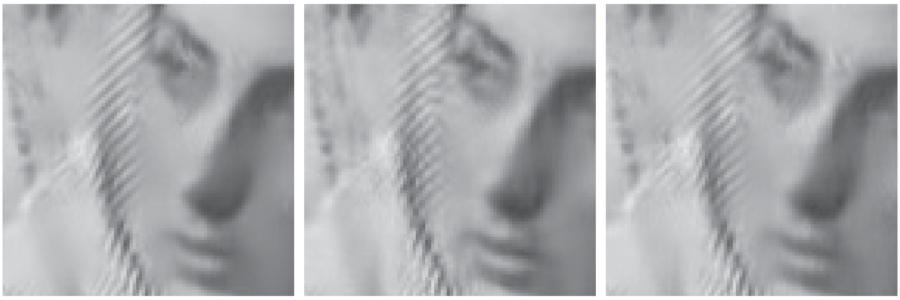
BLS-GSM [PSWS03]
PSNR 29.15, SSIM 0.842



TLS [HP06]
PSNR 29.89, SSIM 0.865

SAFIR [KB08]
PSNR 29.39, SSIM 0.843

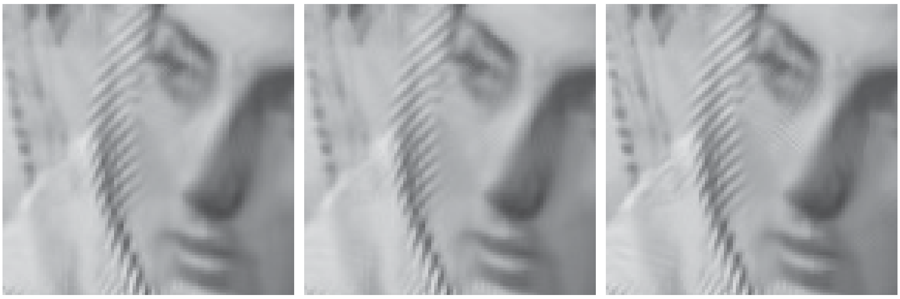
OA-GSM [HS08]
PSNR 29.58, SSIM 0.856



K-SVD [EA06]
PSNR 29.58, SSIM 0.850

P.SA-DCT [FKE07a]
PSNR 28.95, SSIM 0.862

MS-K-SVD [MSE08]
PSNR 30.34, SSIM 0.878



BM3D [P2]
PSNR 30.71, SSIM 0.887

SA-BM3D [P7]
PSNR 30.65, SSIM 0.888

BM3D-SAPCA [P8]
PSNR 31.00, SSIM 0.894

Figure 4.6: Fragment of *Barbara* filtered by the current state-of-the-art denoising methods; in addition, the noisy (σ=25) and the original fragments are also shown.

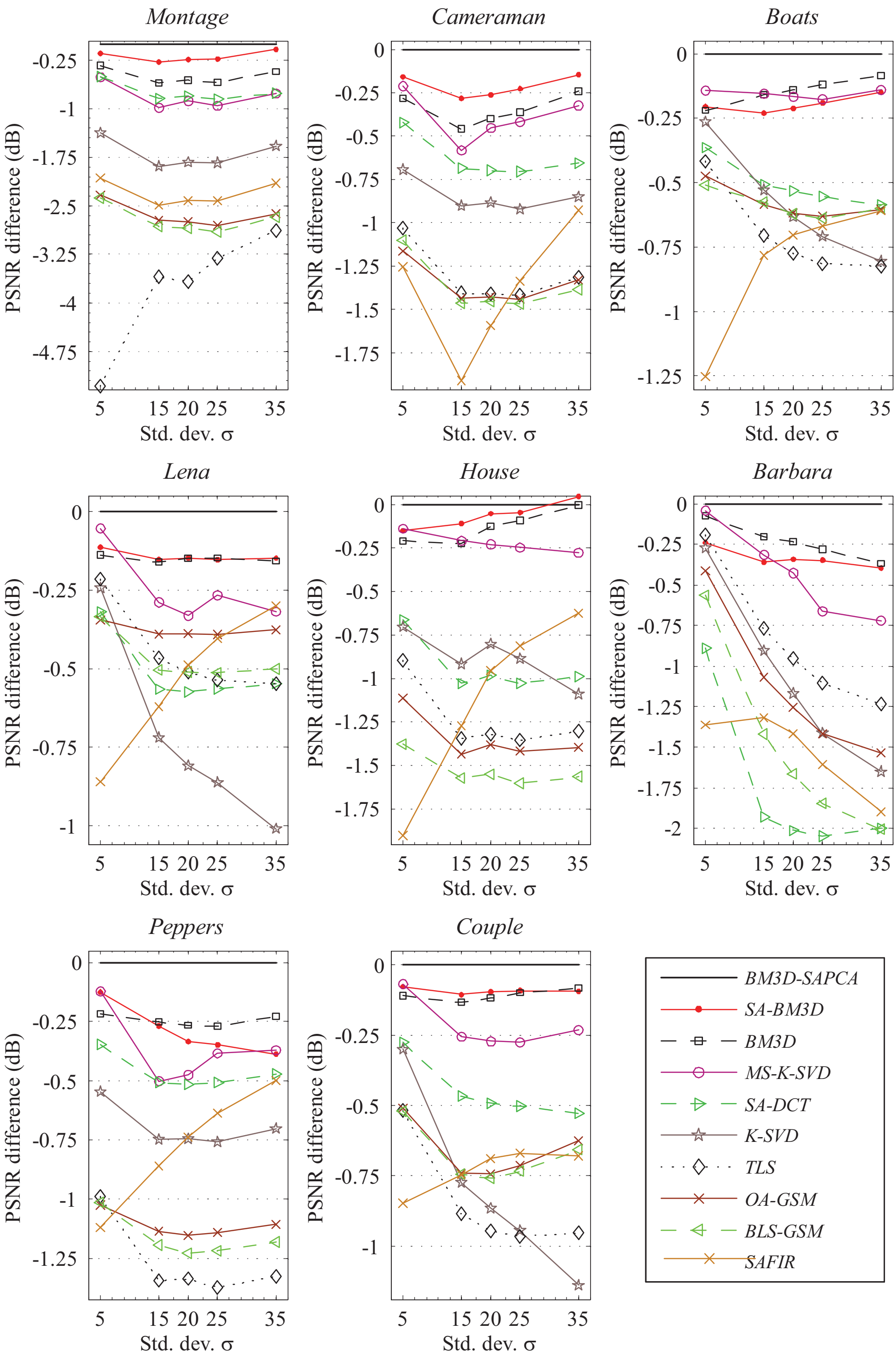


Figure 4.7: Comparison of the PSNR results of the considered state-of-the-art methods for eight test images. For improved visibility, we set the results of the *BM3D-SAPCA* as reference zero and present only the PSNR differences with this particular method, which, in almost all cases, outperforms the others.

Chapter 5

Applications of the *BM3D* Filter

In this chapter we consider application of the *BM3D* filtering to:

- video denoising (Section 5.1),
- raw image denoising (Section 5.2),
- RGB-image denoising (Section 5.3),
- image deblurring (Section 5.4),
- image sharpening (Section 5.5).

These applications, except for the denoising of raw sensor images, are based on the articles that are part of this compound thesis. In addition to these applications, however, there exist quite a few other applications of the *BM3D* filter, some of which are:

- cross-color filtering of noisy raw images [DVF⁺09],
- multiframe raw-image denoising [BF08],
- image and video super-resolution [DFKE08],
- image deblurring via nonlocal variational minimization [KE08],
- noise variance estimation [DF09].

5.1 Video denoising

In this section we consider the denoising of AWGN from video data, a problem that is nearly as significant a research topic as the denoising of AWGN from images. While the various findings about natural image statistics discussed in Section 2.1 are also relevant for individual video frames, the video data has one key property that is of notable importance in video processing — i.e. correlation between successive frames that capture the same scene. That is, video data exhibits strong correlations in both *spatial* (intra-frame) and *temporal* (inter-frame) dimensions. In particular, the authors of [VHR98] have shown that the 3-D spatio-temporal independent components of natural image sequences are oscillations with varying scales and orientations, in analogy with the independent components of images. Similar results were obtained in [Ols03a] for the basis elements of overcomplete spatio-temporal linear representations that maximize sparsity. In the recent study [WL09], image sequences were shown to exhibit a strong prior of temporal smoothness, manifested by low temporal variation of the phases in complex-wavelet transform domain. Not surprisingly, temporal correlations are of fundamental importance to video coding (MPEG 1, 2, and 4, etc.), where predictions from neighboring frames are exploited. These predictions are realized by estimating the motion of small blocks between successive frames, which allows to code motion vectors rather than the blocks themselves and thus significant compression is achieved.

Linear transforms are an important and commonly used tool for obtaining sparse representations of video data, where both the inter- and the intra-frame correlations discussed above are typically exploited. Indeed, most of the current developments in video denoising, such as [BZE06, ZPP06, RAS07, SL03, GSP07], employ some transforms, such as multiscale decompositions, the DCT, etc. Wavelet decompositions in both spatial and temporal dimensions were used in [BZE06], where noise was attenuated by a two-threshold shrinkage of the spatial wavelet spectrum and by hard-thresholding of the temporal wavelet spectrum where the threshold is adaptive with respect to detected motion. The 3-D dual-tree complex-wavelet decomposition used in [SL03] allowed for directional selectivity (in three dimensions) that resulted in sparse representation of objects that are moving across successive frames as well as of objects characterized by intra-frame correlations. Wavelet packets in 3-D spatio-temporal domain, combined with adaptive soft-thresholding were proposed in [RYW04] for video denoising. A wavelet decomposition in spatial domain was used [GSP07] in conjunction with the 1-D DCT in temporal dimension, where the shrinkage threshold is adaptive and captures inter-scale wavelet coefficient dependencies. Inter-frame statistical modeling of wavelet coefficients was employed by [RAS07]. There are also methods that exploit wavelet decomposition only in spatial domain. In [ZPP06], an overcomplete wavelet decomposition is applied on each frame; the noise attenuation is done by motion-adaptive recursive temporal filtering on each subband, followed by adaptive smoothing of the (spatial) wavelet coefficients. The 2-D dual-tree complex wavelet was applied on each frame in [JFW06], motion was estimated using multiresolution block-matching and then noise was attenuated by an adaptive 2-D shrinkage combined with temporal Kalman filtering. Adaptive transforms by training a dictionary of atoms was recently employed in [PE09]. The obtained adaptive dictionary is composed of 3-D atoms, which can represent sparsely spatio-temporal patches. This work is an extension of the *K-SVD* [EA06] filter to image sequences and achieves results that are among the state-of-the-art.

The non-local image modeling has a quite natural application to video processing and denoising since structural similarities between image patches can be sought not only in one frame but in many successive ones. Closely related to the nonlocal filtering scheme, the video-denoising method proposed in [RE05] and later extended in [S2] exploits grouping of similar blocks (restricted to one block per frame) and denoising by hard-thresholding in 3-D DCT domain — in analogy with the *BM3D* nonlocal transform-domain image denoising. Among the first methods to explicitly exploit the nonlocal modeling for video-denoising, however, were the nonlocal spatial estimators [BCM05a, MS05]. The video-denoising counterpart of the *NL-means* [BCM05a] extends the search for similar blocks to spatio-temporal search neighborhoods. A similar denoising scheme [MS05] appeared at approximately the same time, which emphasized that the nonlocal modeling is particularly effective in exploiting correlations that are due to motion of objects across successive frames. Later developments in nonlocal spatial filtering have since been proposed in [BKB07, BCM08, SM08]. The space-time adapted patch-based video restoration [BKB07] proposed weighted averaging of image patches that belong to an adaptively-grown neighborhood around the current pixel. The recent extension [SM08] of this method showed that it can be improved by exploiting higher order regression model of the true signal within each neighborhood.

Extending the previous works [RE05] and [S2], in [P4] we proposed the *VBM3D* video-denoising method, which can be interpreted as a generalization of the *BM3D* filter to video denoising. The *VBM3D* follows the two-step *BM3D* algorithm (Section 4.2) but performs the search for similar blocks not only within a single frame but also within neighboring frames. The resultant search neighborhood is therefore a spatio-temporal (3-D) one. In addition, for the *VBM3D* we proposed a procedure termed *predictive search block-matching* which searches for similar blocks in *adaptive* spatio-temporal neighborhoods, which are adaptively to motion. This procedure reduces significantly the computational cost and enables efficient capturing of similar blocks that follow the motion of objects across frames. As in the *BM3D* filter, collaborative filtering by transform domain shrinkage is applied to produce estimates of all grouped blocks. The *VBM3D* was shown [P4] to be very effective for video denoising and it is still regarded as state-of-the-art, just as its image-denoising counterpart.

5.1.1 Results

We consider experiments performed with the *VBM3D* method, an implementation of which is publicly available [DF] and all results presented here and in [P4] can be reproduced with it. Since in [P4] we already evaluated the PSNR performance of the *VBM3D* filter for a variety of standard image sequences, here we concentrate on comparisons with other works, some of which have been developed after [P4] was published. The results of the 3DWTF [SL03] and the *WRSTF* [ZPP06] methods, as well as the original and noisy test sequences used in our experiments are all courtesy of Dr. Zlokolica, the first author of [ZPP06]. The rest of the results were obtained from the authors of each of the methods. In Fig. 5.1 we compare the PSNR-per-frame for the three image sequences that we consider (i.e., *Salesman*, *Flower garden*, and *Tennis*). These results show that the *VBM3D* outperforms the other works in most of the cases, except for the Tennis sequence where the *3D-K-SVD* is the best performing method for some of the frames but still, the *VBM3D* results are relatively very close (within 0.3 dB difference). Figs. 5.2,

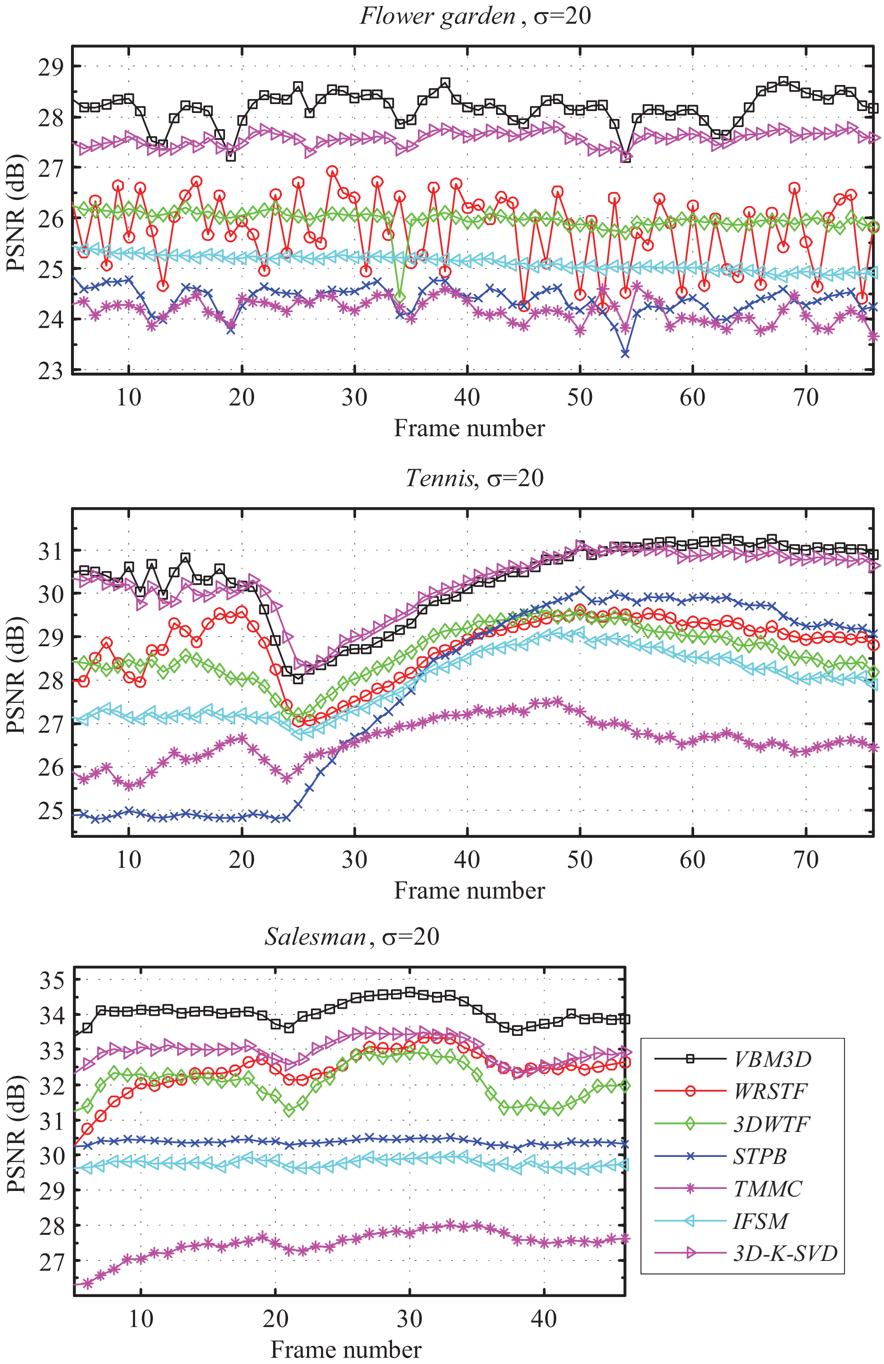


Figure 5.1: PSNR-per-frame comparison between some of the current state-of-the-art video denoising methods for *Flower garden*, *Tennis*, and *Salesman* sequences, corrupted by AWGN with σ=20.

Original fragment

Noisy ($\sigma = 20$)

STPB [BKB07]
PSNR 24.55

TMMC [GAML07]
PSNR 24.16

IFSM [RAS07]
PSNR 25.22

3DWTF [SL03]
PSNR 26.05

WRSTF [ZPP06]
PSNR 26.40

3D-K-SVD [PE09]
PSNR 27.55

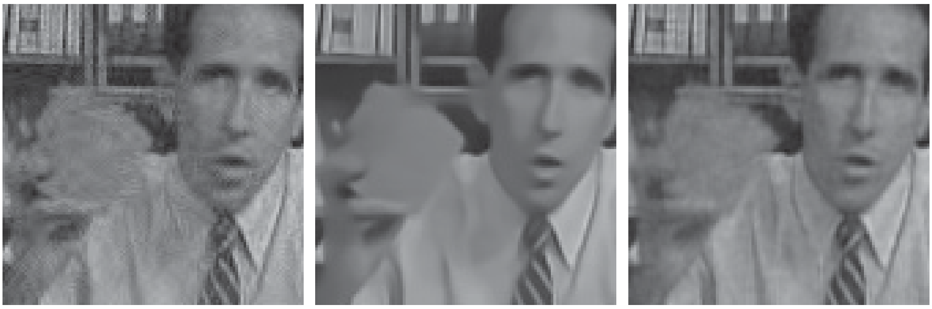
VBM3D [P4]
PSNR 28.38

Figure 5.2: Fragment of *Flower garden* sequence filtered by some of the current state-of-the-art denoising methods; in addition, the noisy (σ=20) and the original fragments are also shown.

Original fragment

Noisy ($\sigma = 20$)

TMMC [GAML07]
PSNR 27.78



IFSM [RAS07]
PSNR 29.90

STPB [BKB07]
PSNR 30.47

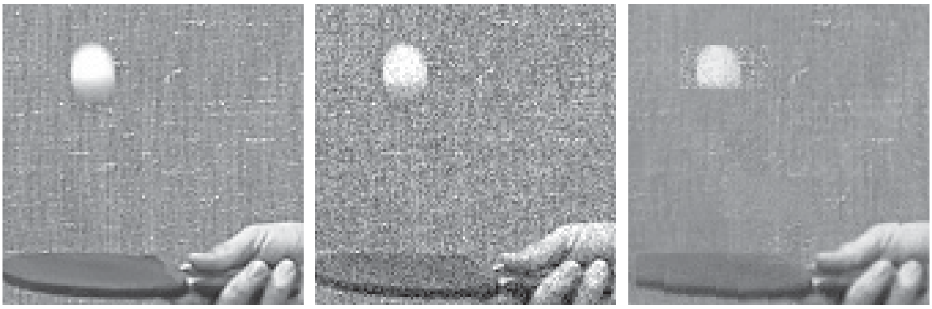
3DWTF [SL03]
PSNR 32.90

WRSTF [ZPP06]
PSNR 33.09

3D-K-SVD [PE09]
PSNR 33.43

VBM3D [P4]
PSNR 34.64

Figure 5.3: Fragment from frame 30 of *Salesman* sequence filtered by some of the current state-of-the-art denoising methods; in addition, the noisy ($\sigma = 20$) and the original fragments are also shown.



Original fragment
Noisy ($\sigma = 20$)
TMMC [GAML07]
PSNR 26.56

IFSM [RAS07]
PSNR 27.30
STPB [BKB07]
PSNR 26.69
3DWTF [SL03]
PSNR 28.03

WRSTF [ZPP06]
PSNR 27.50
3D-K-SVD [PE09]
PSNR 29.00
VBM3D [P4]
PSNR 28.71

Figure 5.4: Fragment from frame 30 of *Tennis* sequence filtered by some of the current state-of-the-art denoising methods; in addition, the noisy (σ=20) and the original fragments are also shown.

5.3, and 5.4 present a visual comparison denoising results for fragments of frame 30 of each of the three considered video sequences. In particular, we notice that both the *3D-K-SVD* and the *VBM3D* achieve better detail preservation for *Flower garden* than the other methods. For *Salesman*, on the other hand, the *VBM3D* is the best performing method especially in reconstructing the hand of the man, which is a fast moving object in this particular sequence. The static background is equally well reconstructed by the *3D-K-SVD*. Finally, for the fragment of *Tennis*, we notice that the *VBM3D* reconstructs well the objects but the background, which resembles a noise-like pattern, is somewhat better reconstructed by the *3D-K-SVD*.

5.2 Practical denoising of digital photographs

As discussed in Section 1.1, the acquisition of digital photographs involves optics, image sensor, and image postprocessing chain. Since denoising is part of the postprocessing, there arises the question at which point should the denoising be performed — as the very first operation or after some other processing has already been done. Let us recall that a noise model of raw sensor images was given in Section 1.1, i.e. uncorrelated signal-dependent (Poissonian-Gaussian) noise given by Eq. (2.4). If a raw image that follows this model undergoes some of the various postprocessing operations in an image postprocessing chain [KWCL06], the noise is no longer uncorrelated. The correlations depend on the particular processing applied and, in general, are not trivially modeled, especially when nonlinear processing is used. Thus, in practical image denoising of digital photographs, the case of denoising applied directly on raw sensor data is preferable since a valid noise model is available.

Expectably, the *BM3D* filter cannot be directly applied on noisy images that contain signal-dependent noise since it assumes AWGN model, for which the noise has constant variance. To take into account the signal-dependent nature of the noise, two practical approaches can be considered. The first and quite natural approach is to use a filter which is specifically designed to attenuate signal-dependent noise, such as [FBKE06, FKE07b]. The second approach is to successively do the following:

- apply a variance-stabilizing transformation (e.g. [Ans48, PS81, Foi09]) on the noisy image, which aims at producing an image with constant variance,
- apply a filter for AWGN,
- inverse the variance stabilizing transformation.

Since the goal of this work is to demonstrate the effectiveness of the *BM3D* filter for denoising raw sensor images, we undertake the second approach mentioned above, where we use the variance stabilizing transformation proposed from [Foi09] and the *BM3D* filter from [P2].

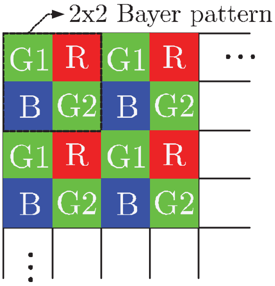


Figure 5.5: The Bayer CFA pattern.

5.2.1 Processing of raw Bayer images

It is worth discussing some properties of raw sensor images before we discuss their denoising. The most widely adopted technique to reproduce colors in digital image acquisition devices [Nak06] is the application of a color filter array (CFA), i.e. an optical filter that specifies the sensitivity of each pixel to certain spectral band in the visible range. The overall array is obtained by a repetition of a particular pattern, a consequence of which is that a raw CFA image resembles a mosaic. Arguably the most widely adopted pattern, the Bayer pattern is formed by a repetition of a 2×2 array that contains four components: two green pixels along a diagonal, one red, and one blue; the Bayer pattern is illustrated in Fig. 5.5. In this work, as the name of this subsection suggests, we consider exclusively denoising of raw Bayer images. Demosaicking is the operation of interpolating the missing color components; a review of recent demosaicking methods can be found in [Pal07].

Since the *BM3D* filter does not assume any particular spatial smoothness, it can be applied directly on raw Bayer images, i.e. images with mosaic appearance where the four components are interleaved according to the pattern from Fig. 5.5. Let us discuss why it is not surprising that the *BM3D* filter can be applied on raw CFA images. Indeed, the regularity of the CFA pattern enables its sparse transform-domain representation — similarly to what happens with uniformly regular areas that are also represented sparsely. Even more important than the regularity of the mosaic structure is the similarity between image patches — a key requirement for the *BM3D* algorithm to effectively attenuate noise. Indeed, due to the repetitive nature of the CFA, the existence of similar image patches in the (unknown) image results in correlations between the corresponding patches that are sampled according the CFA. Whereas we consider only raw Bayer images, the approach presented in this work is not restricted to this particular CFA and, in fact, the structure of the Bayer pattern is not utilized at all by the filtering scheme proposed above.

As mentioned earlier we consider raw images corrupted by the Gaussian-Poissonian noise introduced in Section 2.2.3. In order to apply the *BM3D* filter on such images, we need to perform variance stabilization as in [Foi09], which requires the noise-model parameters to be either a priori known or estimated. For this work we use the estimation procedure from [FTKE08] which estimates the noise parameters from the input noisy image. It is noteworthy that the noise-model estimation from [FTKE08] also models possible clipping of the data due to under- or over-exposures.

A method for denoising raw sensor images that uses the *BM3D* filter in a manner that is very similar to what we propose above has been proposed in [DVF$^+$09]. This method applies a slightly modified *BM3D* filter on a raw Bayer image prior to demosaicking. This slight modification concerns the block-matching being applied only on blocks that are aligned at the same Bayer component. In other words, only blocks that contain identical Bayer pattern are considered; e.g. if the upper-left pixel of a block is a red, then the matching is done only on blocks with red upper-left pixels. The same work presents an interesting experimental result: the *BM3D* filter is more effective when applied directly on a raw Bayer image rather than when applied independently on each of the four Bayer components.

5.2.2 Results

We present results that were obtained with the publicly available Matlab script which implements the variance stabilizing and the declipping transformations from [Foi09], which script uses the (clipped) noise modeling from [FTKE08] and also invokes the standard *BM3D* which is available at [DF]. We consider four raw Bayer images with resolution 5 MPix, the G1 green channels of each are in the upper-left parts of Figs. 5.8, 5.9, 5.6, and 5.7. These images are taken in relatively dimly illuminated conditions, which has resulted in relatively low SNR, which can be observed in the green components of each raw image. In the above mentioned figures, we present the denoised counterparts of the noisy images. In order to better visualize the improvement of the denoising, we have applied an identical postprocessing chain to each of the filtered and the original images and have thus obtained their corresponding RGB-images. Fragments of the RGB images are used to further show the significant improvement after applying the proposed denoising scheme.

5.3 Denoising of color RGB-images

In Section 5.2 we considered denoising of raw Bayer images where a practical signal-dependent noise model was considered. The situation when raw images are not provided and only a noisy RGB image is available is considered in [P3]. There we propose an extension of the *BM3D* filter, termed *CBM3D*, for denoising of AWGN from RGB images. The very first step of the *CBM3D* is a luminance-chrominance color-space transformation (e.g., *YCbCr*, opponent, *YIQ*, etc.) on the input noisy image in order to take advantage of the high correlation between the red, the green, and the blue components. The main peculiarity of the *CBM3D* filter is a technique that in [P3] was termed *grouping constraint on the chrominances*. This grouping constraint means that (for a given reference block) the block-matching is done only in the luminance and based on the it, exactly the same grouping is performed for each of the three channels in order to obtain three groups. That is, the grouping constraint imposes the structural similarities from the luminance to both chrominances. The main justification of this procedure is that the luminance typically has higher SNR than the chrominances and also the luminance contains

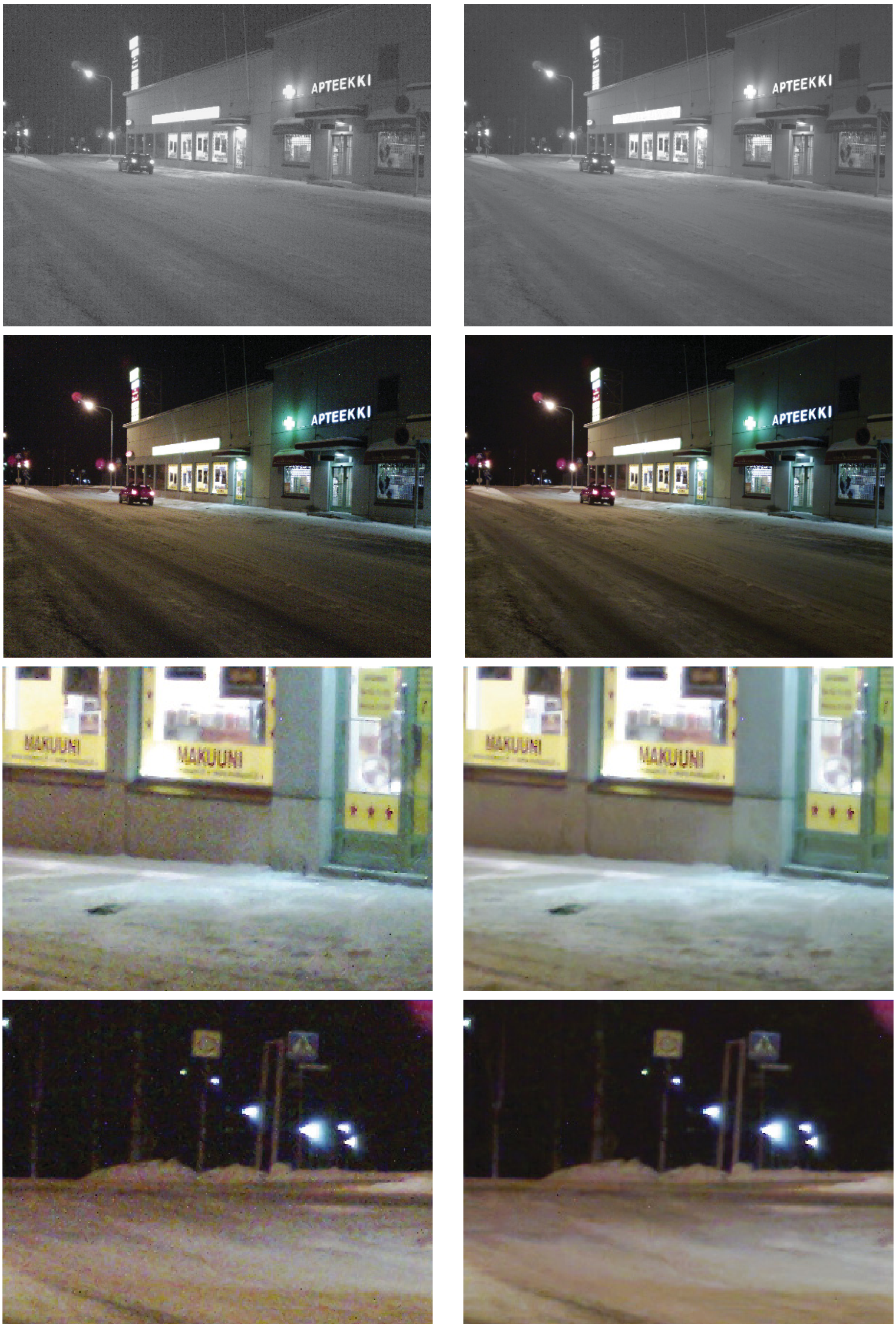


Figure 5.6: Results of applying the *BM3D* filter in conjucntion with the variance stabilizing and declipping transformations from [Foi09] on a 5 MPix raw Bayer image taken in a dim envirnoment using analog gain 8. The top row contains the G1 green channels (gamma corrected for better visibility) of the original and the denoised raw images; the second row contains RGB images obtained by applying an identical image processing chain both to the original and to the denoised raw images. In addition, the last two rows show enlarged fragments of these RGB images.

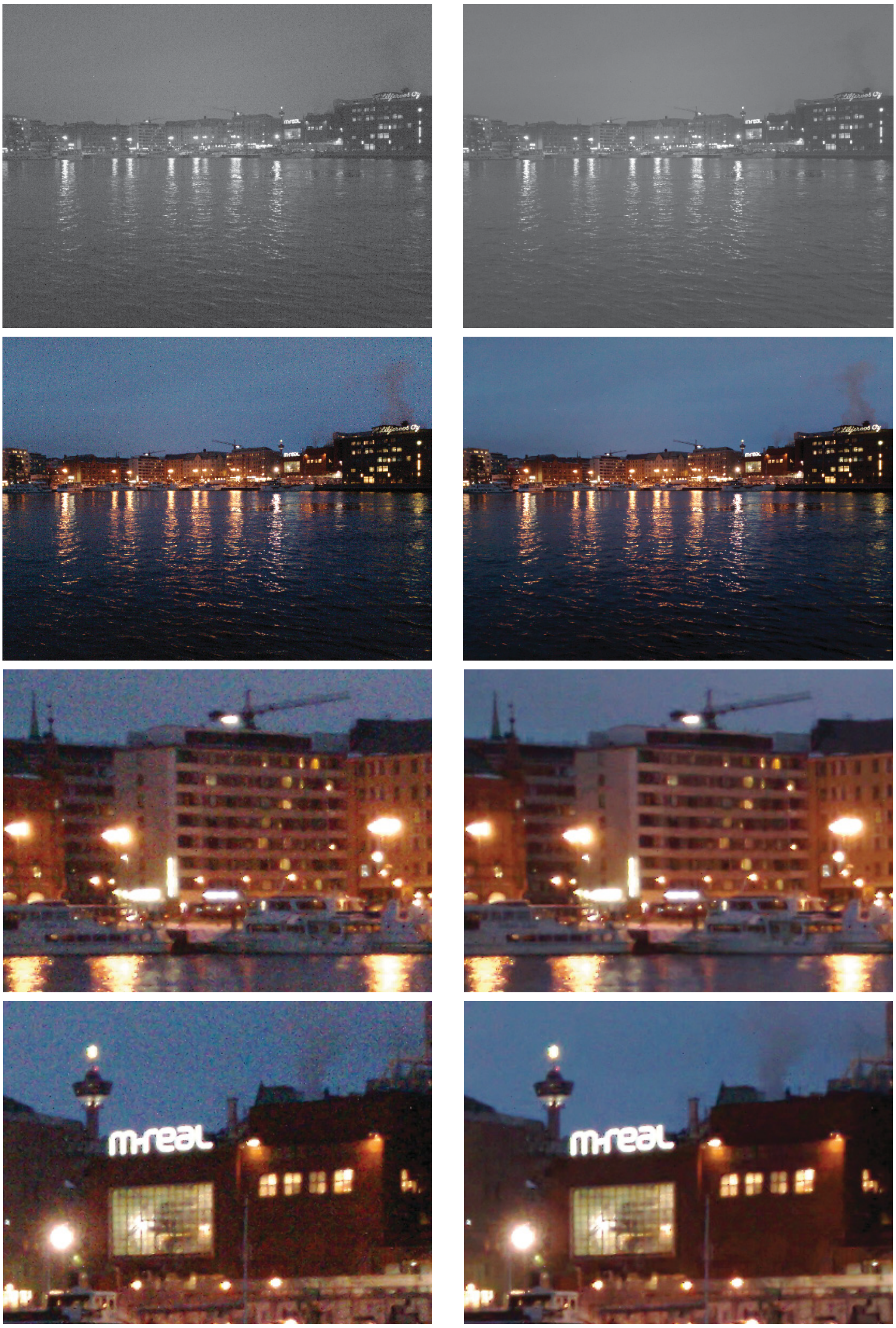


Figure 5.7: Results of applying the *BM3D* filter in conjucntion with the variance stabilizing and declipping transformations from [Foi09] on a 5 MPix raw Bayer image taken in a dim envirnoment using analog gain 8. The top row contains the G1 green channels (gamma corrected for better visibility) of the original and the denoised raw images; the second row contains RGB images obtained by applying an identical image processing chain both to the original and to the denoised raw images. In addition, the last two rows show enlarged fragments of these RGB images.

Figure 5.8: Results of applying the *BM3D* filter in conjucntion with the variance stabilizing and declipping transformations from [Foi09] on a 5 MPix raw Bayer image taken in a dim envirnoment using analog gain 16. The top row contains the G1 green channels (gamma corrected for better visibility) of the original and the denoised raw images; the second row contains RGB images obtained by applying an identical image processing chain both to the original and to the denoised raw images. In addition, the last two rows show enlarged fragments of these RGB images.

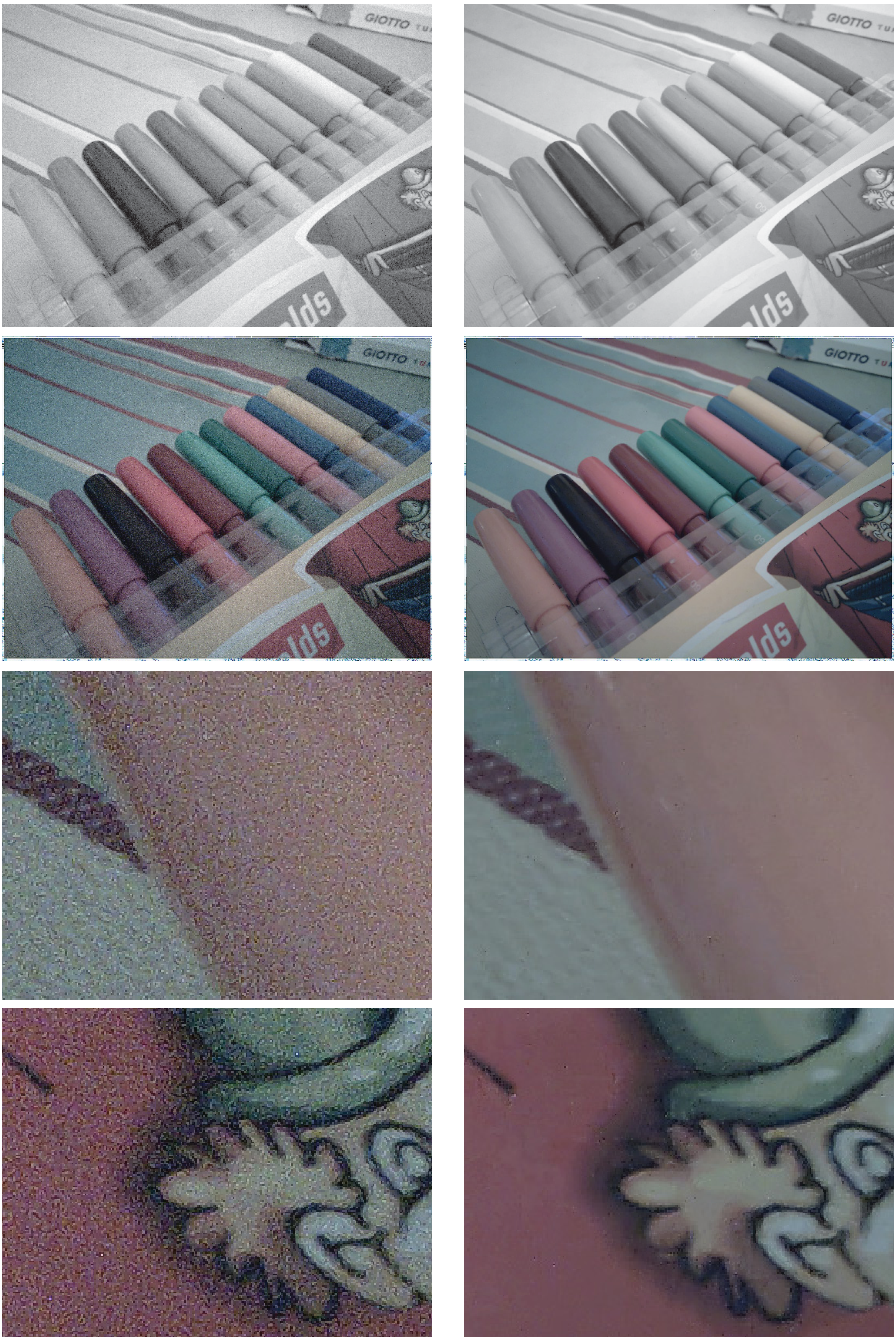


Figure 5.9: Results of applying the *BM3D* filter in conjucntion with the variance stabilizing and declipping transformations from [Foi09] on a 5 MPix raw Bayer image taken in a dim envirnoment using analog gain 10. The top row contains the G1 green channels (gamma corrected for better visibility) of the original and the denoised raw images; the second row contains RGB images obtained by applying an identical image processing chain both to the original and to the denoised raw images. In addition, the last two rows show enlarged fragments of these RGB images.

the most important structures of the image such as edges; further justification is given in [P3]. It is noteworthy that the general approach of applying a constraint on the chrominances was initially proposed in [FKE06, FKE07a].

5.3.1 Results

In this section we show results that complement the ones presented in [P3]. In particular, we consider the RGB images from the Kodak dataset. All results presented here and also in [P3] are obtained by the *CBM3D* Matlab script available in [DF].

Table 5.1 presents PSNR results of the *CBM3D* algorithm applied to RGB-images from the Kodak dataset. We consider noise levels $\sigma = 5, 15, 20, 25, 35$; these values of σ are chosen as a practical range of noise levels, since for $\sigma > 35$, the visual quality is not satisfactory (having PSNR well below 30 dB).

In accordance with the results of the grayscale *BM3D* filter, the proposed *CBM3D* filter achieves state-of-the-art performance in denoising RGB images — in terms of both PSNR and subjective visual quality. This is achieved at a reasonable computational complexity and, in addition, further complexity scalability can be realized by exploiting the complexity/performance trade-off of the *BM3D* presented in P2.

5.4 Image deblurring

Image blurring is a common degradation in imaging, which can have various sources, such as:

- relative motion between scene and camera or object motion during exposure time and
- out-of-focus objects.

In addition to being a degradation, the blur can be induced on purpose by the lens as in the case of extended depth of field imaging systems [DC95]. In all situations, however, it is desirable to inverse the blurring. To define the problem, a model for the blurring is necessary. In many practical cases, blurring can be modeled by a convolution of the true (unknown) image with a fixed point-spread function (PSF) and addition of noise. Such a model is given by

$$z(x) = (y * v)(x) + \eta(x), \tag{5.1}$$

where y is the true (non-degraded) image, v is a blur PSF, η is i.i.d. Gaussian noise with zero mean and variance σ^2, and $x \in X$ is a 2-D coordinate in the image domain X. Deblurring, or non-blind deblurring in particular, is the operation of estimating the blurred signal y when given z, v, and σ. Blind deblurring implies that the PSF v and σ are unknown and need to be estimated; the estimation of the PSF is a significant research topic itself. In this work we consider non-blind deblurring.

AWGN $\sigma = 25$; Output PSNR 31.78 dB

AWGN $\sigma = 25$; Output PSNR: 30.93 dB

Figure 5.10: Noisy and denoised (by the *CBM3D* filter) images 19 and 22 from the Kodak dataset.

AWGN $\sigma = 15$; Output PSNR 34.97 dB

AWGN $\sigma = 15$; Output PSNR: 32.70 dB

Figure 5.11: Noisy and denoised (by the *CBM3D* filter) images 4 and 8 from the Kodak dataset.

Im.	σ				
	5	*15*	*20*	*25*	*35*
1	39.37	32.13	30.40	29.13	27.31
2	40.34	34.58	33.33	32.40	31.07
3	42.47	37.05	35.64	34.54	32.62
4	40.57	34.97	33.66	32.67	31.02
5	39.45	32.70	31.0	29.73	27.61
6	40.00	33.38	31.78	30.59	28.78
7	42.10	36.41	34.87	33.66	31.64
8	39.15	32.70	31.12	29.88	27.82
9	41.18	36.47	35.12	34.06	32.28
10	41.20	36.27	34.90	33.81	31.97
11	40.25	33.88	32.38	31.25	29.52
12	41.34	35.95	34.70	33.76	32.24
13	38.25	30.79	28.98	27.64	25.70
14	39.42	32.76	31.19	30.03	28.24
15	41.01	35.42	34.09	33.08	31.47
16	41.02	34.88	33.42	32.33	30.64
17	40.95	35.41	34.01	32.93	31.16
18	39.01	32.62	31.04	29.83	28.00
19	40.15	34.22	32.79	31.78	30.19
20	41.28	35.83	34.44	33.45	31.84
21	39.96	33.80	32.21	30.99	29.17
22	39.59	33.43	31.98	30.93	29.36
23	41.96	37.14	35.83	34.79	33.09
24	39.77	33.00	31.33	30.09	28.19

Table 5.1: Output PSNR of the *CBM3D* algorithm proposed in [P3] for all 24 RGB-images from the Kodak dataset.

Deblurring is in general an ill-posed problem. Thus, even noise with very small magnitude, such as quantization noise for example, can cause extreme degradations after naive inversion. Regularization, which we discussed in Section 3.1.4, is a well known [Vog02, Han97] and extensively studied tool that can be exploited to alleviate the ill-posedness of the deblurring. It imposes some regularity on the obtained solution; e.g., Tikhonov regularization imposes smoothness, total variation imposes preservation of salient details and smoothness of details with small variation. A huge number of regularization schemes exist [Vog02, Han97].

5.4.1 Regularization by denoising

In this work we are interested in a relatively recent technique [NCB04] that exploits denoising to improve the regularization of image deblurring. In particular,this technique has become a basis of the current best-performing restoration methods [FDKE06, GCMP07, KE08], and [P6]. Indeed, the effectiveness of this deblurring scheme depends greatly on the effectiveness of the exploited denoising filter. Various denoising methods have been employed to improve the regularization of image deblurring. Filtering in multiscale transform domain (e.g., wavelets and pyramid transforms) was shown [NCB99, NCB04, GCMP07] to be effective for this purpose. Another class of denoising techniques used for regularized inversion [KFEA04, KEA05, FDKE06] are based on the *LPA-ICI* [KEA06] which exploits a non-parametric local polynomial fit in anisotropic estimation neighborhoods; the best results of these class of methods are achieved by the *P.SA-DCT* deblurring [FDKE06]. The *BM3D* filter, in particular, has been utilized by [KE08] and by [P6]. In agreement with the state-of-the-art image-denoising results of the *BM3D*, the deblurring methods that use it are also among the current state-of-the-art in image deblurring.

5.4.2 *BM3D* extension for denoising additive colored noise

In [P6], we extended the *BM3D* filter to denoising of additive colored noise, the model of which was given in Section 2.2.2. This extension was used in [P6] to improve the standard Tikhonov regularization by imposing the nonlocal image prior corresponding to the *BM3D* filter. In order to enable attenuation of colored noise, the *BM3D* algorithm was modified in two aspects. First, the book-matching was modified to use the block-distance from Eq. (6) in [P6], i.e. a weighted l^2-norm of the difference between the spectra of two blocks, where the weights are inversely proportional to the variances of the spectrum elements. Second, the shrinkage used for collaborative filtering was modified to take into account the different variances of the 3-D transform coefficients, whereas in the case of white noise, the variance is constant.

5.4.3 Results

Table 1 from [P6] gives a comparison of the proposed work and other prior works. Since the publication date of [P6], there have appeared new better performing methods among which we wish to mention [KE08], where some of the collaborators of the author have utilized the *BM3D* filter in an iterative variational minimization of a particular prior on sparsity.

In this section we present results that complement the ones from [P6] by considering test images from the Kodak dataset (not the standard test images used in [P6]). To carry out these experiments we used the publicly available Matlab function in [DF]. Since we consider grayscale-image deblurring, we used the luminance channel of each test image, which we compute as the average of the red, the green, and the blue channels. We consider the following PSFs, which are among the ones used in [P6]:

- PSF 1: $v(x_1, x_2) = 1/\left(1 + x_1^2 + x_2^2\right), x_1, x_2 = -7, \ldots, 7$,
- PSF 2: v is a 9×9 uniform kernel (boxcar),
- PSF 3: $v = [1\ 4\ 6\ 4\ 1]^T\,[1\ 4\ 6\ 4\ 1]/256$,

All PSFs are normalized so that $\sum v = 1$.

Figs. 5.12 and 5.13 present images that are artificially corrupted according to the model from Eq. (5.1), where the combinations of PSF and σ are chosen as in

[P6]. The reconstructed images show satisfactory detail preservation in accordance with the mentioned significant increase of the SNR. The enlarged fragments give further information about the preservation of fine details.

Notably, the results that we present here and also the ones from [P6] are all obtained without any modification to the fixed regularization parameters. As mentioned in [P6], this is a significant achievement since the proper selection of regularization parameters in standard regularization schemes (such as e.g. Tikhonov regularization) is a challenging problem [Han97] that in general, does no have an established solution. In fact, the regularization parameters depend potentially on the image, on the PSF, and on σ.

5.5 Image sharpening

Image enhancement by sharpening of image details is an application of high practical importance. It is often used as a preprocessing step prior to segmentation in the context of image analysis, classification, pattern recognition. In addition, sharpening can improve the visual perception of images with poor contrast, which is a common task performed in image postprocessing chains. Various methods for image sharpening exist; traditional ones include high-pass linear filtering and elementwise histogram-based transformations such as histogram-equalization, -matching, and -shaping. Current advances [HHKA05, ASP07] in the field exploit image-adaptive filtering and also linear transforms such as wavelet decompositions and trigonometric transforms (DCT, DFT, etc.) to improve the efficiency of the sharpening. Sharpening in transform domain is typically achieved by applying a nonlinear operator to the transform spectrum. Example of a well established such technique is *alpha rooting* [Pra78, AE92], which involves taking the power of the spectrum magnitude according to Eq. (1) in [P5]. Linear transforms have been shown in the recent work [ASP07] to achieve good sharpening when used in conjunction with histogram-based methods. Other relatively recent developments in image enhancement utilize fuzzy logic [Rus02], weighted order-statistic filtering [AB06], and adaptive unsharp masking [PRM00].

Inherent limitation of most sharpening methods is the amplification of noise, [McC80] which is inevitably present in any practical application. In fact, the separation of noise and image details when performing sharpening is a problem that is equivalent to denoising. Thus, sharpening and denoising have often been performed simultaneously, e.g., as in [Rus02, AB06]. From another point of view, for methods that do not take noise into account, the noise amplification can be regarded as a limitation to the amount of applicable sharpening — i.e. beyond a certain point of sharpening, the noise becomes excessively strong.

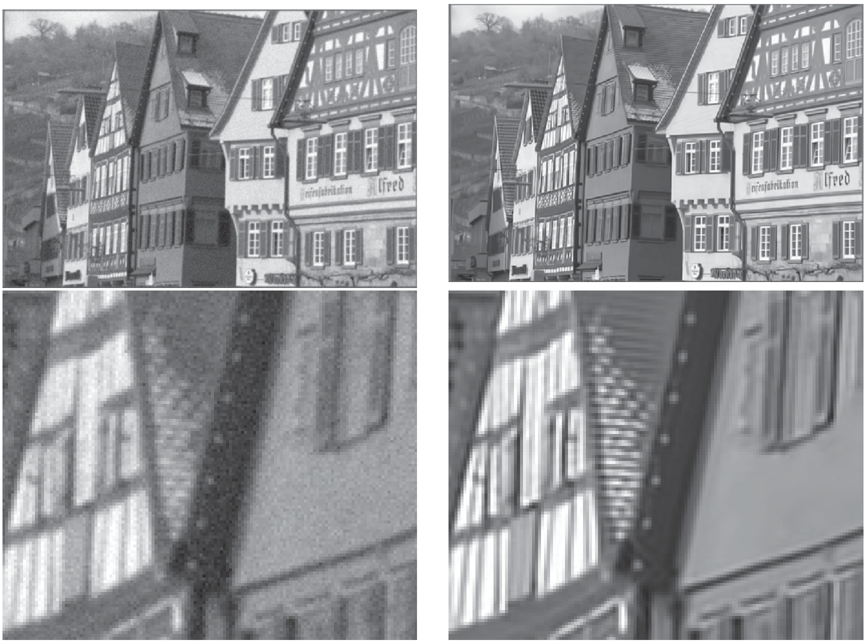
Since the *BM3D* filter has been shown to be very effective for noise attenuation, in [P5] we combined it with alpha-rooting in order to simultaneously sharpen image details and attenuate noise. For noise attenuation, we rely on collaborative hard-thresholding of 3-D spectra, exactly as in the first step of the *BM3D* filter. Subsequently, sharpening is realized by applying the nonlinear alpha-rooting, given in Eq. (1) of [P5], on the thresholded spectra. The simultaneous application of grouping and collaborative filtering allows for both good noise suppression and effective sharpening of even very fine image details.

Blur: PSF1, $\sigma^2 = 2$; Output ISNR 6.28 dB

Blur: PSF1, $\sigma^2 = 8$; Output ISNR 6.74 dB

Figure 5.12: Deblurring results of the *BM3D-DEB* deconvolution method for Kodak images 7 and 24, shown respectively in the upper and lower parts.

Blur: PSF2, $\sigma^2 = 0.308$; Output ISNR 7.77 dB



Blur: PSF3, $\sigma^2 = 49$; Output PSNR 2.84 dB

Figure 5.13: Deblurring results of the BM3D-DEB deconvolution method for Kodak images 5 and 8, shown respectively in the upper and lower parts.

5.5.1 Results

We present results of experiments performed with the implementation of the proposed sharpening available in [DF]. First, in Fig. 5.14 we demonstrate the application of this method on noise-free images, where only their luminances are processed and the chrominances are not modified. The results show a noticeable contrast improvement without any disturbing artifacts such as ringing around edges. In Figs. 5.16 and 5.15, where we consider grayscale input images with relatively mild AWGN (with $\sigma = 5$ and with $\sigma = 15$). We applied the standard *unsharp masking* sharpening on these images in order to demonstrate the noise amplification problem that was mentioned earlier in this section. The results of the proposed joint denoising and sharpening do not suffer from this noise amplification effect and image details are effectively sharpened even in the presence of noise.

Figure 5.14: The proposed image sharpening applied to the noise-free luminance components of Kodak images 16 and 24; the chrominances were not modified. We assumed σ=1 (on the intensity range [0,255]), which corresponds to a very mild noise that is readily enountered in the otherwise noise-free Kodak images.

Figure 5.15: Illustration of image sharpening applied on the luminance of Kodak image 23 with quite mild AWGN (with $\sigma = 5$); this image is given in the top along with an enlarged fragment of it. The second row is the result of applying the standard unsharp mask method; the original image is sharpened but there are notable noise amplification problems. The third row presents the result of the proposed method, which does not contain noise-amplification artifacts and the image appears sharpened.

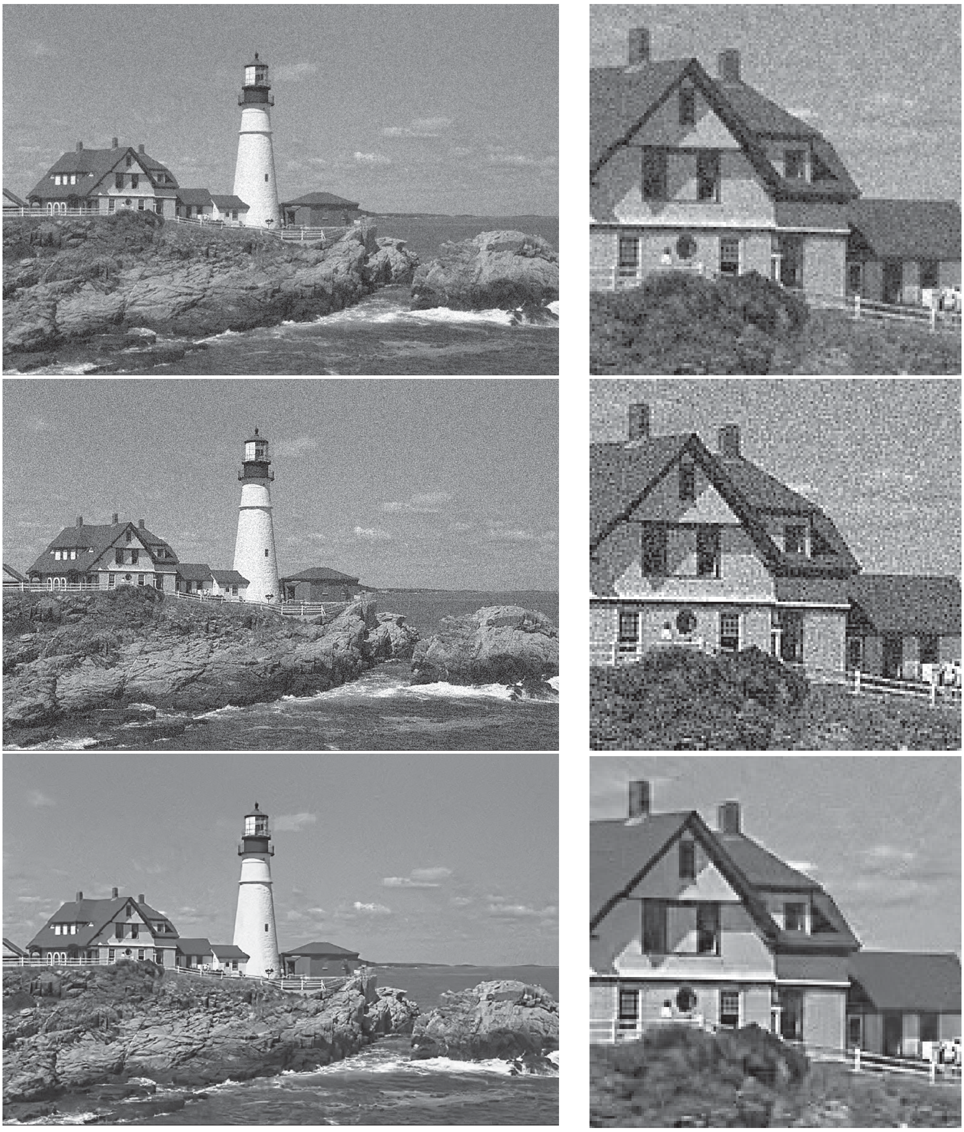


Figure 5.16: Illustration of image sharpening applied on the luminance of Kodak image 21 with quite mild AWGN (with σ=5); this image is given in the top row along with an enlarged fragment of it. The second row contains the result of applying the standard unsharp mask method; the original image is sharpened but there are notable noise amplification problems. The third row presents the result of the proposed method, which does not contain noise amplification artifacts and the image appears sharpened.

Chapter 6

Conclusions

This thesis studies a new class of nonlocal transform-domain filters and their application to various image processing problems. In particular, the *BM3D* filter (Section 4.1, [P1], [P2]) is the main topic of this thesis. The main contribution of this filter is to the problem of obtaining sparse image representations. It exemplifies that nonlocal image modeling can be exploited in conjunction with linear transforms in order to obtain highly sparse transform-domain image representations. The good denoising results of the *BM3D* filter and its successful applications to other problems can be seen as direct consequences of the ability to represent images sparsely. The sparsity improvement of the 3-D transform domain representation used by the *BM3D* filter over the 2-D counterpart is shown in Section 4.3, where the Hoyer sparsity measure is used as an objective measure of sparsity. To further improve the sparsity, the *BM3D* is extended to anisotropic neighborhoods (Section 4.5.1, [P7]) and to data-adaptive PCA transform (Section 4.5.2, [P8]).

In this thesis we have also presented denoising, in general, and the *BM3D* filter in particular, as a fundamental tool in image processing. Whereas we consider exclusively the *BM3D* filter, other denoising filters can, in general, be exploited. Chapter 5 considers a few image-processing applications that benefit from exploiting the *BM3D* filtering scheme. A relatively natural generalization of the *BM3D* filter to video denoising, termed *VBM3D*, was shown to achieve state-of-the-art video-denoising results (Section 5.1, [P4]). The *BM3D* filter in conjunction with a variance-stabilizing transformation achieves very good noise attenuation and detail preservation when applied on raw sensor images (Section 5.2). Denoising of RGB-images by using a luminance-chrominance color space transformation is also identified as a successful application of the *BM3D* filter (Section 5.3, [P3]). Another considered application is image deblurring (Section 5.4, [P6]), where the *BM3D* filter is used to improve the standard Tikhonov regularization. Not only the proposed deblurring method achieves state-of-the-art image deblurring results but it does so by using fixed regularization parameter – independent of either the PSF, the image, or the noise level. The last considered application is joint image sharpening and denoising (Section 5.5, [P5]). It achieves very good sharpening in the presence of noise, which is a particularly challenging task since image sharpening tends to amplify high-frequency noise.

In addition to the applications that are considered in this thesis, the *BM3D* filter is used in other works. A method for estimation of the variance of AWGN in images is proposed in [DF09], where the grouping and the 3-D transform-domain representations are used to effectively separate the noise and image details — where, however, only the statistics of the noise are estimated. An iterative image deblurring via nonlocal variational minimization is proposed in [KE08], which uses a modification of the *BM3D* in each iteration. This method is shown to provide exceptional deblurring results. The *VBM3D* filter is applied for multiframe raw-image denoising [BF08], where a collection of frames roughly corresponds to a video signal (i.e. having correlation between frames). Another application is image- and video-upsampling (super-resolution) considered in [DFKE08].

6.1 Future work

Future work on the *BM3D* filter can be targeted to at least the few directions that we list in the following. The grouping can be modified to enable grouping of image patches with various scales, orientations, shapes. In addition, different block-distance measures can be exploited other than the l^2-distance. The collaborative filtering may be improved by employing an overcomplete transform or even a pre-computed dictionary of atoms as the one obtained by the *K-SVD* [EA06]. The aggregation of the filtered image patches may also benefit from a modification of the averaging weights or from an application of robust weighted averaging (exploiting order statistics). The *BM3D* filter's two-step estimation algorithm may be extended to an iterative scheme where the grouping and the collaborative filtering are improved using results of previous iterations.

We can also foresee some future applications of the *BM3D* filter. One such application can be joint CFA demosaicking and denoising; a significant advance in this direction is already done by the super-resolution method proposed in [DFKE08]. Another application may be HDR composition from a stack of LDR images (e.g. with varying exposures). The ability of the grouping to collect similar blocks (possibly combined with sub-pixel accuracy of the block-matching) may replace the otherwise inevitable registration of the LDR images, which are never perfectly aligned. Sub-pixel accuracy of the block-matching may also be beneficial for video denoising as it is for motion-estimation based video compression.

Bibliography

[AB06] T. C. Aysal and K. E. Barner. Quadratic weighted median filters for edge enhancement of noisy images. *IEEE Trans. Image Process.*, 15(11):3294–3310, Nov. 2006.

[AE92] S. Aghagolzadeh and O. K. Ersoy. Transform image enhancement. *Opt. Eng.*, 31(3):614–626, March 1992.

[AK06] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations (Applied Mathematical Sciences)*. Springer, 2nd edition, 2006.

[Ans48] F. J. Anscombe. The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, 35(3-4):246–254, 1948.

[AR75] N. Ahmed and K. R. Rao. *Orthogonal transforms for digital signal processing*. Springer-Verlag, 1975.

[ASP07] S. Agaian, B. Silver, and K. A. Panetta. Transform coefficient histogram-based image enhancement algorithms using contrast entropy. *IEEE Trans. Image Process.*, 16(3):741–758, March 2007.

[AVT07] S. K. Alexander, E. R. Vrscay, and S. Tsurumi. An examination of the statistical properties of domain-range block matching in fractal image coding. In *Proc. European Signal Process. Conf.*, Poznan, Poland, Sept. 2007.

[BA83] P. J. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.*, 31:532–540, 1983.

[Bar61] H. B. Barlow. *Possible principles underlying the transformations of sensory messages*, pages 217–234. MIT Press, Cambridge, MA, 1961.

[Bar72] H. B. Barlow. Single units and sensation: a neuron doctrine for perceptual psychology. *Perception*, 1(4):371–394, 1972.

[Bar02] D. Barash. A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(6):844–847, 2002.

[BCM05a] A. Buades, B. Coll, and J. M. Morel. Denoising image sequences does not require motion estimation. In *Proc. IEEE Conf. Adv. Video Signal Based Surveil.*, pages 70–74, Palma de Mallorca, Spain, Sept. 2005.

[BCM05b] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multisc. Model. Simul.*, 4(2):490–530, 2005.

[BCM08] A. Buades, B. Coll, and J. M. Morel. Nonlocal image and movie denoising. *Int. J. Comp. Vision*, 76(2):123–139, Feb. 2008.

[BF08] G. Boracchi and A. Foi. Multiframe raw-data denoising based on block-matching and 3-D filtering for low-light imaging and stabilization. In *Proc. Local and Nonlocal Approx. in Image Process.*, Lausanne, Switzerland, Sept. 2008.

[BKB07] J. Boulanger, C. Kervrann, and P. Bouthemy. Space-time adaptation for patch-based image sequence restoration. In *IEEE Trans. Pattern Anal. Machine Inetlligence*, volume 29, pages 1096–1102, June 2007.

[BLIA97] A. J. Blanksby, M. J. Loinaz, D. A. Inglis, and B. D. Ackland. Noise performance of a color cmos photogate image sensor. In *IEEE Int. Electron Devices Meeting Technol. Dig.*, pages 205–208, 1997.

[BM87] G. J. Burton and I. R. Moorhead. Color and spatial structure in natural scenes. *Appl. Opt.*, 26(1):157–170, 1987.

[Bov05] A. C. Bovik. *Handbook of Image and Video Processing.* Elsevier, 2005.

[BS97a] A. J. Bell and T. J. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37:3327–3338, 1997.

[BS97b] R. W. Buccigrossi and E. P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *IEEE Trans. Image Process.*, 8:1688–1701, 1997.

[BS07] J. Bernardo and A. F. M. Smith. *Bayesian Theory (Wiley Series in Probability and Statistics).* Wiley, 2007.

[BZE06] E. J. Balster, Y. F. Zheng, and R. L. Ewing. Combined spatial and temporal domain wavelet shrinkage algorithm for video denoising. *IEEE Trans. Circ. Syst. Video Technol.*, 16(2):220–230, Feb. 2006.

[CD88] W. S. Cleveland and S. J. Devlin. Locally weighted regression: An approach to regression analysis by local fitting. *J. Amer. Stat. Assoc.*, 83(403):596–610, 1988.

[CD95] R. R. Coifman and D. L. Donoho. Translation-invariant de-noising. pages 125–150. Springer-Verlag, 1995.

[CM09a] P. Chatterjee and P. Milanfar. Clustering-Based Denoising With Locally Learned Dictionaries. *IEEE Trans. Image Process.*, 18:1438–1451, July 2009.

[CM09b] P. Chatterjee and P. Milanfar. Is denoising dead? *IEEE Trans. Image Process.*, 2009. Accepted for publication.

[CMF+08] Y. Chen, J. Ma, Q. Feng, L. Luo, P. Shi, and W. Chen. Nonlocal prior bayesian tomographic reconstruction. *J. Math. Imaging Vis.*, 30(2):133–146, 2008.

[CNB98] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Trans. Signal Process.*, 46(4):886–902, 1998.

[CS05] T. F. Chan and J. H. Shen. *Image Processing And Analysis: Variational, PDE, Wavelet, and Stochastic Methods.* SIAM, 2005.

[CW03] L. M. Chalupa and J. S. Werner. *The Visual Neurosciences, 2 Volume Set, (Bradford Books).* The MIT Press, 1 edition, 2003.

[Dau88] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math*, 41:906–966, Feb. 1988.

[DC95] E. R. Dowski and W. T. Cathey. Extended depth of field through wave-front coding. *Appl. Opt.*, 34(11):1859–1866, 1995.

[DC09] V. Dore and M. Cheriet. Robust nl-means filter with optimal pixel-wise smoothing parameter for statistical image denoising. *IEEE Trans. Signal Process.*, 57(5):1703–1716, May 2009.

[DF] K. Dabov and A. Foi. BM3D Filter Matlab Demo Codes. http://www.cs.tut.fi/~foi/GCF-BM3D.

[DF09] A. Danielyan and A. Foi. Noise variance estimation in non-local transform domain. In *Proc. Local and Nonlocal Approx. in Image Process.*, pages 46–55, Tuusula, Finland, Aug. 2009.

[DFKE08] A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian. Image and video super-resolution via spatially adaptive block-matching filtering. In *Proc. Local and Nonlocal Approx. in Image Process.*, Lausanne, Switzerland, Sept. 2008.

[DJ94] D. L. Donoho and Jain M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, Sept. 1994.

[DJ95] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Stat. Assoc.*, 90:1200–24, 1995.

[Don95] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41(3):613–627, May 1995.

[Don98] D. L. Donoho. Ridge functions and orthonormal ridgelets. *J. Approx. Theory*, 111:143–179, 1998.

[Don99] D. L. Donoho. Wedgelets: nearly minimax estimation of edges. *Annals Stat.*, 27:859–897, 1999.

[DV05] M. N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *IEEE Trans. Image Process.*, 14(12):2091–2106, 2005.

[DVF$^+$09] A. Danielyan, M. Vehviläinen, A. Foi, V. Katkovnik, and K. Egiazarian. Cross-color bm3d filtering of noisy raw data. In *Proc. Local and Nonlocal Approx. in Image Process.*, pages 125–129, Tuusula, Finland, Aug. 2009.

[EA06] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.*, 15(12):3736–3745, Dec. 2006.

[ELB08] A. Elmoataz, O. Lezoray, and S. Bougleux. Nonlocal Discrete Regularization on Weighted Graphs: a framework for Image and Manifold Processing. *IEEE Trans. Image Process.*, 17(7):1047–1060, 2008.

[ET96] C. C. Enz and G. C. Temes. Circuit techniques for reducing the effects of op-amp imperfections: Autozeroing, correlated double sampling, and chopper stabilization. *Proc. IEEE*, 84:1584–1614, Nov. 1996.

[FAKE07] A. Foi, S. Alenius, V. Katkovnik, and K. Egiazarian. Noise measurement for raw-data of digital imaging sensors by automatic segmentation of non-uniform targets. *IEEE Sensors J.*, 7(10):1456–1461, Oct. 2007.

[FBKE06] A. Foi, R. Bilcu, V. Katkovnik, and K. Egiazarian. Adaptive-size block transforms for signal-dependent noise removal. In *Proc. NORSIG*, Reykjavik, Iceland, 2006.

[FDKE06] A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian. Shape-Adaptive DCT for denoising and image reconstruction. In *Proc. SPIE Electronic Imaging: Algorithms and Systems V*, volume 6064A-18, San Jose, CA, USA, Jan. 2006.

[FG96] J. Fan and I. Gijbels. *Local Polynomial Modelling and Its Applications.* Chapman and Hall, 1996.

[FH04] D. J. Field and A. Hayes. Contour integration and the lateral connections of V1 neurons. In L. M. Chalupa and J. S. Werner, editors, *The Visual Neurosciences.* MIT Press, 2004.

[Fie87] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, 4(12):2379–2394, 1987.

[Fie93] D. J. Field. *Scale-invariance and Self-similar 'Wavelet' Transforms: an Analysis of Natural Scenes and Mammalian Visual Systems.* Oxford University Press, 1993.

[Fis95] Y. Fisher. *Fractal Image Compression: Theory and Application.* Springer-Verlag, New York, 1995.

[FKE06] A. Foi, V. Katkovnik, and K. Egiazarian. Pointwise Shape-Adaptive DCT denoising with structure preservation in luminance-chrominance space. In *Proc. Video Process. Qual. Metrics Consum. Electr.*, Scottsdale, AZ, USA, Jan. 2006.

[FKE07a] A. Foi, V. Katkovnik, and K. Egiazarian. Pointwise Shape-Adaptive DCT for high-quality denoising and deblocking of grayscale and color images. *IEEE Trans. Image Process.*, 16(5):1395–1411, May 2007.

[FKE07b] A. Foi, V. Katkovnik, and K. Egiazarian. Signal-dependent noise removal in pointwise Shape-Adaptive DCT domain with locally adaptive variance. In *Proc. Euro. Signal Process. Conf.*, Poznan, Poland, Sept. 2007.

[Foi05] A. Foi. *Anisotropic nonparametric image processing: theory, algorithms and applications.* PhD thesis, Politecnico di Milano, Apr. 2005.

[Foi07] A. Foi. *Pointwise shape-adaptive DCT image filtering and signal-dependent noise estimation.* PhD thesis, Tampere University of Technology, Dec. 2007.

[Foi09] Alessandro Foi. Clipped noisy images: Heteroskedastic modeling and practical denoising. *Signal Processing*, 89(12):2609–2629, Dec. 2009.

[FTKE08] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian. Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Trans. Image Process.*, 17(10):1737–1754, Oct. 2008.

[Gab46] D. Gabor. Theory of communication. *IEEE Trans. Commun.*, 93:429–457, 1946.

[GAML07] L. Guo, O. C. Au, M. Ma, and Z. Liang. Temporal video denoising based on multihypothesis motion compensation. *IEEE Trans. Circ. Syst. Video Technol.*, 17(10):1423–1429, Oct. 2007.

[GBP$^+$94] H. Greenspan, S. Belongie, P. Perona, R. Goodman, S. Rakshit, and C. Anderson. Overcomplete steerable pyramid filters and rotation invartiance. In *Proc. IEEE Conf. Comp. Vision Pattern Recogn.*, pages 222–228, Seattle, june 1994.

[GCMP07] J. A. Guerrero-Colon, L. Mancera, and J. Portilla. Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids. *IEEE Trans. Image Process.*, 17(1):27–41, Jan. 2007.

[GCP05] J. A. Guerrero-Colon and J. Portilla. Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids. In *Proc. IEEE Int. Conf. Image Process.*, volume 1, Genova, Italy, Sept. 2005.

[GCSP08] J. A. Guerrero-Colón, E. P. Simoncelli, and J. Portilla. Image denoising using mixtures of Gaussian scale mixtures. In *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008.

[GF07] D. J. Graham and D. J. Field. *Efficient coding of natural images.* Elsevier, 2007.

[GMM01] F. Guichard, L. Moisan, and J. M. Morel. A review of P.D.E. models in image processing and image analysis. *J. Physique IV*, 2002:137–154, 2001.

[GO07] G. Gilboa and S. Osher. Nonlocal linear image regularization and supervised segmentation. *Multisc. Model. Simul.*, 6(2):595–630, 2007.

[GO08] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multisc. Model. Simul.*, 7(3):1005–1028, 2008.

[GPP09] B. Goossens, A. Pizurica, and W. Philips. Image denoising using mixtures of projected Gaussian scale mixtures. *IEEE Trans. Image Process.*, 18(8):1689 –1702, Aug. 2009.

[Gro08] Databeans Research Group. 2008 image sensors. Technical report, 2008.

[GSP07] N. Gupta, M. N. S. Swamy, and E. I. Plotkin. Wavelet domain-based video noise reduction using temporal discrete cosine transform and hierarchically adapted thresholding. *Image Process., IET*, 1(1):2–12, March 2007.

[Gul03] O. Guleryuz. Weighted overcomplete denoising. In *Proc. Asilomar Conf. Signals Syst. Comput.*, volume 2, pages 1992–1996, Pacific Grove, CA, USA, Nov. 2003.

[GW06] R. C. Gonzalez and R. E. Woods. *Digital Image Processing (3rd Edition).* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.

[Han97] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion.* SIAM, Philadelphia, 1997.

[HDF10] S. W. Hasinoff, F. Durand, and W. T. Freeman. Noise-optimal capture for high dynamic range photography. In *CVPR*, 2010.

[HHH09] A. Hyvärinen, J. Hurri, and P. O. Hoyer. *Natural Image Statistics.* Springer-Verlag, 2009.

[HHKA05] S. Hatami, R. Hosseini, M. Kamarei, and H. Ahmadi. Wavelet based fingerprint image enhancement. In *Proc. IEEE Int. Symp. Circ. Syst.*, Kobe, Japan, May 2005.

[HO97] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computat.*, 9:1483–1492, 1997.

[Hoy99] P. O. Hoyer. *Independent Component Analysis in Image Denoising.* PhD thesis, Department of Computer Science and Engineering, Helsinki University of Technology, 1999.

[Hoy04] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints. *J. Mach. Learn. Res.*, 5:1457–1469, 2004.

[HP06] K. Hirakawa and T. W. Parks. Image denoising using total least squares. *IEEE Trans. Image Process.*, 15(9):2730–2742, Sept. 2006.

[HR08] N. Hurley and S. Rickard. Comparing measures of sparsity. In *IEEE Works. Mach. Learn. Signal Process.*, pages 55–60, Oct. 2008.

[HS08] D. K. Hammond and E. P. Simoncelli. Image modeling and denoising with orientation-adapted Gaussian Scale Mixtures. *IEEE Trans. Image Process.*, 17(11):2089–2101, 2008.

[HSJM91] P. Hall, S. J. Sheather, M. C. Jones, and J. S. Marron. On optimal data-based bandwidth selection in kernel density estimation. *Biometrika*, 78:263–269, 1991.

[HW62] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.*, 160:106–154, Jan. 1962.

[JFW06] F. Jin, P. Fieguth, and L. Winger. Wavelet video denoising with regularized multiresolution motion estimation. *EURASIP J. App. Signal Process.*, 2006:Article ID 72705, 11 pages, 2006.

[JMS96] M. C. Jones, J. S. Marron, and S. J. Sheather. A brief survey of bandwidth selection for density estimation. *J. Am. Stat. Assoc.*, 91:401–407, 1996.

[Jol02] I. T. Jolliffe. *Principal Component Analysis.* Springer, second edition, 2002.

[Kat99] V. Katkovnik. A new method for varying adaptive bandwidth. *IEEE Trans. Signal Process.*, 47(9):2567–2571, Sept. 1999.

[Kat09] V. Katkovnik. Nonlocal collaborative l0-norm prior for image denoising. TICSP series 47, Tampere International Center for Signal Process., March 2009.

[KB06] C. Kervrann and J. Boulanger. Optimal spatial adaptation for patch-based image denoising. *IEEE Trans. Image Process.*, 15(10):2866–2878, Oct. 2006.

[KB08] C. Kervrann and J. Boulanger. Local adaptivity to variable smoothness for exemplar-based image regularization and representation. *Int. J. Comput. Vision*, 79(1):45–69, 2008.

[KE08] V. Katkovnik and K. Egiazarian. Nonlocal image deblurring: variational formulation with nonlocal collaborative l0-norm imaging. In *Proc. Local and Nonlocal Approx. in Image Process.*, Tuusula, Finland, Aug. 2008.

[KEA05] V. Katkovnik, K. Egiazarian, and J. Astola. A spatially adaptive nonparametric regression image deblurring. *IEEE Trans. Image Process.*, 14(10):1469–1478, Oct. 2005.

[KEA06] V. Katkovnik, K. Egiazarian, and J. Astola. *Local Approximation Techniques in Signal and Image Processing*, volume PM157. SPIE Press, 2006.

[KFE07] V. Katkovnik, A. Foi, and K. Egiazarian. Mix-distribution modeling for overcomplete denoising. In *Adapt. Learn. Contr. Signal Process.*, Aug. 2007.

[KFEA04] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola. Directional varying scale approximations for anisotropic signal processing. In *Proc. European Signal Process. Conf.*, pages 101–104, Vienna, Austria, Sept. 2004.

[KFEA10] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola. From local kernel to nonlocal multiple-model image denoising. *Int. J. Computer Vision*, 86(1):1–32, 2010.

[Kin01] N. Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Applied Computat. Harm. Anal.*, 10:234–253, 2001.

[Kof35] K. Koffka. *Principles of Gestalt Psychology.* Harcourt, Brace & Company, New York, 1935.

[KOJ05] S. Kindermann, S. Osher, and P. W. Jones. Deblurring and denoising of images by nonlocal functionals. *Multisc. Model. Simul.*, 4(4):1091–1115, 2005.

[KR96] D. C. Knill and W. Richards. *Perception as Bayesian inference.* Cambridge University Press, 1996.

[Kre52] E. R. Kretzmer. Statistics of television signals. *Bell System Technol.*, 31(4):751–763, July 1952.

[KWCL06] W.-C. Kao, S.-H. Wang, L.-Y. Chen, and S.-Y. Lin. Design considerations of color image processing pipeline for digital cameras. *IEEE Trans. Consumer Electr.*, 52(4):1144–1152, 2006.

[LB08] F. Luisier and T. Blu. Sure-let multichannel image denoising: Interscale orthonormal wavelet thresholding. *IEEE Trans. Image Process.*, 17(4):482–492, Apr. 2008.

[LD07] Y. M. Lu and M. N. Do. Multidimensional directional filter banks and surfacelets. *IEEE Trans. Image Process.*, 16(4):918–931, 2007.

[Lee81] J.-S. Lee. Speckle analysis and smoothing of synthetic aperture radar images. *Comp. Graph. Image Process.*, 17(1):24–32, Sept. 1981.

[Lee03] T. S. Lee. Computations in the early visual cortex. *J. Physiol.*, 2003.

[Lep90] O. V. Lepskii. On a problem of adaptive estimation on white gaussian noise. *Theory Probab. Appl.*, 35:454–466, 1990.

[LL98] Lee and B. Lucier. Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Trans. Image Process.*, 7(3):319–333, 1998.

[LLKW05] D. Labate, W-Q. Lim, G. Kutyniok, and G. Weiss. Sparse multidimensional representation using shearlets. In *Proc. SPIE Conf. Wavelet Appl. Signal Image Proc.*, pages 254–262, 2005.

[LPM05] E. Le Pennec and S. G. Mallat. Sparse geometric image representations with bandelets. *IEEE Trans. Image Process.*, 14(4):423–438, 2005.

[LSK$^+$08] Ce Liu, R. Szeliski, S. B. Kang, C. L. Zitnick, and W. T. Freeman. Automatic estimation and removal of noise from a single image. *IEEE Trans. Pattern Anal. Machine Intell.*, 30(2):299–314, Feb. 2008.

[LTE08] O. Lezoray, V. T. Ta, and A. Elmoataz. Nonlocal graph regularization for image colorization. In *Proceedings International Conference on Pattern Recognition*, Tampa, United States, 2008.

[MAJ$^+$98] J. S. Marron, S. Adak, I. M. Johnstone, M. H. Neumann, and P. Patil. Exact risk analysis of wavelet regression. *J. Comput. Graph. Stat.*, 7(3):278–309, 1998.

[Mal89a] S. G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$. *Amer. Math. Soc.*, 315(1):69–87, Sept. 1989.

[Mal89b] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Machine Inetll.*, 11(7):674–693, July 1989.

[Mal08] S. G. Mallat. *A Wavelet Tour of Signal Processing, 3rd ed., Third Edition: The Sparse Way.* Academic Press, 3 edition, 2008.

[Mal09] S. G. Mallat. Geometrical grouplets. *Appl. Computat. Harm. Anal.*, 26(2):161 – 180, 2009.

[Mar82] D. Marr. *Vision.* Freeman and Co., 1982.

[MC97] F. G. Meyer and R. R. Coifman. Brushlets: a tool for directional image analysis and image compression. *Appl. Comp. Harmon. Anal.*, 4:147–187, 1997.

[MCAFM08] A. Maalouf, P. Carré, B. Augereau, and C. Fernandez Maloigne. Inpainting using geometrical grouplets. In *European Signal Processing Conference*, pages 15–20, Lausanne, Switzerland, 08 2008.

[McC80] J. McClellan. Artifacts in alpha-rooting of images. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, volume 5, pages 449–452, Apr. 1980.

[Mey93] Y. Meyer. *Wavelets - Algorithms & Applications.* SIAM, Philadelphia, PA, 1993.

[Mig08] M. Mignotte. A non-local regularization strategy for image deconvolution. *Pat. Recog. Lett.*, 29(16):2206–2212, Dec. 2008.

[MP03] D. D. Muresan and T. W. Parks. Adaptive principal components and image denoising. In *Proc. IEEE Int. Conf. Image Process.*, volume 1, Sept. 2003.

[MS05] M. Mahmoudi and G. Sapiro. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE Signal Process. Letters*, 12(12):839–842, Dec. 2005.

[MSE08] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *Multisc. Model. Simul.*, 7(1):214–241, 2008.

[MY08] S. G. Mallat and G. Yu. Video denoising with grouping bandlets. Technical report, Ecole Polytechnique, 2008.

[MZ93] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.*, 41(12):3397–3415, 1993.

[Nad64] E. A. Nadaraya. On estimating regression. *Theory Probab. Applic.*, 9:141–142, 1964.

[Nak06] J. Nakamura. *Image sensors and signal processing for digital still cameras.* CRC Press, 2006.

[NCB99] R. Neelamani, H. Choi, and R. G. Baraniuk. Wavelet-domain regularized deconvolution for ill-conditioned systems. In *Proc. IEEE Int. Conf. Image Process.*, volume 1, pages 204–208, Kobe, Japan, Oct. 1999.

[NCB04] R. Neelamani, H. Choi, and R. G. Baraniuk. Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems. *IEEE Trans. Signal Process.*, 52(2):418–433, Feb. 2004.

[OF96] B. A. Olshausen and D. J. Field. Natural image statistics and efficient coding. *Network: Computat. Neural Systems*, 7:333–339, 1996.

[Ols03a] B. A. Olshausen. Learning sparse overcomplete representations of time-varying natural images. In *Proc. IEEE Int. Conf. Image Process.*, Sept. 2003.

[Ols03b] B. A. Olshausen. Principles of image representation in visual cortex. In L. M. Chalupa and J. S. Werner, editors, *The Visual Neurosciences.* MIT Press, 2003.

[Pal07] D. Paliy. *Local Approximations in Demosaicing and Deblurring of Digital Sensor Data.* PhD thesis, Tampere University of Technology, Dec. 2007.

[PBC08] G. Peyré, S. Bougleux, and L. Cohen. Non-local regularization of inverse problems. In *Proc. Europ. Conf. Comp. Vision*, pages 57–68, Berlin, Heidelberg, 2008. Springer-Verlag.

[PE09] M. Protter and M. Elad. Image sequence denoising via sparse and redundant representations. *IEEE Trans. Image Process.*, 18(1):27–35, Jan. 2009.

[Pey08] G. Peyré. Image Processing with Non-local Spectral Bases. *J. Multisc. Model. Simul.*, 7(2):703–730, 2008.

[PM90] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(7):629–639, 1990.

[PP06] A. Pizurica and W. Philips. Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising. *IEEE Trans. Image Process.*, 15(3):654–665, March 2006.

[Pra78] W. K. Pratt. *Digital Image Processing.* John Wiley and Sons, New York, 1978.

[PRM00] A. Polesel, G. Ramponi, and V. J. Mathews. Image enhancement via adaptive unsharp masking. *IEEE Trans. Image Process.*, 9(3):505–510, Mar 2000.

[PS81] P. R. Prucnal and B. E. A. Saleh. Transformation of image-signal-dependent noise into image-signal-independent noise. *Opt. Lett.*, 6(7):316–318, 1981.

[PS03] J. Portilla and E. P. Simoncelli. Image restoration using Gaussian scale mixtures in the wavelet domain. In *Proc. IEEE Int. Conf. Image Process.*, volume 2, pages 965–968, Barcelona, Spain, Sept. 2003.

[PSWS03] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli. Image denoising using a scale mixture of Gaussians in the wavelet domain. *IEEE Trans. Image Process.*, 12(11):1338–1351, Nov. 2003.

[Pub09] Electronics.ca Publications. Image sensors 2009: Camera phones continue to dominate shipments worldwide. Technical report, 2009.

[RAS07] S. M. M. Rahman, M. O. Ahmad, and M. N. S. Swamy. Video denoising based on inter-frame statistical modeling of wavelet coefficients. *IEEE Trans. Circ. Syst. Video Technol.*, 17(2):187–198, Feb. 2007.

[RAS08] S. M. M. Rahman, M. O. Ahmad, and M. N. S. Swamy. Bayesian wavelet-based image denoising using the Gauss-Hermite expansion. *IEEE Trans. Image Process.*, 17(10):1755 –1771, Oct. 2008.

[RB05] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *CVPR*, pages 860–867, 2005.

[RCB99] J. K. Romberg, H. Choi, and R. G. Baraniuk. Bayesian wavelet-domain image modeling using hidden Markov trees. In *In Proc. IEEE ICIP, Kobe*, 1999.

[RE05] D. Rusanovskyy and K. Egiazarian. Video denoising algorithm in sliding 3D DCT domain. In *Proc. Adv. Concepts Intell. Video Syst.*, Antwerp, Belgium, Sept. 2005.

[ROF92] L. Rudin, S. Osher, and F. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[RP00] M. Riesenhuber and T. Poggio. Models of object recognition. *Nature Neuroscience*, 3:1199–1204, 2000.

[RSW95] D. Ruppert, S. J. Sheather, and M. P. Wand. An effective bandwidth selector for local least squares regression (Corr: 96V91 p1380). *J. Am. Stat. Assoc.*, 90:1257–1270, 1995.

[RSYD05] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew. Color image processing pipeline. *IEEE Signal Process. Magazine*, 22(1):34–43, 2005.

[Rus02] F. Russo. An image enhancement technique combining sharpening and noise reduction. *IEEE Trans. Instrum. Measure.*, 51(4):824–828, Aug 2002.

[RYW04] N. Rajpoot, Z. Yao, and R. Wilson. Adaptive wavelet restoration of noisy video sequences. In *Proc. IEEE Int. Conf. Image Process.*, volume 2, pages 957–960, Oct. 2004.

[SA96] E. P. Simoncelli and E. H. Adelson. Noise removal via bayesian wavelet coring. In *Proc IEEE Int. Conf. Image Process.*, pages 379–382, 1996.

[Sap06] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis.* Cambridge University Press, New York, NY, USA, 2006.

[SB82] B. Sakitt and H. B. Barlow. A model for the economical encoding of the visual image in cerebral cortex. *Biol. Cybern.*, 43(2):97–108, 1982.

[SCD02] J.-L. Starck, E. J. Candes, and D. L. Donoho. The curvelet transform for image denoising. *IEEE Trans. Image Process.*, 11(6):670–684, 2002.

[Sel08] I. W. Selesnick. The estimation of Laplace random vectors in additive white Gaussian noise. *IEEE Trans. Signal Process.*, 56(8-1):3482–3496, 2008.

[SF95] E. P. Simoncelli and W. T. Freeman. The steerable pyramid: a flexible architecture for multi-scale derivative computation. In *Proc. IEEE Int. Conf. Image Process.*, 1995.

[SFAH92a] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multi-scale transforms. *IEEE Trans. Inf. Theory*, 38(2):587–607, March 1992. Special Issue on Wavelets.

[SFAH92b] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multiscale transforms. *IEEE Trans. Inf. Theory*, 38(2):587–607, 1992.

[Sha99] J. Shao. *Mathematical Statistics.* Springer-Verlag, 1999.

[She03] J. H. Shen. Inpainting and the fundamental problem of image processing. In *SIAM News*, volume 36, 2003.

[SKB01] N. Sochen, R. Kimmel, and A. M. Bruckstein. Diffusions and confusions in signal and image processing. *J. Math. Imaging Vis.*, 14(3):195–209, 2001.

[SL03] I. W. Selesnick and K. Y. Li. Video denoising using 2D and 3D dual-tree complex wavelet transforms. In *Proc. Wavelet Applicat. Signal and Image Process. X, SPIE*, San Diego, USA, Aug. 2003.

[SM95] T. Sikora and B. Makai. Shape-adaptive DCT for generic coding of video. *IEEE Trans. Circ. Syst. Video Technol.*, 5(1):59–62, 1995.

[SM08] H. Seo and P. Milanfar. Video denoising using higher order optimal space-time adaptation. In *Proc. Int. Conf. Acoust. Speech Signal Process.*, pages 1249–1252, March 2008.

[SS02] L. Sendur and I. W. Selesnick. Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency. *IEEE Trans. Signal Process.*, 50(11):2744–2756, Nov. 2002.

[TFG01] H. Tian, B. Fowler, and A. E. Gamal. Analysis of temporal noise in cmos photodiode active pixel sensor. *IEEE J. Solid-State Circ.*, 36(1):92–101, Jan 2001.

[TFM07] H. Takeda, S. Farsiu, and P. Milanfar. Kernel regression for image processing and reconstruction. *IEEE Trans. Image Process.*, 16(2), Feb. 2007.

[The95] A. J. P. Theuwissen. *Solid-State Imaging with Charge-Coupled Devices.* Kluwer Acad. Publishers, Dordrecht, 1995.

[Tik63] A. N. Tikhonov. The regularization of ill-posed problems. *Dokl. Akad. Nauk.*, SSR 153(1):49–52, 1963.

[TM98] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proc. Int. Conf. Comp. Vision*, Washington, DC, USA, 1998.

[TTC92] D. J. Tolhurst, Y. Tadmor, and T. Chao. Amplitude spectra of natural images. *Ophthalmic Physiol. Optics*, 12(2):229–232, 1992.

[VDVK09] D. Van De Ville and M. Kocher. Sure-based non-local means. *IEEE Signal Process. Letters*, 16(11):973–976, Nov. 2009.

[VHR98] J. H. Van Hateren and D. L. Ruderman. Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proc. Royal Soc. London. Series B: Bio. Sciences*, 265(1412):2315–2320, 1998.

[Vog02] C. Vogel. *Computational Methods for Inverse Problems.* SIAM, 2002.

[VVdWPK06] E. Vansteenkiste, D. Van der Weken, W. Philips, and E. E. Kerre. Perceived image quality measurement of state-of-the-art noise reduction schemes. In *Lecture Notes in Computer Science ACIVS*, volume 4179, pages 114–124, Antwerp, Belgium, Sept. 2006.

[Wat64] G. S. Watson. Smooth regression analysis. *Sankhyā Ser.*, 26:359–372, 1964.

[Wat83] A. B. Watson. *Detection and recognition of simple spatial forms.* Springer-Verlag, Berlin, 1983.

[WBSS04] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error measurement to structural similarity. *IEEE Trans. Image Process.*, 13(4), Apr. 2004.

[WF07] Y. Weiss and W. T. Freeman. What makes a good model of natural images? In *CVPR*, pages 1–8, 2007.

[WG04] C. S. Won and R. M. Gray. *Stochastic Image Processing (Information Technology: Transmission, Processing, and Storage).* Kluwer Academic / Plenum Publishers, 2004.

[WL09] Z. Wang and Q. Li. Statistics of natural image sequences: temporal motion smoothness by local phase correlations. In *Proc. SPIE Electronic Imaging: Human Vision Electr. Imaging XIV*, volume 7240, San Jose, CA, USA, Jan. 2009.

[WS00] M. J. Wainwright and E. P. Simoncelli. Scale mixtures of Gaussians and the statistics of natural images. In S. A. Solla, T. K. Leen, and K. Müller, editors, *Adv. Neural Inf. Process. Syst.*, volume 12, pages 855–861. MIT Press, 2000.

[WZ90] B. Wegmann and C. Zetzsche. Statistical dependencies between orientation filter outputs used in human vision based image code. In *Proc. Visual Comm. and Image Process.*, volume 1360, pages 909–922, 1990.

[Yar85] L. P. Yaroslavsky. *Digital Picture Processing.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1985.

[YCP08] F. Yan, L. Cheng, and S. Peng. A new interscale and intrascale orthonormal wavelet thresholding for sure-based image denoising. *IEEE Signal Process. Letters*, 15:139–142, 2008.

[YZW09] H. Yu, L. Zhao, and H. Wang. Image denoising using trivariate shrinkage filter in the wavelet domain and joint bilateral filter in the spatial domain. *IEEE Trans. Image Process.*, 18(10):2364–2369, Oct. 2009.

[ZPP06] V. Zlokolica, A. Pizurica, and W. Philips. Wavelet-domain video denoising based on reliability measures. *IEEE Trans. Circ. Syst. Video Technol.*, 16(8):993–1007, Aug. 2006.

[ÖYE98] R. Öktem, L. Yaroslavsky, and K. Egiazarian. Signal and image denoising in transform domain and wavelet shrinkage: a comparative study. In *Proc. EUSIPCO*, Ghodes, Greece, Sept. 1998.

Publications

Publication [P1]

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," *Proc. SPIE Electronic Imaging* , no. 6064A-30, San Jose, California, USA, January 2006.

Image denoising with block-matching and 3D filtering

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian
Institute of Signal Processing, Tampere University of Technology, Finland
PO BOX 553, 33101 Tampere, Finland
firstname.lastname@tut.fi

ABSTRACT

We present a novel approach to still image denoising based on effective filtering in 3D transform domain by combining sliding-window transform processing with block-matching. We process blocks within the image in a sliding manner and utilize the block-matching concept by searching for blocks which are similar to the currently processed one. The matched blocks are stacked together to form a 3D array and due to the similarity between them, the data in the array exhibit high level of correlation. We exploit this correlation by applying a 3D decorrelating unitary transform and effectively attenuate the noise by shrinkage of the transform coefficients. The subsequent inverse 3D transform yields estimates of all matched blocks. After repeating this procedure for all image blocks in sliding manner, the final estimate is computed as weighed average of all overlapping block-estimates. A fast and efficient algorithm implementing the proposed approach is developed. The experimental results show that the proposed method delivers state-of-art denoising performance, both in terms of objective criteria and visual quality.

Keywords: image denoising, block-matching, 3D transforms

1. INTRODUCTION

Much of the recent research on image denoising has been focused on methods that reduce noise in transform domain. Starting with the milestone work of Donoho,[1, 2] many of the later techniques[3–7] performed denoising in wavelet transform domain. Of these methods, the most successful proved to be the ones[4, 5, 7] based on rather sophisticated modeling of the noise impact on the transform coefficients of overcomplete multiscale decompositions. Not limited to multiscale techniques, the overcomplete representations have traditionally played a significant role in improving the restoration abilities of even the most basic transform-based methods. This is manifested by the sliding-window transform denoising,[8, 9] where the basic idea is to successively denoise overlapping blocks by coefficient shrinkage in local 2D transform domain (e.g. DCT, DFT, etc.). Although the transform-based approaches deliver very good overall performance in terms of objective criteria, they fail to preserve details which are not suitably represented by the used transform and often introduce artifacts that are characteristic of this transform.

A different denoising strategy based on non-local estimation appeared recently,[10, 11] where a pixel of the true image is estimated from regions which are found similar to the region centered at the estimated pixel. These methods, unlike the transform-based ones, introduce very few artifacts in the estimates but often oversmooth image details. Based on an elaborate adaptive weighting scheme, the exemplar-based denoising[10] appears to be the best of them and achieves results competitive to the ones produced by the best transform-based techniques.

The concept of employing similar data patches from different locations is popular in the video processing field under the term of "block-matching", where it is used to improve the coding efficiency by exploiting similarity among blocks which follow the motion of objects in consecutive frames. Traditionally, block-matching has found successful application in conjunction with transform-based techniques. Such applications include video compression (MPEG standards) and also video denoising,[12] where noise is attenuated in 3D DCT domain.

We propose an original image denoising method based on effective filtering in 3D transform domain by combining sliding-window transform processing with block-matching. We undertake the block-matching concept for a single noisy image; as we process image blocks in a sliding manner, we search for blocks that exhibit similarity to the currently-processed one. The matched blocks are stacked together to form a 3D array. In this manner,

Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning
edited by E.R. Dougherty, J.T. Astola, K.O. Egiazarian, N.M. Nasrabadi, S.A. Rizvi
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6064, 606414, © 2006 SPIE-IS&T · 0277-786X/06/$15

we induce high correlation along the dimension of the array in which the blocks are stacked. We exploit this correlation by applying a 3D decorrelating unitary transform which produces a sparse representation of the true signal in 3D transform domain. Efficient noise attenuation is done by applying a shrinkage operator (e.g. hard-thresholding or Wiener filtering) on the transform coefficients. This results in improved denoising performance and effective detail preservation in the local estimates of the matched blocks, which are reconstructed by an inverse 3D transform of the filtered coefficients. After processing all blocks, the final estimate is the weighted average of all overlapping local block-estimates. Because of overcompleteness which is due to the overlap, we avoid blocking artifacts and further improve the estimation ability.

Although the proposed approach is general with respect to the type of noise, for simplicity of exposition, we restrict our attention to the problem of attenuating additive white Gaussian noise (AWGN).

The basic approach and its extension to Wiener filtering are presented in Sections 2 and 3, respectively. An efficient algorithm which implements the proposed approach is developed in Section 4. Finally, Section 5 is devoted to demonstration and discussion of experimental results.

2. DENOISING BY SHRINKAGE IN 3D TRANSFORM DOMAIN WITH BLOCK-MATCHING

Let us introduce the observation model and notation used throughout the paper. We consider noisy observations $z : X \to \mathbb{R}$ of the form $z(x) = y(x) + \eta(x)$, where $x \in X$ is a 2D spatial coordinate that belongs to the image domain $X \subset \mathbb{Z}^2$, y is the true image, and $\eta(x) \sim \mathcal{N}\left(0, \sigma^2\right)$ is white Gaussian noise of variance σ^2. By Z_x we denote a block of fixed size $N_1 \times N_1$ extracted from z, which has $z(x)$ as its upper-left element; alternatively, we say that Z_x is located at x. With $\widehat{y}$ we designate the final estimate of the true image.

Let us state the used assumptions. We assume that some of the blocks (of fixed size $N_1 \times N_1$) of the true image exhibit mutual correlation. We also assume that the selected unitary transform is able to represent sparsely these blocks. However, the diversity of such blocks in natural images often makes the latter assumption unsatisfied in 2D transform domain and fulfilled only in 3D transform domain due to the correlation introduced by block-matching. The standard deviation σ of the AWGN can be accurately estimated (e.g.[1]), therefore we assume its a-priori knowledge.

2.1. Local Estimates

We successively process all overlapping blocks of fixed size in a sliding manner, where "process" stands for the consecutive application of block-matching and denoising in local 3D transform domain. For the sub-subsections to follow, we fix the currently processed block as Z_{x_R}, where $x_R \in X$, and denominate it as "reference block".

2.1.1. Block-matching

Block-matching is employed to find blocks that exhibit high correlation to Z_{x_R}. Because its accuracy is significantly impaired by the presence of noise, we utilize a block-similarity measure which performs a coarse initial denoising in local 2D transform domain. Hence, we define a block-distance measure (inversely proportional to similarity) as

$$d\left(Z_{x_1}, Z_{x_2}\right) = N_1^{-1} \left\| \Upsilon\left(\mathcal{T}_{2D}\left(Z_{x_1}\right), \lambda_{thr2D}\sigma\sqrt{2\log\left(N_1^2\right)}\right) - \Upsilon\left(\mathcal{T}_{2D}\left(Z_{x_2}\right), \lambda_{thr2D}\sigma\sqrt{2\log\left(N_1^2\right)}\right)\right\|_2, \quad (1)$$

where $x_1, x_2 \in X$, $\mathcal{T}_{2D}$ is a 2D linear unitary transform operator (e.g. DCT, DFT, etc.), Υ is a hard-threshold operator, λ_{thr2D} is fixed threshold parameter, and $\|\cdot\|_2$ denotes the L^2-norm. Naturally, Υ is defined as

$$\Upsilon\left(\lambda, \lambda_{thr}\right) = \begin{cases} \lambda, & \text{if } |\lambda| > \lambda_{thr} \\ 0, & \text{otherwise.} \end{cases}$$

The result of the block-matching is a set $\mathcal{S}_{x_R} \subseteq X$ of the coordinates of the blocks that are similar to Z_{x_R} according to our d-distance (1); thus, $\mathcal{S}_{x_R}$ is defined as

$$\mathcal{S}_{x_R} = \left\{x \in X \mid d\left(Z_{x_R}, Z_x\right) < \tau_{match}\right\}, \quad (2)$$

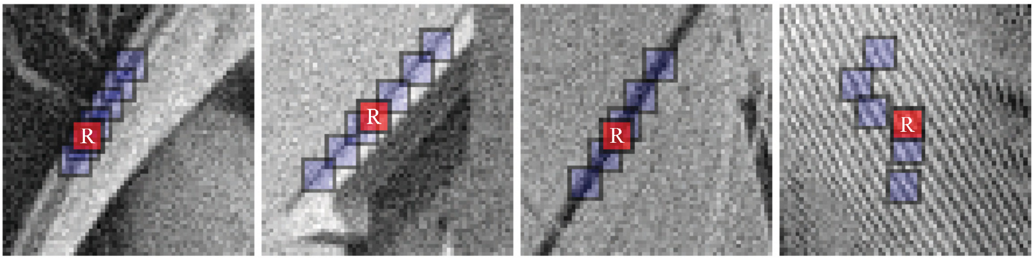


Figure 1. Fragments of *Lena, House, Boats* and *Barbara* corrupted by AWGN of $\sigma = 15$. For each fragment block-matching is illustrated by showing a reference block marked with 'R' and a few of its matched ones.

where τ_{match} is the maximum d-distance for which two blocks are considered similar. Obviously $d\left(Z_{x_R}, Z_{x_R}\right) = 0$, which implies that $|\mathcal{S}_{x_R}| \geq 1$, where $|\mathcal{S}_{x_R}|$ denotes the cardinality of $\mathcal{S}_{x_R}$.

The matching procedure in presence of noise is demonstrated on Figure 1, where we show a few reference blocks and the ones matched as similar to them.

2.1.2. Denoising in 3D transform domain

We stack the matched noisy blocks $Z_{x \in \mathcal{S}_{x_R}}$ (ordering them by increasing d-distance to Z_{x_R}) to form a 3D array of size $N_1 \times N_1 \times |\mathcal{S}_{x_R}|$, which is denoted by $\mathbf{Z}_{\mathcal{S}_{x_R}}$. We apply a unitary 3D transform $\mathcal{T}_{3D}$ on $\mathbf{Z}_{\mathcal{S}_{x_R}}$ in order to attain sparse representation of the true signal. The noise is attenuated by hard-thresholding the transform coefficients. Subsequently, the inverse transform operator $\mathcal{T}_{3D}^{-1}$ yields a 3D array of reconstructed estimates

$$\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}} = \mathcal{T}_{3D}^{-1}\left(\Upsilon\left(\mathcal{T}_{3D}\left(\mathbf{Z}_{\mathcal{S}_{x_R}}\right), \lambda_{thr3D}\sigma\sqrt{2\log\left(N_1^2\right)}\right)\right), \tag{3}$$

where λ_{thr3D} is a fixed threshold parameter. The array $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$ comprises of $|\mathcal{S}_{x_R}|$ stacked local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ of the true image blocks located at $x \in \mathcal{S}_{x_R}$. We define a weight for these local estimates as

$$\omega_{x_R} = \begin{cases} \frac{1}{N_{har}}, & \text{if } N_{har} \geq 1 \\ 1, & \text{otherwise,} \end{cases} \tag{4}$$

where N_{har} is the number of non-zero transform coefficients after hard-thresholding. Observe that $\sigma^2 N_{har}$ is equal[*] to the total variance of $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$. Thus, sparser decompositions of $\mathbf{Z}_{\mathcal{S}_{x_R}}$ result in less noisy estimates which are awarded greater weights by (4).

2.2. Estimate Aggregation

After processing all reference blocks, we have a set of local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$, $\forall x_R \in X$ (and their corresponding weights ω_{x_R}, $\forall x_R \in X$), which constitute an overcomplete representation of the estimated image due to the overlap between the blocks. It is worth mentioning that a few local block estimates might be located at the same coordinate (e.g. $\widehat{Y}^{x_a}_{x_b}$ and $\widehat{Y}^{x_b}_{x_b}$ are both located at x_b but obtained while processing the reference blocks at x_a and x_b, respectively). Let $\widehat{Y}^{x_R}_{x_m}(x)$ be an estimate of $y(x)$, where $x, x_R \in X$, and $x_m \in \mathcal{S}_{x_R}$. We zero-extend $\widehat{Y}^{x_R}_{x_m}(x)$ outside its square support in order to simplify the formulation. The final estimate $\widehat{y}$ is computed as a weighted average of all local ones as given by

$$\widehat{y}(x) = \frac{\sum_{x_R \in X} \sum_{x_m \in \mathcal{S}_{x_R}} \omega_{x_R} \widehat{Y}^{x_R}_{x_m}(x)}{\sum_{x_R \in X} \sum_{x_m \in \mathcal{S}_{x_R}} \omega_{x_R} \chi_{x_m}(x)}, \ \forall x \in X, \tag{5}$$

---

[*] Equality holds only if the matched blocks that build $\mathbf{Z}_{\mathcal{S}_{x_R}}$ are non-overlapping; otherwise, a certain amount of correlation is introduced in the noise.

where $\chi_{x_m} : X \to \{0, 1\}$ is the characteristic function of the square support of a block located at $x_m \in X$.

One can expect substantially overcomplete representation of the signal in regions where a block is matched to many others. On the other hand, if a match is not found for a given reference block, the method reduces to denoising in 2D transform domain. Thus, the overcomplete nature of the method is highly dependent on the block-matching and therefore also on the particular noisy image.

3. WIENER FILTER EXTENSION

Provided that an estimate of the true image is available (e.g. it can be obtained from the method given in the previous section), we can construct an empirical Wiener filter as a natural extension of the above thresholding technique. Because it follows the same approach, we only give the few fundamental modifications that are required for its development and thus omitting repetition of the concept. Let us denote the initial image estimate by $e : X \to \mathbb{R}$. In accordance with our established notation, E_x designates a square block of fixed size $N_1 \times N_1$, extracted from e and located at $x \in X$.

3.1. Modification to Block-Matching

In order to improve the accuracy of block-matching, it is performed within the initial estimate e rather than the noisy image. Accordingly, we replace the thresholding-based d-distance measure from (1) with the normalized L^2-norm of the difference of two blocks with subtracted means. Hence, the definition (2) of $\mathcal{S}_{x_R}$ becomes

$$\mathcal{S}_{x_R} = \left\{ x \in X \mid N_1^{-1} \left\| \left(E_{x_R} - \overline{E_{x_R}} \right) - \left(E_x - \overline{E_x} \right) \right\|_2 < \tau_{match} \right\}, \tag{6}$$

where $\overline{E_{x_R}}$ and $\overline{E_x}$ are the mean values of the blocks E_{x_R} and E_x, respectively. The mean subtraction allows for improved matching of blocks with similar structures but different mean values.

3.2. Modification to Denoising in 3D Transform Domain

The linear Wiener filter replaces the nonlinear hard-thresholding operator. The attenuating coefficients for the Wiener filter are computed in 3D transform domain as

$$\mathbf{W}_{\mathcal{S}_{x_R}} = \frac{\left| \mathcal{T}_{3D} \left(\mathbf{E}_{\mathcal{S}_{x_R}} \right) \right|^2}{\left| \mathcal{T}_{3D} \left(\mathbf{E}_{\mathcal{S}_{x_R}} \right) \right|^2 + \sigma^2},$$

where $\mathbf{E}_{\mathcal{S}_{x_R}}$ is a 3D array built by stacking the matched blocks $E_{x \in \mathcal{S}_{x_R}}$ (in the same manner as $\mathbf{Z}_{\mathcal{S}_{x_R}}$ is built by stacking $Z_{x \in \mathcal{S}_{x_R}}$). We filter the 3D array of noisy observations $\mathbf{Z}_{\mathcal{S}_{x_R}}$ in $\mathcal{T}_{3D}$-transform domain by an elementwise multiplication with $\mathbf{W}_{\mathcal{S}_{x_R}}$. The subsequent inverse transform gives

$$\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}} = \mathcal{T}_{3D}^{-1} \left(\mathbf{W}_{\mathcal{S}_{x_R}} \mathcal{T}_{3D} \left(\mathbf{Z}_{\mathcal{S}_{x_R}} \right) \right), \tag{7}$$

where $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$ comprises of stacked local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ of the true image blocks located at the matched locations $x \in \mathcal{S}_{x_R}$. As in (4), the weight assigned to the estimates is inversely proportional to the total variance of $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$ and defined as

$$\omega_{x_R} = \left(\sum_{i=1}^{N_1} \sum_{j=1}^{N_1} \sum_{t=1}^{\left| \mathcal{S}_{x_R} \right|} \left| \mathbf{W}_{\mathcal{S}_{x_R}} (i, j, t) \right|^2 \right)^{-1}. \tag{8}$$

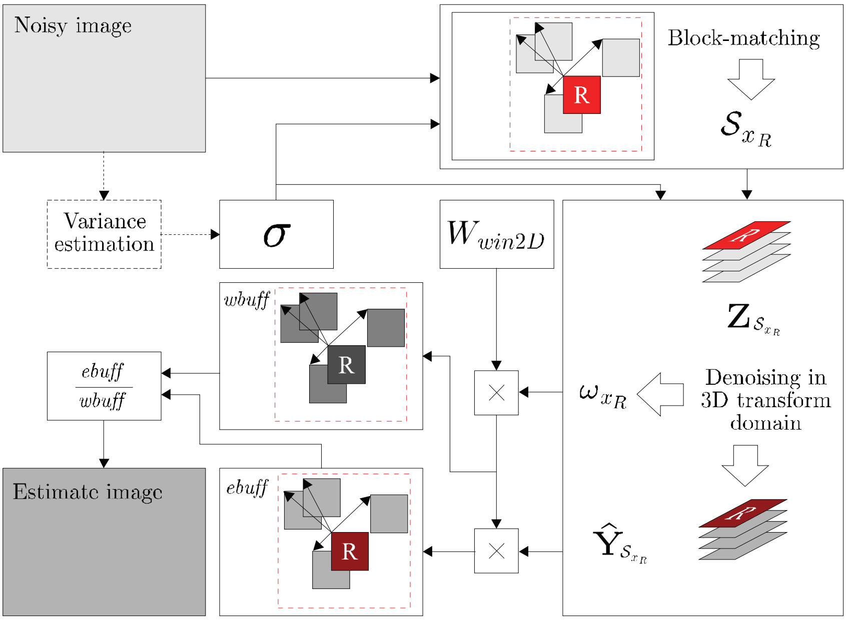


Figure 2. Flowchart for denoising by hard-thresholding in 3D transform domain with block-matching.

4. ALGORITHM

We present an algorithm which employs the hard-thresholding approach (from Section 2) to deliver an initial estimate for the Wiener filtering part (from Section 3) that produces the final estimate. A straightforward implementation of this general approach is computationally demanding. Thus, in order to realize a practical and efficient algorithm, we impose constraints and exploit certain expedients. In this section we introduce these aspects and develop an efficient implementation of the proposed approach.

The choice of the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$ is governed by their energy compaction (sparsity) ability for noise-free image blocks (2D) and stacked blocks (3D), respectively. It is often assumed that neighboring pixels in small blocks extracted from natural images exhibit high correlation; thus, such blocks can be sparsely represented by well-established decorrelating transforms, such as the DCT, the DFT, wavelets, etc. From computational efficiency point of view, however, very important characteristics are the separability and the availability of fast algorithms. Hence, the most natural choice for $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$ is a fast separable transform which allows for sparse representation of the true-image signal in each dimension of the input array.

4.1. Efficient Image Denoising Algorithm with Block-Matching and 3D Filtering

Let us introduce constraints for the complexity of the algorithm. First, we fix the maximum number of matched blocks by setting an integer N_2 to be the upper bound for the cardinality of the sets $\mathcal{S}_{x_R \in X}$. Second, we do block-matching within a local neighborhood of fixed size $N_S \times N_S$ centered about each reference block, instead of doing it in the whole image. Finally, we use N_{step} as a step by which we slide to every next reference block. Accordingly, we introduce $X_R \subseteq X$ as the set of the reference blocks' coordinates, where $|X_R| \approx \frac{|X|}{N_{step}^2}$ (e.g., $N_{step} = 1$ implies $X_R = X$).

In order to reduce the impact of artifacts on the borders of blocks (border effects), we use a Kaiser window W_{win2D} (with a single parameter β) as part of the weights of the local estimates. These artifacts are inherent of many transforms (e.g. DFT) in presence of sharp intensity differences across the borders of a block.

Let the input noisy image be of size $M \times N$, thus $|X| = MN$. We use two buffers of the same size—*ebuff* for estimates and *wbuff* for weights—to represent the summations in the numerator and denominator, respectively, in (5). For simplicity, we extend our notation so that $\mathit{ebuff}\,(x)$ denotes a single pixel at coordinate $x \in X$ and ebuff_x designates a block located at x in *ebuff* (the same notation is to be used for *wbuff*).

A flowchart of the hard-thresholding part of the algorithm is given in Figure 2 (but we do not give such for the Wiener filtering part since it requires only the few changes given in Section 3). Following are the steps of the image denoising algorithm with block-matching and 3D filtering.

1. **Initialization.** Initialize $\mathit{ebuff}\,(x) = 0$ and $\mathit{wbuff}\,(x) = 0$, for all $x \in X$.
2. **Local hard-thresholding estimates.** For each $x_R \in X_R$, do the following sub-steps.
    - (a) *Block-matching.* Compute $\mathcal{S}_{x_R}$ as given in Equation (2) but restrict the search to a local neighborhood of fixed size $N_S \times N_S$ centered about x_R. If $|\mathcal{S}_{x_R}| > N_2$, then let only the coordinates of the N_2 blocks with smallest d-distance to Z_{x_R} remain in $\mathcal{S}_{x_R}$ and exclude the others.
    - (b) *Denoising by hard-thresholding in local 3D transform domain.* Compute the local estimate blocks $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ and their corresponding weight ω_{x_R} as given in (3) and (4), respectively.
    - (c) *Aggregation.* Scale each reconstructed local block estimate $\widehat{Y}^{x_R}_x$, where $x \in \mathcal{S}_{x_R}$, by a block of weights $W(x_R) = \omega_{x_R} W_{win2D}$ and accumulate to the estimate buffer: $\mathit{ebuff}_x = \mathit{ebuff}_x + W(x_R)\,\widehat{Y}^{x_R}_x$, for all $x \in \mathcal{S}_{x_R}$. Accordingly, the weight block is accumulated to same locations as the estimates but in the weights buffer: $\mathit{wbuff}_x = \mathit{wbuff}_x + W(x_R)$, for all $x \in \mathcal{S}_{x_R}$.
3. **Intermediate estimate.** Produce the intermediate estimate $e(x) = \frac{\mathit{ebuff}\,(x)}{\mathit{wbuff}\,(x)}$ for all $x \in X$, which is to be used as initial estimate for the Wiener counterpart.
4. **Local Wiener filtering estimates.** Use e as initial estimate. The buffers are re-initialized: $\mathit{ebuff}\,(x) = 0$ and $\mathit{wbuff}\,(x) = 0$, for all $x \in X$. For each $x_R \in X_R$, do the following sub-steps.
    - (a) *Block-matching.* Compute $\mathcal{S}_{x_R}$ as given in (6) but restrict the search to a local neighborhood of fixed size $N_S \times N_S$ centered about x_R. If $|\mathcal{S}_{x_R}| > N_2$, then let only the coordinates of the N_2 blocks with smallest distance (as defined in Subsection 3.1) to E_{x_R} remain in $\mathcal{S}_{x_R}$ and exclude the others.
    - (b) *Denoising by Wiener filtering in local 3D transform domain.* The local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ and their weight ω_{x_R} are computed as given in (7) and (8), respectively.
    - (c) *Aggregation.* It is identical to step 2c.
5. **Final estimate.** The final estimate is given by $\widehat{y}(x) = \frac{\mathit{ebuff}\,(x)}{\mathit{wbuff}\,(x)}$, for all $x \in X$.

4.2. Complexity

The time complexity order of the algorithm as a function of its parameters is given by

$$\mathcal{O}\left(MN\mathcal{O}_{\mathcal{T}_{2D}}\left(N_1, N_1\right)\right) + \mathcal{O}\left(MN\frac{\left(N_1^2 + N_2\right)N_S^2}{N_{step}^2}\right) + \mathcal{O}\left(MN\frac{\mathcal{O}_{\mathcal{T}_{3D}}\left(N_1, N_1, N_2\right)}{N_{step}^2}\right),$$

where the first two addends are due to block-matching and the third is due to $\mathcal{T}_{3D}$ used for denoising and where $\mathcal{O}_{\mathcal{T}_{2D}}(N_1, N_1)$ and $\mathcal{O}_{\mathcal{T}_{3D}}(N_1, N_1, N_2)$ denote the complexity orders of the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$, respectively. Both $\mathcal{O}_{\mathcal{T}_{2D}}$ and $\mathcal{O}_{\mathcal{T}_{3D}}$ depend on properties of the adopted transforms such as separability and availability of fast algorithms. For example, the DFT has an efficient implementation by means of fast Fourier transform (FFT). The 2D FFT, in particular, has complexity $\mathcal{O}\left(N_1 N_2 \log\left(N_1 N_2\right)\right)$ as opposed to $\mathcal{O}\left(N_1^2 N_2^2\right)$ of a custom non-separable transform. Moreover, an effective trade-off between complexity and denoising performance can be achieved by varying N_{step}.

Table 1. Results in output PSNR (dB) of the denoising algorithm with block-matching and filtering in 3D DFT domain.

	Image						
σ/ PSNR	*Lena* 512×512	*Barbara* 512×512	*House* 256×256	*Peppers* 256×256	*Boats* 512×512	*Couple* 512×512	*Hill* 512×512
5/ 34.15	38.63	38.18	39.54	37.84	37.20	37.40	37.11
10/ 28.13	35.83	34.87	36.37	34.38	33.79	33.88	33.57
15/ 24.61	34.21	33.08	34.75	32.31	31.96	31.93	31.79
20/ 22.11	33.03	31.77	33.54	30.87	30.65	30.58	30.60
25/ 20.17	32.08	30.75	32.67	29.80	29.68	29.57	29.74
30/ 18.59	31.29	29.90	31.95	28.97	28.90	28.75	29.04
35/ 17.25	30.61	29.13	31.21	28.14	28.20	28.03	28.46
50/ 14.16	29.08	27.51	29.65	26.46	26.71	26.46	27.21
100/ 8.13	26.04	24.14	25.92	23.11	24.00	23.60	24.77

5. RESULTS AND DISCUSSION

We present experiments conducted with the algorithm introduced in Section 4, where the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$ are the 2D DFT and the 3D DFT, respectively. All results are produced with the same fixed parameters—but different for the hard-thresholding and Wiener filtering parts. For the hard-thresholding, N_1 is automatically selected in the range $7 \leq N_1 \leq 13$ based on σ, $\tau_{match} = 0.233$, $N_2 = 28$, $N_{step} = 4$, $N_S = 73$, $\beta = 4$, $\lambda_{th2D} = 0.82$, and $\lambda_{th3D} = 0.75$. For the Wiener filtering, N_1 is automatically selected in the range $7 \leq N_1 \leq 11$ based on σ, $\tau_{match} = \frac{\sigma}{4000} + 0.0105$, $N_2 = 72$, $N_{step} = 3$, $N_S = 35$, and $\beta = 3$. In Table 1, we summarize the results of the proposed technique in terms of output peak signal-to-noise ratio (PSNR) in decibels (dB), which is defined as

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{255^2}{|X|^{-1} \sum_{x \in X} \left(y\left(x\right) - \widehat{y}\left(x\right)\right)^2} \right).$$

At `http://www.cs.tut.fi/~foi/3D-DFT`, we provide a collection of the original and denoised test images that were used in our experiments, together with the algorithm implementation (as C++ and MATLAB functions) which produced all reported results. With the mentioned parameters, the execution time of the whole algorithm is less than 9 seconds for an input image of size 256 × 256 on a 3 GHz Pentium machine.

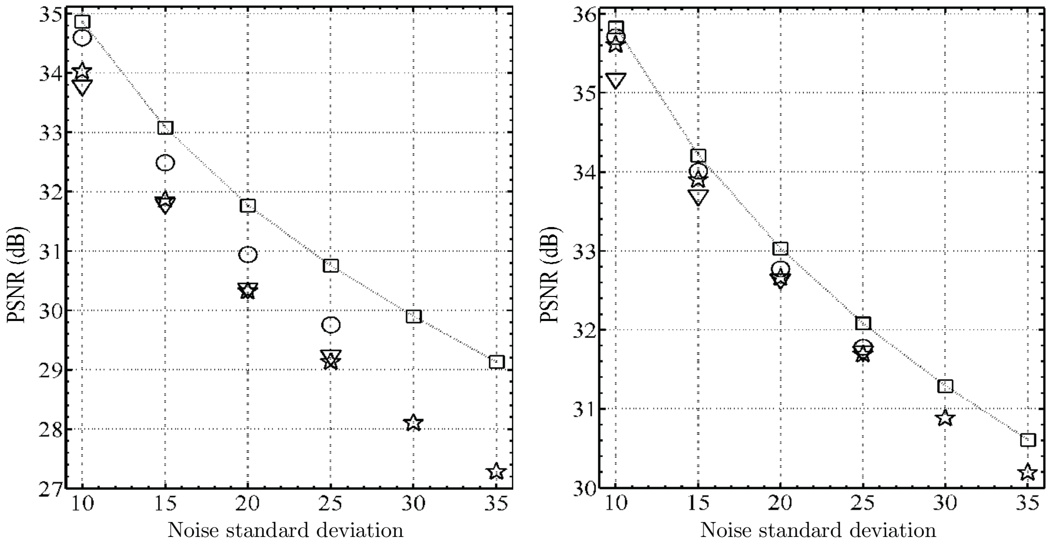
In Figure 3, we compare the output PSNR of our method with the reported ones of three[6, 7, 10] state-of-art techniques known to the authors as best. However, for standard deviations 30 and 35 we could neither find nor reproduce the results of both the *FSP+TUP*[7] and the exemplar-based[10] techniques, thus they are omitted.

In Figure 4, we show noisy ($\sigma = 35$) *House* image and the corresponding denoised one. For this test image, similarity among neighboring blocks is easy to perceive in the uniform regions and in the regular-shaped structures. Hence, those details are well-preserved in our estimate. It is worth referring to Figure 1, where block-matching is illustrated for a fragment of *House.*

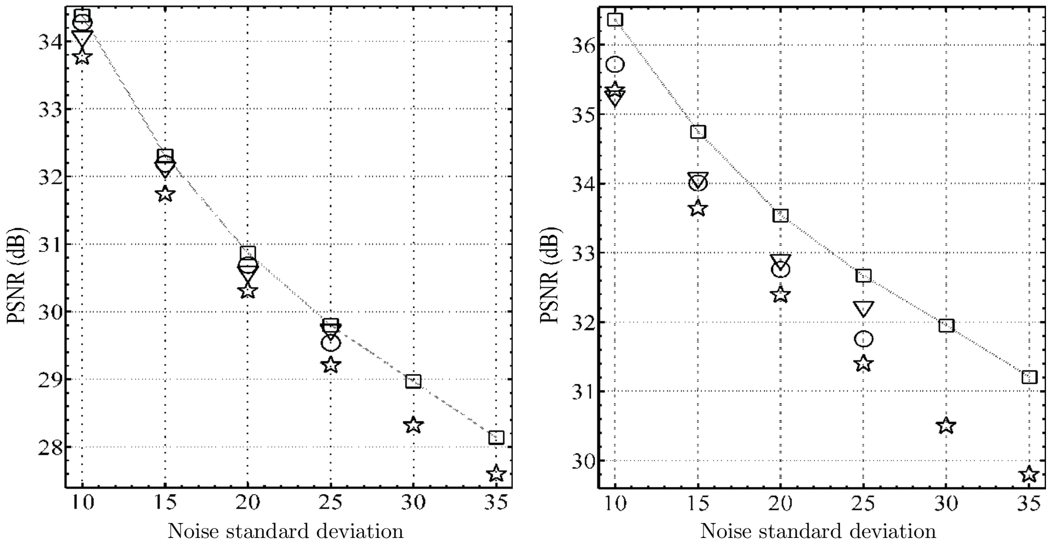
Pairs of noisy ($\sigma = 35$) and denoised *Lena* and *Hill* images are shown in Figures 5 and 6, respectively. The enlarged fragments in each figure help to demonstrate the good quality of the denoised images in terms of faithful detail preservation (stripes on the hat in *Lena* and the pattern on the roof in *Hill*).

We show fragments of noisy ($\sigma = 50$) and denoised *Lena, Barbara, Couple*, and *Boats* images in Figure 7. For this relatively high level of noise, there are very few disturbing artifacts and the proposed technique attains good preservation of: sharp details (the table legs in *Barbara* and the poles in *Boats*), smooth regions (the cheeks of Lena and the suit of the man in *Couple*), and oscillatory patterns (the table cover in *Barbara*). A fragment of *Couple* corrupted by noise of various standard deviations is presented in Figure 8.

In order to demonstrate the capability of the proposed method to preserve textures, we show fragments of heavily noisy ($\sigma = 100$) and denoised *Barbara* in Figure 9. Although the true signal is almost completely buried under noise, the stripes on the clothes are faithfully restored in the estimate.



(a) *Barbara* (b) *Lena*



(c) *Peppers* (d) *House*

Figure 3. Output PSNR as a function of the standard deviation for *Barbara* (a), *Lena* (b), *Peppers* (c), and *House* (d). The notation is: proposed method (squares), *FSP+TUP*[7] (circles), *BLS-GSM*[6] (stars), and exemplar-based[10] (triangles).

Figure 4. On the left are a noisy ($\sigma = 35$) *House* and two enlarged fragments from it; on the right are the denoised image (PSNR 31.21 dB) and the corresponding fragments.

We conclude by remarking that the proposed method outperforms—in terms of objective criteria—all techniques known to us. Moreover, our estimates retain good visual quality even for relatively high levels of noise.

Our current research extends the presented approach by the adoption of variable-sized blocks and shape-adaptive transforms,[13] thus further improving the adaptivity to the structures of the underlying image. Also, application of the technique to more general restoration problems is being considered.

REFERENCES

1. D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Stat. Assoc.*, vol. **90**, pp. 1200–1224, 1995.
2. D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Theory*, vol. **41**, pp. 613–627, 1995.
3. S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Processing*, vol. **9**, pp. 1532–1546, 2000.
4. A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Processing*, vol. **11**, pp. 545–557, 2002.
5. L. Sendur and I. W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Processing Letters*, vol. **9**, pp. 438–441, 2002.
6. J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Processing*, vol. **12**, pp. 1338–1351, 2003.
7. J. A. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *Proc. of IEEE Int'l Conf on Image Processing*, Genoa, Italy, September 2005.

Figure 5. On the left are noisy ($\sigma = 35$) *Lena* and two enlarged fragments from it; on the right are the denoised image (PSNR 30.61 dB) and the corresponding fragments.

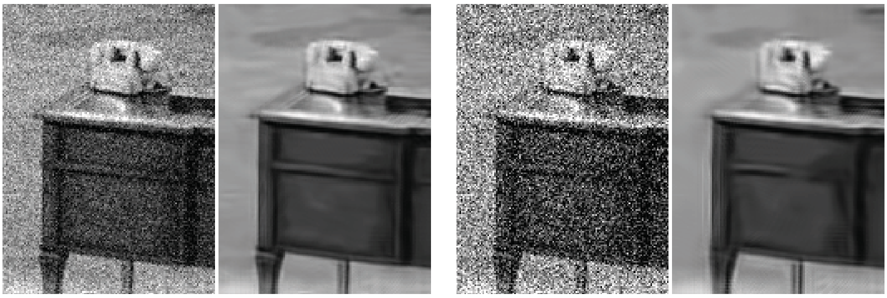
Figure 6. On the left are noisy ($\sigma = 35$) *Hill* and two fragments from it; on the right are the denoised image (PSNR 28.46 dB) and the corresponding fragments from it.

(a) *Lena* ($\sigma = 50$, PSNR 29.08 dB) (b) *Barbara* ($\sigma = 50$, PSNR 27.51 dB)

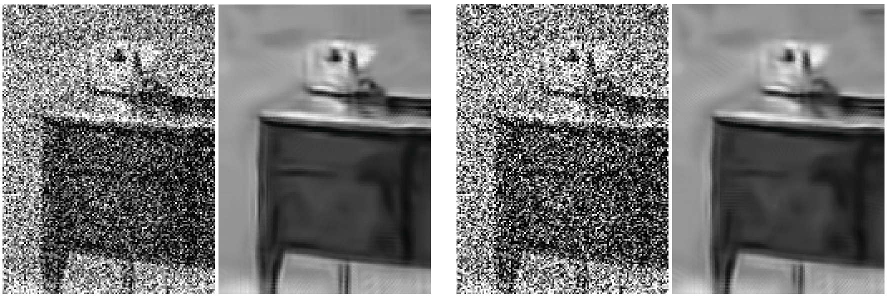
(c) *Couple* ($\sigma = 50$, PSNR 26.46 dB) (d) *Boats* ($\sigma = 50$, PSNR 26.71 dB)

Figure 7. Fragments of noisy ($\sigma = 50$) and denoised test images.

8. L. Yaroslavsky, K. Egiazarian, and J. Astola, "Transform domain image restoration methods: review, comparison and interpretation," in *Nonlinear Image Processing and Pattern Analysis XII*, *Proc. SPIE* **4304**, pp. 155–169, 2001.
9. R. Öktem, L. Yaroslavsky and K. Egiazarian, "Signal and image denoising in transform domain and wavelet shrinkage: a comparative study," in *Proc. of EUSIPCO'98*, Rhodes, Greece, September 1998.
10. C. Kervrann and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation," *Research Report INRIA*, RR-5624, July 2005.
11. A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. **4**, pp. 490–530, 2005.
12. D. Rusanovskyy and K. Egiazarian, "Video denoising algorithm in sliding 3D DCT domain," in *Proc. of ACIVS'05*, Antwerp, Belgium, September 2005.
13. A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-Adaptive DCT for Denoising and Image Reconstruction," in *Electronic Imaging'06, Proc. SPIE* **6064**, no. 6064A-18, San Jose, California USA, 2006.



(a) $\sigma = 25$, PSNR 29.57 dB (b) $\sigma = 50$, PSNR 26.46 dB



(c) $\sigma = 75$, PSNR 24.74 dB (d) $\sigma = 100$, PSNR 23.60 dB

Figure 8. Pairs of fragments of noisy and denoised *Couple* for standard deviations: 25 (a), 50 (b), 75 (c), and 100 (d).

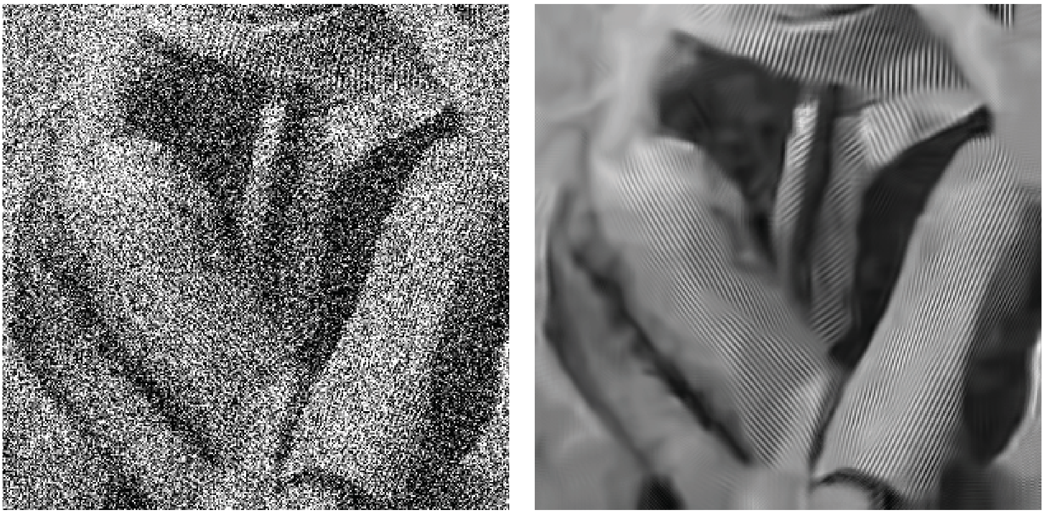


Figure 9. Fragments of noisy ($\sigma = 100$) and denoised (PSNR 24.14 dB) *Barbara*.

Publication [P2]

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

IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 16, NO. 8, AUGUST 2007

Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering

Kostadin Dabov, *Student Member, IEEE*, Alessandro Foi, Vladimir Katkovnik, and
Karen Egiazarian, *Senior Member, IEEE*

Abstract—**We propose a novel image denoising strategy based on an enhanced sparse representation in transform domain. The enhancement of the sparsity is achieved by grouping similar 2-D image fragments (e.g., blocks) into 3-D data arrays which we call "groups." Collaborative filtering is a special procedure developed to deal with these 3-D groups. We realize it using the three successive steps: 3-D transformation of a group, shrinkage of the transform spectrum, and inverse 3-D transformation. The result is a 3-D estimate that consists of the jointly filtered grouped image blocks. By attenuating the noise, the collaborative filtering reveals even the finest details shared by grouped blocks and, at the same time, it preserves the essential unique features of each individual block. The filtered blocks are then returned to their original positions. Because these blocks are overlapping, for each pixel, we obtain many different estimates which need to be combined. Aggregation is a particular averaging procedure which is exploited to take advantage of this redundancy. A significant improvement is obtained by a specially developed collaborative Wiener filtering. An algorithm based on this novel denoising strategy and its efficient implementation are presented in full detail; an extension to color-image denoising is also developed. The experimental results demonstrate that this computationally scalable algorithm achieves state-of-the-art denoising performance in terms of both peak signal-to-noise ratio and subjective visual quality.**

Index Terms—**Adaptive grouping, block matching, image denoising, sparsity, 3-D transform shrinkage.**

I. INTRODUCTION

PLENTY of denoising methods exist, originating from various disciplines such as probability theory, statistics, partial differential equations, linear and nonlinear filtering, and spectral and multiresolution analysis. All these methods rely on some explicit or implicit assumptions about the true (noise-free) signal in order to separate it properly from the random noise.

In particular, the transform-domain denoising methods typically assume that the true signal can be well approximated by a linear combination of few basis elements. That is, the signal is sparsely represented in the transform domain. Hence, by preserving the few high-magnitude transform coefficients that convey mostly the true-signal energy and discarding the rest which are mainly due to noise, the true signal can be effectively estimated. The sparsity of the representation depends on both the transform and the true-signal's properties.

The multiresolution transforms can achieve good sparsity for spatially localized details, such as edges and singularities. Because such details are typically abundant in natural images and convey a significant portion of the information embedded therein, these transforms have found a significant application for image denoising. Recently, a number of advanced denoising methods based on multiresolution transforms have been developed, relying on elaborate statistical dependencies between coefficients of typically overcomplete (e.g., translation-invariant and multiply-oriented) transforms. Examples of such image denoising methods can be seen in [1]–[4].

Not limited to the wavelet techniques, the overcomplete representations have traditionally played an important role in improving the restoration abilities of even the most basic transform-based methods. This is manifested by the sliding-window transform-domain image denoising methods [5], [6] where the basic idea is to apply shrinkage in local (windowed) transform domain. There, the overlap between successive windows accounts for the overcompleteness, while the transform itself is typically orthogonal, e.g., the 2-D DCT.

However, the overcompleteness by itself is not enough to compensate for the ineffective shrinkage if the adopted transform cannot attain a sparse representation of certain image details. For example, the 2-D DCT is not effective in representing sharp transitions and singularities, whereas wavelets would typically perform poorly for textures and smooth transitions. The great variety in natural images makes impossible for any fixed 2-D transform to achieve good sparsity for all cases. Thus, the commonly used orthogonal transforms can achieve sparse representations only for particular image patterns.

The adaptive principal components of local image patches was proposed by Muresan and Parks [7] as a tool to overcome the mentioned drawbacks of standard orthogonal transforms. This approach produces good results for highly-structured image patterns. However, the computation of the correct PCA basis is essentially deteriorated by the presence of noise. With similar intentions, the K-SVD algorithm [8] by Elad and Aharon utilizes highly overcomplete dictionaries obtained via a preliminary training procedure. A shortcoming of these techniques is that both the PCA and learned dictionaries impose a very high computational burden.

Another approach [9] is to exploit a shape-adaptive transform on neighborhoods whose shapes are adaptive to salient image details and, thus, contain mostly homogeneous signal.

Manuscript received December 29, 2006; revised March 24, 2007. This work was supported by the Academy of Finland, Project 213462 (Finnish Centre of Excellence program 2006–2011). K. Dabov was supported by Tampere Graduate School in Information Science and Engineering (TISE). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stanley J. Reeves.

The authors are with the Institute of Signal Processing, Tampere University of Technology, 33101 Tampere, Finland (e-mail: kostadin.dabov@tut.fi; alessandro.foi@tut.fi; vladimir.katkovnik@tut.fi; karen.egiazarian@tut.fi).

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.2007.901238

The shape-adaptive transform can achieve a very sparse representation of the true signal in these adaptive neighborhoods.

Recently, an elaborate adaptive spatial estimation strategy, the *nonlocal means*, was introduced [10]. This approach is different from the transform domain ones. Its basic idea is to build a pointwise estimate of the image where each pixel is obtained as a weighted average of pixels centered at regions that are similar to the region centered at the estimated pixel. The estimates are nonlocal as in principle the averages can be calculated over all pixels of the image. A significant extension of this approach is the exemplar-based estimator [11], which exploits pairwise hypothesis testing to define adaptive nonlocal estimation neighborhoods and achieves results competitive to the ones produced by the best transform-based techniques.

In this paper, we propose a novel image denoising strategy based on an enhanced sparse representation in transform-domain. The enhancement of the sparsity is achieved by *grouping* similar 2-D fragments of the image into 3-D data arrays which we call "groups." *Collaborative filtering* is a special procedure developed to deal with these 3-D groups. It includes three successive steps: 3-D transformation of a group, shrinkage of transform spectrum, and inverse 3-D transformation. Thus, we obtain the 3-D estimate of the group which consists of an array of jointly filtered 2-D fragments. Due to the similarity between the grouped fragments, the transform can achieve a highly sparse representation of the true signal so that the noise can be well separated by shrinkage. In this way, the collaborative filtering reveals even the finest details shared by grouped fragments and at the same time it preserves the essential unique features of each individual fragment.

An image denoising algorithm based on this novel strategy is developed and described in detail. It generalizes and improves our preliminary algorithm introduced in [12]. A very efficient algorithm implementation offering effective complexity/performance tradeoff is developed. Experimental results demonstrate that it achieves outstanding denoising performance in terms of both peak signal-to-noise ratio (PSNR) and subjective visual quality, superior to the current state-of-the-art. Extension to color-image denoising based on [13] is also presented.

The paper is organized as follows. We introduce the grouping and collaborative filtering concepts in Section II. The developed image denoising algorithm is described in Section III. An efficient and scalable realization of this algorithm can be found in Section IV and its extension to color-image denoising is given in Section V. Experimental results are presented in Section VI. Section VII gives an overall discussion of the developed approach and Section VIII contains relevant conclusions.

II. GROUPING AND COLLABORATIVE FILTERING

We denominate *grouping* the concept of collecting similar d-dimensional fragments of a given signal into a $d+1$-dimensional data structure that we term "group." In the case of images for example, the signal fragments can be arbitrary 2-D neighborhoods (e.g., image patches or blocks). There, a group is a 3-D array formed by stacking together similar image neighborhoods. If the neighborhoods have the same shape and size, the formed 3-D array is a generalized cylinder. The importance of grouping is to enable the use of a higher dimensional filtering of each group, which exploits the potential similarity (correlation, affinity, etc.) between grouped fragments in order to estimate the true signal in each of them. This approach we denominate *collaborative filtering*.

A. Grouping

Grouping can be realized by various techniques; e.g., K-means clustering [14], self-organizing maps [15], fuzzy clustering [16], vector quantization [17], and others. There exist a vast literature on the topic; we refer the reader to [18] for a detailed and systematic overview of these approaches.

Similarity between signal fragments is typically computed as the inverse of some distance measure. Hence, a smaller distance implies higher similarity. Various distance measures can be employed, such as the ℓ^p-norm of the difference between two signal fragments. Other examples are the weighted Euclidean distance ($p = 2$) used in the nonlocal means estimator [10], and also the normalized distance used in the exemplar-based estimator [11]. When processing complex or uncertain (e.g., noisy) data, it might be necessary to first extract some features from the signal and then to measure the distance for these features only [18].

B. Grouping by Matching

Grouping techniques such as vector quantization or K-means clustering are essentially based on the idea of partitioning. It means that they build groups or clusters (classes) which are disjoint, in such a way that each fragment belongs to one and only one group. Constructing disjoint groups whose elements enjoy high mutual similarity typically requires recursive procedures and can be computationally demanding [18]. Furthermore, the partitioning causes unequal treatment of the different fragments because the ones that are close to the centroid of the group are better represented than those far from it. This happens always, even in the special case where all fragments of the signal are equidistantly distributed.

A much simpler and effective grouping of mutually similar signal fragments can be realized by *matching* where, in contrast to the above partitioning methods, the formed groups are not necessarily disjoint. Matching is a method for finding signal fragments similar to a given *reference* one. That is achieved by pairwise testing the similarity between the reference fragment and candidate fragments located at different spatial locations. The fragments whose distance (i.e., dissimilarity) from the reference one is smaller than a given threshold are considered mutually similar and are subsequently grouped. The similarity plays the role of the membership function for the considered group and the reference fragment can be considered as some sort of "centroid" for the group. Any signal fragment can be used as a reference one, and, thus, a group can be constructed for it.

We remark that for most distance measures, establishing a bound on the distance between the reference fragment and all of the matched ones means that the distance between any two fragments in that group is also bounded. Roughly speaking, this bound is the diameter of the group. While for an arbitrary distance measure such a statement may not hold precisely, for the case of metrics (e.g., ℓ^p-norms) it is just a direct consequence of the triangle inequality.

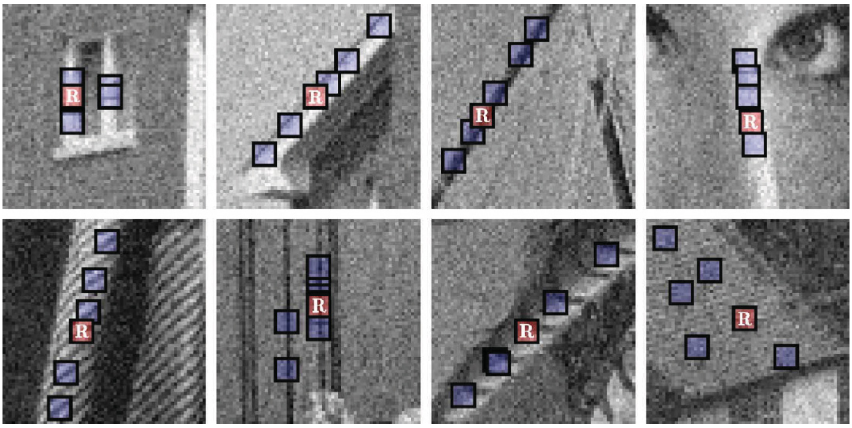


Fig. 1. Illustration of grouping blocks from noisy natural images corrupted by white Gaussian noise with standard deviation 15 and zero mean. Each fragment shows a reference block marked with "R" and a few of the blocks matched to it.

Block-matching (BM) is a particular matching approach that has been extensively used for motion estimation in video compression (MPEG 1, 2, and 4, and H.26x). As a particular way of grouping, it is used to find similar blocks, which are then stacked together in a 3-D array (i.e., a group). An illustrative example of grouping by block-matching for images is given in Fig. 1, where we show a few reference blocks and the ones matched as similar to them.

C. Collaborative Filtering

Given a group of n fragments, the collaborative filtering of the group produces n estimates, one for each of the grouped fragments. In general, these estimates can be different. The term "collaborative" is taken literally, in the sense that each grouped fragment collaborates for the filtering of all others, and vice versa.

Let us consider an illustrative example of collaborative filtering for the estimation of the image in Fig. 2 from an observation (not shown) corrupted by additive zero-mean independent noise. In particular, let us focus on the already grouped blocks shown in the same figure. These blocks exhibit perfect mutual similarity, which makes the elementwise averaging (i.e., averaging between pixels at the same relative positions) a suitable estimator. Hence, for each group, this collaborative averaging produces estimates of all grouped blocks. Because the corresponding noise-free blocks are assumed to be identical, the estimates are unbiased. Therefore, the final estimation error is due only to the residual variance which is inversely proportional to the number of blocks in the group. Regardless of how complex the signal fragments are, we can obtain very good estimates provided that the groups contain a large number of fragments.

However, perfectly identical blocks are unlikely in natural images. If nonidentical fragments are allowed within the same group, the estimates obtained by elementwise averaging become biased. The bias error can account for the largest share of the overall final error in the estimates, unless one uses an estimator that allows for producing a different estimate of each grouped fragment. Therefore, a more effective collaborative filtering strategy than averaging should be employed.

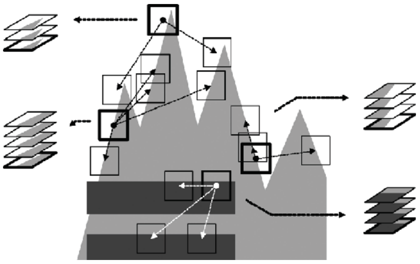


Fig. 2. Simple example of grouping in an artificial image, where for each reference block (with thick borders) there exist perfectly similar ones.

D. Collaborative Filtering by Shrinkage in Transform Domain

An effective collaborative filtering can be realized as shrinkage in transform domain. Assuming $d+1$-dimensional groups of similar signal fragments are already formed, the collaborative shrinkage comprises of the following steps.

- Apply a $d+1$-dimensional linear transform to the group.
- Shrink (e.g., by soft- and hard-thresholding or Wiener filtering) the transform coefficients to attenuate the noise.
- Invert the linear transform to produce estimates of all grouped fragments.

This collaborative transform-domain shrinkage can be particularly effective when applied to groups of natural image fragments, e.g., the ones in Fig. 1. These groups are characterized by both:

- intrafragment correlation which appears between the pixels of each grouped fragment—a peculiarity of natural images;
- interfragment correlation which appears between the corresponding pixels of different fragments—a result of the similarity between grouped fragments.

The 3-D transform can take advantage of both kinds of correlation and, thus, produce a sparse representation of the true signal in the group. This sparsity makes the shrinkage very effective in attenuating the noise while preserving the features of the signal.

Let us give a simple illustration of the benefit of this collaborative shrinkage by considering the grouped image blocks shown in Fig. 1. Let us first consider the case when no collaborative filtering is performed but instead a 2-D transform is applied separately to each individual block in a given group of n fragments. Since these grouped blocks are very similar, for any of them we should get approximately the same number, say α, of significant transform coefficients. It means that the whole group of n fragments is represented by $n\alpha$ coefficients. In contrast, in the case of collaborative filtering, in addition to the 2-D transform, we apply a 1-D transform across the grouped blocks (equivalent to applying a separable 3-D transform to the whole group). If this 1-D transform has a DC-basis element, then because of the high similarity between the blocks, there are approximately[1] only α significant coefficients that represent the whole group instead of $n\alpha$. Hence, the grouping enhances the sparsity, which increases with the number of grouped blocks.

As Fig. 1 demonstrates, a strong similarity between small image blocks at different spatial locations is indeed very common in natural images. It is a characteristic of blocks that belong to uniform areas, edges, textures, smooth intensity gradients, etc. Therefore, the existence of mutually similar blocks can be taken as a very realistic assumption when modeling natural images, which strongly motivates the use of grouping and collaborative filtering for an image denoising algorithm.

III. ALGORITHM

In the proposed algorithm, the grouping is realized by block-matching and the collaborative filtering is accomplished by shrinkage in a 3-D transform domain. The used image fragments are square blocks of fixed size. The general procedure carried out in the algorithm is as follows. The input noisy image is processed by successively extracting reference blocks from it and for each such block:

- find blocks that are similar to the reference one (block-matching) and stack them together to form a 3-D array (group);
- perform collaborative filtering of the group and return the obtained 2-D estimates of all grouped blocks to their original locations.

After processing all reference blocks, the obtained block estimates can overlap, and, thus, there are multiple estimates for each pixel. We aggregate these estimates to form an estimate of the whole image.

This general procedure is implemented in two different forms to compose a two-step algorithm. This algorithm is illustrated in Fig. 3 and proceeds as follows.

Step 1) *Basic estimate*.
  a) *Block-wise estimates*. For each block in the noisy image, do the following.
    i) *Grouping*. Find blocks that are similar to the currently processed one and then stack them together in a 3-D array (group).
    ii) *Collaborative hard-thresholding*. Apply a 3-D transform to the formed group, attenuate the noise by hard-thresholding of the transform coefficients, invert the 3-D transform to produce estimates of all grouped blocks, and return the estimates of the blocks to their original positions.
  b) *Aggregation*. Compute the basic estimate of the true-image by weighted averaging all of the obtained block-wise estimates that are overlapping.

Step 2) *Final estimate*: Using the basic estimate, perform improved grouping and collaborative Wiener filtering.
  a) *Block-wise estimates*. For each block, do the following.
    i) *Grouping*. Use BM within the basic estimate to find the locations of the blocks similar to the currently processed one. Using these locations, form two groups (3-D arrays), one from the noisy image and one from the basic estimate.
    ii) *Collaborative Wiener filtering*. Apply a 3-D transform on both groups. Perform Wiener filtering on the noisy one using the energy spectrum of the basic estimate as the true (pilot) energy spectrum. Produce estimates of all grouped blocks by applying the inverse 3-D transform on the filtered coefficients and return the estimates of the blocks to their original positions.
  b) *Aggregation*. Compute a final estimate of the true-image by aggregating all of the obtained local estimates using a weighted average.

There are two significant motivations for the second step in the above algorithm:

- using the basic estimate instead of the noisy image allows to improve the grouping by block-matching;
- using the basic estimate as the pilot signal for the empirical Wiener filtering is much more effective and accurate than the simple hard-thresholding of the 3-D spectrum of the noisy data.

Observation Model and Notation: We consider a noisy image $z : X \rightarrow \mathbb{R}$ of the form

$$z(x) = y(x) + \eta(x), \quad x \in X$$

where x is a 2-D spatial coordinate that belongs to the image domain $X \subset \mathbb{Z}^2$, y is the true image, and η is i.i.d. zero-mean Gaussian noise with variance σ^2, $\eta(\cdot) \sim \mathcal{N}(0, \sigma^2)$. With Z_x we denote a block of fixed size $N_1 \times N_1$ extracted from z, where x is the coordinate of the top-left corner of the block. Alternatively, we say that Z_x is located at x in z. A group of collected 2-D blocks is denoted by a bold-face capital letter with a subscript

---

[1] This is just a qualitative statement because the actual number of significant coefficients depends on the normalization of the transforms and on the thresholds used for the 2-D and 3-D cases.

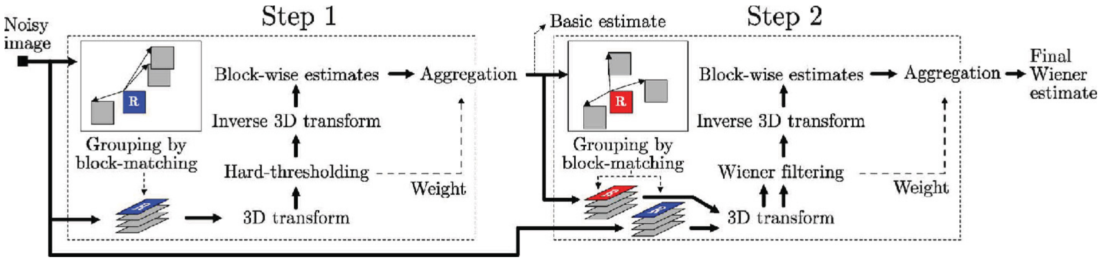


Fig. 3. Flowchart of the proposed image denoising algorithm. The operations surrounded by dashed lines are repeated for each processed block (marked with "R").

that is the set of its grouped blocks' coordinates, e.g., $\mathbf{Z}_S$ is a 3-D array composed of blocks Z_x located at $x \in S \subseteq X$. In order to distinguish between parameters used in the first and in the second step, we respectively use the superscripts "ht" (hard-thresholding) and "wie" (Wiener filtering). For example, N_1^{ht} is the block size used in Step 1 and N_1^{wie} is the block size used in Step 2. Analogously, we denote the basic estimate with $\widehat{y}^{\text{basic}}$ and the final estimate with $\widehat{y}^{\text{final}}$.

The following subsections present in detail the steps of the proposed denoising method.

A. *Steps 1a and 2a: Block-Wise Estimates*

In this step, we process reference image blocks in a sliding-window manner. Here, "process" stands for performing grouping and estimating the true signal of all grouped blocks by:

- collaborative hard-thresholding in Step 1aii;
- collaborative Wiener filtering in Step 2aii.

The resultant estimates are denominated "block-wise estimates."

Because Steps 1a and 2a bear the same structure, we respectively present them in the following two sections. Therein, we fix the currently processed image block as Z_{x_R} (located at the current coordinate $x_R \in X$) and denominate it "reference block."

1) Steps 1ai and 1aii: Grouping and Collaborative Hard-Thresholding: We realize grouping by block-matching within the noisy image z, as discussed in Section II-B. That is, only blocks whose distance (dissimilarity) with respect to the reference one is smaller than a fixed threshold are considered similar and grouped. In particular, we use the ℓ^2-distance as a measure of dissimilarity.

Ideally, if the true-image y would be available, the block-distance could be calculated as

$$d^{\text{ideal}}\left(Z_{x_R}, Z_x\right) = \frac{\|Y_{x_R} - Y_x\|_2^2}{\left(N_1^{\text{ht}}\right)^2} \tag{1}$$

where $\|\cdot\|_2$ denotes the ℓ^2-norm and the blocks Y_{x_R} and Y_x are respectively located at x_R and $x \in X$ in y. However, only the noisy image z is available and the distance can only be calculated from the noisy blocks Z_{x_R} and Z_x as

$$d^{\text{noisy}}\left(Z_{x_R}, Z_x\right) = \frac{\|Z_{x_R} - Z_x\|_2^2}{\left(N_1^{\text{ht}}\right)^2}. \tag{2}$$

If the blocks Z_{x_R} and Z_x do not overlap, this distance is a non-central chi-squared random variable with mean

$$E\left\{d^{\text{noisy}}\left(Z_{x_R}, Z_x\right)\right\} = d^{\text{ideal}}\left(Z_{x_R}, Z_x\right) + 2\sigma^2$$

and variance

$$var\left\{d^{\text{noisy}}\left(Z_{x_R}, Z_x\right)\right\} = \frac{8\sigma^4}{\left(N_1^{\text{ht}}\right)^2} + \frac{8\sigma^2 d^{\text{ideal}}\left(Z_{x_R}, Z_x\right)}{\left(N_1^{\text{ht}}\right)^2}. \tag{3}$$

The variance grows asymptotically with $\mathcal{O}(\sigma^4)$. Thus, for relatively large σ or small N_1^{ht}, the probability densities of the different $d^{\text{noisy}}(Z_{x_R}, Z_x)$ are likely to overlap heavily and this results in erroneous grouping.[2] That is, blocks with greater ideal distances than the threshold are matched as similar, whereas blocks with smaller such distances are left out.

To avoid the above problem, we propose to measure the block-distance using a coarse prefiltering. This prefiltering is realized by applying a normalized 2-D linear transform on both blocks and then hard-thresholding the obtained coefficients, which results in

$$d\left(Z_{x_R}, Z_x\right) = \frac{\left\|\Upsilon'\left(\mathcal{T}_{\text{2D}}^{\text{ht}}\left(Z_{x_R}\right)\right) - \Upsilon'\left(\mathcal{T}_{\text{2D}}^{\text{ht}}(Z_x)\right)\right\|_2^2}{\left(N_1^{\text{ht}}\right)^2} \tag{4}$$

where Υ' is the hard-thresholding operator with threshold $\lambda_{\text{2D}}\sigma$ and $\mathcal{T}_{\text{2D}}^{\text{ht}}$ denotes the normalized 2-D linear transform.[3]

Using the d-distance (4), the result of BM is a set that contains the coordinates of the blocks that are similar to Z_{x_R}

$$S_{x_R}^{\text{ht}} = \left\{x \in X : d\left(Z_{x_R}, Z_x\right) \leq \tau_{\text{match}}^{\text{ht}}\right\} \tag{5}$$

where the fixed $\tau_{\text{match}}^{\text{ht}}$ is the maximum d-distance for which two blocks are considered similar. The parameter $\tau_{\text{match}}^{\text{ht}}$ is selected from deterministic speculations about the acceptable value of the ideal difference, mainly ignoring the noisy components of the signal. Obviously $d(Z_{x_R}, Z_{x_R}) = 0$, which implies that $|S_{x_R}^{\text{ht}}| \geq 1$, where $|S_{x_R}^{\text{ht}}|$ denotes the cardinality of $S_{x_R}^{\text{ht}}$. After obtaining $S_{x_R}^{\text{ht}}$, a group is formed by stacking the matched noisy blocks $Z_{x \in S_{x_R}^{\text{ht}}}$ to form a 3-D array of size $N_1^{\text{ht}} \times N_1^{\text{ht}} \times |S_{x_R}^{\text{ht}}|$,

---

[2] The effect of this is the sharp drop of the output-PSNR observed for two of the graphs in Fig. 9 at about $\sigma = 40$.

[3] For simplicity, we do not invert the transform $\mathcal{T}_{2D}^{ht}$ and compute the distance directly from the spectral coefficients. When $\mathcal{T}_{2D}^{ht}$ is orthonormal, the distance coincides with the ℓ^2-distance calculated between the denoised block-estimates in space domain.

which we denote $\mathbf{Z}_{S^{\text{ht}}_{x_R}}$. The matched blocks can in general overlap. We do not restrict the ordering, which is discussed in Section IV-B.

The collaborative filtering of $\mathbf{Z}_{S^{\text{ht}}_{x_R}}$ is realized by hard-thresholding in 3-D transform domain. The adopted normalized 3-D linear transform, denoted $\mathcal{T}^{\text{ht}}_{\text{3D}}$, is expected to take advantage of the two types of correlation, discussed in Section II-D, and attain good sparsity for the true signal group $\mathbf{Y}_{S^{\text{ht}}_{x_R}}$. This allows for effective noise attenuation by hard-thresholding, followed by inverse transform that yields a 3-D array of block-wise estimates

$$\widehat{\mathbf{Y}}^{\text{ht}}_{S^{\text{ht}}_{x_R}} = \mathcal{T}_{\text{3D}}^{\text{ht}^{-1}}\left(\Upsilon\left(\mathcal{T}_{\text{3D}}^{\text{ht}}\left(\mathbf{Z}_{S^{\text{ht}}_{x_R}}\right)\right)\right) \tag{6}$$

where Υ is a hard-threshold operator with threshold $\lambda_{\text{3D}}\sigma$. The array $\widehat{\mathbf{Y}}^{\text{ht}}_{S^{\text{ht}}_{x_R}}$ comprises of $|S^{\text{ht}}_{x_R}|$ stacked block-wise estimates $\widehat{Y}^{\text{ht},x_R}_{x}$, $\forall x \in S^{\text{ht}}_{x_R}$. In $\widehat{Y}^{\text{ht},x_R}_{x}$, the subscript x denotes the location of this block-estimate and the superscript x_R indicates the reference block.

2) Steps 2ai and 2aii: Grouping and Collaborative Wiener Filtering: Given the basic estimate $\widehat{y}^{\text{basic}}$ of the true image obtained in Step 1b, the denoising can be improved by performing grouping within this basic estimate and collaborative empirical Wiener filtering.

Because the noise in $\widehat{y}^{\text{basic}}$ is assumed to be significantly attenuated, we replace the thresholding-based d-distance (4) with the normalized squared ℓ^2-distance computed within the basic estimate. This is a close approximation of the ideal distance (1). Hence, the coordinates of the matched blocks are the elements of the set

$$S^{\text{wie}}_{x_R} = \left\{ x \in X : \frac{\left\|\widehat{Y}^{\text{basic}}_{x_R} - \widehat{Y}^{\text{basic}}_{x}\right\|_2^2}{\left(N_1^{\text{wie}}\right)^2} < \tau^{\text{wie}}_{\text{match}} \right\}. \tag{7}$$

We use the set $S^{\text{wie}}_{x_R}$ in order to form two groups, one from the basic estimate and one from the noisy observation:

- $\widehat{\mathbf{Y}}^{\text{basic}}_{S^{\text{wie}}_{x_R}}$ by stacking together the basic estimate blocks $\widehat{Y}^{\text{basic}}_{x \in S^{\text{wie}}_{x_R}}$;
- $\mathbf{Z}_{S^{\text{wie}}_{x_R}}$ by stacking together the noisy blocks $Z_{x \in S^{\text{wie}}_{x_R}}$.

We define the empirical Wiener shrinkage coefficients from the energy of the 3-D transform coefficients of the basic estimate group as

$$\mathbf{W}_{S^{\text{wie}}_{x_R}} = \frac{\left|\mathcal{T}^{\text{wie}}_{\text{3D}}\left(\widehat{\mathbf{Y}}^{\text{basic}}_{S^{\text{wie}}_{x_R}}\right)\right|^2}{\left|\mathcal{T}^{\text{wie}}_{\text{3D}}\left(\widehat{\mathbf{Y}}^{\text{basic}}_{S^{\text{wie}}_{x_R}}\right)\right|^2 + \sigma^2}. \tag{8}$$

Then the collaborative Wiener filtering of $\mathbf{Z}_{S^{\text{wie}}_{x_R}}$ is realized as the element-by-element multiplication of the 3-D transform coefficients $\mathcal{T}^{\text{wie}}_{\text{3D}}(\mathbf{Z}_{S^{\text{wie}}_{x_R}})$ of the noisy data with the Wiener shrinkage coefficients $\mathbf{W}_{S^{\text{wie}}_{x_R}}$. Subsequently, the inverse transform $\mathcal{T}_{\text{3D}}^{\text{wie}^{-1}}$ produces the group of estimates

$$\widehat{\mathbf{Y}}^{wie}_{S^{\text{wie}}_{x_R}} = \mathcal{T}_{\text{3D}}^{\text{wie}^{-1}}\left(\mathbf{W}_{S^{\text{wie}}_{x_R}}\mathcal{T}^{\text{wie}}_{\text{3D}}\left(\mathbf{Z}_{S^{\text{wie}}_{x_R}}\right)\right). \tag{9}$$

This group comprises of the block-wise estimates $\widehat{Y}^{wie,x_R}_{x}$ located at the matched locations $x \in S^{\text{wie}}_{x_R}$.

B. Steps 1b and 2b: Global Estimate by Aggregation

Each collection of block-wise estimates $\widehat{Y}^{\text{ht},x_R}_{x \in S^{\text{ht}}_{x_R}}$ and $\widehat{Y}^{\text{wie},x_R}_{x \in S^{\text{wie}}_{x_R}}$, $\forall x_R \in X$, obtained respectively in Steps 1a and 2a, is an overcomplete representation of the true-image because in general the block-wise estimates can overlap. In addition, more than one block-estimate can be located at exactly the same coordinate, e.g., $\widehat{Y}^{\text{ht},x_a}_{x_b}$ and $\widehat{Y}^{\text{ht},x_b}_{x_b}$ are both located at x_b but obtained while processing the reference blocks at x_a and x_b, respectively. One can expect substantially overcomplete representation of the signal in regions where there are plenty of overlapping block-wise estimates, i.e., where a block is matched (similar) to many others. Hence, the redundancy of the method depends on the grouping and, therefore, also on the particular image.

To compute the basic and the final estimates of the true-image in Steps 1b and 2b, respectively, we aggregate the corresponding block-wise estimates $\widehat{Y}^{\text{ht},x_R}_{x \in S^{\text{ht}}_{x_R}}$ and $\widehat{Y}^{\text{wie},x_R}_{x \in S^{\text{wie}}_{x_R}}$, $\forall x_R \in X$. This aggregation is performed by a weighted averaging at those pixel positions where there are overlapping block-wise estimates. The selection of weights is discussed in the following section.

1) Aggregation Weights: In general, the block-wise estimates are statistically correlated, biased, and have different variance for each pixel. However, it is quite demanding to take into consideration all these effects. Similarly to [6] and [9], we found that a satisfactory choice for aggregation weights would be ones that are inversely proportional to the total sample variance of the corresponding block-wise estimates. That is, noisier block-wise estimates should be awarded smaller weights. If the additive noise in the groups $\mathbf{Z}_{S^{\text{ht}}_{x_R}}$ and $\mathbf{Z}_{S^{\text{wie}}_{x_R}}$ is independent, the total sample variance in the corresponding groups of estimates (6) and (9) is respectively equal to $\sigma^2 N^{x_R}_{\text{har}}$ and $\sigma^2 \|\mathbf{W}_{S^{\text{wie}}_{x_R}}\|_2^2$, where $N^{x_R}_{\text{har}}$ is the number of retained (nonzero) coefficients after hard-thresholding and $\mathbf{W}_{S^{\text{wie}}_{x_R}}$ are the Wiener filter coefficients (8). Based on this, in Step 1b for each $x_R \in X$, we assign the weight

$$w^{\text{ht}}_{x_R} = \begin{cases} \frac{1}{\sigma^2 N^{x_R}_{\text{har}}}, & \text{if } N^{x_R}_{\text{har}} \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{10}$$

for the group of estimates $\widehat{Y}^{\text{ht},x_R}_{x \in S^{\text{ht}}_{x_R}}$. Similarly, in Step 2b for each $x_R \in X$, we assign the weight

$$w^{\text{wie}}_{x_R} = \sigma^{-2}\left\|\mathbf{W}_{S^{\text{wie}}_{x_R}}\right\|_2^{-2} \tag{11}$$

for the group of estimates $\widehat{Y}^{\text{wie},x_R}_{x \in S^{\text{wie}}_{x_R}}$.

We remark that independence of the noise in a group is only achieved when the noisy blocks that build this group do not overlap each other. Therefore, on the one hand, the cost of ensuring independence would constitute a severe restriction for the BM, i.e., allowing matching only among nonoverlapping blocks. On the other hand, if the possible overlaps are considered, the computation of the individual variance of each transform coefficient in $\mathcal{T}^{\text{ht}}_{\text{3D}}(\mathbf{Z}_{S^{\text{ht}}_{x_R}})$ or $\mathcal{T}^{\text{wie}}_{\text{3D}}(\mathbf{Z}_{S^{\text{wie}}_{x_R}})$ becomes a prohibitive complication that requires considering the covariance terms in the corresponding transform coefficients. In our algorithm we use overlapping blocks but do not consider the covariances. Hence, the proposed weights (10) and (11) are only loosely inversely proportional to the corresponding total sample variances.

2) Aggregation by Weighted Average: The global basic estimate $\widehat{y}^{\text{basic}}$ is computed by a weighted average of the block-wise estimates $\widehat{Y}^{\text{ht},x_R}_{x\in S^{\text{ht}}_{x_R}}$ obtained in Step 1a, using the weights $w^{\text{ht}}_{x_R}$ defined in (10), i.e.,

$$\widehat{y}^{\text{basic}}(x) = \frac{\sum\limits_{x_R\in X}\sum\limits_{x_m\in S^{\text{ht}}_{x_R}} w^{\text{ht}}_{x_R}\widehat{Y}^{\text{ht},x_R}_{x_m}(x)}{\sum\limits_{x_R\in X}\sum\limits_{x_m\in S^{\text{ht}}_{x_R}} w^{\text{ht}}_{x_R}\chi_{x_m}(x)}, \quad \forall x\in X \qquad (12)$$

where $\chi_{x_m} : X \rightarrow \{0,1\}$ is the characteristic function of the square support of a block located at $x_m \in X$, and the block-wise estimates $\widehat{Y}^{\text{ht},x_R}_{x_m}$ are zero-padded outside of their support.

The global final estimate $\widehat{y}^{\text{final}}$ is computed by (12), where $\widehat{y}^{\text{basic}}$, $\widehat{Y}^{\text{ht},x_R}_{x_m}$, $S^{\text{ht}}_{x_R}$, and $w^{\text{ht}}_{x_R}$ are replaced respectively by $\widehat{y}^{\text{final}}$, $\widehat{Y}^{\text{wie},x_R}_{x_m}$, $S^{\text{wie}}_{x_R}$, and $w^{\text{wie}}_{x_R}$.

IV. FAST AND EFFICIENT REALIZATION

A straightforward implementation of the method presented in the previous section is highly computationally demanding. In order to realize a practical and efficient algorithm, we impose constraints and exploit certain expedients which we present in the following list.

Reduce the number of processed blocks.

- Rather than sliding by one pixel to every next reference block, use a step of $N_{\text{step}} \in \mathbb{N}$ pixels in both horizontal and vertical directions. Hence, the number of reference blocks is decreased from approximately $|X|$ to $|X|/N^2_{\text{step}}$.

Reduce the complexity of grouping.

- Restrict the maximum size of a group by setting an upper bound $N_2 \in \mathbb{N}$ on the number of grouped blocks; i.e., ensuring $|S_{x_R\in X}| \leq N_2$.
- Search for candidate matching blocks in a local neighborhood of restricted size $N_S \times N_S$ centered about the currently processed coordinate $x_R \in X$.
- To further speed-up the BM, we use *predictive search*, i.e., the search neighborhoods are nonrectangular and depend on the previously matched blocks. We form such a neighborhood as the union of $N_{PR} \times N_{PR}$ (where $N_{PR} \ll N_S$) ones centered at the previous matched coordinates correspondingly shifted by N_{step} in the direction of processing the image, e.g., in horizontal direction for raster scan. For every N_{FS}th processed block, we nevertheless perform an exhaustive-search BM in the larger $N_S \times N_S$ neighborhood. In particular, $N_{FS} = 1$ implies that only exhaustive-search in $N_S \times N_S$ is used.

Reduce the complexity of applying transforms.

- Restrict the transforms $\mathcal{T}^{\text{ht}}_{3\text{D}}$ and $\mathcal{T}^{\text{wie}}_{3\text{D}}$ to the class of separable transforms and use respectively $\mathcal{T}^{\text{ht}}_{2\text{D}}$ and $\mathcal{T}^{\text{wie}}_{2\text{D}}$ across the matched blocks and a 1-D transform, $\mathcal{T}_{1\text{D}}$, along the third dimension of a group, along which the blocks are stacked.
- The spectra $\mathcal{T}^{\text{ht}}_{2\text{D}}(Z_x)$, $\mathcal{T}^{\text{wie}}_{2\text{D}}(Z_x)$, and $\mathcal{T}^{\text{wie}}_{2\text{D}}(\widehat{Y}^{\text{basic}}_x)$ are precomputed for each block in a neighborhood $N_S \times N_S$ around the currently processed coordinate. Later, these are reused for subsequent reference blocks whose $N_S \times N_S$ neighborhoods overlap the current one. Thus, these transforms are computed exactly once for each processed coordinate; e.g., they are not recomputed each time in (4). In addition, in (6), (8), and (9), we compute the forward $\mathcal{T}^{\text{ht}}_{3\text{D}}$ and $\mathcal{T}^{\text{wie}}_{3\text{D}}$ transforms simply by applying $\mathcal{T}_{1\text{D}}$ across precomputed $\mathcal{T}^{\text{ht}}_{2\text{D}}$- and $\mathcal{T}^{\text{wie}}_{2\text{D}}$-transformed blocks, respectively.

Realize efficiently the aggregation.

- First, in Steps 1aii and 2aii, the obtained block-wise estimates are weighted and accumulated in a buffer (with the size of the image). At the same time, the corresponding weights are accumulated at the same locations in another buffer. Then, in Steps 1b and 2b, the aggregation (12) is finally realized by a simple element-wise division between the two buffers.

Reduce the border effects.

- Use a $N_1 \times N_1$ Kaiser window (with parameter β) as part of the weights in (12) in order to reduce border effects which can appear when certain 2-D transforms (e.g., the 2-D DCT, the 2-D DFT, or periodized wavelets) are used.

A. Complexity

The time complexity of the algorithm is $\mathcal{O}(|X|)$ and, thus, depends linearly on the size of the input image, as all parameters are fixed.

Given the restrictions introduced in the previous subsection, without exploiting predictive-search BM, the number of operations per pixel is approximately

$$3\mathcal{C}_{\mathcal{T}_{2\text{D}}} + \frac{2\left(N_1^2 + N_2\right)N_S^2}{N^2_{\text{step}}} + \frac{3\left(N_2\mathcal{C}_{\mathcal{T}_{2\text{D}}} + N_1^2\mathcal{C}_{\mathcal{T}_{1\text{D}}}\right)}{N^2_{\text{step}}}$$

where for simplicity we omit the superscripts "ht" and "wie" from the parameters/operators, and where:

- the first addend is due to precomputing $\mathcal{T}_{2\text{D}}$ for each sliding block (within a $N_S \times N_S$ neighborhood);
- the second is due to grouping by exhaustive-search BM in a $N_S \times N_S$ neighborhood;
- the third addend is due to the transforms $\mathcal{T}_{3\text{D}}$ that is a separable composition of $\mathcal{T}_{2\text{D}}$ and $\mathcal{T}_{1\text{D}}$.

Above, $\mathcal{C}_{\mathcal{T}}$ denotes the number of arithmetic operations required for a transform $\mathcal{T}$; it depends on properties such as availability of fast algorithms, separability, etc. For example, the DFT can be computed efficiently by a fast Fourier transform algorithm and a dyadic wavelet decomposition can be realized efficiently using iterated filterbanks.

By exploiting fast separable transforms and the predictive-search BM, we can significantly reduce the complexity of the algorithm.

B. Parameter Selection

We studied the proposed algorithm using various transforms and parameters. As a results, we propose sets of parameters that are categorized in two profiles, "Normal" and "Fast," presented in Table I. The main characteristics of these profiles are as follows.

TABLE I
PARAMETER SETS FOR THE FAST AND NORMAL PROFILES

		Fast Profile	Normal Profile	
			$\sigma \leq 40$	$\sigma > 40$
Approx. exec. time for a 256×256 grayscale image on 1.5 GHz Celeron M		0.7 sec	4.1 sec	5.8 sec
Parameters for Step 1 (*ht*)	$\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$	2D-Bior1.5	2D-Bior1.5	2D-DCT
	N_1^{ht}	8	8	12
	N_2^{ht}	16	16	16
	$N_{\mathrm{step}}^{\mathrm{ht}}$	6	3	4
	$N_{\mathrm{S}}^{\mathrm{ht}}$	25	39	39
	$N_{\mathrm{FS}}^{\mathrm{ht}}$	6	1	1
	$N_{\mathrm{PR}}^{\mathrm{ht}}$	3	-	-
	β^{ht}	2.0	2.0	2.0
	λ_{2D}	0	0	2.0
	λ_{3D}	2.7	2.7	2.8
	$\tau_{\mathrm{match}}^{\mathrm{ht}}$	2500	2500	5000
Parameters for Step 2 (*wie*)	$\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$	2D-DCT	2D-DCT	2D-DCT
	N_1^{wie}	8	8	11
	N_2^{wie}	16	32	32
	$N_{\mathrm{step}}^{\mathrm{wie}}$	5	3	6
	$N_{\mathrm{S}}^{\mathrm{wie}}$	25	39	39
	$N_{\mathrm{FS}}^{\mathrm{wie}}$	5	1	1
	$N_{\mathrm{PR}}^{\mathrm{wie}}$	2	-	-
	$\tau_{\mathrm{match}}^{\mathrm{wie}}$	400	400	3500
	β^{wie}	2.0	2.0	2.0
Common	$\mathcal{T}_{\mathrm{1D}}$	1D-Haar	1D-Haar	1D-Haar

- Normal Profile. This profile offers a reasonable compromise between computational complexity and denoising performance. It is divided in two cases depending on the level of noise.
  - — $\sigma \leq 40$ the noise is not too severe to affect the correctness of the grouping; hence, the thresholding in the d-distance (4) is disabled by setting $\lambda_{\mathrm{2D}} = 0$ and relatively small block sizes are used, N_1^{ht}, $N_1^{\mathrm{wie}} = 8$.
  - — $\sigma > 40$ corresponds to high level of noise; hence, $\lambda_{\mathrm{2D}} = 2$ is used to improve the correctness of the grouping and larger block sizes are used, $N_1^{\mathrm{ht}} = 12$ and $N_1^{\mathrm{wie}} = 11$.
- Fast Profile. Provides lower computational complexity at the cost of decreased denoising performance. It exploits the proposed fast predictive-search BM (unlike the Normal Profile, which uses only the exhaustive-search BM).

The benefit of using thresholding $(\lambda_{\mathrm{2D}} = 2)$ for the d-distance and larger block sizes when $\sigma > 40$ is illustrated in Fig. 9 and discussed in Section VI.

To show how the denoising performance depends on the choice of the transforms $\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$, $\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$, and $\mathcal{T}_{\mathrm{1D}}$, we present some experimental results in Table II. As already stated, the 3-D transforms $\mathcal{T}_{\mathrm{3D}}^{\mathrm{ht}}$ and $\mathcal{T}_{\mathrm{3D}}^{\mathrm{wie}}$ used in Steps 1 and 2 of our method are formed by a separable composition of $\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$ and $\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$, respectively, with $\mathcal{T}_{\mathrm{1D}}$. Furthermore, both $\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$ and $\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$ are separable compositions of 1-D transforms such as the ones specified in the table. The following normalized transforms were used in our experiment.

- DST, DCT: The discrete sine and cosine transforms.
- WHT: The Walsh–Hadamard transform.
- A few full dyadic wavelet decompositions using the following.
  - — Dbp: The Daubechies wavelet with p vanishing moments, where $p = 1,2,4,6$; when $p = 1$, it coincides with the Haar wavelet.
  - — Bior1.N_{r}: A bi-orthogonal spline wavelet, where the vanishing moments of the decomposing and the reconstructing wavelet functions are 1 and N_{r}, respectively.
- DC+rand: An orthonormal transform that has a DC basis element and the rest of its basis elements have random nature, i.e., obtained by orthonormalization of realizations of a white Gaussian process.

TABLE II
DEPENDENCY OF THE OUTPUT PSNR (dB) ON THE USED TRANSFORMS. THE COLUMNS CORRESPONDING TO $\mathcal{T}_{2D}^{ht}$ CONTAIN PSNR RESULTS OF THE BASIC ESTIMATE $\widehat{y}^{\mathrm{basic}}$ AND ALL OTHER COLUMNS CONTAIN RESULTS OF THE FINAL ESTIMATE $\widehat{y}^{\mathrm{final}}$. THE NOISE IN THE OBSERVATIONS HAD $\sigma = 25$

Transform	*Boats*			*Lena*		
	$\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$	$\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$	$\mathcal{T}_{\mathrm{1D}}$	$\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$	$\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$	$\mathcal{T}_{\mathrm{1D}}$
Haar	29.31	29.84	**29.91**	31.24	31.93	**32.08**
Db2	29.22	29.83	29.90	31.19	31.97	32.06
Db4	29.34	29.88	29.89	31.31	32.01	32.06
Db6	29.30	29.86	29.89	31.28	31.98	32.06
Bior1.3	29.42	29.87	29.90	31.35	31.96	32.06
Bior1.5	**29.43**	29.88	29.90	31.37	31.97	32.06
WHT	29.22	29.84	29.88	31.24	32.00	32.07
DCT	29.35	**29.91**	29.88	**31.42**	**32.08**	32.07
DST	29.33	**29.91**	29.79	31.36	31.97	31.92
DC+rand	29.07	29.75	29.88	31.06	31.88	32.06
DC-only	-	-	28.03	-	-	30.65

In addition, only for $\mathcal{T}_{\mathrm{1D}}$, we experimented with elementwise averaging, i.e., preserving only the DC in the third dimension (and discarding all other transform coefficients), hence its name "DC-only." For this case, all grouped blocks are estimated by elementwise averaging, exactly as in the illustrative example of Section II-C.

In Table II, we present results corresponding to various $\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$, $\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$, and $\mathcal{T}_{\mathrm{1D}}$. There, the Normal Profile parameters were used in all cases, where only the transform corresponding to a particular table column was changed. Boldface result correspond to the best performing transform. We observe that the choice of $\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$ and $\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$ does not have a significant impact on the denoising performance. Even the "DC+rand" transform, whose basis elements except for the DC are random, shows only a modest PSNR decrease in the range 0.1–0.4 dB. This can be explained by the fact that the collaborative filtering depends mainly on $\mathcal{T}_{\mathrm{1D}}$ for exploiting the interfragment correlation among grouped blocks. The estimation ability does not significantly depend on the energy compaction capabilities of $\mathcal{T}_{\mathrm{2D}}^{\mathrm{ht}}$ and $\mathcal{T}_{\mathrm{2D}}^{\mathrm{wie}}$. In this sense, the interfragment correlation appears as a much more important feature than the intrafragment correlation.

Let us now focus on the results corresponding to the various $\mathcal{T}_{\mathrm{1D}}$ transforms in Table II. One can distinguish the moderately worse performance of the DST as compared with not only the other standard transforms but also with the "DC+rand." We argue that the reason for this is the lack of DC basis element in the DST—in contrast with all other transforms, which have this element. Why is the DC of $\mathcal{T}_{\mathrm{1D}}$ important? Roughly speaking, this is so because the DC basis element captures the similarity between elements along the 3rd dimension of a group. Since

TABLE III
GRAYSCALE-IMAGE DENOISING: OUTPUT PSNR (dB) OF THE PROPOSED BM3D ALGORITHM

σ / PSNR	*C.man* 256^2	*House* 256^2	*Peppers* 256^2	*Montage* 256^2	*Lena* 512^2	*Barbara* 512^2	*Boats* 512^2	*F.print* 512^2	*Man* 512^2	*Couple* 512^2	*Hill* 512^2	*Lake* 512^2
2 / 42.11	43.96	44.63	43.48	46.47	43.59	43.66	43.18	42.90	43.61	43.17	43.04	43.02
5 / 34.16	38.29	39.83	38.12	41.14	38.72	38.31	37.28	36.51	37.82	37.52	37.14	36.58
10 / 28.14	34.18	36.71	34.68	37.35	35.93	34.98	33.92	32.46	33.98	34.04	33.62	32.85
15 / 24.61	31.91	34.94	32.70	35.15	34.27	33.11	32.14	30.28	31.93	32.11	31.86	31.08
20 / 22.11	30.48	33.77	31.29	33.61	33.05	31.78	30.88	28.81	30.59	30.76	30.72	29.87
25 / 20.18	29.45	32.86	30.16	32.37	32.08	30.72	29.91	27.70	29.62	29.72	29.85	28.94
30 / 18.59	28.64	32.09	29.28	31.37	31.26	29.81	29.12	26.83	28.86	28.87	29.16	28.18
35 / 17.25	27.93	31.38	28.52	30.46	30.56	28.98	28.43	26.09	28.22	28.15	28.56	27.50
50 / 14.16	25.84	29.37	26.41	27.35	28.86	27.17	26.64	24.36	26.59	26.38	27.08	25.78
75 / 10.63	24.05	27.20	24.48	25.04	27.02	25.10	24.96	22.68	25.10	24.63	25.58	24.11
100 / 8.14	22.81	25.50	22.91	23.38	25.57	23.49	23.74	21.33	23.97	23.37	24.45	22.91

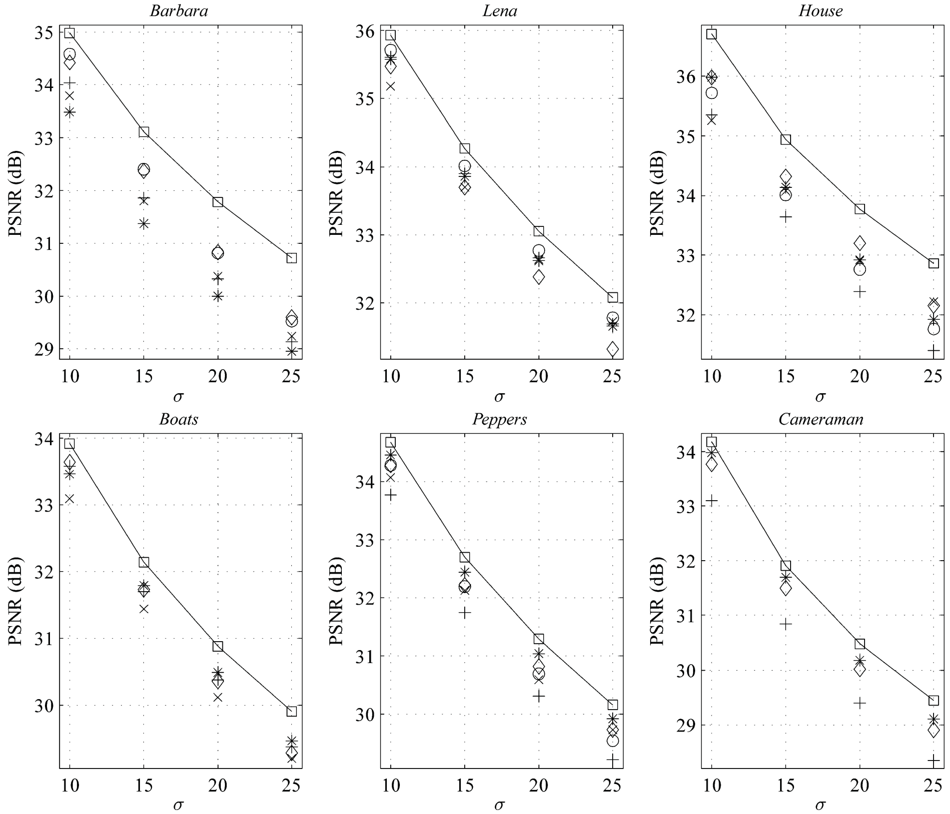


Fig. 4. Grayscale-image denoising: output PSNR as a function of σ for the following methods. "□": proposed BM3D; "○": FSP+TUP BLS-GSM [4]; "+": BLS-GSM [3]; "×": exemplar-based [11]; "◇": K-SVD [8]; "∗": pointwise SA-DCT [9]. (Note that the result of [4] for *Boats* and the results of [4] and [11] for *Cameraman* are missing since they were neither reported in the corresponding articles, nor were implementations of these methods publicly available.)

the grouped blocks are similar, so are their corresponding 2-D spectra and the DC terms reflect this similarity.

However, as it has been discussed in Section II, the existence of perfectly matching blocks is unlikely. In order to avoid trivial groups containing only the reference block, a strictly positive threshold is used in (5) and (7). Additionally, as follows from (3), the accuracy of the block-distance is affected by the noise. In practice this means that within a group there can be blocks for which the underlying true signal Y_x is much farther from Y_{x_R} than $\tau_{\mathrm{match}}^{\mathrm{ht}}$. Therefore, the sole DC element is not able to capture the potential differences between grouped blocks. This is confirmed by the poor results of the "DC-only" for $\mathcal{T}_{1\mathrm{D}}$. The availability of additional basis elements in any of the other transforms, even the random ones in "DC+rand," results in big performance improvement over the "DC-only."

We experimentally found that the ordering of blocks in the group does not have a significant effect on the estimation ability of the algorithm. This is confirmed by the results of the

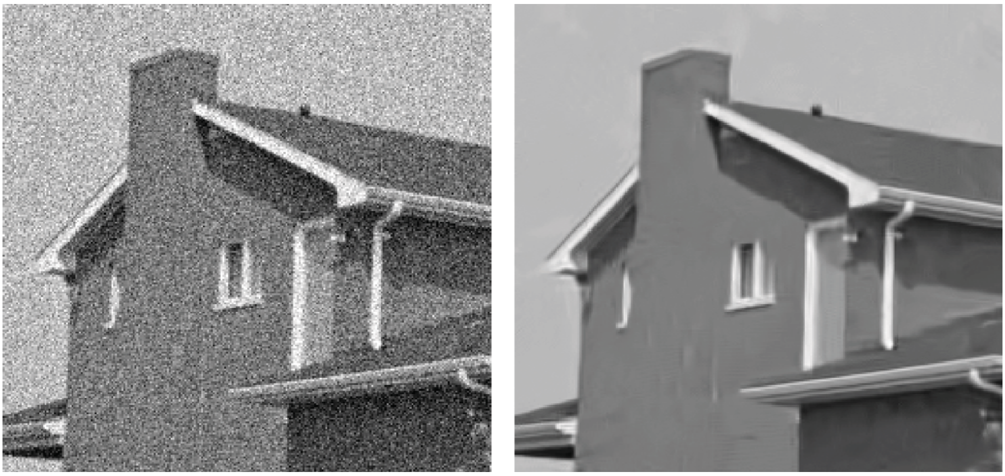
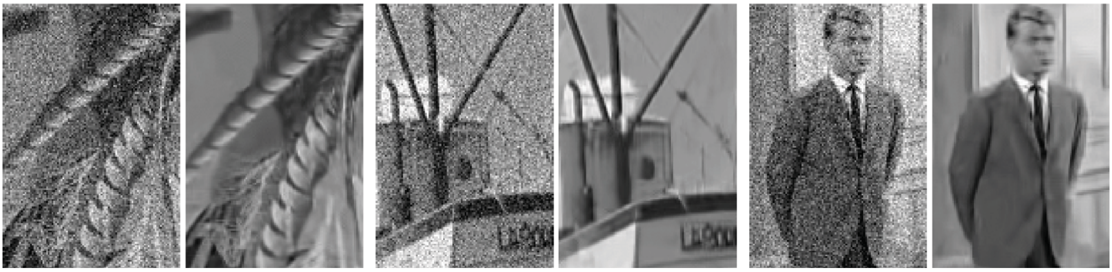


Fig. 5. Noisy ($\sigma = 25$) grayscale *House* image and the BM3D estimate (PSNR 32.86 dB).

(a) *Lena* (PSNR 32.08 dB) (b) *Barbara* (PSNR 30.73 dB) (c) *Cameraman* (PSNR 29.45 dB)



(d) *Man* (PSNR 29.62 dB) (e) *Boats* (PSNR 29.91 dB) (f) *Couple* (PSNR 29.72 dB)

Fig. 6. Fragments of noisy ($\sigma = 25$, PSNR 20.18 dB) grayscale images and the corresponding BM3D estimates. (a) *Lena* (PSNR 32.08 dB); (b) *Barbara* (PSNR 30.73 dB); (c) *Cameraman* (PSNR 29.45 dB); (d) *Man* (PSNR 29.62 dB); (e) *Boats* (PSNR 29.91 dB); (f) *Couple* (PSNR 29.72 dB).

"DC+rand" for $\mathcal{T}_{1\mathrm{D}}$ which achieves the same results as any of the other (structured, nonrandom) orthogonal transforms. For this transform, the ordering is *irrelevant* for the DC and is relevant only for the other basis elements which, however, are generated randomly. Hence, we may conclude that the ordering of the blocks in the groups does not influence the final results. Given this and because in our implementation the BM already produces a collection of blocks ordered by their block-distance, we resort to using exactly this ordering. Naturally, first in a group is always the reference block as the distance to itself is trivially equal to zero.

Note that, even though a group is constructed based on the similarity with respect to a given reference block, this does not imply that this block is better represented by the group than any of the others. For example, it can happen that all the matched blocks (except the reference block) are quite dissimilar from the reference one but tightly similar to each other. Such a group could be termed as "unbalanced."

We choose the Haar full dyadic decomposition for $\mathcal{T}_{1\mathrm{D}}$ because it can be efficiently implemented with iterated filter-banks using 2-tap analysis/synthesis filters. To apply such an orthonormal full dyadic decomposition, the transform size must

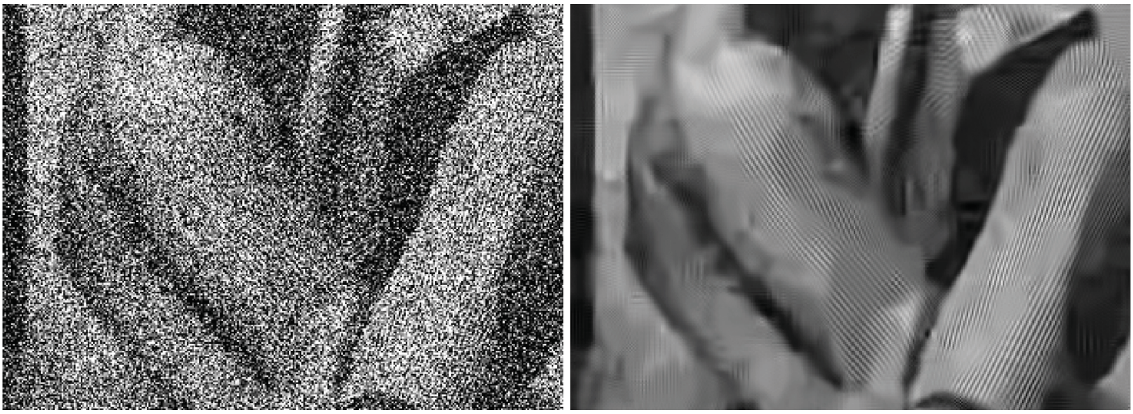


Fig. 7. On the left: Fragment of a noisy ($\sigma = 100$, PSNR 8.14 dB) grayscale *Barbara*; on the right: the corresponding fragment of the BM3D estimate (PSNR 23.49 dB).

be a power of 2. We enforced this requirement by restricting the number of elements of both S_x^{ht} (5) and S_x^{wie} (7) to be the largest power of 2 smaller than or equal to the original number of elements in S_x^{ht} and S_x^{wie}, respectively.

V. EXTENSION TO COLOR-IMAGE DENOISING

We consider a natural *RGB* image with additive i.i.d. zero-mean Gaussian noise in each of its channels. Let a luminance-chrominance transformation be applied on such a noisy image, where the luminance channel is denoted with Y and the chrominance channels are denoted with U and V. Prominent examples of such transformations are the $YCbCr$ and the opponent color transformations, whose transform matrices are, respectively

$$A_{YCbCr} = \begin{bmatrix} 0.30 & 0.59 & 0.11 \\ -0.17 & -0.33 & 0.50 \\ 0.50 & -0.42 & -0.08 \end{bmatrix}$$
$$A_{\mathrm{opp}} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{\sqrt{6}} & 0 & \frac{-1}{\sqrt{6}} \\ \frac{1}{3\sqrt{2}} & \frac{-\sqrt{2}}{3} & \frac{1}{3\sqrt{2}} \end{bmatrix}. \tag{13}$$

Due to properties of the underlying natural color image, such as high correlation between its R, G, and B channels, the following observations can be made.

- Y has higher signal-to-noise ratio (SNR) than U and V (decorrelation of the R, G, and B channels).
- Y contains most of the valuable information (edges, shades, objects, texture patterns, etc.).
- U and V contain mostly low-frequency information (very often these channels come from undersampled data).
- Iso-luminant regions with variation only in U and V are unlikely.

A straightforward extension of the developed grayscale denoising method for color-image denoising would be to apply it separately on each of the Y, U, and V channels. This naive approach, however, would suffer from the lower SNR in the chrominances since the grouping is sensitive to the level of noise. Because a proper grouping is essential for the effectiveness of our method, we propose to perform the grouping only once for the luminance Y and reuse exactly the same grouping when applying collaborative filtering on the chrominances U and V. That is, the sets of grouped blocks' coordinates from (5) and (7) are found for Y, respectively in Steps 1ai and 2ai, and reused for both U and V; using these sets, the collaborative filtering (Steps 1aii and 2aii) and the aggregation (Steps 1b and 2b) are performed separately on each of the three channels. The *grouping constraint* on the chrominances is based on the assumption that if the luminances of two blocks are mutually similar, then their chrominances are also mutually similar. Furthermore, given that grouping by block-matching takes approximately half of the execution time of the BM3D, the grouping constraint enables a computational reduction of approximately one third as compared to applying the grayscale BM3D separately on the three channels.

VI. RESULTS

In this section, we present and discuss the experimental results obtained by the developed algorithms; the grayscale version is denominated *block-matching and 3-D filtering* (BM3D) and the color version is accordingly abbreviated C-BM3D. For all experiments, we used the Matlab codes available at http://www.cs.tut.fi/~foi/GCF-BM3D. At this website, we also provide further results and the original and denoised test images used in our experiments. Unless specified otherwise, we use the parameters of the "Normal Profile" from Table I for both the BM3D and the C-BM3D.

A. Grayscale-Image Denoising

The output PSNR results of the BM3D algorithm for a standard set of grayscale images are given in Table III. The PSNR of an estimate $\widehat{y}$ of a true image y, is computed according to the standard formula

$$\mathrm{PSNR}(\widehat{y}) = 10\log_{10}\left(\frac{255^2}{|X|^{-1}\sum_{x\in X}\left(y(x)-\widehat{y}(x)\right)^2}\right).$$

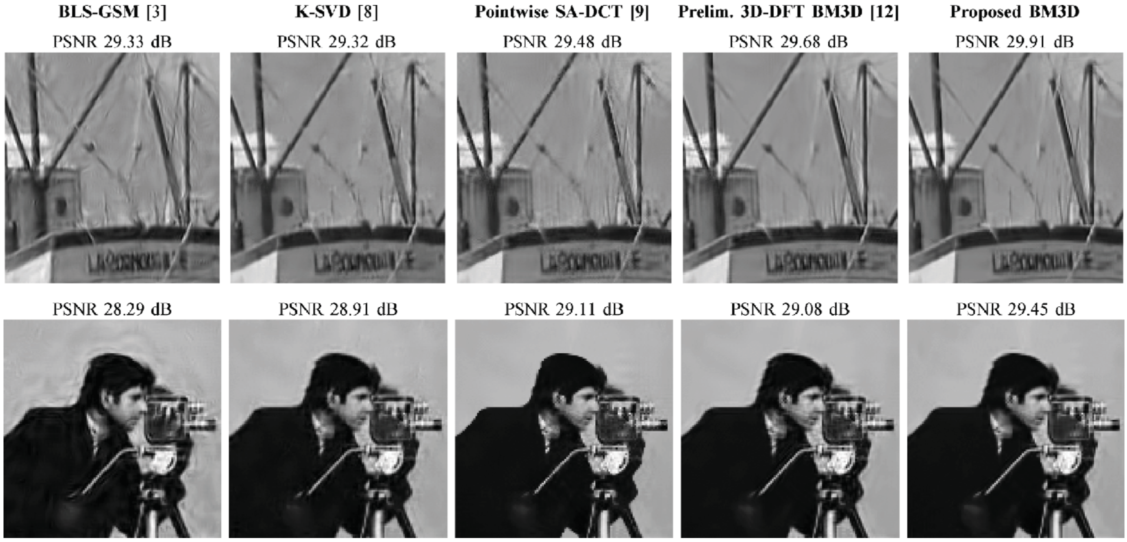


Fig. 8. Fragments of the grayscale (top row) *Boats* and (bottom row) *Cameraman* denoised by (from left to right): [3], [8], [9], [12], and the proposed BM3D for noise with $\sigma = 25$ (fragments of the noisy images can be seen in Fig. 6).

In Fig. 4, we compare the output PSNR results of the proposed BM3D with those of the state-of-the-art techniques BLS-GSM [3], FSP+TUP BLS-GSM [4], exemplar-based [11], K-SVD [8], Pointwise SA-DCT [9]; for the K-SVD method [8], we report its best results, which are those obtained with an adaptive dictionary trained on the noisy image. It can be seen from the figure that the proposed BM3D demonstrates the best performance and uniformly outperforms all of the other techniques. In particular, a significant improvement is observed for *House* and *Barbara* since these images contain structured objects (edges in *House* and textures in *Barbara*) which enable a very effective grouping and collaborative filtering.

In Fig. 5, we show a noisy $(\sigma = 25)$ *House* image and the corresponding BM3D estimate. In this test image, similarity among neighboring blocks is easy to perceive in the uniform regions and along the regular-shaped structures, some of which are illustrated in Fig. 1. Hence, such details are well-preserved in the estimate.

The denoising performance of the BM3D algorithm is further illustrated in Fig. 6, where we show fragments of a few noisy $(\sigma = 25)$ test images and fragments of the corresponding denoised ones. The denoised images show good preservation of:

- uniform areas and smooth intensity transitions (cheeks of *Lena*, and the backgrounds of the other images);
- textures and repeating patterns (the scarf in *Barbara*);
- sharp edges and singularities (borders of objects in *Cameraman* and *Boats*).

A denoising example for an extreme level of noise such as $\sigma = 100$ is shown in Fig. 7. Given that the original image is almost completely buried into noise, the produced estimate shows reasonable detail preservation. In particular, repeated patterns, such as the stripes on the clothes, are faithfully reconstructed.

Regarding the subjective visual quality, we find that various image details are well preserved and at the same time very few artifacts are introduced; one can observe this in Figs. 6–8. The state-of-the-art subjective visual quality of our algorithm is confirmed by the result of the psycho-visual experiment carried out by Vansteenkiste *et al.* [19]. There, 35 evaluators classified the preliminary version [12] of the BM3D algorithm as the best among 8 evaluated state-of-the-art techniques. The criteria in this evaluation were *perceived noisiness*, *perceived blurriness*, and *overall visual quality*. Furthermore, we consider the subjective visual quality of the current BM3D algorithm to be significantly better (in terms of detail preservation) than that of its preliminary version evaluated in [19]. In Fig. 8, we show images denoised by the current and by the preliminary versions of the BM3D algorithm. A close inspection reveals that the images denoised by the current BM3D have both fewer ringing artifacts and better preservation of details.

We show the PSNR performance of the Fast and Normal BM3D Profiles in Fig. 9. The two cases of the Normal Profile from Table I are considered separately for $\sigma \in [10, 75]$ in order to show the sharp PSNR drop of the "$\sigma \leq 40$" graph at about $\sigma = 40$ due to erroneous grouping. On the other hand, for the "$\sigma > 40$" graph, where the thresholding-based d-distance (4) is used with a relatively large block-size N_1, one can observe that there is no sharp PSNR drop. It is noteworthy that, for up to moderate levels of noise such as $\sigma < 35$, the PSNR difference between the Fast and the Normal Profiles is in the range 0.05–0.2 dB. This can be an acceptable price for the 6-fold reduction of the execution time shown in Table I; more precisely, the approximate execution time (for denoising a 256 × 256 image calculated on a 1.5-GHz Celeron M) decreases from 4.1 s for the Normal Profile to 0.7 s for the Fast Profile. The BM3D algorithm allows for further complexity/performance tradeoff by varying N_{step}. As a rough comparison, the execution times (for denoising a 256 × 256 image on a 1.5-GHz Celeron M) of the other methods considered in Fig. 4 were: 22.1 s for the

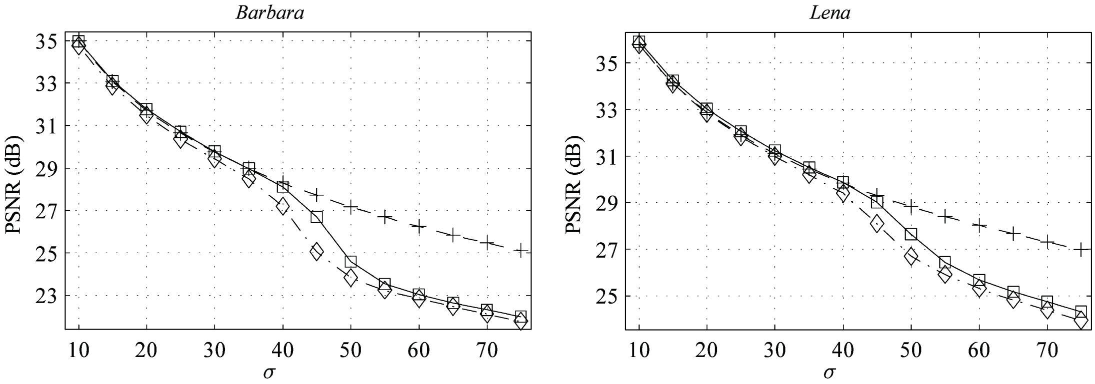


Fig. 9. Comparison between the output PSNR corresponding to the profiles in Table I. Notation is: "◇" for fast profile, "□" for the normal profile in the case "$\sigma \leq 40$" and "+" in the case "$\sigma > 40$"; both instances of the normal profile are shown for all considered values of σ in the range [10, 75].

TABLE IV
COLOR-IMAGE DENOISING: OUTPUT PSNR OF THE PROPOSED C-BM3D ALGORITHM

σ / PSNR	*Lena*	*Peppers*	*Baboon*	*F16*	*House*
5 / 34.15	37.82	36.82	35.25	39.68	38.97
10 / 28.13	35.22	33.78	30.64	36.69	36.23
15 / 24.61	33.94	32.60	28.39	35.00	34.85
20 / 22.11	33.02	31.83	26.97	33.77	33.84
25 / 20.17	32.27	31.20	25.95	32.78	33.03
30 / 18.59	31.59	30.62	25.14	31.94	32.33
35 / 17.25	30.91	30.00	24.46	31.13	31.58
50 / 14.15	29.72	28.68	23.14	29.41	30.22
75 / 10.63	28.19	27.12	21.71	27.60	28.33

BLS-GSM, 6.2 s for the SA-DCT filter, 9–30 min (depending on σ) for training the adaptive K-SVD on an input noisy image, and 25–120 s to perform the filtering using the found dictionary. The execution time of the exemplar-based method was reported in [11] to be about 1 min when measured on a 2-GHz Pentium IV. The execution time of the FSP+TUP BLS-GSM was not reported; however, it is a two-step BLS-GSM extension that should not be faster than the BLS-GSM.

B. *Color-Image Denoising*

We performed experiments with the C-BM3D using the opponent color space transformation (13) and the Normal Profile algorithm parameters. In all experiments, we considered noisy images with i.i.d. zero-mean Gaussian noise of variance σ^2 in each of their R, G, and B channels. The PSNR for *RGB* images is computed using the standard formula

$$10\log_{10}\left(\frac{255^2}{(3|X|)^{-1}\sum_{c=R,G,B}\sum_{x\in X}\left(y_c(x)-\widehat{y}_c(x)\right)^2}\right)$$

where the subscript $c \in \{R, G, B\}$ denotes the color channel. Table IV presents the output-PSNR results of the proposed C-BM3D algorithm for a few standard test images. A comparison with the two recent state-of-the-art methods [9], [20] is given in Table V. One can see that the proposed algorithm outperforms them for the three test images considered there.

TABLE V
COLOR-IMAGE DENOISING: OUTPUT-PSNR COMPARISON WITH THE TWO STATE-OF-THE-ART RECENT METHODS [20] AND [9]

Image	Method	Standard deviation σ			
		10	*15*	*20*	*25*
Lena 512^2 RGB	Proposed C-BM3D	**35.22**	**33.94**	**33.02**	**32.27**
	P.wise SA-DCT [9]	34.95	33.58	32.61	31.85
	ProbShrink-MB [20]	34.60	33.03	31.92	31.04
Peppers 512^2 RGB	Proposed C-BM3D	**33.78**	**32.60**	**31.83**	**31.20**
	P.wise SA-DCT [9]	33.70	32.42	31.57	30.90
	ProbShrink-MB [20]	33.44	32.05	31.12	30.35
Baboon 512^2 RGB	Proposed C-BM3D	**30.64**	**28.39**	**26.97**	**25.95**
	P.wise SA-DCT [9]	30.62	28.33	26.89	25.86
	ProbShrink-MB [20]	30.17	27.83	26.38	25.27

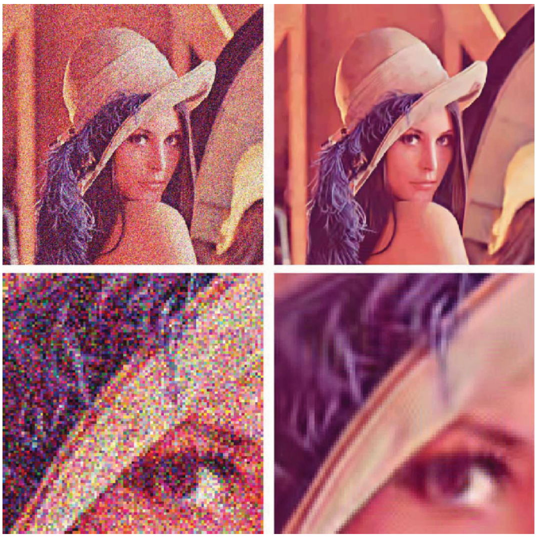


Fig. 10. Color-image denoising: On the left are a noisy *Lena* image ($\sigma = 50$, PSNR 14.15 dB) and a fragment of it; on the right are the C-BM3D estimate (PSNR 29.72 dB) and the corresponding fragment.

The visual quality can be inspected from Fig. 10 where a noisy (with $\sigma = 50$) color *Lena* and the C-BM3D estimate are shown. One can observe the faithfully preserved details on the hat, the sharp edges, and the smooth regions. The approximate execution time of the C-BM3D for a 256 × 256 RGB image was 7.6 s on a 1.5-GHz Celeron M.

VII. DISCUSSION

The approach presented in this paper is an evolution of our work on local approximation techniques. It started from the classical local polynomial approximation with a simple symmetric neighborhood. The adaptive pointwise varying size of this neighborhood was a first step to practically efficient algorithms. A next step was devoted to anisotropic estimation based on adaptive starshaped neighborhoods allowing nonsymmetric estimation areas. The nonsymmetry of these estimates is a key-point in designing estimators relevant to natural images. This development has been summarized in the recent book [21].

These techniques are based on fixed-order approximations. For image processing, these approximations are in practice reduced to zero and first order polynomials. It became clear that the developed neighborhood adaptivity had practically exhausted its estimation potential.

The breakthrough appears when the adaptive order local approximations are introduced. First, it was done in terms of the orthonormal transform with varying window size [22]. The hard-thresholding of the spectrum of these transforms means that some terms in the approximating series are adaptively dropped, and, thus, the order of the model becomes data dependent [23]. The most efficient development of the idea of the adaptive order estimation in local neighborhoods was the pointwise shape-adaptive DCT filter [9], where the orthonormal transform is calculated in adaptive shape neighborhoods defined by special statistical rules.

The next essential step in the development of the local approximations is presented in this paper. The spatial adaptivity is realized by selection of sets of blocks similar to a given reference one. Thus, local estimates become nonlocal. The selected blocks are grouped in 3-D arrays, jointly filtered, and aggregated at the places where they were taken from. The joint filtering of the blocks in the 3-D arrays is realized by shrinkage of the spectrum items; thus, the idea of the order adaptive estimation is exploited again but in quite a specific way. The main advantages of this approach are the nonlocality and the collaborative filtering. The latter results in effective preservation of local features in image blocks and very efficient denoising.

We wish to mention the work of a few other authors in order to clarify the context of our contribution and to state what makes it different from other similar approaches.

Since our method and the nonlocal estimators [10] and [11] are based on the same assumptions about the signal, it is worth comparing this class of techniques with our method. The weighted mean used in the nonlocal estimation corresponds to a zero-order polynomial approximation. Its effectiveness depends on an elaborate computation of adaptive weights, depending on the similarity between image patches centered at the estimated pixel and the ones used in the averaging. Our approach is different; by using a more flexible set of the basis functions (embedded in the transform), we enable order-adaptivity of the model and a more efficient exploitation of the similarity between grouped blocks. This is realized by collaborative filtering that allows for high-order estimates (not only weighted means) to be calculated for all grouped blocks.

The algorithm proposed in [8] is derived from a global optimization formulation. The image is segmented in a set of overlapping blocks and the filtering is enabled by fitting a minimum complexity model to each of these blocks. The final image estimate is obtained by fusing these models. A very good performance of the algorithm mainly follows from using a set of basis functions (dictionaries) obtained by training. In contrast, our collaborative filtering is essentially different because the model induced by hard-thresholding has low-complexity *only in relation to the group as a whole*. For the block-wise estimates and for the image overall, the model can instead be highly complex and redundant as each block can enter in many groups and, thus, can participate in many collaborative estimates. This redundancy gives a very good noise attenuation and allows to avoid artifacts typical for the standard thresholding schemes. Thus, we may say that instead of some low-complexity modeling as in [8], we exploit specific overcomplete representations.

The collaborative Wiener filtering used in the second step and the aggregation of block-wise estimates using adaptive weights are major features of our approach. The Wiener filtering uses the power spectrum of the basic estimate to filter the formed groups. As a result, the estimation improves significantly over the hard-thresholding used in the first step. The improvement in PSNR can be seen from Table II (by comparing the numbers in the column of "$\mathcal{T}_{2\mathrm{D}}^{\mathrm{ht}}$" with the numbers in any of the other two columns "$\mathcal{T}_{2\mathrm{D}}^{\mathrm{wie}}$" or "$\mathcal{T}_{1\mathrm{D}}$"); one can observe that the improvement is substantial, typically greater than 0.5 dB.

The basis functions used in our algorithm are standard ones, computationally efficient, and image independent. We believe that the proposed denoising method could be improved by using more sophisticated bases such as adaptive PCA [7], or overcomplete learned dictionaries [8]. However, the computational complexity would significantly increase because these transforms are typically nonseparable and do not have fast algorithms. As it is shown in the previous section, even with the currently used standard transforms, our algorithm already demonstrates better performance than both [8] and [11].

The proposed extension to color images is nontrivial because we do not apply the grayscale BM3D independently on the three luminance-chrominance channels, but we impose a *grouping constraint* on both chrominances. The grouping constraint means that the grouping is done only once, in the luminance (which typically has a higher SNR than the chrominances), and exactly the same grouping is reused for collaborative filtering in both chrominances. It is worth comparing the performance of the proposed C-BM3D versus the independent application of the grayscale BM3D on the individual color channels. This is done in Table VI which shows that the C-BM3D achieves 0.2–0.4 dB better PSNR than the independent application of the BM3D on the opponent color channels and 0.3–0.8 dB better PSNR than the independent application of the BM3D on the *RGB* channels. This improvement shows the significant benefit

TABLE VI
PSNR RESULTS OF THREE DIFFERENT APPROACHES TO COLOR-IMAGE DENOISING. THE NOISE WAS ADDED IN *RGB* WITH $\sigma = 25$ AND ALL PSNR (dB) VALUES WERE ALSO COMPUTED IN *RGB* SPACE

Approach to color-image denoising	*Lena*	*House*	*Peppers*
BM3D independently on the R, G, and B color channels	31.44	32.18	30.93
BM3D independently on each opponent color channel	32.01	32.64	31.01
C-BM3D, with grouping constraint in opponent color space	**32.27**	**33.03**	**31.20**

of using the grouping constraint on the chrominances in the C-BM3D.

We note that a similar idea of filtering the chrominances using information from the luminance was exploited already in the Pointwise SA-DCT denoising method [9]. There, adaptive-shape estimation neighborhoods are determined only for Y and then reused for both U and V. The PSNR improvement (0.1–0.4 dB) of the proposed approach compared with [9] is consistent with the improvement between the grayscale versions of these two methods.

VIII. CONCLUSION

The image modeling and estimation algorithm developed in this paper can be interpreted as a novel approach to nonlocal adaptive nonparametric filtering. The algorithm demonstrates state-of-the-art performance. To the best of our knowledge, the PSNR results shown in Tables III and IV are the highest for denoising additive white Gaussian noise from grayscale and color images, respectively. Furthermore, the algorithm achieves these results at reasonable computational cost and allows for effective complexity/performance tradeoff, as shown in Table I.

The proposed approach can be adapted to various noise models such as additive colored noise, non-Gaussian noise, etc., by modifying the calculation of coefficients' variances in the basic and Wiener parts of the algorithm. In addition, the developed method can be modified for denoising 1-D-signals and video, for image restoration, as well as for other problems that can benefit from highly sparse signal representations.

REFERENCES

[1] L. Sendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans. Signal Process.*, vol. 50, no. 11, pp. 2744–2756, Nov. 2002.

[2] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Process.*, vol. 11, no. 5, pp. 545–557, May 2002.

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

[4] J. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," presented at the IEEE Int. Conf. Image Process., Genova, Italy, Sep. 2005.

[5] L. Yaroslavsky, K. Egiazarian, and J. Astola, "Transform domain image restoration methods: Review, comparison and interpretation," in *Proc. Nonlinear Image Process. and Pattern Analysis XII*, 2001, vol. 4304, pp. 155–169.

[6] O. Guleryuz, "Weighted overcomplete denoising," in *Proc. Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 2003, vol. 2, pp. 1992–1996.

[7] D. Muresan and T. Parks, "Adaptive principal components and image denoising," presented at the IEEE Int. Conf. Image Processing, Sep. 2003.

[8] 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.

[9] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, May 2007.

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

[11] 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.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," presented at the SPIE Electronic Imaging: Algorithms and Systems V, San Jose, CA, Jan. 2006.

[13] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space," presented at the IEEE Int. Conf. Image Process., San Antonio, TX, Sep. 2007.

[14] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. Berkeley Symp. Math. Statist. Prob.*, Berkeley, CA, 1967, pp. 281–297.

[15] T. Kohonen, *Self-Organizing Maps*, ser. Information Sciences, 2nd ed. Heidelberg, Germany: Springer, 1997, vol. 30.

[16] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler, *Fuzzy Cluster Analysis*. Chichester, U.K.: Wiley, 1999.

[17] A. Gersho, "On the structure of vector quantizers," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 157–166, Feb. 1982.

[18] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.

[19] E. Vansteenkiste, D. Van der Weken, W. Philips, and E. E. Kerre, "Perceived image quality measurement of state-of-the-art noise reduction schemes," in *Proc. Lecture Notes in Computer Science ACIVS*, Antwerp, Belgium, Sept. 2006, vol. 4179, pp. 114–124.

[20] A. Pizurica and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 654–665, Mar. 2006.

[21] V. Katkovnik, K. Egiazarian, and J. Astola, *Local Approximation Techniques in Signal and Image Process*. Bellingham, WA: SPIE, 2006, vol. PM157.

[22] V. Katkovnik, K. Egiazarian, and J. Astola, "Adaptive window size image de-noising based on intersection of confidence intervals (ICI) rule," *Math. Imag. Vis.*, vol. 16, no. 3, pp. 223–235, May 2002.

[23] A. Foi and V. Katkovnik, "From local polynomial approximation to pointwise shape-adaptive transforms: An evolutionary nonparametric regression perspective," presented at the Int. TICSP Workshop Spectral Methods Multirate Signal Process., Florence, Italy, Sep. 2006.

Kostadin Dabov (S'07) received the M.Sc. degree in digital signal processing from the Institute of Signal Processing, Tampere University of Technology, Tampere, Finland, in April 2006, where he is currently pursuing the D.Tech. degree in signal restoration with locally adaptive transform-based techniques.

His interests include signal enhancement and restoration and efficient design and realization of signal processing algorithms.

Alessandro Foi received the M.Sc. degree in mathematics from the Università degli Studi di Milano, Milano, Italy, in 2001, and the Ph.D. degree in mathematics from the Politecnico di Milano in 2005.

His research interests include mathematical and statistical methods for signal processing, functional analysis, and harmonic analysis. Currently, he is a Researcher at the Institute of Signal Processing, Tampere University of Technology, Tampere, Finland. His work focuses on spatially adaptive algorithms for anisotropic denoising and deblurring of digital images and on noise modeling for digital imaging sensors.

Vladimir Katkovnik received the M.Sc., Ph.D., and D.Sc. degrees in technical cybernetics from the Leningrad Polytechnic Institute, Leningrad, Russia, in 1960, 1964, and 1974, respectively.

From 1964 to 1991, he held the positions of Associate Professor and Professor at the Department of Mechanics and Control Processes, Leningrad Polytechnic Institute. From 1991 to 1999, he was a Professor of statistics with the Department of the University of South Africa, Pretoria. From 2001 to 2003, he was a Professor of mechatronics with the Kwangju Institute of Science and Technology, Korea. From 2000 to 2001, and since 2003, he has been a Research Professor with the Institute of Signal Processing, Tampere University of Technology, Tampere, Finland. He has published seven books and more than 200 papers. His research interests include stochastic signal processing, linear and nonlinear filtering, nonparametric estimation, imaging, nonstationary systems, and time-frequency analysis.

Karen Egiazarian (SM'96) was born in Yerevan, Armenia, in 1959. He received the M.Sc. degree in mathematics from Yerevan State University in 1981, the Ph.D. degree in physics and mathematics from Moscow State University, Moscow, Russia, in 1986, and the D.Tech. degree from the Tampere University of Technology (TUT), Tampere, Finland, in 1994.

He was a Senior Researcher with the Department of Digital Signal Processing, Institute of Information Problems and Automation, National Academy of Sciences of Armenia. Since 1996, he has been an Assistant Professor with the Institute of Signal Processing, Tampere University of Technology, where he is currently a Professor, leading the Transforms and Spectral Methods Group. His research interests are in the areas of applied mathematics, signal processing, and digital logic.

Publication [P3]

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space," *Proc. IEEE Int. Conf. Image Process.*, San Antonio, TX, USA, September 2007.

COLOR IMAGE DENOISING VIA SPARSE 3D COLLABORATIVE FILTERING WITH GROUPING CONSTRAINT IN LUMINANCE-CHROMINANCE SPACE

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian

Institute of Signal Processing, Tampere University of Technology
P.O. Box 553, 33101 Tampere, Finland
firstname.lastname@tut.fi

ABSTRACT

We propose an effective color image denoising method that exploits filtering in highly sparse local 3D transform domain in each channel of a luminance-chrominance color space. For each image block in each channel, a 3D array is formed by stacking together blocks similar to it, a process that we call “grouping”. The high similarity between grouped blocks in each 3D array enables a highly sparse representation of the true signal in a 3D transform domain and thus a subsequent shrinkage of the transform spectra results in effective noise attenuation. The peculiarity of the proposed method is the application of a “grouping constraint” on the chrominances by reusing exactly the same grouping as for the luminance. The results demonstrate the effectiveness of the proposed grouping constraint and show that the developed denoising algorithm achieves state-of-the-art performance in terms of both peak signal-to-noise ratio and visual quality.

Index Terms— color image denoising, adaptive grouping, block-matching, shrinkage.

1. INTRODUCTION

Recently, we proposed the block-matching and 3D filtering (BM3D) grayscale-image denoising method [1, 2]. It achieves excellent results in terms of both mean-squared-error and subjective visual quality [3]. These results, to the best of our knowledge, not only overcome those by any other prior algorithm but are a significant step beyond the current state-of-the-art in image denoising (e.g., [4]).

The BM3D is based on *grouping*, a procedure that finds mutually similar 2D image blocks and stacks them together in 3D arrays (groups). Due to the similarity between such grouped blocks, a 3D transform produces very sparse representations of the groups. It is well known that sparsity is a fundamental property allowing for effective denoising. The sparsity depends on the similarity, and thus on the grouping. Therefore, a proper grouping is crucial for the BM3D. However, measuring mutual similarity of image blocks is difficult when only noisy data are available. In this sense, grouping should be applied in a wise manner in order to achieve the desired sparsity.

In this paper we extend the grayscale BM3D method to color-image denoising. Given an *RGB* image corrupted by additive white Gaussian noise, we first transform the image to a luminance-chrominance color space (such as *YCbCr*, opponent, *YIQ*, etc.). Our generalization of the BM3D is non-trivial because we do not apply the grayscale BM3D independently on the three luminance-chrominance channels but we impose a *grouping constraint* on both chrominances. The grouping constraint means that the grouping is done only once, in the luminance (which typically has a higher SNR than the chrominances), and exactly the same grouping is reused for filtering both chrominances.

This constraint is based on the assumption that the underlying image structures (objects, edges, patterns, etc.) which determine the existence of mutually similar blocks are the same across all color channels. This assumption is typically fulfilled for natural images. Loosely speaking, we treat grouping as a color-invariant process. The constraint on the chrominances increases the stability of the grouping with respect to noise. This results in more accurate and sparsely represented groups in the chrominances and thus in more effective denoising.

2. BLOCK-MATCHING AND 3D FILTERING (BM3D)

We briefly explain the grayscale BM3D method; for a comprehensive study one can refer to [2].

Images are processed in a sliding-window manner, where for each processed window (block), used as a reference one, a 3D array is formed by stacking together blocks similar to it. We call this process “grouping” and the obtained 3D arrays we denominate “groups”.

2.1. Grouping by block-matching

The grouping in BM3D is realized by block-matching. It searches in a local neighborhood for similar blocks by measuring the dissimilarity to the reference one. If the dissimilarity is smaller than a fixed threshold, the block is considered similar and is used for composing the group. The evaluation of the dissimilarity between noisy image blocks is sensitive to the noise. In particular, if the mean square difference is used as dissimilarity measure, then its variance grows quadratically with respect to the variance of the noise.

2.2. Collaborative filtering

The high similarity between grouped blocks in each 3D array enables a highly sparse representation of the true signal in a 3D transform domain. Therefore, a subsequent shrinkage of the transform spectra results in effective noise attenuation and the inverse of the 3D transform produces estimates of all blocks in the group. These estimates can be different for different blocks. Since each grouped block collaborates for the filtering of all other blocks in the group and vice versa, this filtering approach is called “collaborative filtering”.

This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program 2006 – 2011).

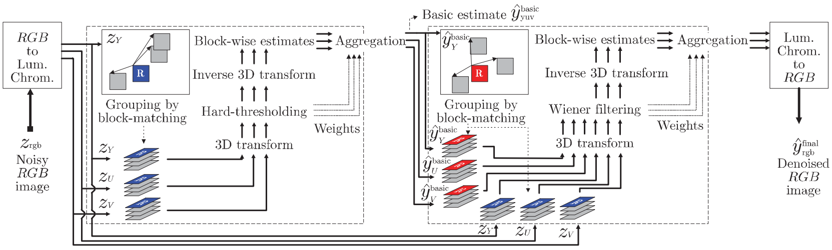


Fig. 1. Flowchart of the proposed color-image denoising method. Operations surrounded by dashed lines are repeated for each reference block.

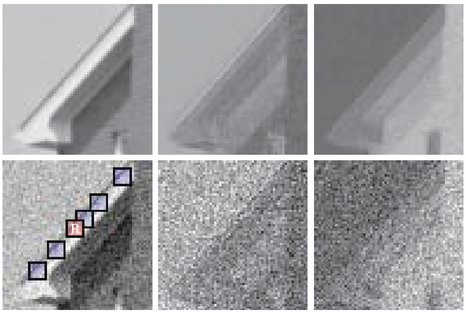


Fig. 2. The top row contains noise-free *Y*, *Cb*, and *Cr* channels of a fragment of *House* and the bottom row contains corresponding noisy ones (noise with standard deviation 22 is added in *RGB* space). Block-matching in the luminance *Y* is illustrated by showing a reference block (denoted by 'R') and the ones matched as similar to it; the same matched locations are used for both chrominances due to the grouping constraint.

3. GROUPING CONSTRAINT ON THE CHROMINANCES

Let us perform a luminance-chrominance transformation of an *RGB* natural image corrupted by independent noise, denoting the luminance with *Y* and the chrominances with *U* and *V*. We consider in particular the *YCbCr* and opponent color transformations, whose transform matrices are respectively

$$A_{YCbCr}=\begin{bmatrix}0.30 & 0.59 & 0.11\\ -0.17 & -0.33 & 0.50\\ 0.50 & -0.42 & -0.08\end{bmatrix},\quad A_{opp}=\begin{bmatrix}\frac{1}{3} & \frac{1}{3} & \frac{1}{3}\\ \frac{1}{\sqrt{6}} & 0 & \frac{-1}{\sqrt{6}}\\ \frac{1}{3\sqrt{2}} & \frac{-\sqrt{2}}{3} & \frac{1}{3\sqrt{2}}\end{bmatrix}. \qquad (1)$$

Due to properties of the underlying natural color image, such as high correlation between its *R*, *G*, and *B* channels, the following observations can be made:

- *Y* has higher SNR than *U* and *V* (decorrelation of the *R*, *G*, and *B* channels);
- *Y* contains most of the valuable information (edges, shades, objects, texture patterns, etc.);
- *U* and *V* contain mostly low-frequency information (very often these channels come from undersampled data);
- iso-luminant regions with variation only in *U* and *V* are unlikely.

A naive, straightforward way to use the BM3D for color-image denoising would be to apply it separately on each of the *Y*, *U* and *V* channels. This approach would suffer from the lower SNR in the chrominances since the grouping is quite sensitive to the level of noise. Because a proper grouping is essential for the BM3D, we propose to perform the grouping only once for *Y* and reuse exactly the same grouping when applying BM3D on *U* and *V*. That is, we impose a *grouping constraint* on the chrominances based on the assumption that if the luminances of two blocks are similar, then their chrominances are also similar. An illustration of block-matching in the luminance (in *YCbCr* color space) is given in Figure 2. The observations that we made earlier about the *Y*, *U*, and *V* channels can be confirmed from this figure (e.g. *Y* has a higher SNR than both *U* and *V*, and it also contains the most valuable information about the image structures).

Furthermore, given that grouping by block-matching takes approximately half of the execution time of the BM3D, the grouping constraint enables a computational reduction of approximately one third as compared to applying the grayscale BM3D separately on *Y*, *U* and *V*.

4. C-BM3D ALGORITHM FOR COLOR-IMAGE DENOISING

We consider a noisy *RGB* image z_{rgb} modeled as $z_{\text{rgb}} = y_{\text{rgb}} + \eta_{\text{rgb}}$, where $y_{\text{rgb}} = [y_R, y_G, y_B]$ is the true image and $\eta_{\text{rgb}} = [\eta_R, \eta_G, \eta_B]$ is independent Gaussian noise, where $\eta_C(\cdot) \sim \mathcal{N}\left(0, \sigma_C^2\right)$ for $C=$ *R*,*G*,*B*; the variances σ_R^2, σ_G^2, and σ_B^2 can be different. Following is the proposed algorithm, denominated Color-BM3D (C-BM3D), whose flowchart is shown in Figure 1.

1. Transform z_{rgb} to a luminance-chrominance space, the result is $z_{\text{yuv}} = [z_Y, z_U, z_V]$. The applied transform is not restricted; e.g., *YCbCr*, opponent, or others can be used.
2. *Obtain the basic estimate, denoted* $\widehat{y}^{\text{basic}}_{\text{yuv}} = \left[\widehat{y}^{\text{basic}}_Y, \widehat{y}^{\text{basic}}_U, \widehat{y}^{\text{basic}}_V\right]$.
   (a) *Block-wise estimates*. For each block in z_Y, do the following. Use block-matching to find the locations of the blocks in z_Y that are similar to the currently processed one. For each of the *Y*, *U*, and *V* channels, form a 3D array (group) by stacking blocks located at the obtained locations in z_Y, z_U, and z_V, respectively. Subsequently,

Method σ	*Lena* (512×512 RGB)				*Peppers* (512×512 RGB)				*Baboon* (512×512 RGB)			
	10	*15*	*20*	*25*	*10*	*15*	*20*	*25*	*10*	*15*	*20*	*25*
C-BM3D (proposed)	**35.22**	**33.94**	**33.02**	**32.27**	**33.78**	**32.60**	**31.83**	**31.20**	30.64	**28.39**	**26.97**	**25.95**
P.wise SA-DCT [5, 6]	34.95	33.58	32.61	31.85	33.70	32.42	31.57	30.90	30.62	28.33	26.89	25.86
ProbShrink-MB [7]	34.60	33.03	31.92	31.04	33.44	32.05	31.12	30.35	30.17	27.83	26.38	25.27
VMMSE [8]	34.02	31.89	30.24	28.88	33.12	31.13	29.67	28.45	**30.68**	28.24	26.63	25.36

Table 1. Comparison of the output PSNR of a few denoising methods.

for each of these three formed groups, apply a 3D transform, attenuate the noise by hard-thresholding its transform spectrum, and invert the 3D transform to produce estimates of all grouped blocks.

(b) *Aggregation*. Compute each of $\widehat{y}_Y^{\text{basic}}$, $\widehat{y}_U^{\text{basic}}$, and $\widehat{y}_V^{\text{basic}}$ by a weighted average of the obtained block-wise estimates corresponding to the *Y*, *U*, and *V* channels, respectively.

3. *Obtain the final estimate* $\widehat{y}_{\text{yuv}}^{\text{final}}$, by using $\widehat{y}_{\text{yuv}}^{\text{basic}}$ to further improve the grouping and to perform collaborative Wiener filtering.

    (a) *Block-wise estimates*. For each block in $\widehat{y}_Y^{\text{basic}}$, do the following. Use block-matching to find the locations of the blocks in $\widehat{y}_Y^{\text{basic}}$ that are similar to the currently processed one. For each of the *Y*, *U*, and *V* channels, form a pair of 3D arrays by stacking blocks located at the obtained locations in z_Y and $\widehat{y}_Y^{\text{basic}}$, z_U and $\widehat{y}_U^{\text{basic}}$, and z_V and $\widehat{y}_V^{\text{basic}}$, respectively. This results in the formation of six 3D arrays (groups). Subsequently, apply a 3D transform on each of these six 3D arrays and perform 3D Wiener filtering on the three noisy ones using the energy spectra of the ones from the basic estimate. Then, invert the 3D transform to produce estimates of all grouped blocks for each of the color channels.

    (b) *Aggregation*. Compute $\widehat{y}_Y^{\text{final}}$, $\widehat{y}_U^{\text{final}}$, and $\widehat{y}_V^{\text{final}}$ by a weighted average of the obtained block-wise estimates corresponding to the *Y*, *U*, and *V* channels, respectively.

4. Obtain the final denoised image $\widehat{y}_{\text{rgb}}^{\text{final}}$ by transforming $\widehat{y}_{\text{yuv}}^{\text{final}}$ to *RGB* space.

We note that the shrinkage (hard-thresholding and Wiener filtering) uses the variance of the noise in each of the *Y*, *U*, and *V* channels, which are computed by $\left[\sigma_Y^2, \sigma_U^2, \sigma_V^2\right]^T = A^2\left[\sigma_R^2, \sigma_G^2, \sigma_B^2\right]^T$, where A^2 is the used luminance-chrominance transform matrix, e.g. Eq. (1), whose elements are squared and where the superscript ^{T} denotes transposition.

5. EXPERIMENTAL RESULTS

The implementation of the C-BM3D algorithm used in the following experiments is publicly available[1] and it is based on the "Normal Profile" of the BM3D algorithm specified in [2] without any modification of its parameters. The computational complexity of the algorithm is quite reasonable; e.g., a 256×256 *RGB* image is denoised for about 5 seconds on a 3 GHz Pentium PC. The used test color images are *House*, *Lena*, *Peppers*, *F16*, *Baboon* and the 12th image from the Kodak PhotoCD dataset. In all experiments the noise variance is the same for each of the *R*, *G*, and *B* channels, $\sigma_R^2 = \sigma_G^2 = \sigma_B^2 = \sigma^2$. We always calculate the peak signal-to-noise ratio (PSNR) in the *RGB* space. Unless specified otherwise, we utilize luminance-chrominance transformation to the opponent color space.

σ	*Lena*	*Peppers*	*Baboon*	*F16*	*House*	*Kod. 12*
5	37.82	36.82	35.25	39.68	38.97	41.34
10	35.22	33.78	30.64	36.68	36.23	37.84
15	33.94	32.60	28.39	34.99	34.85	35.95
20	33.02	31.83	26.97	33.77	33.84	34.70
25	32.27	31.20	25.95	32.78	33.03	33.76
30	31.59	30.61	25.14	31.93	32.34	32.96
35	30.91	30.00	24.46	31.13	31.58	32.24
50	29.72	28.68	23.14	29.40	30.22	30.74
75	28.19	27.12	21.70	27.60	28.33	29.31

Table 2. Output PSNR of the proposed C-BM3D algorithm.

In Table 2, we present peak signal-to-noise ratio (PSNR) results of the proposed algorithm, where the PSNR is computed in the *RGB* space. A comparison with other methods is given in Table 1; to our knowledge, [7, 6] represent the current state-of-the-art in color-image denoising. One can see that the proposed algorithm outperforms all of the other methods for the three test images presented there, except for *Baboon* in the case of $\sigma = 10$.

Figure 3 shows a noisy ($\sigma = 35$) *House* image and the corresponding C-BM3D estimate. Most of the image details are well-preserved in the denoised image (e.g. edges, singularities, uniform areas) and in the same time there are almost no visible artifacts. The enlarged fragment in the same figure shows how effective is the proposed method for well structured details, such as the roof of the house.

6. DISCUSSION AND CONCLUSIONS

It is worth comparing the performance of the proposed C-BM3D versus the independent application of the grayscale BM3D on the individual color channels. Table 3 shows that the C-BM3D achieves 0.2 – 0.4 dB better PSNR than the independent application of BM3D on the opponent color channels and 0.3 – 0.8 dB better PSNR than the independent application of BM3D on the *RGB* channels. This improvement is not insignificant and shows the benefit of the proposed grouping constraint on the chrominances.

Although iso-luminant image regions with variation only in the chrominances are unlikely in natural images, we show such example in Figure 4 for *YCbCr* color space. The luminance of the swimsuit is nearly constant in contrast to the Cr chrominance channel that contains a sharp transition. Note also the significantly lower SNR of the chrominance channel as compared with the luminance. Because of the imposed grouping constraint, one can expect that the sharp transition in the chrominance is not well reconstructed. However,

---

[1] Matlab code available at `www.cs.tut.fi/~foi/GCF-BM3D`.

Fig. 3. On the left: noisy ($\sigma = 35$) *House* and a fragment of it; on the right: the corresponding denoised image (PSNR 31.58 dB) and fragment.

Approach	*Lena*	*House*	*Peppers*
C-BM3D, with grouping constraint in opponent color space	**32.27**	**33.03**	**31.20**
BM3D independently on the opponent color channels	32.01	32.64	31.01
BM3D independently on the *R*, *G*, and *B* color channels	31.44	32.18	30.93

Table 3. PSNR results of C-BM3D and of the independent application of the grayscale BM3D on the noisy color channels; the noise in *RGB* has $\sigma = 25$; all PSNR values were computed in *RGB* space.

even in such a case of grouping potentially dissimilar blocks, the proposed method can still achieve satisfactory reconstruction. The reason for this is that the BM3D uses a complete 3D transform to represent each group. The shrinkage will not be able to attenuate the noise as effectively but the chrominance details will be preserved, as shown in the denoised swimsuit in Figure 4.

We note that a similar idea of filtering the chrominances using information from the luminance was exploited already in the Pointwise SA-DCT denoising method [5, 6]. There, adaptive-shape estimation neighborhoods are determined only for *Y* and then reused for both *U* and *V*. The PSNR improvement (0.1 – 0.4 dB) of the proposed approach compared with [5, 6] is consistent with the improvement between the grayscale versions of these two methods.

The proposed C-BM3D achieves state-of-the-art performance in terms of both PSNR and subjective visual quality. This is achieved at a reasonable computational complexity. In addition, effective complexity scalability can be realized by exploiting the complexity/performance trade-off of the BM3D presented in [2].

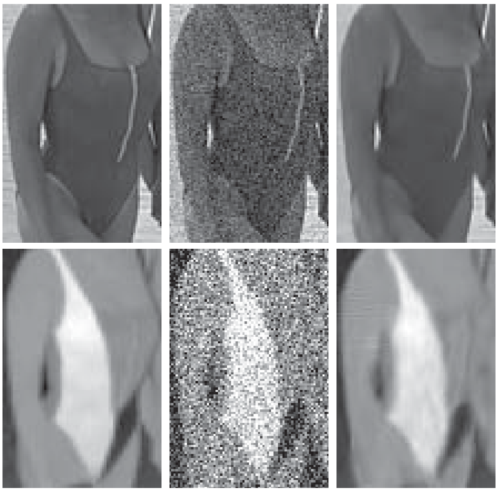


Fig. 4. Original (left), noisy (middle), and denoised (right) fragments of the luminance (top) and the chrominance *Cr* (bottom) of the 12th Kodak test image in *YCbCr* color space. The noise (added in *RGB* space) has $\sigma = 25$.

7. REFERENCES

[1] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering," in *Proc. SPIE Electronic Imaging: Algorithms and Systems V*, vol. 6064A-30, San Jose, CA, USA, January 2006.

[2] ——, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, December 2006, accepted, preprint at www.cs.tut.fi/˜foi/GCF-BM3D.

[3] E. Vansteenkiste, D. Van der Weken, W. Philips, and E. E. Kerre, "Perceived image quality measurement of state-of-the-art noise reduction schemes," in *Lecture Notes in Computer Science ACIVS*, vol. 4179, Antwerp, Belgium, September 2006, pp. 114–124.

[4] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli, "Image denoising using a scale mixture of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, November 2003.

[5] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT denoising with structure preservation in luminance-chrominance space," in *Proc. Video Process. Qual. Metrics Cons. Electr.*, Scottsdale, AZ, USA, January 2006.

[6] ——, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, May 2007.

[7] A. Pizurica and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 654–665, March 2006.

[8] P. Scheunders and J. Driesen, "Least squares inter-band denoising of color and multispectral images," in *Proc. IEEE Int. Conf. Image Process.*, Singapore, October 2004.

Publication [P4]

K. Dabov, A. Foi, and K. Egiazarian, “Video denoising by sparse 3D transform-domain collaborative filtering,” *Proc. European Signal Processing Conference*, Poznan, Poland, September 2007.

First published in the Proceedings of the 15th European Signal Processing Conference (EUSIPCO-2007) in 2007, published by EURASIP

VIDEO DENOISING BY SPARSE 3D TRANSFORM-DOMAIN COLLABORATIVE FILTERING

Kostadin Dabov, Alessandro Foi, and Karen Egiazarian

Institute of Signal Processing, Tampere University of Technology
P.O. Box 553, 33101 Tampere, Finland
firstname.lastname@tut.fi

ABSTRACT

We propose an effective video denoising method based on highly sparse signal representation in local 3D transform domain. A noisy video is processed in blockwise manner and for each processed block we form a 3D data array that we call "group" by stacking together blocks found similar to the currently processed one. This grouping is realized as a spatio-temporal predictive-search block-matching, similar to techniques used for motion estimation. Each formed 3D group is filtered by a 3D transform-domain shrinkage (hard-thresholding and Wiener filtering), the result of which are estimates of all grouped blocks. This filtering—that we term "collaborative filtering"—exploits the correlation between grouped blocks and the corresponding highly sparse representation of the true signal in the transform domain. Since, in general, the obtained block estimates are mutually overlapping, we aggregate them by a weighted average in order to form a non-redundant estimate of the video. Significant improvement of this approach is achieved by using a two-step algorithm where an intermediate estimate is produced by grouping and collaborative hard-thresholding and then used both for improving the grouping and for applying collaborative empirical Wiener filtering. We develop an efficient realization of this video denoising algorithm. The experimental results show that at reasonable computational cost it achieves state-of-the-art denoising performance in terms of both peak signal-to-noise ratio and subjective visual quality.

1. INTRODUCTION

Many video denoising methods have been proposed in the last few years. Prominent examples of the current developments in the field are the wavelet based techniques [1, 2, 3, 4, 5]. These methods typically utilize both the sparsity and the statistical properties of a multiresolution representation as well as the inherent correlations between frames in temporal dimension. A recent denoising strategy, the non-local spatial estimation [6], has also been adapted to video denoising [7]. In this approach, similarity between 2D patches is used to determine the weights in a weighted averaging between the central pixels of these patches. For image denoising, the similarity is measured for all patches in a 2D local neighborhood centered at the currently processed coordinate. For video denoising, a 3D such neighborhood is used. The effectiveness of this method depends on the presence of many similar true-signal blocks.

Based on the same assumption as the one used in the non-local estimation, i.e. that there exist mutually similar blocks in natural images, in [8] we proposed an image denoising method. There, for each processed block, we perform two special procedures — *grouping* and *collaborative filtering*. Grouping finds mutually similar 2D blocks and then stacks them together in a 3D array that we call *group*. The benefit of grouping highly similar signal fragments together is the increased correlation of the true signal in the formed 3D array. Collaborative filtering takes advantage of this increased correlation to effectively suppress the noise and produces estimates of each of the grouped blocks. We showed [8] that this approach is very effective for image denoising.

In this paper, we apply the concepts of grouping and collaborative filtering to video denoising. Grouping is performed by a specially developed predictive-search block-matching technique that significantly reduces the computational cost of the search for similar blocks. We develop a two-step video-denoising algorithm where the predictive-search block-matching is combined with collaborative hard-thresholding in the first step and with collaborative Wiener filtering in the second step. At a reasonable computational cost, this algorithm achieves state-of-the-art denoising results in terms of both PSNR and visual quality. This work generalizes the denoising approach from [8] and improves on the video denoising algorithm proposed in [9].

2. GROUPING AND COLLABORATIVE FILTERING FOR VIDEO DENOISING

The concepts of grouping and collaborative filtering are both extensively studied in [8]. Therefore, in this section we only give a general overview in the context of video denoising. A noisy video signal is processed in block-wise manner (processed blocks can overlap), where the currently processed block is denominated *reference block*. For each reference block, grouping is performed followed by collaborative filtering.

Analogously to [8], we realize grouping by block-matching, a procedure that tests the similarity between the reference block and ones that belong to a predefined search neighborhood. The similarity is typically computed as the inverse of some distance (dissimilarity) measure. The distance that we adopt in the sequel is the ℓ^2-norm of the difference between two blocks. Given the nature of video, the search neighborhood is a 3D domain that spans both the temporal and the two spatial dimensions. In this work, we propose to search for similar blocks by *predictive-search block-matching*. The peculiarity of this technique, fully explained in Section 4, is the adoption of data-adaptive spatio-temporal 3D search neighborhoods. They are adaptive to similarities between and within the frames, and thus to motion in the video. It allows for a significant complexity reduction as compared with full-search in non-adaptive neighborhoods.

In [8] we demonstrated that transform-domain shrinkage can be utilized as an effective realization of collaborative filtering. It comprises three steps; first, a 3D transform is applied on a group to produce a highly sparse representation of the true signal in it; second, shrinkage (e.g., hard thresholding or Wiener filtering) is performed on the transform coefficients; and third, an inverse 3D transform produces estimates of all grouped blocks. By exploiting the similarity

---

This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program [2006 - 2011]).

among grouped blocks, the transform produces a sparse representation of the true signal in the transform domain. This enables the subsequent shrinkage to efficiently attenuate the noise and at the same time to preserve the most significant portion of the true-signal spectrum.

After performing grouping and collaborative filtering for each reference block, a collection of overlapping blockwise estimates is obtained. This collection forms a redundant estimate of the true signal. In order to form a non-redundant estimate, the blockwise estimates need to be aggregated. As in [8], we propose aggregation by weighted averaging where the weights are inversely proportional to the squared ℓ^2-norm of the shrunk groups' spectra and thus loosely reciprocal to the total variance of each filtered group.

Using the above procedures, we develop a two-step video-denoising algorithm whose structure is analogous to that of the image-denoising algorithm [8]. The first step produces a basic (intermediate) estimate of the video signal by applying the proposed denoising scheme using grouping and collaborative hard-thresholding. The second step uses the basic estimate to improve the denoising in the following two aspects. First, grouping is performed within the basic estimate rather than within the noisy video, and second, the hard-thresholding is replaced by empirical Wiener filtering that uses the spectra of groups from the basic estimate.

3. ALGORITHM

We consider an observed noisy video $z(x) = y(x) + \eta(x)$, where y is the true video signal, $\eta(\cdot) \sim \mathcal{N}(0, \sigma^2)$ is an i.i.d. Gaussian noise sample and $x = (x_1, x_2, t) \in X$ are coordinates in the spatio-temporal 3D domain $X \subset \mathbb{Z}^3$. The first two components $(x_1, x_2) \in \mathbb{Z}^2$ are the spatial coordinates and the third one, $t \in \mathbb{Z}$, is the time (frame) index. The variance σ^2 is assumed a priori known. The proposed two-step denoising algorithm is presented in the right column of this page. It is also illustrated in Figure 1.

4. GROUPING BY PREDICTIVE-SEARCH BLOCK-MATCHING

A straightforward approach is to use a fixed-size 3D search neighborhood for the grouping by block-matching. However, capturing blocks of a moving object across many frames requires large spatial dimensions of such search neighborhood. On the one hand, using large sizes imposes a rather high complexity burden, and on the other hand, using small ones results in unsatisfactory grouping and poor denoising results.

In order to efficiently capture blocks that are part of objects which move across subsequent frames, we propose to use predictive-search block-matching, an inductive procedure that finds similar (matching) blocks by searching in a data-adaptive spatio-temporal subdomain of the video sequence. For a given reference block located at $x = (x_1, x_2, t_0)$, when using a temporal window of $2N_{FR}+1$ frames, the predictive-search block-matching comprises the following steps.

- Starting with frame t_0, an exhaustive-search block-matching is performed in a nonadaptive $N_S \times N_S$ neighborhood centered about (x_1, x_2). The result are the spatial locations of the N_B blocks (within this neighborhood) which exhibit highest similarity to the reference one. These locations are collected in the set $S^{t_0} \subset \mathbb{Z}^3$.
- The *predictive search* in frame $t_0 + k$, $0 < |k| \leq N_{FR}$, is defined inductively based on the matching results from the previously processed frame $t_0 + k - sign(k)$, i.e. from the preceding frame for $k > 0$ or from the subsequent frame for $k < 0$. This search for similar blocks takes place within the union of $N_{PR} \times N_{PR}$ neighborhoods centered at the spatial coordinates of the previously found locations $x \in S^{t_0+k-sign(k)}$. That is, these locations predict

---

V-BM3D video denoising algorithm

Step 1. Obtain a basic estimate using grouping and collaborative hard-thresholding.

1.1. For each coordinate $x \in X_R$ do:

(a) $S_x = PS\text{-}BM(Z_x)$,

(b) $\hat{\mathsf{Y}}_{S_x} = T_{3D}^{-1}\left(HARD\text{-}THR\left(T_{3D}\left(\mathsf{Z}_{S_x}\right), \lambda_{3D}\sigma\right)\right)$, where $\hat{\mathsf{Y}}_{S_x}$ is a group of blockwise estimates $\hat{Y}^x_{x'}$, $\forall x' \in S_x$.

1.2. Produce the basic estimate $\hat{y}^{basic}$ by aggregation of the blockwise estimates $\hat{Y}^x_{x'}$, $\forall x \in X_R$ and $\forall x' \in S_x$ using weighted averaging with $weight\left(\hat{Y}^x_{x'}\right) = \frac{1}{\sigma^2 N_{har}(x)} W_{2D}$.

Step 2. Obtain the final estimate by grouping within the basic estimate and collaborative Wiener filtering that uses the spectra of the corresponding groups from the basic estimate.

2.1. For each coordinate $x \in X_R$ do:

(a) $S_x = PS\text{-}BM\left(\hat{Y}^{basic}_x\right)$,

(b) $\hat{\mathsf{Y}}_{S_x} = T_{3D}^{-1}\left(T_{3D}\left(\mathsf{Z}_{S_x}\right) \frac{\left[T_{3D}\left(\hat{\mathsf{Y}}^{basic}_{S_x}\right)\right]^2}{\left[T_{3D}\left(\hat{\mathsf{Y}}^{basic}_{S_x}\right)\right]^2 + \sigma^2}\right)$.

2.2. Produce the final estimate $\hat{y}^{final}$ by aggregation of $\hat{Y}^x_{x'}$ $\forall x \in X_R$ and $\forall x' \in S_x$ using weighted averaging with $weight\left(\hat{Y}^x_{x'}\right) = \sigma^{-2} \left\| \frac{\left[T_{3D}\left(\hat{\mathsf{Y}}^{basic}_{S_x}\right)\right]^2}{\left[T_{3D}\left(\hat{\mathsf{Y}}^{basic}_{S_x}\right)\right]^2 + \sigma^2} \right\|_2^{-2} W_{2D}$, where $\|\cdot\|_2$ denotes ℓ^2-norm.

Notation:

- $X_R \subset X$ is a set that contains the coordinates of the processed reference blocks. We build it by taking each N_{step} element of X along both spatial dimensions, hence $|X_R| \approx \frac{|X|}{N^2_{step}}$.
- Z_x denotes a block of size $N_1 \times N_1$ in z, whose upper-left corner is at x. Similar notation is used for $\hat{Y}^x_{x'}$ and $\hat{Y}^{basic}_x$; the former is an estimate for the block located at x', obtained while processing reference block Z_x and the latter is a block located at x extracted from the basic estimate y^{basic}.
- $S_x = PS\text{-}BM(Z_x)$ performs predictive-search block-matching (Section 4) using Z_x as a reference block, the result of which is the set S_x containing the coordinates of the matched blocks. For Step 2, the search is performed in the basic estimate instead of in the noisy video.
- Z_{S_x} denotes a group (i.e. a 3D array) formed by stacking together the blocks $Z_{x \in S_x}$; the same notation is used for $\hat{\mathsf{Y}}^x_{S_x}$ and $\hat{\mathsf{Y}}^{basic}_{S_x}$. The size of these groups is $N_1 \times N_1 \times |S_x|$.
- $T_{3D}\left(\mathsf{Z}_{S_x}\right)$ is the spectrum of Z_{S_x} using a 3D linear transform T_{3D} which should have a DC basis element (e.g. the 3D-DCT, the 3D-DFT, etc.).
- $HARD\text{-}THR\left(T_{3D}\left(\mathsf{Z}_{S_x}\right), \lambda_{3D}\sigma\right)$ applies hard-thresholding on the transform coefficients (except for the DC) using threshold $\lambda_{3D}\sigma$ where λ_{3D} is a fixed threshold parameter.
- $N_{har}(x)$ is the number of nonzero coefficient retained after hard-thresholding $T_{3D}\left(\mathsf{Z}_{S_x}\right)$; since the DC is always preserved, $N_{har}(x) > 0$, ensuring division by zero never occurs in Step 1.1.a.
- W_{2D} is a 2D Kaiser window of size $N_1 \times N_1$.

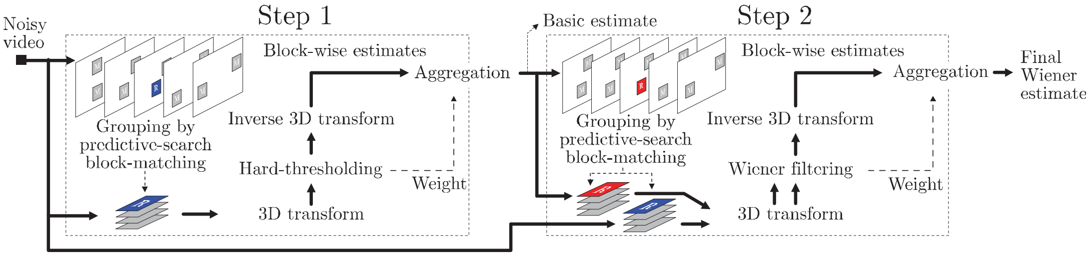


Figure 1: Flowchart of the proposed BM3D video denoising method. The operation enclosed by dashed lines are repeated for each reference block. Grouping is illustrated by showing a reference block marked with 'R' and the matched ones in a temporal window of 5 frames ($N_{FR} = 2$).

where similar blocks are likely to be present in the current frame (i.e. frame $t_0 + k$) and thus one can afford to have $N_{PR} < N_S$. The result for the current frame are the N_B locations of the blocks that exhibit highest similarity to the reference one; they are collected in the set S^{t_0+k}.

After performing the predictive-search block-matching for all of the frames $t_0 + k$ for $k = -N_{FR}, \ldots, N_{FR}$, we form a single set $S_x \subset \mathbb{Z}^3$ that contains at most N_2 of all $x' \in \bigcup_{k=-N_{FR}}^{N_{FR}} S^{t_0+k}$ that have the smallest corresponding block-distances to the reference block, which distances should also be smaller than a predefined threshold, τ_{match}. A group is later formed by stacking together blocks located at $x' \in S_x$. The exact ordering of the blocks within the 3D groups is not important, as shown in [8]. In the worst case, no matching blocks are found and then the group will contain only one block — the reference one — since its distance to itself is zero and therefore x will always be included in S_x.

Except for the frame t_0 in the procedure presented above, the spatial search neighborhoods are data-adaptive as they depend on previously matched locations. This adaptivity can be interpreted as following the motion of objects across frames. It is worth noting that similar approach has already been used for motion estimation [10] and also for fractal based image coding [11].

5. RESULTS

We present experimental results obtained with the proposed V-BM3D algorithm. A Matlab implementation of the V-BM3D that can reproduce these results is publicly available at `http://www.cs.tut.fi/~foi/GCF-BM3D`. There, one can find original and processed test sequences, details of which can be seen in Table 1.

The same algorithm parameters were used in all experiments. Here we give the most essential ones, as the rest can be seen in the provided Matlab script. The temporal window used 9 frames, i.e. $N_{FR} = 4$. The predictive-search block-matching used $N_S = 7$, $N_{PR} = 5$, and $N_B = 2$; the maximum number of matched blocks was $N_2 = 8$, and the threshold $\lambda_{3D} = 2.7$. Some of the parameters differed for the two steps; i.e., for Step 1, $N_1 = 8$, $N_{step} = 6$, and for Step 2, $N_1 = 7$, $N_{step} = 4$. The transforms were the same as in [8]: for Step 1, T_{3D} is a separable composition of a 1D biorthogonal wavelet full-dyadic decomposition in both spatial dimensions and a 1D Haar wavelet full-dyadic decomposition in the third (temporal) dimension; for Step 2, T_{3D} uses the 2D DCT in spatial domain and the same Haar decomposition in the temporal one. To increase the number of non-overlapping blocks in the groups and hence have more uncorrelated noise in them, we slightly modified the used distance measure. The modification was a subtraction of a small value d_s ($d_s = 3$ for Step 1 and $d_s = 7$ for Step 2) from the distance computed for blocks that are at the spatial coordinate of the reference one but in different frames.

In Table 1 we present the PSNR (dB) results of the proposed algorithm for a few sequences; there, the PSNR was measured globally on each whole sequence. In Figure 2, we compare our method with the 3DWTF [5] and the WRSTF [3], which are among the state-of-the-art in video denoising. For this comparison, we applied our method on noisy sequences and compared with the ones denoised by the other two methods. These sequences had been made publicly available by Dr. V. Zlokolica at `http://telin.ugent.be/~vzlokoli/PHD`, for which we are thankful. We note that the pixel intensities of the input noisy videos are quantized to integers in the range $[0, 255]$, unlike in the case of the results in Table 1. In Figure 2 one can observe that the proposed V-BM3D produces significantly higher PSNR than the other two methods for each frame of the three considered sequences, with a difference well higher than 1 dB for most of the frames. Moreover, this was achieved at similar execution times as compared with the WRSTF; i.e., the proposed V-BM3D (implemented as a Matlab MEX-function) filters a CIF (288×352) frame for 0.7 seconds on a 1.8 GHz Intel Core Solo machine and the WRSTF was reported [3] to do the same for 0.86 seconds on an Athlon64 (4000+) 2.4 GHz machine. Figure 3 gives a visual comparison for a fragment of the 77th frame of *Tennis* denoised with each of the considered techniques. The proposed method shows superior preservation of fine image details and at the same time it introduces significantly less artifacts.

6. DISCUSSION

Let us compare the proposed predictive-search block-matching with the motion estimation methods based on block-matching. Indeed, the predictive-search block-matching proposed in this work can be viewed as a sophisticated motion estimation which is not restricted to only one matched block per frame. That is, N_B blocks per frame can be used in the proposed grouping scheme. This can be beneficial in situations when there are only very few (or none) similar blocks along the temporal dimension, e.g. in the case of frame change. In that case, mutually similar blocks at different spatial locations within the same frame are exploited when forming groups and hence better sparsity is achieved by applying a 3D transform. This can be particularly effective when grouping blocks that are parts of, e.g., edges, textures, and uniform regions.

The second step of the proposed method is very impor-

σ/PSNR	Video name:	*Salesm.*	*Tennis*	*Fl.Gard.*	*Miss Am.*	*Coastg.*	*Foreman*	*Bus*	*Bicycle*
	Frame size:	288×352	240×352	240×352	288×360	144×176	288×352	288×352	576×720
	Frames:	50	150	150	150	300	300	150	30
5 / 34.15		40.44	38.47	36.49	41.58	38.25	39.77	37.55	40.89
10 / 28.13		37.21	34.68	32.11	39.61	34.78	36.46	33.32	37.62
15 / 24.61		35.44	32.63	29.81	38.64	33.00	34.64	31.05	35.67
20 / 22.11		34.04	31.20	28.24	37.85	31.71	33.30	29.57	34.18
25 / 20.17		32.79	30.11	27.00	37.10	30.62	32.19	28.48	32.90
30 / 18.59		31.68	29.22	25.89	36.41	29.68	31.27	27.59	31.77
35 / 17.25		30.72	28.56	25.16	35.87	28.92	30.56	26.91	30.85

Table 1: Output PSNR (dB) of the proposed V-BM3D algorithm for a few image sequences; the noise is i.i.d. Gaussian with variance σ^2 and zero mean. The PSNR was computed globally on each whole sequence.

σ	PSNR after Step 1	PSNR after Step 2
10	35.44	36.46
15	33.59	34.64
20	32.12	33.30
25	30.86	32.19

Table 2: PSNR improvement after applying Step 2 of our algorithm on the *Foreman* test sequence.

tant for the effectiveness of the overall approach. This is due to the improved grouping and the improved shrinkage by empirical Wiener filtering both of which are made possible by utilizing the basic (intermediate) estimate. In Table 2 one can compare the PSNR corresponding to both the basic estimate and the final one; the improvement is substantial, exceeding 1 dB in all cases shown there.

Since our approach uses the same assumptions that are used for the non-local estimation denoising, it is worth comparing the two approaches. The non-local means uses weighted averaging to obtain the final pixel estimate, where the weights depend on the similarity between the 2D patches (e.g. blocks) centered at the averaged pixels and the patch centered at the estimated pixel. In order to achieve good performance, this approach needs to capture plenty of very similar (in the ideal case, identical) patches. In the proposed method, we use a rather different approach where instead of a simple weighted average we use a complete decorrelating linear transform. The higher-order terms of the transform can approximate also variations between the spectral components of the grouped blocks, enabling a good filtering also for relatively dissimilar blocks. The subsequent shrinkage allows to take advantage of the sparsity by preserving the high-magnitude coefficients and truncating the ones with small magnitudes that are mostly due to noise.

The weights in the adopted aggregation (Steps 1.2 and 2.2 of the algorithm) favour blockwise estimates coming from sparsely represented groups. Such groups have few nonzero coefficients after hard-thresholding (Step 1.1.b) and few Wiener attenuation coefficients close to unity (Step 2.1.b).

Computational scalability of the V-BM3D can be achieved as in the image denoising counterpart of the algorithm [8] by varying certain parameters. The most important parameters that allow for such trade-off between denoising quality and complexity are the sliding step N_{step} and the block-matching parameters N_S, N_{PR}, and N_B.

7. CONCLUSIONS

In this work, we proposed a video denoising method that is both computationally efficient and achieves state-of-the-art results in terms of both PSNR and visual quality. These results are consistent with the ones already obtained by the image denoising counterpart [8] of the approach. We are currently working on extensions of the proposed method. A detailed analysis of its complexity and implementation issues as well as its application to color-video denoising will be reported in a forthcoming full-length publication.

REFERENCES

[1] E. Balster, Y. Zheng, and R. Ewing, “Combined spatial and temporal domain wavelet shrinkage algorithm for video denoising,” *IEEE Trans. Circuits Syst. Video Tech.*, vol. 16, no. 2, pp. 220–230, February 2006.

[2] F. Jin, P. Fieguth, and L. Winger, “Wavelet video denoising with regularized multiresolution motion estimation,” *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. Article ID 72 705, 11 pages, 2006.

[3] V. Zlokolica, A. Pizurica, and W. Philips, “Wavelet-domain video denoising based on reliability measures,” *IEEE Trans. Circuits Syst. Video Tech.*, vol. 16, no. 8, pp. 993–1007, August 2006.

[4] S. M. M. Rahman, M. O. Ahmad, and M. N. S. Swamy, “Video denoising based on inter-frame statistical modeling of wavelet coefficients,” *IEEE Trans. Circuits Syst. Video Tech.*, vol. 17, no. 2, pp. 187–198, February 2007.

[5] I. Selesnick and K. Li, “Video denoising using 2D and 3D dual-tree complex wavelet transforms,” in *Proc. Wavelet Applicat. Signal Image Process. X, SPIE*, San Diego, USA, August 2003.

[6] A. Buades, B. Coll, and J. M. Morel, “A review of image denoising algorithms, with a new one,” *Multisc. Model. Simulat.*, vol. 4, no. 2, pp. 490–530, 2005.

[7] A. Buades, B. Coll, and J. Morel, “Denoising image sequences does not require motion estimation,” in *Proc. IEEE Conf. Adv. Video Signal Based Surveil., AVSS*, Palma de Mallorca, Spain, September 2005, pp. 70–74.

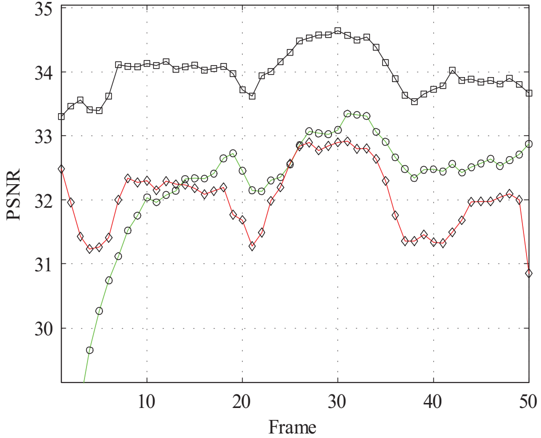
[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3D transform-domain collaborative filtering,” *IEEE Trans. Image Process.*, vol. 16, no. 8, August 2007, to appear in.

[9] D. Rusanovskyy, K. Dabov, and K. Egiazarian, “Moving-window varying size 3D transform-based video denoising,” in *Proc. Int. Workshop on Video Process. and Quality Metrics*, Scottsdale, Arizona, USA, January 2006.

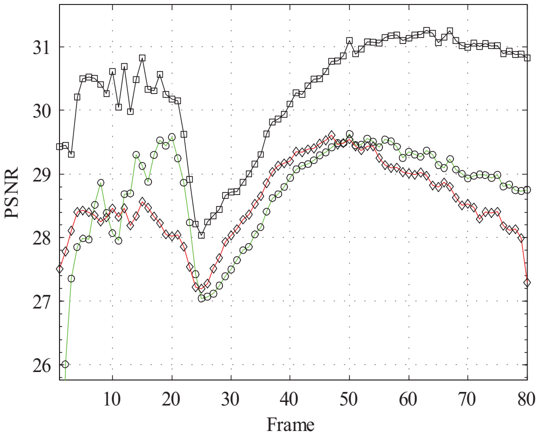
[10] C.-L. Fang, W.-Y. Chen, Y.-C. Liu, and T.-H. Tsai, “A new adaptive return prediction search algorithm for block matching,” in *Proc. IEEE Pacific Rim Conference on Multimedia*, vol. 2532/2002, Hsinchu, Taiwan, December 2002, pp. 120–126.

[11] C.-C. Wan and C.-H. Hsieh, “An efficient fractal image-coding method using interblock correlation search,” *IEEE Trans. Circuits Syst. Video Tech.*, vol. 11, no. 2, pp. 257–261, February 2001.

Salesman ($\sigma = 20$)



Tennis ($\sigma = 20$)



Flower garden ($\sigma = 20$)

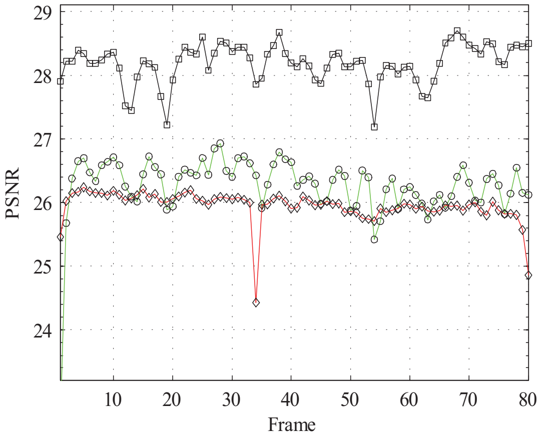


Figure 2: Per-frame PSNR comparison of the proposed V-BM3D ('□' marker) with the WRSTF [3] ('○' marker) and the 3DWTF [5] ('◇' marker).

3DWTF [5] (PSNR 28.12 dB)

WRSTF [3] (PSNR 28.83 dB)

Proposed V-BM3D (PSNR 30.93 dB)

Figure 3: Fragment of the 77th frame of *Tennis* denoised by the 3DWTF, the WRSTF, and the proposed V-BM3D; the noise had $\sigma = 20$. The output PSNR (for this frame only) is given in parentheses for each of the methods.

Publication [P5]

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Joint image sharpening and denoising by 3D transform-domain collaborative filtering," *Proc. Int. TICSP Workshop Spectral Meth. Multirate Signal Process.*, Moscow, Russia, September 2007.

JOINT IMAGE SHARPENING AND DENOISING BY 3D TRANSFORM-DOMAIN COLLABORATIVE FILTERING

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian

Institute of Signal Processing, Tampere University of Technology,
P.O. Box 553, Tampere 33101, Finland
firstname.lastname@tut.fi

ABSTRACT

In order to simultaneously sharpen image details and attenuate noise, we propose to combine the recent block-matching and 3D filtering (BM3D) denoising approach, based on 3D transform-domain collaborative filtering, with alpha-rooting, a transform-domain sharpening technique. The BM3D exploits grouping of similar image blocks into 3D arrays (groups) on which collaborative filtering (by hard-thresholding) is applied. We propose two approaches of sharpening by alpha-rooting; the first applies alpha-rooting individually on the 2D transform spectra of each grouped block; the second applies alpha-rooting on the 3D-transform spectra of each 3D array in order to sharpen fine image details shared by all grouped blocks and further enhance the interblock differences. The conducted experiments with the proposed method show that it can preserve and sharpen fine image details and effectively attenuate noise.

1. INTRODUCTION

The problem of sharpening image details while attenuating noise arises in practice as a preprocessing step prior to segmentation, binarization (by thresholding) in the context of image analysis, classification, pattern recognition. The procedure is also very useful in consumer applications, where it is utilized to obtain more visually appealing digital photographs. Various methods for image sharpening exist. Traditional ones rely on linear filtering that boosts higher frequencies or on elementwise transformations such as the histogram-based methods (equalization, matching, and shaping). Current advances [1, 2, 3] in the field exploit transforms such as wavelet decompositions and trigonometric transforms (DCT, DFT, etc.) to improve the efficiency of the sharpening. The sharpening is typically achieved by amplifying certain parts of the transform spectrum. Example of a well established such technique is the alpha-rooting [4, 5], where the amplification is achieved by taking the α-root of the magnitude of the original coefficients, for some constant $\alpha > 1$. A recent work [3] showed that histograms of the logarithm of a transform spectrum can also be utilized for sharpening using methods such as histogram matching and shaping.

An inherent drawback of most sharpening methods is the amplification of the noise component that is inevitably present when dealing with practical applications.

Recently, we proposed a highly effective image denoising method [6], namely the block-matching and 3D filtering (BM3D), based on filtering of enhanced sparse non-local image representations. Using block-wise processing, we group blocks similar to the currently processed one into 3D arrays, termed “groups”. Subsequently, we apply what we call “collaborative filtering” on each of these groups. In [6] we showed that this filtering can be efficiently realized by shrinkage (e.g., hard-thresholding) in 3D-transform domain. It is exactly in the 3D-transform domain where the enhanced sparse representation of the true signal is attained. Hence, the shrinkage of the transform coefficients allows for both good noise attenuation and faithful detail preservation. The result of the collaborative filtering of a group is a set of grouped block estimates, which are then returned to their original location in the image domain. Since the block estimates from a given group can mutually overlap and also overlap with ones from different groups, we aggregate them by a weighted averaging in order to obtain a single estimate of each image pixel.

In this paper we propose to combine the BM3D denoising approach with alpha-rooting, in order to simultaneously sharpen image details and attenuate noise. Hard-thresholding is applied on the 3D-transform spectrum of each group to attenuate the noise. Subsequently, sharpening is realized by modifying the thresholded transform coefficients of the grouped blocks. We show experimental results corresponding to two approaches, one with alpha-rooting performed on the 3D transform spectrum and the other with alpha-rooting applied separately on the 2D transform spectrum of each grouped block. The joint application of the collaborative filtering and alpha-rooting allows for both good noise suppression and effective preservation and sharpening of even very fine image details. It is worth noting that the adopted 3D transform is a separable composition of a 2D transform (on each block) and a 1D transform (in the “temporal” dimension, along which blocks are stacked), which makes possible the application of alpha-rooting on 2D transform spectra. In the following sections we present the developed method and show that

---

This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program [2006 - 2011]).

it can be very effective when applied on both artificially degraded images and real retinal medical images.

2. PROPOSED METHOD

We consider a noisy image $z = y+\eta$, where y is the noise-free image (with poor contrast) and η is i.i.d. Gaussian noise with zero mean and variance σ^2. Following are the two variations of the proposed method, denominated BM3D-SH3D and BM3D-SH2D. The former performs alpha-rooting on 3D transform spectra and the latter on 2D transform spectra. Their corresponding flowcharts are shown in Figure 1.

1. Process overlapping blocks in a raster scan. For each such block, do the following:
   (a) Use block-matching to find the locations of the blocks in z that are similar to the currently processed one. Form a 3D array (group) by stacking the blocks located at the obtained locations.
   (b) Apply a 3D transform on the formed group.
   (c) Attenuate the noise by hard-thresholding the 3D transform spectrum.
   (d) Sharpening. We propose the following two alternatives.
      - BM3D-SH3D. Apply alpha-rooting on the hard-thresholded 3D transform spectrum and invert the 3D transform to produce filtered grouped blocks.
      - BM3D-SH2D. Invert the 1D transforms along the temporal dimension of the formed 3D array, then perform alpha-rooting separately on the 2D transform spectra of each grouped block and subsequently invert the 2D transforms to produce filtered grouped blocks.
2. Return the filtered blocks to their original locations in the image domain and compute the resultant filtered image by a weighted average of these filtered blocks.

Except for the alpha-rooting and a modification of the aggregation weights, both described below, the rest of the steps of the algorithm are taken without modification from the first step of BM3D [6] (the one that uses hard-thresholding). We refer the reader to [6] for details.

Given a transform spectrum t of a signal, which contains a DC coefficient termed $t(0)$, the alpha-rooting is performed as

$$t_{sh}(i) = \begin{cases} \operatorname{sign}[t(i)]\,|t(0)|\left|\frac{t(i)}{t(0)}\right|^{\frac{1}{\alpha}}, & \text{if } t(0) \neq 0 \\ t(i), & \text{otherwise,} \end{cases} \tag{1}$$

where t_{sh} is the spectrum of the resultant signal and where an exponent $\alpha > 1$ results in sharpening of image details [5]. This simple technique is applied for both the BM3D-SH3D and the BM3D-SH2D on 3D and 2D transform spectra, respectively. The alpha-rooting is motivated by fact that, except for the DC, the transform basis elements extract differential information of the signal. Its effectiveness is based on amplifying these coefficients according to Equation (1). Performing hard-thresholding prior to alpha-rooting resolves problems [7] of this sharpening technique regarding noise amplification (which causes spike artifacts).

As shown in [6], the overlapping filtered blocks can be effectively aggregated using weights that are inversely proportional to the total variance of the filtered groups to which the respective blocks belong. While for the denoising method described there, this total variance can be approximated as σ^2 times the number of retained (i.e. non-zero) coefficients after thresholding, in the case of joint denoising and sharpening, due to the amplification caused by alpha-rooting, the variance of each sharpened coefficient in the group could be different from the original σ^2.

We derive a simple estimator for the total residual variance in each group. Let t be the hard-thresholded 3D transform spectrum of a group and t_{sh} be the spectrum of the sharpened signal obtained by Equation 1. If $t(0) \neq 0$, we can rewrite $t_{sh}(i)$ as the product of two independent terms $\varrho_0 = |t(0)|^{1-\frac{1}{\alpha}}$ and $\rho_i = \operatorname{sign}[t(i)]\,|t(i)|^{\frac{1}{\alpha}}$. Assuming that the moduli of $t(0)$ and $t(i)$ are much larger than their standard deviation σ (which can be a reasonable assumption, since $t(0)$ is a DC coefficient and $t(i)$ survived thresholding), the variances of the product of these two terms can be roughly approximated (using the first partial derivatives) as follows

$$\begin{aligned} \operatorname{var}\{t_{sh}(i)\} &\simeq \left(\varrho_0'(E\{t(0)\})\,E\{\rho_i\}\right)^2 \operatorname{var}\{t(0)\} + \\ &\quad + \left(\rho_i'(E\{t(i)\})\,E\{\varrho_0\}\right)^2 \operatorname{var}\{t(i)\} \\ &\simeq \left(1-\frac{1}{\alpha}\right)^2 |E\{t(0)\}|^{-\frac{2}{\alpha}} E^2\{\rho_i\}\,\sigma^2 + \\ &\quad + \frac{1}{\alpha^2}|E\{t(i)\}|^{\frac{2}{\alpha}-2} E^2\{\varrho_0\}\,\sigma^2 \\ &\simeq \left(1-\frac{1}{\alpha}\right)^2 \left|\frac{E\{t(i)\}}{E\{t(0)\}}\right|^{\frac{2}{\alpha}} \sigma^2 + \\ &\quad + \frac{1}{\alpha^2}|E\{t(i)\}|^{\frac{2}{\alpha}-2} |E\{t(0)\}|^{2-\frac{2}{\alpha}}\,\sigma^2 \\ &\simeq \left(1-\frac{1}{\alpha}\right)^2 |t(0)|^{-\frac{2}{\alpha}} |t(i)|^{\frac{2}{\alpha}}\,\sigma^2 + \\ &\quad + \frac{1}{\alpha^2}|t(i)|^{\frac{2}{\alpha}-2} |t(0)|^{2-\frac{2}{\alpha}}\,\sigma^2 \\ &= \omega_i\sigma^2 \end{aligned}$$

Thus, the total variance of the filtered (i.e., thresholded and sharpened) group is approximated as

$$v = \sigma^2 + \sum_{t(i)\neq 0, i>0} \omega_i\sigma^2. \tag{2}$$

Consequently, the weights used for the aggregation are defined as the reciprocal of the above v. Note that these

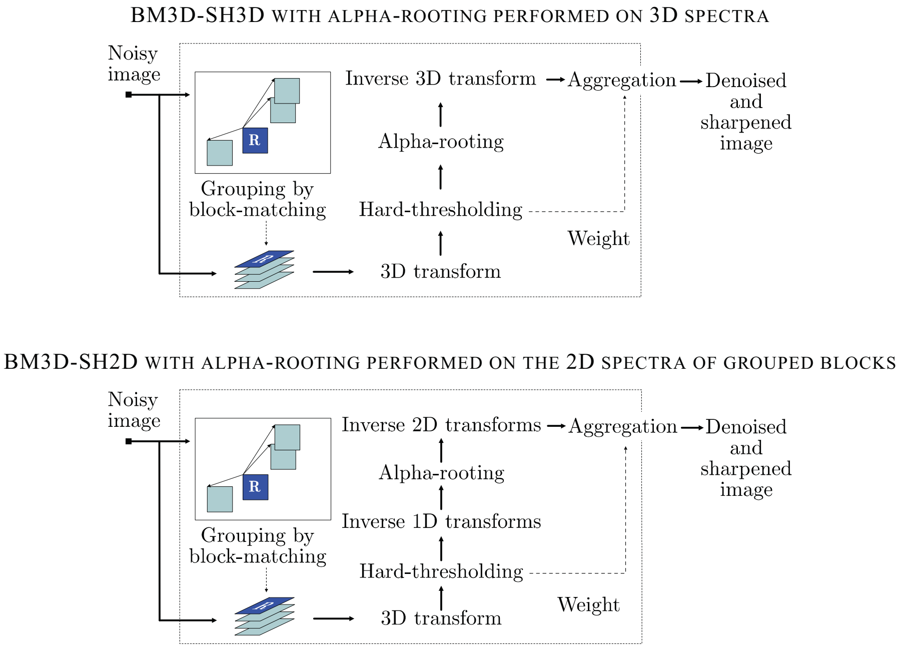


Figure 1. Flowcharts of the two variations of the proposed method. Top: BM3D-SH3D, which applies alpha-rooting on the 3D spectrum of each group; bottom: BM3D-SH2D, which applies alpha-rooting on the 2D spectrum of each grouped block.

weights are used only for BM3D-SH3D and for the BM3D-SH2D, for simplicity, we resort to the same weights used in BM3D [6]. (i.e. $\omega_i = 1$). Further, one can observe that if $\alpha = 1$, we obtain $\omega_i = 1$. Thus, (2) can be interpreted as a generalization of the weights used in [6].

3. RESULTS

We present experimental results from the two variations of the proposed method. Their implementation along with the original noise-free and the filtered images are available online[1]. We used the same algorithm parameters as in BM3D [6]; one can refer to the provided Matlab codes for additional details. The exponent α sets the desired level of sharpening. An illustration of applying the proposed method for a few values of α is given in Figure 2. If not specified otherwise, in our experiments we used $\alpha = 1.5$.

In Figures 4 and 5 one can see the results of applying the proposed method on images degraded by noise with $\sigma = 10, 20$. It can be observed that fine details are well preserved and sharpened while the noise is suppressed. It is worth recalling that the BM3D is particularly effective for objects that are formed by blocks for which there can be found plenty of similar blocks at different spatial locations. Such objects are edges, repeating patterns, and textures. Hence, these (e.g., the repeating circular objects in *Harbour* and the edges of the building in *Pentagon*) are well preserved and subsequently sharpened.

Figure 3 shows an example of denoising and sharpening a fragment from a real retinal image with added noise of $\sigma = 10, 20$. The resultant filtered images preserve and sharpen most of the image details while efficiently suppress noise. Moreover, the filtered images clearly reveal details which are hard to see even in the original image.

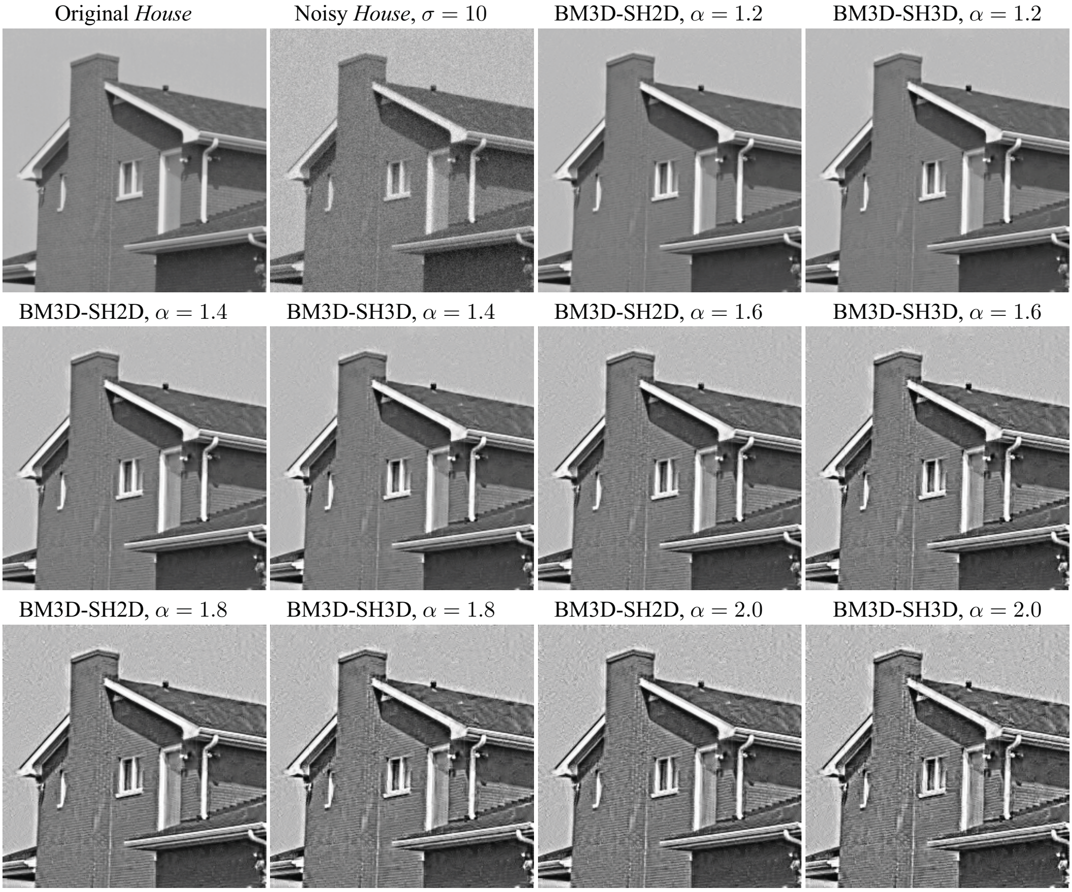
4. DISCUSSION AND CONCLUSIONS

We developed a method for joint denoising and sharpening of grayscale images. It inherits the outstanding denoising potential of the BM3D [6] and in combination with alpha-rooting, it achieves perceptually appealing filtered images where fine details are effectively sharpened.

Let us discuss the two variations of the proposed method. The BM3D-SH3D differs from BM3D-SH2D in that alpha-rooting is applied on the 3D transform spectrum which conveys information of the temporal differences in a group in addition to the spatial one (we call “temporal” the dimension along which blocks are stacked together in a group). In this 3D transform domain, the application of Equation (1) can amplify also differential information along this temporal dimension (and not along the spatial ones). It is noteworthy that in the case of ideal grouping, i.e. having identical blocks in a group, the two approaches produce identical filtered blocks (and differ only in the weighted aggregation of these blocks). The visual results of both approaches (Figures 4, 5, 3, 7) are quite similar by subjective evaluation. Therefore, we abstain from concluding which of them is better in practice. Instead, we

---

[1] Matlab code available at `www.cs.tut.fi/~foi/GCF-BM3D`.


Original *House* | Noisy *House*, $\sigma = 10$ | BM3D-SH2D, $\alpha = 1.2$ | BM3D-SH3D, $\alpha = 1.2$

BM3D-SH2D, $\alpha = 1.4$ | BM3D-SH3D, $\alpha = 1.4$ | BM3D-SH2D, $\alpha = 1.6$ | BM3D-SH3D, $\alpha = 1.6$

BM3D-SH2D, $\alpha = 1.8$ | BM3D-SH3D, $\alpha = 1.8$ | BM3D-SH2D, $\alpha = 2.0$ | BM3D-SH3D, $\alpha = 2.0$

Figure 2. Result of filtering with BM3D-SH2D and BM3D-SH3D for $\alpha = 1.2, 1.4, 1.6, 1.8, 2.0$. Higher values of α correspond to stronger sharpening of the details. The processed image is *House* with noise of standard deviation $\sigma = 10$.

only point to some of the noticeable differences. In the enlarged fragment of *Pentagon* in Figure 5, one can observe that for $\sigma = 10$ the BM3D-SH3D suppresses slightly better the noise in the smooth areas, mainly due to the aggregation weights defined from the total variance estimate in Equation (2). In Figure 7, we see that the BM3D-SH2D can preserve more effectively the oscillating curve. In addition, Figure 6 presents a plot of line 365 of *Pentagon*, which shows that BM3D-SH3D attains slightly better sharpening results than the BM3D-SH2D.

The proposed method can be extended for color data, where the denoising is performed on all three luminance-chrominance color channels as in [8] and the alpha-rooting is performed on all channels or only on the luminance one, depending on the application.

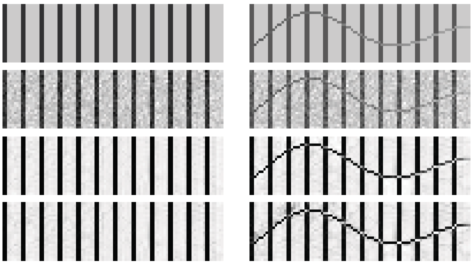


Figure 7. First row: noise-free artificial test images 24×96 pixels; second row: corresponding noisy images with $\sigma = 15$; third row: results of applying BM3D-SH2D; fourth row: results of applying BM3D-SH3D. For the experiments in this figure, both algorithms used $\alpha = 2$ and the sliding step of the BM3D was set to unity.

5. REFERENCES

[1] D.-Y. Tsai, Y. L. M. Sekiya, S. Sakaguchi, and I. Yamada, "A method of medical image enhancement using wavelet analysis," in *Proc. IEEE Int. Conf. Signal Process.*, vol. 1, August 2002, pp. 723–726.

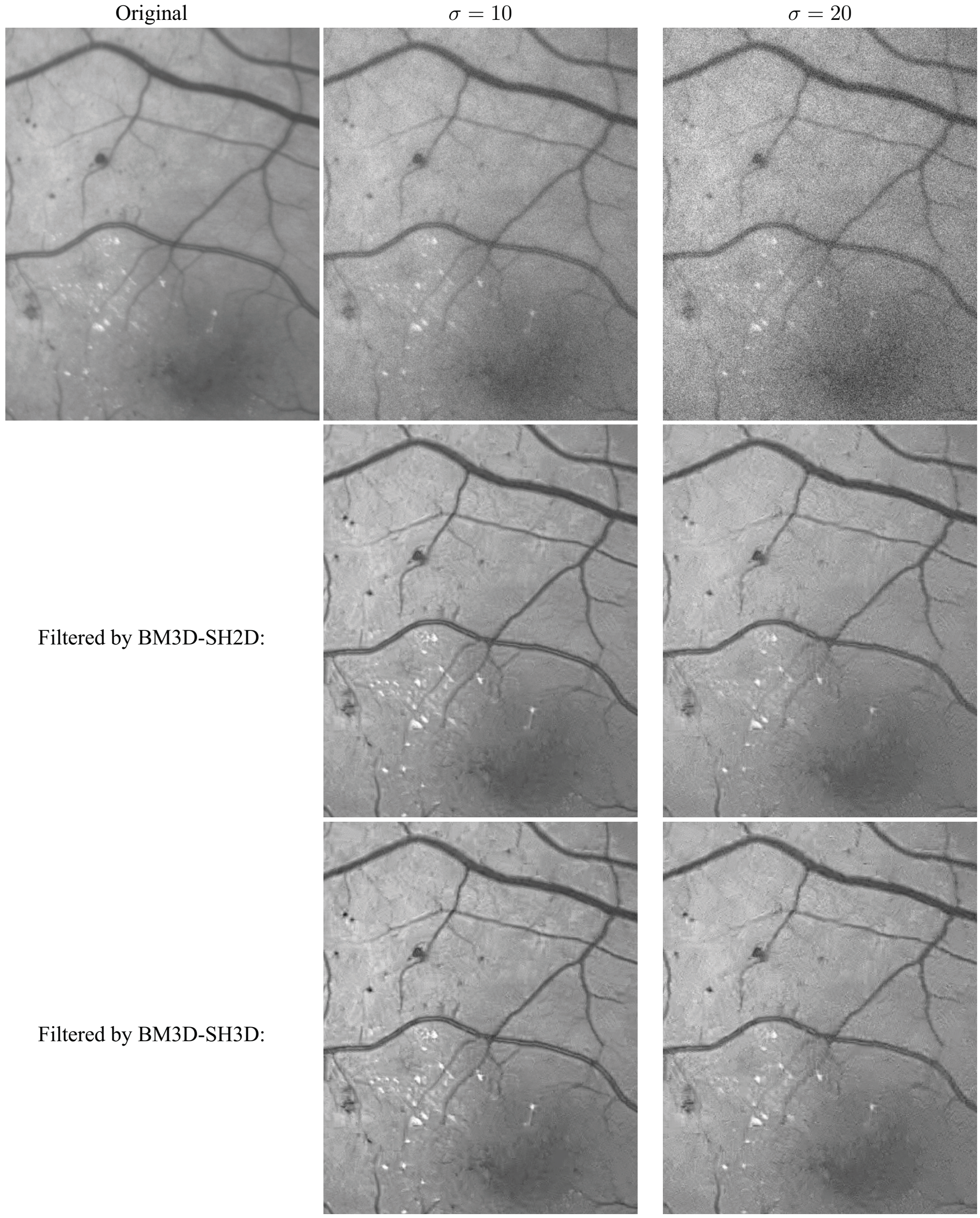


Figure 3. Results of applying the proposed methods on a fragment of a retinal fundus image. The second and third colums correspond to noise with $\sigma = 10$ and $\sigma = 20$, respectively.

[2] S. Hatami, R. Hosseini, M. Kamarei, and H. Ahmadi, "Wavelet based fingerprint image enhancement," in *Proc. IEEE Int. Symp. Circ. Syst. (ISCAS)*, Kobe, Japan, May 2005.

[3] S. Agaian, B. Silver, and K. A. Panetta, "Transform coefficient histogram-based image enhancement algorithms using contrast entropy," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 741–758, March 2007.

[4] W. K. Pratt, *Digital Image Processing*. John Wiley and Sons, New York, 1978.

[5] S. Aghagolzadeh and O. K. Ersoy, "Transform image

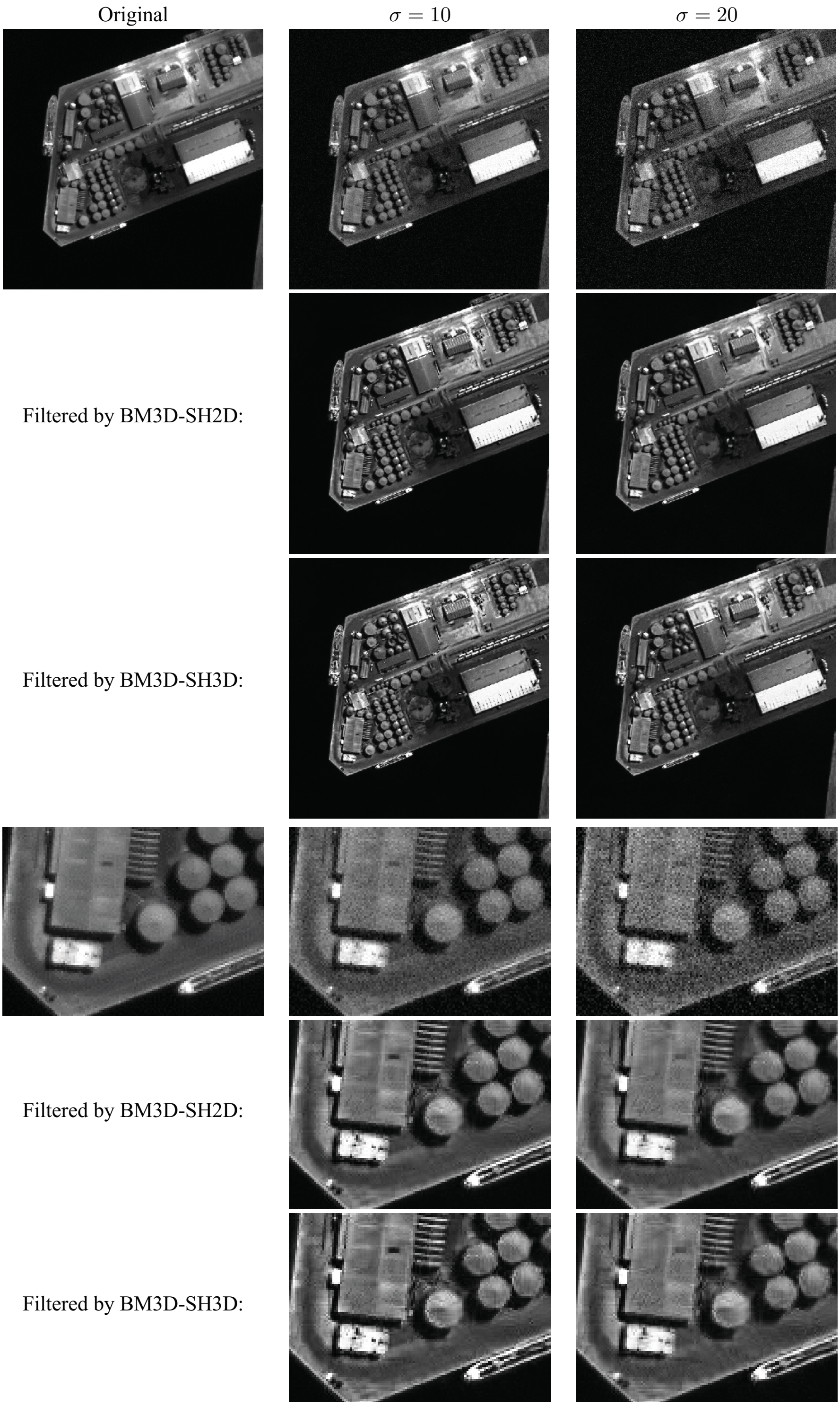


Figure 4. Results of applying the proposed methods on the noisy *Harbour* image (upper half) and a fragment of it (lower half). The second and third colums correspond to noise with $\sigma = 10$ and $\sigma = 20$, respectively.

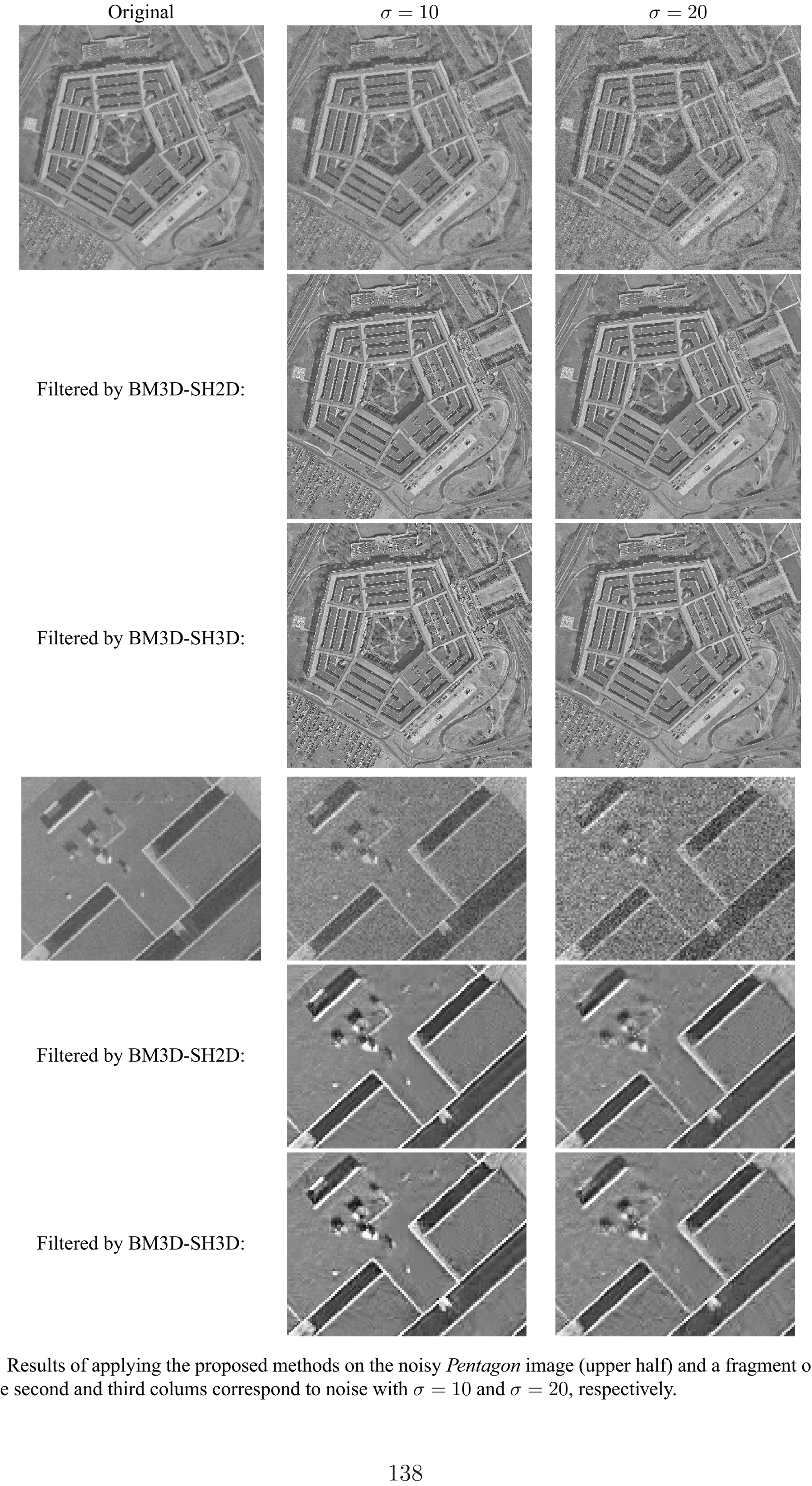


Figure 5. Results of applying the proposed methods on the noisy *Pentagon* image (upper half) and a fragment of it (lower half). The second and third colums correspond to noise with $\sigma = 10$ and $\sigma = 20$, respectively.

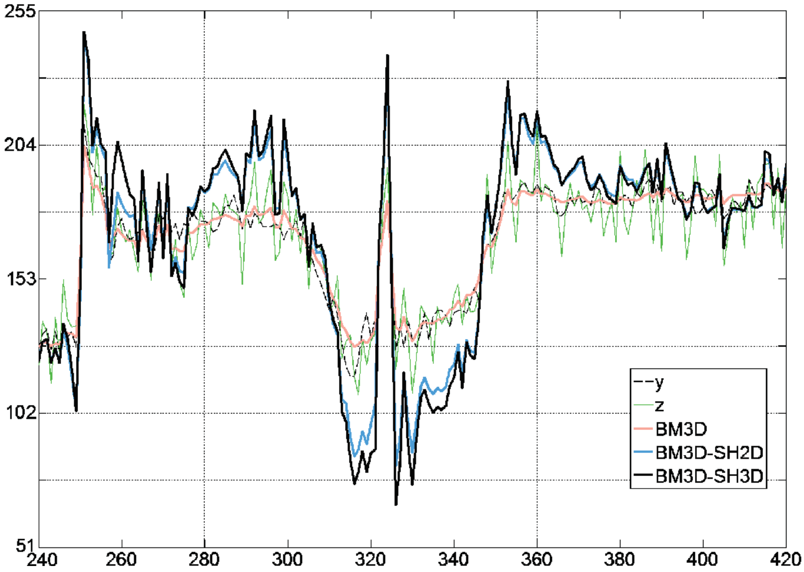


Figure 6. Line 365 of *Pentagon* image for: the noise-free image y, a noisy ($\sigma = 10$) image z, the result of BM3D denoising with no sharpening, the result of joint denoising and sharpening with BM3D-SH2D, and the result of joint denoising and sharpening with BM3D-SH3D.

enhancement," *Opt. Eng.*, vol. 31, no. 3, pp. 614–626, March 1992.

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

[7] J. McClellan, "Artifacts in alpha-rooting of images," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, vol. 5, April 1980, pp. 449–452.

[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space," in *Proc. IEEE Int. Conf. Image Process.*, San Antonio, Texas, September 2007.

Publication [P6]

K. Dabov, A. Foi, and K. Egiazarian, “Image restoration by sparse 3D transform-domain collaborative filtering,” *Proc. SPIE Electronic Imaging*, no. 6812-07, San Jose, California, USA, January 2008.

Image restoration by sparse 3D transform-domain collaborative filtering

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian

Department of Signal Processing, Tampere University of Technology

P.O. Box 553, 33101 Tampere, Finland

firstname.lastname@tut.fi

ABSTRACT

We propose an image restoration technique exploiting regularized inversion and the recent block-matching and 3D filtering (BM3D) denoising filter. The BM3D employs a non-local modeling of images by collecting similar image patches in 3D arrays. The so-called collaborative filtering applied on such a 3D array is realized by transform-domain shrinkage. In this work, we propose an extension of the BM3D filter for colored noise, which we use in a two-step deblurring algorithm to improve the regularization after inversion in discrete Fourier domain. The first step of the algorithm is a regularized inversion using BM3D with collaborative hard-thresholding and the seconds step is a regularized Wiener inversion using BM3D with collaborative Wiener filtering. The experimental results show that the proposed technique is competitive with and in most cases outperforms the current best image restoration methods in terms of improvement in signal-to-noise ratio.

Keywords: image restoration, deconvolution, deblurring, block-matching, collaborative filtering

1. INTRODUCTION

Image blurring is a common degradation in imaging. In many cases, the blurring can be assumed space-invariant and thus modeled as a convolution of the true image with a fixed point-spread function (PSF). Such a model is given by

$$z(x) = (y \circledast v)(x) + \eta(x), \tag{1}$$

where y is the true (non-degraded) image, v is a blur PSF, η is i.i.d. Gaussian noise with zero mean and variance σ^2, and $x \in X$ is a 2D coordinate in the image domain X. The inversion of the blurring is in general an ill-posed problem; thus, even noise with very small magnitude, such as truncation noise due to limited-precision arithmetic, can cause extreme degradations after naive inversion. Regularization is a well known and extensively studied approach to alleviate this problem. It imposes some regularity conditions (e.g., smoothness) on the obtained image estimate and/or on its derivatives. Numerous approaches that employ regularization have been proposed; an introduction can be found for example in the books.[1, 2] In particular, an image restoration scheme that comprises of regularized inversion followed by denoising has been a basis of the current best-performing restoration methods.[3, 4] Such denoising after the inversion can be considered as part of the regularization since it attenuates the noise in the obtained solution (i.e. the solution is smoothed).

Various denoising methods can be employed to suppress the noise after the inversion. Filtering in multiresolution transform domain (e.g., overcomplete wavelet and pyramid transforms) was shown[4–6] to be effective for this purpose. In particular, the SV-GSM,[4] which employs Gaussian scale mixtures in overcomplete directional and multiresolution pyramids, is among the current best image deblurring methods. Another denoising technique used after regularized inversion[3, 7, 8] is the LPA-ICI[9] which exploits a non-parametric local polynomial fit in anisotropic estimation neighborhoods. The best results of the methods based on LPA-ICI were achieved by the shape-adaptive discrete cosine transform (SA-DCT) deblurring[3] where the denoising is realized by shrinkage of the SA-DCT applied on local neighborhoods whose arbitrary shapes are defined by the LPA-ICI.

This work was partly supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program [2006 - 2011]); the work of K. Dabov was supported by the Tampere Graduate School in Information Science and Engineering (TISE).

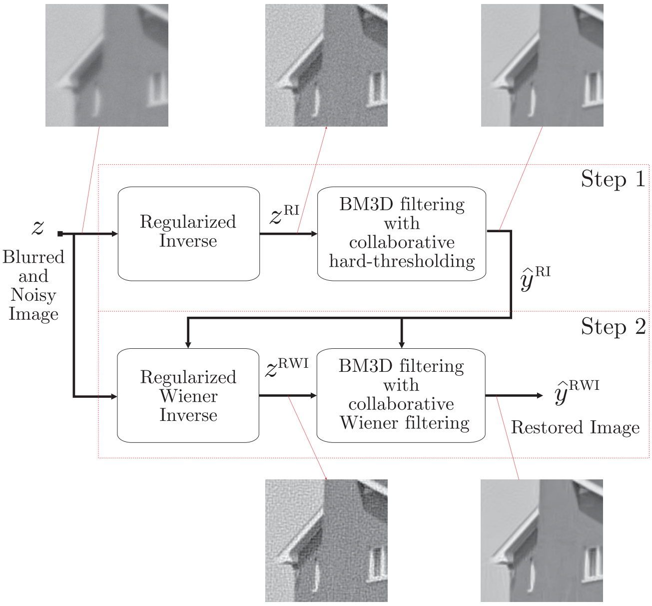


Figure 1. Flowchart of the proposed deconvolution algorithm. A fragment of *House* illustrates the images after each operation.

In this work we follow the above restoration scheme (regularized inversion followed by denoising) exploiting an extension of the block-matching and 3D filtering[10] (BM3D). This filter is based on the assumption that there exist mutually similar patches within a natural image — the same assumption used in other non-local image filters such as.[11, 12] The BM3D processes a noisy image in a sliding-window (block) manner, where block-matching is performed to find blocks similar to the currently processed one. The blocks are then stacked together to form a 3D array and the noise is attenuated by shrinkage in a 3D-transform domain. This results in a 3D array of filtered blocks. A denoised image is produced by aggregating the filtered blocks to their original locations using weighted averaging. This filter was shown[10] to be highly effective for attenuation of additive i.i.d. Gaussian (white) noise. The contribution of this work includes

- extension of the BM3D filter for additive colored noise, and
- image deblurring method that exploits the extended BM3D filter for improving the regularization after regularized inversion in Fourier transform domain.

The paper is organized as follows. The developed image restoration method and the extension of the BM3D filter are presented in Sections 2. Simulation results and a brief discussion are given in Section 3 and relevant conclusions are made in Section 4.

2. IMAGE RESTORATION WITH REGULARIZATION BY BM3D FILTERING

The observation model given in Equation (1) can be expressed in discrete Fourier transform (DFT) domain as

$$Z = YV + \tilde{\eta}, \tag{2}$$

where Y, V, and $\tilde{\eta}$ are the DFT spectra of y, v, and η, respectively. Capital letters denote DFT of a signal; e.g. $Z = \mathcal{F}\{z\}$, $V = \mathcal{F}\{v\}$; the only exception in that notation is for $\tilde{\eta} = \mathcal{F}\{\eta\}$. Due to the normalization of the forward DFT, the variance of $\tilde{\eta}$ is $|X|\,\sigma^2$, where $|X|$ is the cardinality of the set X (i.e., $|X|$ is the number of pixels in the input image).

Given the input blurred and noisy image z, the blur PSF v, and the noise variance σ^2, we apply the following two-step image deblurring algorithm, which is illustrated in Figure 1.

Proposed two-step image deblurring algorithm

Step 1. Regularized Inversion (RI) using BM3D with collaborative hard-thresholding.

1.1. The regularized inverse z^{RI} is computed in DFT domain as

$$\begin{aligned} T^{\mathrm{RI}} &= \frac{\bar{V}}{|V|^2 + \alpha_{\mathrm{RI}}\,|X|\,\sigma^2} \\ z^{\mathrm{RI}} &= \mathcal{F}^{-1}\left\{T^{\mathrm{RI}} Z\right\} \\ &= \mathcal{F}^{-1}\left\{Y \frac{|V|^2}{|V|^2 + \alpha_{\mathrm{RI}}\,|X|\,\sigma^2}\right\} + \mathcal{F}^{-1}\left\{\tilde{\eta} T^{\mathrm{RI}}\right\}, \end{aligned} \tag{3}$$

where α_{RI} is a regularization parameter determined empirically. Note that the obtained inverse z^{RI} is the sum of $\mathcal{F}^{-1}\left\{Y \frac{|V|^2}{|V|^2+\alpha_{\mathrm{RI}}|X|\sigma^2}\right\}$, a biased estimate of y, and the colored noise $\mathcal{F}^{-1}\left\{\tilde{\eta} T^{\mathrm{RI}}\right\}$.

1.2. Attenuate the colored noise in z^{RI} given by Eq. (3) using BM3D with collaborative hard-thresholding (see Section 2.1); the denoised image is denoted $\hat{y}^{\mathrm{RI}}$.

Step 2. Regularized Wiener inversion (RWI) using BM3D with collaborative Wiener filtering.

2.1. Using $\hat{y}^{\mathrm{RI}}$ as a reference estimate, compute the regularized Wiener inverse z^{RWI} as

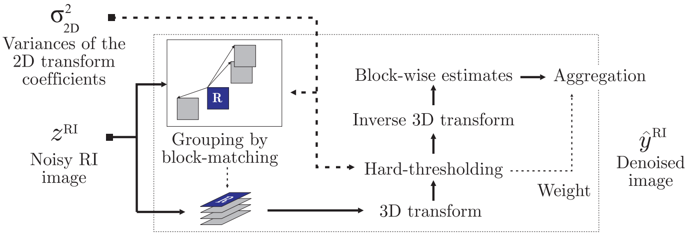
$$\begin{aligned} T^{\mathrm{RWI}} &= \frac{\bar{V}\left|\hat{Y}^{\mathrm{RI}}\right|^2}{\left|V\hat{Y}^{\mathrm{RI}}\right|^2 + \alpha_{\mathrm{RWI}}\,|X|\,\sigma^2}, \\ z^{\mathrm{RWI}} &= \mathcal{F}^{-1}\left\{T^{\mathrm{RWI}} Z\right\}, \\ z^{\mathrm{RWI}} &= \mathcal{F}^{-1}\left\{Y \frac{\left|V\hat{Y}^{\mathrm{RI}}\right|^2}{\left|V\hat{Y}^{\mathrm{RI}}\right|^2 + \alpha_{\mathrm{RWI}}\,|X|\,\sigma^2}\right\} + \mathcal{F}^{-1}\left\{\tilde{\eta} T^{\mathrm{RWI}}\right\} \end{aligned} \tag{4}$$

where, analogously to Eq. (3), α_{RWI} is a regularization parameter and z^{RWI} is the sum of a biased estimate of y and colored noise.

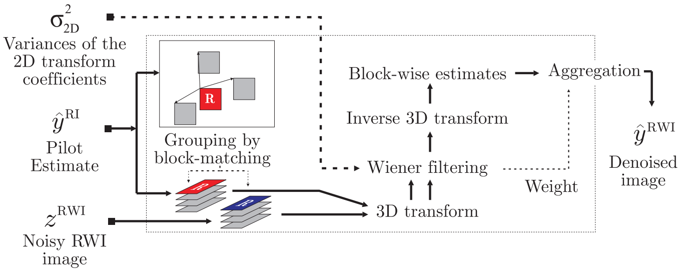
2.2. Attenuate the colored noise in z^{RWI} using BM3D with collaborative Wiener filtering (see Section 2.2) which also uses $\hat{y}^{\mathrm{RI}}$ as a pilot estimate. The result $\hat{y}^{\mathrm{RWI}}$ of this denoising is the final restored image.

The BM3D filtering of the colored noise (Steps 1.2 and 2.2) plays the role of a further regularization of the sought solution. It allows the use of relatively small regularization parameters in the Fourier-domain inverses, hence reducing the bias in the estimates z^{RI} and z^{RWI}, which are instead essentially noisy. The BM3D denoising filter[10] is originally developed for additive white Gaussian noise. Thus, to enable the attenuation of colored noise, we propose some modifications to the original filter.

Before we present the extensions that enable attenuation of colored noise, we recall how the BM3D filter works; for details of the original method one can refer to.[10] The BM3D processes an input image in a sliding-window manner, where the window (block) has a fixed size $N_1 \times N_1$. For each processed block a 3D array is



(a) BM3D with collaborative hard-thresholding



(b) BM3D with collaborative Wiener filtering

Figure 2. Flowcharts of the BM3D filter extentions for colored-noise removal.

formed by stacking together blocks (from various image locations) which are similar to the current one. This process is called "grouping" and is realized by block-matching. Consequently, a separable 3D transform $\mathcal{T}_{3D}$ is applied on the 3D array in such a manner that first a 2D transform, $\mathcal{T}_{2D}$, is applied on each block in the group and then a 1D transform, $\mathcal{T}_{1D}$, is applied in the third dimension. The noise is attenuated by shrinkage (e.g. hard-thresholding or empirical Wiener filtering) of the $\mathcal{T}_{3D}$-transform spectrum. Subsequently, the transform $\mathcal{T}_{3D}$ is inverted and each of the filtered blocks in the group is returned to its original location. After processing the whole image, since the filtered blocks can (and usually do) mutually overlap, they are aggregated by weighted averaging to form a final denoised image.

If the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{1D}$ are orthonormal, the grouped blocks are non-overlapping, and the noise in the input image is i.i.d. Gaussian, then the noise in the $\mathcal{T}_{3D}$-transform domain is also i.i.d. Gaussian with the same constant variance. However, if the noise is colored as in the case of Eq. (3), then the variances $\sigma^2_{2D}(i)$, for $i = 1, \ldots, N_1^2$, of the $\mathcal{T}_{2D}$-transform coefficients are in general non constant. In the following subsections, we extend the BM3D filter to attenuate such colored noise. We note that the developed extensions are not necessarily restricted to the considered image restoration scheme but are applicable to filtering of colored noise in general.

Let us introduce the notation used in what follows. With z_x^{RI} we denote a 2D block of fixed size $N_1 \times N_1$ extracted from z^{RI}, where $x \in X$ is the coordinate of the top-left corner of the block. Let us note that this block notation is different from the one (capital letter with subscript) used in[10] since the capital letter in this paper is reserved for the DFT of an image. A group of collected 2D blocks is denoted by a bold-face letter with a subscript indicating the set of its grouped blocks' coordinates: e.g., $\mathbf{z}_S^{RI}$ is a 3D array composed of the blocks z_x^{RI}, $\forall x \in S \subseteq X$.

2.1. BM3D with collaborative hard-thresholding (Step 1.2)

This filtering is applied on the noisy z^{RI} given by Eq. (3). The variances of the coefficients of a $\mathcal{T}_{2D}$-transform (applied to an arbitrary image block) are computed as

$$\sigma^2_{2D}(i) = \frac{\sigma^2}{|X|}\left\|T^{RI}\mathcal{F}\left\{\psi^{(i)}_{\mathcal{T}_{2D}}\right\}\right\|_2^2, \quad \forall i = 1, \ldots, N_1^2, \tag{5}$$

where $\psi^{(i)}_{\mathcal{T}_{2\mathrm{D}}}$ is the i-th basis element of $\mathcal{T}_{2\mathrm{D}}$. The flowchart of the BM3D with collaborative hard-thresholding extended for color-noise attenuation is given in Figure 2(a).

The variances $\sigma^2_{2\mathrm{D}}$ are used in the block-matching to reduce the influence of noisier transform coefficients when determining the block-distance. To accomplish this, the block-distance is computed as the ℓ^2-norm of the difference between the two $\mathcal{T}_{2\mathrm{D}}$-transformed blocks scaled by the corresponding standard deviations of the $\mathcal{T}_{2\mathrm{D}}$-transform coefficients. Thus, the distance is given by

$$d\left(z^{\mathrm{RI}}_{x_R}, z^{\mathrm{RI}}_{x}\right) = N_1^{-2} \left\| \frac{\mathcal{T}_{2\mathrm{D}}\left\{z^{\mathrm{RI}}_{x_R}\right\} - \mathcal{T}_{2\mathrm{D}}\left\{z^{\mathrm{RI}}_{x}\right\}}{\sigma_{2\mathrm{D}}} \right\|_2^2, \tag{6}$$

where $z^{\mathrm{RI}}_{x_R}$ is the current reference block, z^{RI}_{x} is an arbitrary block in the search neighborhood, and the operations between the three $N_1 \times N_1$ arrays $\mathcal{T}_{2\mathrm{D}}\left\{z^{\mathrm{RI}}_{x_R}\right\}$, $\mathcal{T}_{2\mathrm{D}}\left\{z^{\mathrm{RI}}_{x}\right\}$, and $\sigma_{2\mathrm{D}}$ are elementwise. After the best-matching blocks are found (their coordinates are saved as the elements of the set S_{x_R}) and grouped together in a 3D array, collaborative hard-thresholding is applied. It consists of applying the 3D transform $\mathcal{T}_{3\mathrm{D}}$ on the 3D group, hard-thresholding its spectrum, and then inverting the $\mathcal{T}_{3\mathrm{D}}$. To attenuate the colored noise, the hard-threshold is made dependent on the variance of each $\mathcal{T}_{3\mathrm{D}}$-transform coefficient. Due to the separability of $\mathcal{T}_{3\mathrm{D}}$, this variance depends only on the corresponding 2D coordinate within the $\mathcal{T}_{3\mathrm{D}}$-spectrum; thus, along the third dimension of a group the variance and hence the threshold are the same. The hard-thresholding is performed by an elementwise multiplication of the $\mathcal{T}_{3\mathrm{D}}$-spectrum $\mathcal{T}_{3\mathrm{D}}\left\{\mathbf{z}^{\mathrm{RI}}_{S_{x_R}}\right\}$ with the 3D array $\mathbf{h}_{x_R}$ defined as

$$\mathbf{h}_{x_R}(i,j) = \begin{cases} 1, & \text{if } \left|\mathcal{T}_{3\mathrm{D}}\left\{\mathbf{z}^{\mathrm{RI}}_{S_{x_R}}\right\}(i,j)\right| > \lambda_{3\mathrm{D}}\sigma_{2\mathrm{D}}(i) \\ 0, & \text{otherwise,} \end{cases} \quad, \quad \forall i = 1, \ldots, N_1^2, \, \forall j = 1, \ldots, \left|S_{x_R}\right|,$$

where i is a spatial-coordinate index and j is an index of the coefficients in the third dimension, $\lambda_{3\mathrm{D}}$ is a fixed threshold coefficient and $\left|S_{x_R}\right|$ denotes the cardinality of the set S_{x_R}.

After all reference blocks are processed, the filtered blocks are aggregated by a weighted averaging, producing the denoised image $\hat{y}^{\mathrm{RI}}$. The weight for all filtered blocks in an arbitrary 3D group is the inverse of the sum of the variances of the non-zero transform coefficients after hard-thresholding; for a 3D group using $x_R \in X$ as reference, the weight is

$$w^{\mathrm{ht}}_{x_R} = \frac{1}{\sum\limits_{\substack{i=1,\ldots,N_1^2 \\ j=1,\ldots,\left|S_{x_R}\right|}} \mathbf{h}_{x_R}(i,j)\,\sigma^2_{2\mathrm{D}}(i)}.$$

2.2. BM3D with collaborative Wiener filtering (Step 2.2)

The BM3D with collaborative empirical Wiener filtering uses $\hat{y}^{\mathrm{RI}}$ as a reference estimate of the true image y. Since the grouping by block-matching is performed on this estimate and not on the noisy image, there is no need to modify the distance calculation as in Eq. (6). The only modification from Step 2 of the original BM3D filter concerns the different variances of the $\mathcal{T}_{3\mathrm{D}}$-transform coefficients in the empirical Wiener filtering. This filtering is performed by an elementwise multiplication of the $\mathcal{T}_{3\mathrm{D}}$-spectrum $\mathcal{T}_{3\mathrm{D}}\left\{\mathbf{z}^{\mathrm{RWI}}_{S_{x_R}}\right\}$ with the Wiener attenuation coefficients $\mathbf{w}_{x_R}$ defined as

$$\mathbf{w}_{x_R}(i,j) = \frac{\left|\mathcal{T}_{3\mathrm{D}}\left\{\widehat{\mathbf{y}}^{\mathrm{RI}}_{S_{x_R}}\right\}(i,j)\right|^2}{\left|\mathcal{T}_{3\mathrm{D}}\left\{\widehat{\mathbf{y}}^{\mathrm{RI}}_{S_{x_R}}\right\}(i,j)\right|^2 + \sigma^2_{2\mathrm{D}}(i)}, \quad \forall i = 1, \ldots, N_1^2, \forall j = 1, \ldots, \left|S_{x_R}\right|,$$

where, similarly to Eq. (5), the variances $\sigma^2_{2\mathrm{D}}$ of the $\mathcal{T}_{2\mathrm{D}}$-transform coefficients are computed as

$$\sigma^2_{2\mathrm{D}}(i) = \frac{\sigma^2}{|X|} \left\| T^{\mathrm{RWI}} \mathcal{F}\left\{\psi^{(i)}_{\mathcal{T}_{2\mathrm{D}}}\right\} \right\|_2^2, \quad \forall i = 1, \ldots, N_1^2. \tag{7}$$

For an arbitrary $x_R \in X$, the aggregation weight for its corresponding filtered 3D group is

$$w_{x_R}^{\text{wie}} = \frac{1}{\sum\limits_{\substack{i=1,\ldots,N_1^2 \\ j=1,\ldots,\left|S_{x_R}\right|}} \mathbf{w}_{x_R}^2\left(i,j\right)\sigma_{2\text{D}}^2\left(i\right)}.$$

The flowchart of the BM3D with collaborative Wiener filtering extended for color-noise attenuation is given in Figure 2(b).

3. RESULTS AND DISCUSSION

We present simulation results of the proposed algorithm, whose Matlab implementation is available online.[13] All parameters, obtained after a rough empirical optimization, are fixed in all experiments (invariant of the noise variance σ^2, blur PSF v, and image y) and can be inspected from the provided implementation. In our experiments, we used the same blur PSFs and noise combinations as in.[4] In particular, these PSFs are:

- PSF 1: $v\left(x_1,x_2\right) = 1/\left(1+x_1^2+x_2^2\right), x_1, x_2 = -7,\ldots,7$,
- PSF 2: v is a 9×9 uniform kernel (boxcar),
- PSF 3: $v = \left[1\ 4\ 6\ 4\ 1\right]^T\left[1\ 4\ 6\ 4\ 1\right]/256$,
- PSF 4: v is a Gaussian PSF with standard deviation 1.6,
- PSF 5: v is a Gaussian PSF with standard deviation 0.4.

All PSFs are normalized so that $\sum v = 1$.

Table 1 presents a comparison of the improvement in signal-to-noise ratio (ISNR) for a few methods[3, 4, 6, 14–16] among which are the current best.[3, 4] The results of ForWaRD[6] were obtained with the Matlab codes[17] made available by its authors, for which we used automatically estimated regularization parameters. The results of the SA-DCT deblurring[3] were produced with the Matlab implementation,[18] where however we used fixed regularization parameters in all experiment in order to have fair comparison (rather than using regularization parameters dependent on the PSF and noise). The results of the GSM method[19] and the SV-GSM[4] are taken from.[4] In most of the experiments, the proposed method outperforms the other techniques in terms of ISNR. We note that the results of the four standard experiments used in the literature (e.g.,[3, 6, 7, 20]) on image restoration are included in Table 1 as follows.

- Experiment 1: PSF2, $\sigma^2 = 0.308$, and *Cameraman* image.
- Experiment 2: PSF1, $\sigma^2 = 2$, and *Cameraman* image.
- Experiment 3: PSF1, $\sigma^2 = 8$, and *Cameraman* image.
- Experiment 4: PSF3, $\sigma^2 = 49$, and *Lena* image.

The visual quality of some of the restored images can be evaluated from Figures 4, 5, and 6. One can see that fine details are well preserved and there are few artifacts in the deblurred images. In particular, ringing can be seen in some images such as the ones shown in Figure 3, where a comparison with the SA-DCT deblurring[3] is made. The ringing is stronger (and ISNR is lower) in the estimate obtained by the proposed technique. We explain this as follows; let us recall that each of the noisy images z^{RI} and z^{RWI} (input to the extended BM3D filter) is sum of a bias and additive colored noise; the exact models of z^{RI} and z^{RWI} are given by Eq. (3) and (4), respectively. The ringing is part of the bias and thus it is not modeled as additive colored noise. Hence, if the ringing magnitude is relatively high, the BM3D fails to attenuate it and it is preserved in the final estimate, as in Figure 3.

By comparing the results corresponding to $\hat{y}^{\text{RI}}$ and $\hat{y}^{\text{RWI}}$ in Table 2, one can see the improvement in ISNR after applying the second step (RWI using BM3D with collaborative Wiener filtering) of our two-step restoration scheme. This improvement is significant and can be explained as follows. First, the regularized Wiener inverse is more effective than the regularized inverse because it uses the estimated power spectrum for the inversion given

Blur: PSF2, $\sigma^2 = 0.308$ (PSNR 20.76 dB)

SA-DCT deblurring[3] (ISNR 8.55 dB)

Proposed method (ISNR 8.34 dB)

Fragments of the SA-DCT result

Fragments of the result by the proposed method

Figure 3. Comparison of the proposed method with the SA-DCT deconvolution method for *Cameraman* and PSF 2 blur kernel.

in Eq. (4). Second, the block-matching in the BM3D filtering is more accurate because it is performed within the available estimate $\hat{y}^{\mathrm{RI}}$ rather than within the input noisy image z^{RWI}. Third, the empirical Wiener filtering used by the BM3D in that step is more effective than the simple hard-thresholding used in the first step. In fact, the first step can be considered as an adaptation step that significantly improves the actual restoration performed by the second step.

In Table 2, we also provide (in the row corresponding to $\hat{y}^{\mathrm{RWI}}_{\mathrm{naive}}$) the results of the naive approach of using the original BM3D filter[10] rather than the one extended for colored noise. This filter was applied on z^{RI} and z^{RWI} by assuming additive i.i.d. Gaussian noise, whose variance was computed as $\sigma^2_{\mathrm{WGN}} = N_1^{-2} \sum_{i=1}^{N_1^2} \sigma^2_{\mathrm{2D}}(i)$, where $\sigma^2_{\mathrm{2D}}(\cdot)$ is defined in Eq. (5) and (7) for z^{RI} and z^{RWI}, respectively. This variance calculation was empirically found to be better (in terms of ISNR) than estimating a noise variance from the noisy images z^{RI} and z^{RWI}. The benefit of using the BM3D for colored noise reaches 1 dB; in particular, the benefit is substantial for those experiments where the noise in z^{RI} and z^{RWI} is highly colored.

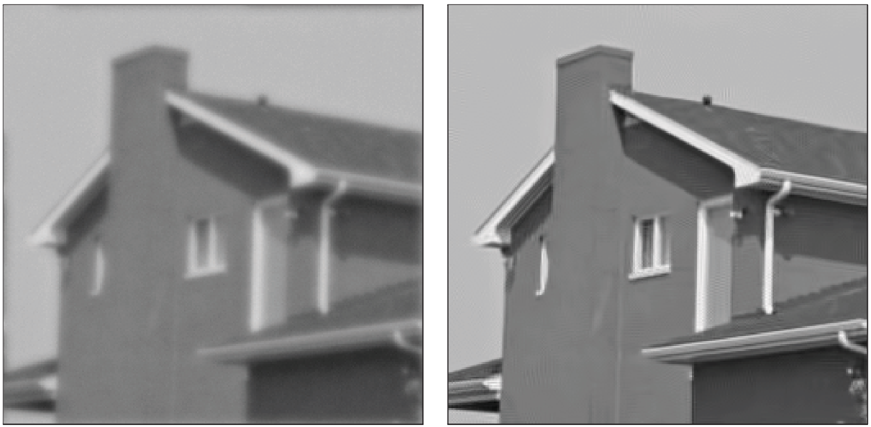
4. CONCLUSIONS

The developed image deblurring method outperforms the current best techniques in most of the experiments. This performance is in line with the BM3D denoising filter[10] which is among the current best denoising filters. The proposed colored-noise extension of the BM3D is not restricted to the developed deblurring method and it can in general be applied to filter colored noise.

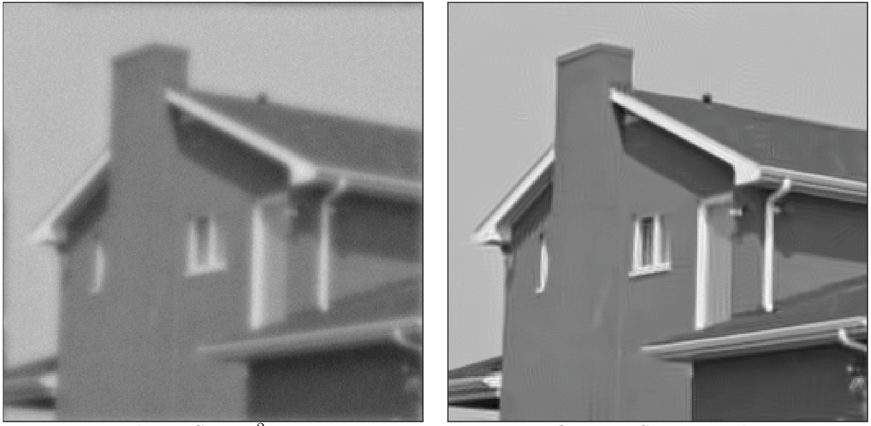
Future developments might target attenuation of ringing artifacts by exploiting the SA-DCT transform[21] which, as shown in Figure 3, is effective in suppressing them.

Blur	PSF 1		PSF 2	PSF 3	PSF 4	PSF 5
σ^2	2	8	0.308	49	4	64
Cameraman						
Input PSNR	22.23	22.16	20.76	24.62	23.36	29.82
ForWaRD[6]	6.76	5.08	7.34	2.40	3.14	3.92
GSM[19]	6.84	5.29	-1.61	2.56	2.83	3.81
EM[16]	6.93	4.88	7.59	-	-	-
Segm.-based Reg.[15]	7.23	-	8.04	-	-	-
GEM[14]	7.47	5.17	8.10	-	-	-
BOA[20]	7.46	5.24	8.16	-	-	-
Anis. LPA-ICI[7]	7.82	5.98	8.29	-	-	-
SV-GSM[4]	7.45	5.55	7.33	2.73	3.25	4.19
SA-DCT[3]	8.11	6.33	**8.55**	**3.37**	3.72	**4.71**
Proposed	**8.19**	**6.40**	8.34	3.34	**3.73**	4.70
Lena						
Input PSNR	27.25	27.04	25.84	28.81	29.16	30.03
Segm.-based Reg.[15]	-	-	-	1.34	-	-
GEM[14]	-	-	-	2.73	-	-
BOA[20]	-	-	-	2.84	-	-
ForWaRD[6]	6.05	4.90	6.97	2.93	3.50	5.42
EM[16]	-	-	-	2.94	-	-
Anis. LPA-ICI[7]	-	-	-	3.90	-	-
SA-DCT[3]	7.55	6.10	7.79	4.49	4.08	5.84
Proposed	**7.95**	**6.53**	**7.97**	**4.81**	**4.37**	**6.40**
House						
Input PSNR	25.61	25.46	24.11	28.06	27.81	29.98
ForWaRD[6]	7.35	6.03	9.56	3.19	3.85	5.52
GSM[19]	8.46	6.93	-0.44	4.37	4.34	5.98
SV-GSM[4]	8.64	7.03	9.04	4.30	4.11	6.02
SA-DCT[3]	9.02	7.74	10.50	4.99	**4.65**	5.96
Proposed	**9.32**	**8.14**	**10.85**	**5.13**	4.56	**7.21**
Barbara						
Input PSNR	23.34	23.25	22.49	24.22	23.77	29.78
ForWaRD[6]	3.69	1.87	4.02	0.94	0.98	3.15
GSM[19]	5.70	3.28	-0.27	1.44	0.95	4.91
SV-GSM[4]	6.85	3.80	5.07	**1.94**	**1.36**	5.27
SA-DCT[3]	5.45	2.54	4.79	1.31	1.02	3.83
Proposed	**7.80**	**3.94**	**5.86**	1.90	1.28	**5.80**

Table 1. Comparison of the output ISNR [dB] of a few deconvolution methods (only the rows corresponding to "Input PSNR" contain PSNR [dB] of the input blurry images).



Blur: PSF1, $\sigma^2 = 2$ Output ISNR 9.32 dB



Blur: PSF1, $\sigma^2 = 8$ Output ISNR 8.14 dB

Blur: PSF2, $\sigma^2 = 0.308$ Output ISNR 10.85 dB

Figure 4. Deblurring results of the proposed method for *House*.

Blur: PSF3, $\sigma^2 = 49$ Output ISNR 4.81 dB

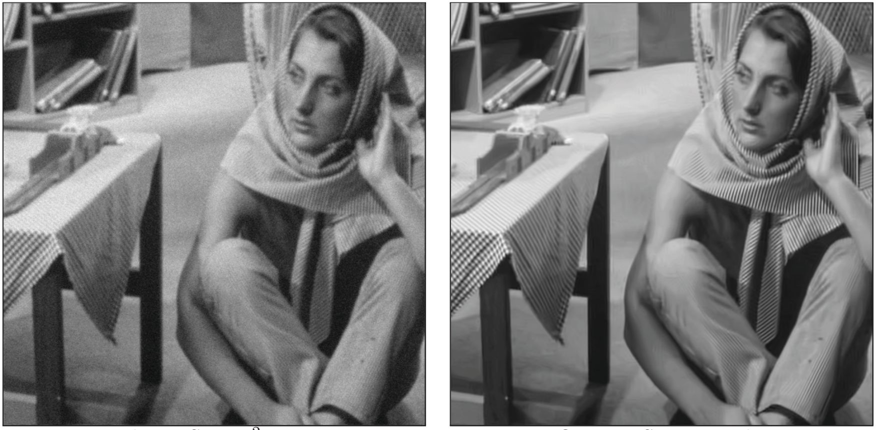
Blur: PSF4, $\sigma^2 = 4$ Output ISNR 4.37 dB

Blur: PSF5, $\sigma^2 = 64$ Output ISNR 6.40 dB

Figure 5. Deblurring results of the proposed method for *Lena*.

Blur: PSF1, $\sigma^2 = 8$ Output ISNR 3.94 dB

Blur: PSF2, $\sigma^2 = 0.308$ Output ISNR 5.86 dB



Blur: PSF3, $\sigma^2 = 49$ Output ISNR 1.90 dB

Figure 6. Deblurring results of the proposed method for *Barbara*.

Blur →	PSF 1		PSF 2	PSF 3	PSF 4	PSF 5
σ^2 →	2	8	0.308	49	4	64
$\hat{y}^{\mathrm{RI}}$	7.13	5.16	7.52	2.31	3.23	2.46
$\hat{y}^{\mathrm{RWI}}$	**8.19**	**6.40**	**8.34**	**3.34**	**3.73**	**4.70**
$\hat{y}^{\mathrm{RWI}}_{\mathrm{naive}}$	7.17	6.25	8.14	2.57	2.71	4.63

Table 2. ISNR comparison for: the basic estimate $\hat{y}^{\mathrm{RI}}$; the final estimate $\hat{y}^{\mathrm{RWI}}$; the final estimate $\hat{y}^{\mathrm{RWI}}_{\mathrm{naive}}$ obtained using the original BM3D filter instead of the one extended for colored noise. The test image was *Cameraman*.

REFERENCES

1. C. Vogel, *Computational Methods for Inverse Problems*, SIAM, 2002.
2. P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*, SIAM, Philadelphia, 1997.
3. A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-Adaptive DCT for denoising and image reconstruction," in *Proc. SPIE Electronic Imaging: Algorithms and Systems V*, **6064A-18**, (San Jose, CA, USA), January 2006.
4. J. A. Guerrero-Colon, L. Mancera, and J. Portilla, "Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids," *IEEE Trans. Image Process.* **17**, pp. 27–41, January 2007.
5. R. Neelamani, H. Choi, and R. G. Baraniuk, "Wavelet-domain regularized deconvolution for ill-conditioned systems," in *Proc. IEEE Int. Conf. Image Process.*, **1**, pp. 204–208, (Kobe, Japan), October 1999.
6. R. Neelamani, H. Choi, and R. G. Baraniuk, "Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Trans. Signal Process.* **52**, pp. 418–433, February 2004.
7. V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "Directional varying scale approximations for anisotropic signal processing," in *Proc. European Signal Process. Conf.*, pp. 101–104, (Vienna, Austria), September 2004.
8. V. Katkovnik, K. Egiazarian, and J. Astola, "A spatially adaptive nonparametric regression image deblurring," *IEEE Trans. Image Process.* **14**, pp. 1469–1478, October 2005.
9. V. Katkovnik, K. Egiazarian, and J. Astola, *Local Approximation Techniques in Signal and Image Process.*, vol. PM157, SPIE Press, 2006.
10. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.* **16**, pp. 2080–2095, August 2007.
11. A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling and Simulation* **4**(2), pp. 490–530, 2005.
12. C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Trans. Image Process.* **15**, pp. 2866–2878, October 2006.
13. K. Dabov and A. Foi, "BM3D Filter Matlab Demo Codes." http://www.cs.tut.fi/~foi/GCF-BM3D.
14. J. Dias, "Fast GEM wavelet-based image deconvolution algorithm," in *Proc. IEEE Int. Conf. Image Process.*, **3**, pp. 961–964, (Barcelona, Spain), September 2003.
15. M. Mignotte, "An adaptive segmentation-based regularization term for image restoration," in *Proc. IEEE Int. Conf. Image Process.*, **1**, pp. 901–904, (Genova, Italy), September 2005.
16. M. A. T. Figueiredo and R. D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Process.* **12**, pp. 906–916, August 2003.
17. R. Neelamani, "Fourier-Wavelet Regularized Deconvolution (ForWaRD) Software for 1-D Signals and Images." http://www.dsp.rice.edu/software/ward.shtml.
18. A. Foi and K. Dabov, "Pointwise Shape-Adaptive DCT Demobox." http://www.cs.tut.fi/~foi/SA-DCT.
19. J. Portilla and E. P. Simoncelli, "Image restoration using Gaussian scale mixtures in the wavelet domain," in *Proc. IEEE Int. Conf. Image Process.*, **2**, pp. 965–968, (Barcelona, Spain), September 2003.
20. M. A. T. Figueiredo and R. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *Proc. IEEE Int. Conf. Image Process.*, **2**, pp. 782–785, (Genova, Italy), September 2005.
21. A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise Shape-Adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.* **16**, pp. 1395–1411, May 2007.

Publication [P7]

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “A nonlocal and shape-adaptive transform-domain collaborative filtering”, *Proc. Int. Workshop on Local and Non-Local Approx. in Image Process.*, Lausanne, Switzerland, August 2008.

A NONLOCAL AND SHAPE-ADAPTIVE TRANSFORM-DOMAIN COLLABORATIVE FILTERING

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian

Department of Signal Processing, Tampere University of Technology
P.O. Box 553, 33101, Tampere, Finland
email: firstname.lastname@tut.fi

ABSTRACT

We propose an image denoising method that exploits both nonlocal image modeling and locally adaptive anisotropic estimation. The method uses grouping of adaptive-shape neighborhoods whose surrounding square supersets have been found similar by a block-matching procedure. The data defined on these grouped neighborhoods is stacked together, resulting in 3-D data structures which are generalized cylinders with adaptive-shape cross sections. Because of the similarity, which follows from the matching, and because of the adaptive selection of the shape of the neighborhoods, these 3-D groups are characterized by a high correlation along all the three dimensions. We apply a 3-D decorrelating transform, computed as a separable composition of the Shape-Adaptive DCT (SA-DCT) and a 1-D orthonormal transform, and subsequently attenuate the noise by spectrum shrinkage with hard-thresholding or Wiener filtering. Inversion of the 3-D transform produces individual estimates for all grouped neighborhoods. These estimates are returned to their original locations and aggregated with other estimates coming from different groups.

Overall, this method generalizes two existing filters: the BM3D filter, which uses grouping of fixed-size square blocks, and the Poinwise SA-DCT filter, which exploits shrinkage on adaptive-shape supports. We show that the developed method inherits the strengths of both filters, resulting in a very effective and flexible tool for image denoising.

1. INTRODUCTION

Image denoising is a fundamental problem in image processing and a lot of research has been dedicated it. Some of the recent and most successful advances in the field are the methods [15, 7, 3, 5, 1, 13, 2, 14]. Among these, are two methods by the present authors, the BM3D filter [2] and the Pointwise SA-DCT filter (P.SA-DCT) [5], which are used as a basis for this work.

The BM3D filter exploits a specific nonlocal image modeling [9] through a procedure termed *grouping and collaborative filtering*. Grouping finds mutually similar 2-D image blocks and stacks them together in 3-D arrays. Collaborative filtering produces individual estimates of all grouped blocks by filtering them jointly, through transform-domain shrinkage of the 3-D arrays (groups). In doing so, BM3D relies on two strong characteristics of natural images. First, the abundance of mutually similar patches and, second, that the content of small blocks is locally highly correlated. On these assumptions, the group enjoys correlation in all three dimensions and a sparse representation of the true signal is obtained by applying a decorrelating 3-D transform on the group. The subsequent shrinkage achieves effective noise attenuation thanks to sparsity. We have shown [2] that the BM3D filter is a very efficient and powerful denoiser. Its results are still beyond the capabilities of most of the more recent and advanced algorithms. Of course, BM3D is particularly successful when plenty of matching blocks can be found (e.g., textures, regular shaped image structures, or uniform areas), i.e. where the nonlocal modeling is suitable. On the other hand, the assumption that image content is highly correlated on a square block of fixed size is sometimes very artificial. Blocks containing small image details, singularities, or sharp and curved edges are examples where a non-adaptive transform is not able to deliver a sparse representation. Thus, for these blocks, the BM3D filter may introduce certain artifacts and the denoising is not very effective. Unfortunately, these are often the very parts of the image where the visual attention is mainly focused.

The Pointwise SA-DCT filter [5] works differently. It exploits *pointwise-adaptive arbitrarily shaped neighborhoods*, adaptive to image content in such a way that the signal contained in a neighborhood is highly homogeneous. The subsequent application of the (2-D) shape-adaptive discrete cosine transform (SA-DCT) [12] on such a neighborhood results in a sparse representation of the true signal, which enables effective noise attenuation by transform shrinkage. Due to the adaptivity of neighborhoods to local image details, the P.SA-DCT filter achieves exceptional preservation of edges and singularities, as shown [5, 4]. However, a drawback of the nonparametric local-homogeneous image model, assumed by the Pointwise SA-DCT filter, becomes evident for texture-rich regions. There, local homogeneity is very limited and, thus, the performance of the filter is not satisfactory [4]. Moreover, the P.SA-DCT is a *local* filter, and as such it cannot take advantage of the abundance of repeated structures and patterns found in natural images.

From the mentioned properties of the BM3D and the P.SA-DCT filters, one can infer that their strengths and weaknesses are complementing each other. This is the motivation that leads us to propose in this work an image denoising method that simultaneously generalizes these two filters. The new method performs *grouping* of mutually similar *adaptive-shape neighborhoods*. Thus, the 3-D groups become generalized cylinders with adaptive-shape cross sections (as opposed to the groups in BM3D, which are square prisms with fixed-size square cross sections). In this manner, both the nonlocal image modeling and the local adaptivity to image features are exploited. In particular, the spatial correlation within the 2-D neighborhoods is increased as compared with the BM3D. After the grouping, we apply a 3-D transform that is a separable composition of the SA-DCT and a 1-D orthonormal transform. Noise is then attenuated by spectrum shrinkage with either hard-thresholding or Wiener filtering. Similar to the BM3D, inversion of the 3-D transform produces individual estimates for all grouped neighborhoods. These estimates are returned to their original locations and aggregated with estimates coming from other groups.

We show that this developed method, denominated SA-BM3D, inherits the strengths of both the P.SA-DCT and the BM3D filters and at the same time overcomes their weaknesses.

2. PRELIMINARIES

We consider noisy observations z of the form

$$z(x) = y(x) + \eta(x), \quad x \in X, \tag{1}$$

where $y : X \to \mathbb{R}$ is the original grayscale image, η is i.i.d. Gaussian white noise, $\eta(\cdot) \sim \mathcal{N}(0, \sigma^2)$, and x is a spatial variable belonging

---

This work was supported by the Academy of Finland (application no. 213462, Finnish Programme for Centres of Excellence in Research 2006-2011, and application no. 118312, Finland Distinguished Professor Programme 2007-2010); the first author was supported by TISE graduate school.

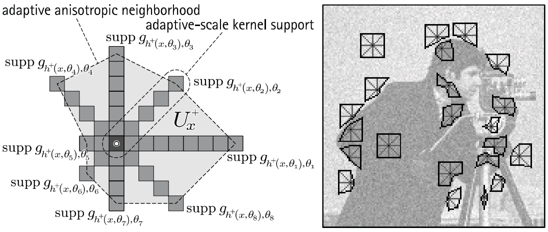


Figure 1: Fast implementation of the LPA-ICI anisotropic neighborhoods. “Linewise” one-dimensional directional LPA kernels are used for 8 directions. The anisotropic neighborhood U_x^+ is constructed as the polygonal hull of the adaptive-scale kernels' supports (left). Thus, only the adaptive scales h^+ are needed to construct the neighborhood. Some examples of the anisotropic neighborhoods $\tilde{U}_x^+$ used for SA-DCT filtering of the noisy *Cameraman* image (right).

to the image domain $X \subset \mathbb{Z}^2$.

Given a function $f : X \to \mathbb{R}$, a subset $U \subset X$, and a function $g : U \to \mathbb{R}$, we denote by $f_{|U} : U \to \mathbb{R}$ the restriction of f on U, $f_{|U}(x) = f(x)\ \forall x \in U$, and by $g^{|X} : X \to \mathbb{R}$ the zero-extension of g to X, $\left(g^{|X}\right)_{|U} = g$ and $g^{|X}(x) = 0\ \forall x \in X \setminus U$. The characteristic (indicator) function of U is defined as $\chi_U = 1_{|U}{}^{|X}$. We denote by $|U|$ the cardinality (i.e. the number of elements) of U. The symbol “$\circledast$” stands for the convolution operation.

2.1 Adaptive anisotropic neighborhoods

As in [5], we use the local polynomial approximation-intersection of confidence intervals (LPA-ICI) technique [11] in order to associate, to each $x \in X$, and adaptive-shape neighborhood in which the image is homogenous.

2.1.1 Directional pointwise adaptive scales by LPA-ICI

For each of the eight directions $\theta_k = \frac{(k-1)}{4}\pi$, $k = 1, \ldots, 8$, a varying-scale family of narrow “linewise” directional-LPA [10] convolution kernels $\{g_{h,\theta_k}\}_{h \in H}$ is used to obtain a corresponding set of directional varying-scale estimates $\{\hat{y}_{h,\theta_k}\}_{h \in H}$, $\hat{y}_{h,\theta_k} = z \circledast g_{h,\theta_k}$, $h \in H$, where $H \subset \mathbb{R}^+$ is the set of scales. These estimates are then compared according to the ICI rule [6, 8, 11], and as a result an adaptive scale $h^+(x, \theta_k) \in H$ is defined for every $x \in X$ and for every direction θ_k.

2.1.2 Adaptive-shape neighborhood

The anisotropic neighborhood U_x^+ is the octagon constructed as the polygonal hull of $\left\{\operatorname{supp} g_{h^+(x,\theta_k),\theta_k}\right\}_{k=1}^{8}$. Such neighborhoods are shown in Figure 1. We note that, in our particular implementation, the value of the adaptive-scale $h^+(x, \theta_k)$ coincides with the length (measured in pixels) of the directional window in the direction θ_k (i.e. with the length of the support of the corresponding directional kernel). Thus, in order to construct any neighborhood U_x^+ it suffices to know only the adaptive scales $\left\{h^+(x, \theta_k)\right\}_{k=1}^{8}$ for all $x \in X$.

Let us remind that being convolution kernels, the LPA kernels g_{h,θ_k} are always “centered” at the origin, therefore U_x^+ is always a neighborhood of the origin. The actual adaptive neighborhood of x, which contains the observations that are used for estimation, is instead

$$\tilde{U}_x^+ = \left\{v \in X : (x - v) \in U_x^+\right\},$$

in other words $\tilde{U}_x^+$ (with tilde) is obtained by translation and mirroring of U_x^+ (without tilde). In both symbols, the subscript “x” denotes the point for which the adaptive scales are obtained while the “+” is used to distinguish the adaptive neighborhoods from the non-adaptive ones. We remark that neighborhoods $\tilde{U}_{x'}^+, \tilde{U}_{x''}^+$ corresponding to adjacent or nearby points x', x'' do usually overlap unless an edge or sharp transition exists between the two points.

Additionally, in this work, we also use the following more general mirrored translates of U_x^+:

$$\begin{aligned} \tilde{U}_{x,x_R}^+ &= \left\{v \in X : (x - v) \in U_{x_R}^+\right\} = \\ &= \left\{v \in X : (x_R - x + v) \in \tilde{U}_{x_R}^+\right\}. \end{aligned}$$

$\tilde{U}_{x,x_R}^+$ is an adaptive neighborhood of x which differs from $\tilde{U}_x^+$ in that $\tilde{U}_{x,x_R}^+$ uses the adaptive scales corresponding to the point x_R and not those corresponding to x itself. Obviously, $\tilde{U}_{x,x}^+ = \tilde{U}_x^+$.

2.2 Shape-Adaptive DCT transform

As in [5], we consider the orthonormal SA-DCT with DC-separation [12]. It means that before applying the orthonormal SA-DCT, we first subtract the data of its mean, which constitutes an extra coefficient that will be processed independently from the actual SA-DCT coefficients. Once the DC is separated, the orthonormal SA-DCT is applied on zero-mean data. The SA-DCT is computed by cascaded application of one-dimensional varying-length orthonormal DCT transforms first on the columns and then on the rows (or vice versa[1]) that constitute the considered region. This process is illustrated in Figure 2. We denote by T_U^{SA} the orthonormal SA-DCT transform corresponding to a region $U \subset X$ and by $V_U \subset \mathbb{Z}^2$ the domain of the T_U^{SA} transform coefficients. Let $T_U^{\mathrm{SA}-1}$ be the inverse transform of T_U^{SA}. The mean of z on U is denoted as $m_U(z) = \frac{1}{|U|}\sum_{x \in U} z(x)$.

We refer the interested reader to [5] for more subtle details about the selection between column/row or row/column processing and the particular coefficient alignment strategies used within the SA-DCT.

3. SA-BM3D ALGORITHM

Similar to the BM3D, the proposed SA-BM3D algorithm (illustrated in Figure 3) exploits a two-step approach where

- in the first step, block-matching is performed on the noisy image and the noise is attenuated by collaborative hard-thresholding,
- in the second step, block-matching is performed on the initial (basic) estimate obtained in the first step, and the noise is attenuated by collaborative empirical Wiener filtering.

Each of the two steps is presented in the following subsections.

3.1 SA-BM3D with hard thresholding (Step 1)

Once the LPA-ICI adaptive scales $\left\{h^+(x, \theta_k)\right\}_{k=1}^{8}$, and thus the adaptive-shape neighborhoods U_x^+, have been found (Section 2.1), for each $x \in X$, the following operations are performed:

- shape-adaptive grouping;
- collaborative hard-thresholding;
- aggregation.

These operations are explained in detail in the following subsections; therein we fix the currently processed coordinate as $x_R \in X$ and denominate it *reference point.*

3.1.1 Shape-adaptive grouping via block-matching

The adaptive neighborhoods $\tilde{U}_x^+$ can be too small for reliable patch-matching, especially when filtering tiny image details in heavy noise. Therefore, the matching for $\tilde{U}_x^+$ needs to be carried out for

---

[1] Note that even though the SA-DCT is implemented like a separable 2-D transform (using cascaded 1-D transforms on columns and rows), in general it is not separable and different sets of transform coefficients can be obtained when processing the considered region either first column-wise and then row-wise or first row-wise and then column-wise.

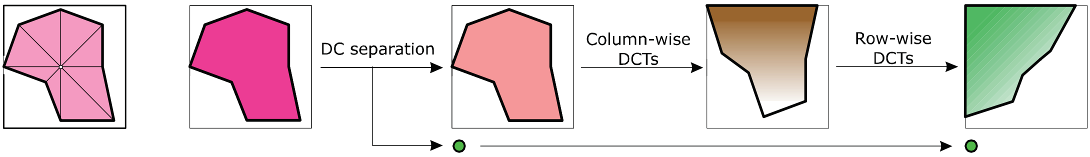


Figure 2: Illustration of the forward SA-DCT with DC-separation.

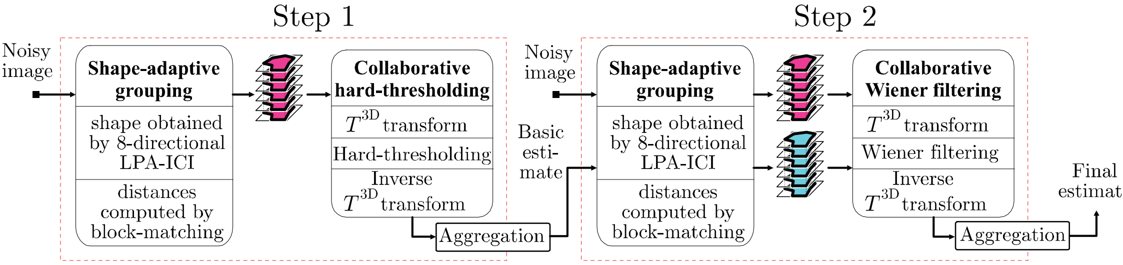


Figure 3: Flowchart of the proposed SA-BM3D image denoising method. Operations surrounded by dashed lines are repeated for each processed coordinate of the input image.

a superset. In particular, we use square blocks as supersets and perform a block-matching procedure like the one in the BM3D algorithm [2]. In what follows, we define a mapping that associates to every reference point $x_R \in X$ a block that can be used as the reference one in the matching needed for the grouping of $\tilde{U}^+_{x_R}$.

Mapping. Let $x \in X$ and denote by $\tilde{B}_x \subset \mathbb{Z}^2$ be the square block of size $(2h_{\max} - 1) \times (2h_{\max} - 1)$ centered at x, where $h_{\max} = \max\{H\}$. Let $\mathbb{B}$ be the collection of all such blocks which are entirely contained in X, $\mathbb{B} = \{\tilde{B}_x : x \in X,\ \tilde{B}_x \subset X\}$. Note that for $h_{\max} > 1$ the cardinality of $\mathbb{B}$ is strictly smaller than $|X|$ because if x is close enough to the boundary ∂X of X, then $\tilde{B}_x$ would cross ∂X and hence $\tilde{B}_x \not\subseteq X$ and $\tilde{B}_x \notin \mathbb{B}$. We indicate by $X_{\mathbb{B}} \subset X$ the set of points for which we can construct a block belonging to $\mathbb{B}$, $X_{\mathbb{B}} = \{x \in X : \tilde{B}_x \in \mathbb{B}\}$. To every $x \in X$ we can associate a point $x_{\mathbb{B}} \in X_{\mathbb{B}}$ such that the magnitude $\|\delta_{\mathbb{B}}(x)\|_2$ of $\delta_{\mathbb{B}}(x) = x_{\mathbb{B}} - x$ is minimal. For a rectangular X, because of convexity of $X_{\mathbb{B}}$, the surjective mapping $x \mapsto x_{\mathbb{B}}$ is univocally defined and so is $\delta_{\mathbb{B}}(x)$. Note that $\delta_{\mathbb{B}}(x) \neq 0$ only for x sufficiently close to the boundary ∂X of X.

Block-matching. For each point $x \in X_{\mathbb{B}}$, we produce grouping by block-matching within the image z. That is, for each block $\tilde{B}_x$ we look for "similar" blocks $\tilde{B}_{x'}$ whose range distance $d_z(x, x')$ with respect to $\tilde{B}_x$,

$$d_z(x, x') = \left\| z_{|\tilde{B}_x} - z_{|\tilde{B}_{x'}} \right\|_2,$$

is smaller than a fixed threshold $\tau^{\text{ht}}_{\text{match}} \geq 0$. Thus, we construct the set S_x so that it contains the central points of the found blocks:

$$S_x = \left\{x' \in X_{\mathbb{B}} : d_z(x, x') \leq \tau^{\text{ht}}_{\text{match}}\right\}. \tag{2}$$

The threshold $\tau^{\text{ht}}_{\text{match}}$ is the maximum d_z-distance for which two blocks are considered similar[2]. Obviously $d_z(x, x) = 0$, which implies that the cardinality $|S_x| \geq 1$ for any $x \in X_{\mathbb{B}}$.

Shape-adaptive grouping. Let now $x_R \in X$ be a reference point and define $x^{\delta}_R = x_R + \delta_{\mathbb{B}}(x_R)$. Using the result $S_{x^{\delta}_R}$ from the block-matching, we associate to the reference point x_R not only its own adaptive neighborhood $\tilde{U}^+_{x_R}$, but a whole collection (disjoint union) $\widetilde{\mathbb{U}}_{x_R}$ of neighborhoods having the same shape and defined as[3]

$$\widetilde{\mathbb{U}}_{x_R} = \coprod_{x + \delta_{\mathbb{B}}(x_R) \in S_{x^{\delta}_R}} \tilde{U}^+_{x, x_R}. \tag{3}$$

Let us observe that all neighborhoods in $\widetilde{\mathbb{U}}_{x_R}$ have the same shape, which is completely determined by the adaptive scales $\{h^+(x_R, \theta_k)\}_{k=1}^{8}$ at x_R. This also implies that for all $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$ the domain of the corresponding SA-DCT coefficients $V_{\tilde{U}^+_{x,x_R}}$ is one and the same and coincides with $V_{\tilde{U}^+_{x_R}}$.

At the current reference point x_R, a group is built by stacking together the noisy patches $z_{|\tilde{U}^+_{x,x_R}}$, $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$. This group is a 3-D data array defined on the generalized cylinder $\tilde{U}^+_{x_R} \times \left\{1, \ldots, \left|S_{x^{\delta}_R}\right|\right\}$, as illustrated in Figure 4. In compact form, the group is denoted as $\mathbf{Z}_{x_R} : \widetilde{\mathbb{U}}_{x_R} \to \mathbb{R}$.

3.1.2 Collaborative hard-thresholding

Given a group $\mathbf{Z}_{x_R}$, collaborative filtering is realized as shrinkage in a 3-D transform domain. Here, the 3-D transform $T^{3\text{D}}$ is a composition of the (2-D) SA-DCT with DC-separation on each neighborhood of $\widetilde{\mathbb{U}}_{x_R}$ with an orthonormal 1-D transform $T^{1\text{D}}$ applied along the third dimension of the group. As in the BM3D algorithm [2], we require $T^{1\text{D}}$ to have a DC term. The three steps of the collaborative hard-thresholding are as follows.

1. Forward $T^{3\text{D}}$ transform (illustrated in Figure 4).

- For each $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$:
  - compute mean value $m_{\tilde{U}^+_{x,x_R}}(z) = \frac{1}{\left|\tilde{U}^+_{x,x_R}\right|} \sum_{v \in \tilde{U}^+_{x,x_R}} z(v)$;

---

[2] The influence of noise on d_z is studied in [2]. In case of heavy noise, to reduce its influence on d_z, we can exploit a coarse prefiltering embedded within d_z, as described in [2].

[3] The set (3) must not be interpreted as the mere union of the neighborhoods $\tilde{U}^+_{x,x_R}$ such that $x + \delta_{\mathbb{B}}(x_R) \in S_{x^{\delta}_R}$. While, for simplicity, we may write $\widetilde{\mathbb{U}}_{x_R} = \{\tilde{U}^+_{x,x_R} : x + \delta_{\mathbb{B}}(x_R) \in S_{x^{\delta}_R}\}$, a more proper notation is actually

$$\widetilde{\mathbb{U}}_{x_R} = \left\{\left(\tilde{U}^+_{x,x_R}, x\right) : x + \delta_{\mathbb{B}}(x_R) \in S_{x^{\delta}_R}\right\} \subset X \times X,$$

because we need to distinguish between different neighborhoods coming from different elements of $S_{x^{\delta}_R}$.

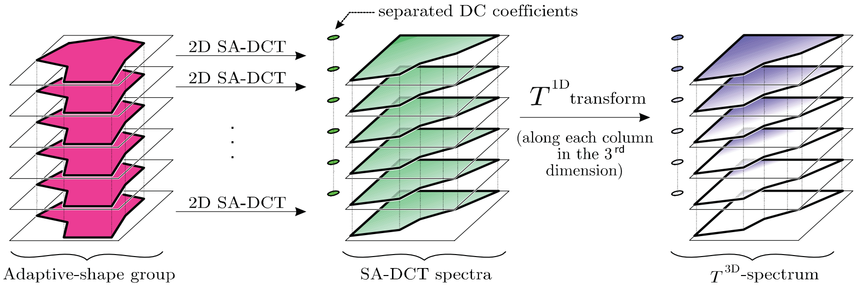


Figure 4: Illustration of applying $T^{3\mathrm{D}}$ on the adaptive-shape group $\mathbf{Z}_{x_R}$.

  - compute the SA-DCT spectrum $\varphi_{z,x,x_R} : V_{\tilde{U}^+_{x_R}} \to \mathbb{R}$ as
$$T^{\mathsf{SA}}_{\tilde{U}^+_{x,x_R}}\left(z_{|\tilde{U}^+_{x,x_R}} - m_{\tilde{U}^+_{x,x_R}}(z)\right);$$
  - define DC coefficient $\varphi^{\mathrm{DC}}_{z,x,x_R} = \left|\tilde{U}^+_{x_R}\right|^{1/2} m_{\tilde{U}^+_{x,x_R}}(z)$.

  At this stage, for each $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$, we have obtained one DC coefficient $\varphi^{\mathrm{DC}}_{z,x,x_R}$ and $\left|\tilde{U}^+_{x_R}\right|$ SA-DCT coefficients φ_{z,x,x_R}. Thus, we have a total of $\left(1+\left|\tilde{U}^+_{x_R}\right|\right)\left|S_{x_R^\delta}\right|$ coefficients, structured as a 3-D array $\left(\{\mu\} \amalg V_{\tilde{U}^+_{x_R}}\right) \times \left\{1,\ldots,\left|S_{x_R^\delta}\right|\right\}$, where $\{\mu\}$ is a singleton placeholder for separated DC coefficients $\varphi^{\mathrm{DC}}_{z,x,x_R}$.

- Apply $T^{1\mathrm{D}}$ along the vector $\varphi^{\mathrm{DC}}_{z,x,x_R} : \{\mu\} \times \left\{1,\ldots,\left|S_{x_R^\delta}\right|\right\} \to \mathbb{R}$ and along the vectors $\varphi_{z,x,x_R} : \{\nu\} \times \left\{1,\ldots,\left|S_{x_R^\delta}\right|\right\}$, $\nu \in V_{\tilde{U}^+_{x_R}}$.

  This second stage provides us with the full 3-D spectrum $T^{3\mathrm{D}}\left(\mathbf{Z}_{x_R}\right)$ of the group $\mathbf{Z}_{x_R}$. We emphasize again that there are $\left(1+\left|\tilde{U}^+_{x_R}\right|\right)\left|S_{x_R^\delta}\right|$ spectral coefficients.

2. Shrinkage by hard-thresholding

   As in [2], we perform hard-thresholding of the spectrum $T^{3\mathrm{D}}\left(\mathbf{Z}_{\widetilde{\mathbb{U}}_{x_R}}\right)$ using the threshold $\gamma_{\mathrm{thr}}\sigma$, where $\gamma_{\mathrm{thr}} > 0$ is a fixed constant. We threshold all coefficients except the DC coefficient in the $T^{3\mathrm{D}}$-spectrum (i.e. the DC coefficient of the $T^{1\mathrm{D}}$-spectrum of the vector $\varphi^{\mathrm{DC}}_{z,x,x_R}$), which systematically is always preserved. After thresholding, we have $N^{\mathrm{har}}_{x_R} \geq 1$ non-thresholded coefficients ("number of harmonics").

3. Inverse $T^{3\mathrm{D}}$ transform

   Inversion of the $T^{3\mathrm{D}}$ transform is computed as follows.

- Apply the inverse of $T^{1\mathrm{D}}$ and obtain filtered vectors $\hat{\varphi}^{\mathrm{DC}}_{y,x,x_R}$ and $\hat{\varphi}_{y,x,x_R}$ which are estimates of the DCs and SA-DCT spectra of the hypothetical group $\mathbf{Y}_{x_R}$ formed by stacking together patches $y_{|\tilde{U}^+_{x,x_R}}$, $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$, from the unknown noise-free image y.
- For each $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$:
  - define the estimate of the mean $m_{\tilde{U}^+_{x,x_R}}(y)$ as
$$\widehat{m_{\tilde{U}^+_{x,x_R}}}(y) = \left|\tilde{U}^+_{x_R}\right|^{-1/2} \hat{\varphi}^{\mathrm{DC}}_{y,x,x_R};$$
  - compute a local estimate $\hat{y}_{x,x_R} : \tilde{U}^+_{x,x_R} \to \mathbb{R}$ of $y_{|\tilde{U}^+_{x,x_R}}$ by inverse SA-DCT of $\hat{\varphi}_{y,x,x_R}$ followed by addition of the mean, $\hat{y}_{x,x_R} = T^{\mathsf{SA}\,-1}_{\tilde{U}^+_{x,x_R}}\left(\hat{\varphi}_{y,x,x_R}\right) + \widehat{m_{\tilde{U}^+_{x,x_R}}}(y)$.

In this way, we obtain local estimates for each of the neighborhoods in $\widetilde{\mathbb{U}}_{x_R}$. All these estimates can be denoted in compact group form as $\widehat{\mathbf{Y}}_{x_R} : \widetilde{\mathbb{U}}_{x_R} \to \mathbb{R}$.

3.1.3 *Aggregation*

After performing grouping and collaborative filtering for all $x_R \in X$, we have obtained for each x_R a group $\widehat{\mathbf{Y}}_{x_R}$ of $\left|S_{x_R^\delta}\right|$ of distinct *local* estimates (i.e. filtered data on the adaptive-shape neighborhoods) of y. Overall, we end up with $\sum_{x_R \in X}\left|S_{x_R^\delta}\right|$ local estimates. We note that each local estimate is supported in one of the $|X|$ adaptive neighborhoods $\tilde{U}^+_x$, for $x \in X$. However, provided $x_R \neq x'_R$, even when two neighborhoods $\tilde{U}^+_{x,x_R} = \tilde{U}^+_{x,x'_R}$ coincide, the respective estimates $\hat{y}_{x,x_R}$ and $\hat{y}_{x,x'_R}$ can be different, as they are obtained from filtering possibly different groups $\mathbf{Z}_{x_R}$ and $\mathbf{Z}_{x'_R}$. Thus, the collection of local estimates $\coprod_{x_R \in X}\widehat{\mathbf{Y}}_{x_R} = \coprod_{x_R \in X,\ x+\delta_{\mathbb{B}}(x_R) \in S_{x_R^\delta}} \hat{y}_{x,x_R}$ is a highly redundant and rich representation of the original image y.

In order to obtain a single *global* estimate $\hat{y}^{\mathrm{ht}} : X \to \mathbb{R}$ defined on the whole image domain, all these local estimates are averaged together using adaptive weights $w_{x_R} > 0$ in the following convex combination:

$$\hat{y}^{\mathrm{ht}} = \frac{\sum_{x_R \in X}\sum_{x+\delta_{\mathbb{B}}(x_R) \in S_{x_R^\delta}} w_{x_R}\hat{y}_{x,x_R}{}^{|X}}{\sum_{x_R \in X}\sum_{x+\delta_{\mathbb{B}}(x_R) \in S_{x_R^\delta}} w_{x_R}\chi_{\tilde{U}^+_{x,x_R}}}, \tag{4}$$

where the weights w_{x_R} are defined by

$$w_{x_R} = \frac{\sigma^{-2}}{N^{\mathrm{har}}_{x_R}\,|\tilde{U}^+_{x_R}|}. \tag{5}$$

3.2 SA-BM3D with Wiener filtering (Step 2)

As in [2, 5], the denoising performance is significantly improved by applying a second step with empirical Wiener filtering. The initial estimate $\hat{y}^{\mathrm{ht}}$ obtained in the first step is used for two purposes; first, the shape-adaptive grouping exploits block-matching performed on $\hat{y}^{\mathrm{ht}}$ rather than on the noisy image and, second, the magnitude of the spectrum of $\hat{y}^{\mathrm{ht}}$ is used to perform empirical Wiener filtering (rather than hard-thresholding as in the first step). For a reference point $x_R \in X$, the shape-adaptive grouping and the collaborative Wiener filtering are given in the following two subsections.

3.2.1 *Shape-adaptive grouping*

Because the noise in $\hat{y}^{\mathrm{ht}}$ is assumed to be significantly attenuated, we can obtain more accurate block-matching by replacing the distance d_z by a distance $d_{\hat{y}^{\mathrm{ht}}}$ where the similarity between blocks is evaluated on $\hat{y}^{\mathrm{ht}}$ instead of z:

$$d_{\hat{y}^{\mathrm{ht}}}\left(x, x'\right) = \left\|\hat{y}^{\mathrm{ht}}{}_{|\tilde{B}_x} - \hat{y}^{\mathrm{ht}}{}_{|\tilde{B}_{x'}}\right\|_2.$$

Thus, we redefine the sets S_x (which contain the coordinates of the matched blocks) as

$$S_x = \left\{x' \in X_{\mathbb{B}} : d_{\hat{y}^{\mathrm{ht}}}\left(x, x'\right) \leq \tau^{\mathrm{wie}}_{\mathrm{match}}\right\}, \tag{6}$$

where $\tau^{\mathrm{wie}}_{\mathrm{match}} > 0$ is another threshold for the range distance between blocks.

Using (6), we construct the collection of neighborhoods $\widetilde{\mathbb{U}}_{x_R}$ as in (3). Subsequently, $\widetilde{\mathbb{U}}_{x_R}$ is used to build two groups, first, the group $\mathbf{Z}_{x_R} : \widetilde{\mathbb{U}}_{x_R} \to \mathbb{R}$ by stacking together the noisy patches $z_{|\tilde{U}^+_{x,x_R}}$, $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$, and second, $\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R} : \widetilde{\mathbb{U}}_{x_R} \to \mathbb{R}$ by stacking together the initial estimate patches $\hat{y}^{\mathrm{ht}}_{|\tilde{U}^+_{x,x_R}}$, $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$. Both groups are defined on the same domain $\widetilde{\mathbb{U}}_{x_R}$.

3.2.2 Collaborative Wiener filtering

The shrinkage of the 3-D spectrum of the group $\mathbf{Z}_{x_R}$ is performed by empirical Wiener filtering which uses the magnitude of the T^{3D} spectrum of the group $\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}$. Because of the DC-separation, some care is required when these 3-D spectra are defined.

In particular, while the 3-D spectrum of $\mathbf{Z}_{x_R}$ is defined as in Section 3.1.2, the spectrum of $\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}$ is computed by subtracting from $\hat{y}^{\mathrm{ht}}_{|\tilde{U}^+_{x,x_R}}$ the mean $m_{\tilde{U}^+_{x,x_R}}(z)$, $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$. Additionally, the shrinkage coefficients for the DC-vector are computed using directly the corresponding DC-vector with the means of $\hat{y}^{\mathrm{ht}}_{|\tilde{U}^+_{x,x_R}}$. Let us describe precisely these operations.

1. Forward T^{3D} transform

- For each $\tilde{U}^+_{x,x_R} \in \widetilde{\mathbb{U}}_{x_R}$:
  - compute mean value $m_{\tilde{U}^+_{x,x_R}}(z) = \frac{1}{\left|\tilde{U}^+_{x,x_R}\right|}\sum_{v\in\tilde{U}^+_{x,x_R}} z\left(v\right)$;
  - compute mean value $m_{\tilde{U}^+_{x,x_R}}(\hat{y}^{\mathrm{ht}}) = \frac{1}{\left|\tilde{U}^+_{x,x_R}\right|}\sum_{v\in\tilde{U}^+_{x,x_R}} \hat{y}^{\mathrm{ht}}(v)$;
  - compute the SA-DCT spectrum $\varphi_{z,x,x_R} : V_{\tilde{U}^+_{x_R}} \to \mathbb{R}$ as
  $$T^{\mathrm{SA}}_{\tilde{U}^+_{x,x_R}}\left(z_{|\tilde{U}^+_{x,x_R}} - m_{\tilde{U}^+_{x,x_R}}(z)\right);$$
  - compute the SA-DCT spectrum $\varphi_{\hat{y}^{\mathrm{ht}},x,x_R} : V_{\tilde{U}^+_{x_R}} \to \mathbb{R}$ as
  $$T^{\mathrm{SA}}_{\tilde{U}^+_{x,x_R}}\left(\hat{y}^{\mathrm{ht}}_{|\tilde{U}^+_{x,x_R}} - m_{\tilde{U}^+_{x,x_R}}(z)\right);$$

  (note that we subtract $m_{\tilde{U}^+_{x,x_R}}(z)$ from both $z_{|\tilde{U}^+_{x,x_R}}$ and $\hat{y}^{\mathrm{ht}}_{|\tilde{U}^+_{x,x_R}}$)
  - define DC coefficient $\varphi^{\mathrm{DC}}_{z,x,x_R} = \left|\tilde{U}^+_{x_R}\right|^{1/2} m_{\tilde{U}^+_{x,x_R}}(z)$.
  - define DC coefficient $\varphi^{\mathrm{DC}}_{\hat{y}^{\mathrm{ht}},x,x_R} = \left|\tilde{U}^+_{x_R}\right|^{1/2} m_{\tilde{U}^+_{x,x_R}}\left(\hat{y}^{\mathrm{ht}}\right)$.
- Like in Section 3.1.2, apply T^{1D} along the vectors $\varphi^{\mathrm{DC}}_{z,x,x_R}$, $\varphi^{\mathrm{DC}}_{\hat{y}^{\mathrm{ht}},x,x_R}$, φ_{z,x,x_R} and $\varphi_{\hat{y}^{\mathrm{ht}},x,x_R}$.

This second stage provides us with the full 3-D spectrum $T^{\mathrm{3D}}\left(\mathbf{Z}_{x_R}\right)$ of $\mathbf{Z}_{x_R}$ and with a special reference 3-D spectrum $\breve{T}^{\mathrm{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)$ of $\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}$. We need to distinguish between T^{3D} and $\breve{T}^{\mathrm{3D}}$ because from $\hat{y}^{\mathrm{ht}}_{|\tilde{U}^+_{x,x_R}}$ we subtract $m_{\tilde{U}^+_{x,x_R}}(z)$ instead of the mean $m_{\tilde{U}^+_{x,x_R}}\left(\hat{y}^{\mathrm{ht}}\right)$. The asymmetry of the DC-separation in the definition of the 3-D transforms for the groups $\mathbf{Z}_{x_R}$ and $\widehat{\mathbf{Y}}_{x_R}$ is to circumvent the intrinsic redundancy in the SA-DCT with DC-separation, which has as spectrum of $\left|\tilde{U}^+_{x_R}\right| + 1$ elements when $\left|\tilde{U}^+_{x_R}\right|$ would suffice. Essentially, by subtracting the same means from both groups, we treat these subtracted means as deterministic terms that would then be inactive in the Wiener filtering of the other coefficients.

Image	σ	BM3D	P.SA-DCT	SA-BM3D
Barbara	10	**34.98** **0.9421**	33.50 0.9342	34.75 0.9419
	20	**31.78** 0.9054	30.00 0.8862	31.65 **0.9056**
	30	**29.81** 0.8687	28.10 0.8365	**29.81** **0.8709**
Lena	10	35.93 0.9166	35.58 0.9140	**35.95** **0.9175**
	20	33.05 0.8772	32.63 0.8718	**33.08** **0.8785**
	30	31.26 0.8449	30.86 0.8393	**31.33** **0.8474**
Montage	10	37.35 0.9679	37.12 0.9657	**37.57** **0.9682**
	20	33.61 0.9404	33.36 0.9362	**33.92** **0.9407**
	30	31.37 0.9114	31.06 0.9075	**31.70** **0.9138**
Cameraman	10	34.18 0.9319	33.98 0.9316	**34.35** **0.9331**
	20	30.48 0.8755	30.18 0.8752	**30.61** **0.8772**
	30	28.64 0.8375	28.24 0.8318	**28.72** **0.8384**

Table 1: PSNR (upper entries) and SSIM (lower entries) comparison between the proposed SA-BM3D, the BM3D, and the P.SA-DCT filters.

2. Wiener filtering

We obtain an estimate $\widehat{T^{\mathrm{3D}}\left(\mathbf{Y}_{x_R}\right)}$ of the T^{3D}-spectrum of $\mathbf{Y}_{x_R}$ as

$$\widehat{T^{\mathrm{3D}}\left(\mathbf{Y}_{x_R}\right)} = \mathbf{W}_{x_R} T^{\mathrm{3D}}\left(\mathbf{Z}_{x_R}\right),$$

where the group $\mathbf{W}_{x_R}$ composed of the empirical Wiener shrinkage coefficients is defined by

$$\mathbf{W}_{x_R} = \frac{\left(\breve{T}^{\mathrm{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)\right)^2}{\left(\breve{T}^{\mathrm{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)\right)^2 + \sigma^2}. \tag{7}$$

Let us remark that here all multiplications and divisions of groups are element-by-element operations.

3. Inverse T^{3D} transform (computed exactly as in Section 3.1.2).

3.2.3 Aggregation

As in Section 3.1.3, after processing all coordinates $x_R \in X$, we obtain local estimates on adaptive-shape neighborhoods from all the groups $\widehat{\mathbf{Y}}_{x_R}$, $x_R \in X$. In order to obtain a single *global* estimate $\hat{y}^{\mathrm{wie}} : X \to \mathbb{R}$ defined on the whole image domain, we average together all the local estimates using a weighted averaging identical to (4). However, the weights w_{x_R} are now defined by

$$w_{x_R} = \frac{\sigma^{-2}}{\left\|\mathbf{W}_{x_R}\right\|_2^2 |\tilde{U}^+_{x_R}|}, \tag{8}$$

where the squared ℓ^2-norm of $\mathbf{W}_{x_R}$ is naturally computed as the sum of the square of each individual Wiener shrinkage coefficient contained in the group.

4. RESULTS

We present experimental results obtained with a preliminary version of the proposed SA-BM3D algorithm. The algorithm parameters are fixed for all experiments. Here we list some of the most

Image σ	10	20	30	40	50
Lena	35.95	33.08	31.33	29.98	28.50
	0.9175	0.8785	0.8474	0.8198	0.7822
Cameraman	34.35	30.61	28.72	27.29	25.99
	0.9331	0.8772	0.8384	0.8089	0.7664
Barbara	34.75	31.65	29.81	27.69	25.29
	0.9419	0.9056	0.8709	0.8202	0.7218
Peppers	34.72	31.22	29.16	27.67	26.54
	0.9283	0.8856	0.8501	0.8214	0.7961
House	36.85	33.86	32.12	30.74	28.69
	0.9275	0.8781	30.74	0.8283	0.7897
Montage	37.57	33.92	31.70	29.94	28.24
	0.9682	0.9407	0.9138	0.8884	0.8509
Boats	33.88	30.81	29.04	27.70	26.38
	0.8893	0.8247	0.7773	0.7364	0.6856

Table 2: PSNR (upper entries) and SSIM (lower entries) results of the proposed SA-BM3D for various σ and test images.

important ones. The 1-D transform T^{1D} (part of the separable T^{3D}) was the 1-D Haar wavelet full-dyadic decomposition. The scales $H = \{1, 2, 3, 4, 5, 6\}$ and $H = \{1, 2, 3, 4\}$ are respectively used in the first and the second step. The maximum number of matched blocks $N_2 = 16$ in the first step and $N_2 = 32$ in the second. If there are fewer matches, only the best matching 2^j are kept, $j \in \mathbb{N}$, so to be able to apply the Haar transform.

In Table 1 we provide a comparison of the PSNR and the mean structural similarity index map [16] (SSIM) results of the SA-BM3D, BM3D and P.SA-DCT methods. A visual comparison is given in Figures 5 and 6, where we show enlarged details for *Barbara*, *Montage*, and *Cameraman*. In these figures we use a relatively high standard-deviation of the noise, $\sigma = 35$, in order to emphasize the differences in the results by each method. One can make the following observations.

- Textures in *Barbara* are preserved equally well by both the BM3D and the SA-BM3D — and significantly better as compared with the P.SA-DCT filter.
- Sharp edges in *Cameraman* are reconstructed equally well by both the P.SA-DCT and the SA-BM3D filters, while the BM3D filter introduces some ringing and blurring.
- The PSNR and SSIM results of the SA-BM3D are comparable or better than the ones of the BM3D and the P.SA-DCT filters. In particular, the problems of the P.SA-DCT filter in reconstructing textures, resulting in poor PSNR for *Barbara*, has been overcome by the SA-BM3D, whose results are comparable with the ones of the BM3D filter. An interesting result can be seen in Table 1 for *Barbara* in the case of $\sigma = 20, 30$ and in Figure 5 for $\sigma = 35$; in these cases the BM3D performs best in PSNR but the SA-BM3D is best in SSIM[4].

In addition, in Table 2 we give PSNR and SSIM results of the SA-BM3D for a broader set of test images and various values of σ.

5. CONCLUSIONS

As a simultaneous generalization of the BM3D and the Pointwise SA-DCT filters, the proposed denoising method exploits both:

- nonlocal image modeling realized by *finding and grouping* of similar image neighborhoods (as in the BM3D filter);
- pointwise adaptive anisotropic estimation realized by grouping of *adaptive-shape image neighborhoods* (as in the P.SA-DCT filter).

The developed algorithm was shown in Section 4 to inherit the strengths of both methods (i.e. BM3D's good reconstruction of textures and regular image structures and P.SA-DCT's good reconstruction of sharp edges and image singularities) and at the same compensating for their deficiencies (i.e. P.SA-DCT's ineffectiveness for textures and the BM3D's blurring and detail loss around sharp curved edges and singularities).

REFERENCES

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

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

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

[4] A. Foi, *Pointwise Shape-Adaptive DCT Image Filtering and Signal-Dependent Noise Estimation*. D.Sc.Tech. Thesis, Institute of Signal Processing, Tampere University of Technology, Publication 710, 2007.

[5] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise Shape-Adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411, May 2007.

[6] A. Goldenshluger and A. Nemirovski, "On spatial adaptive estimation of nonparametric regression," *Math. Meth. Statistics*, vol. 6, pp. 135–170, 1997.

[7] J. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, Genova, Italy, September 2005.

[8] V. Katkovnik, "A new method for varying adaptive bandwidth selection," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2567–2571, September 1999.

[9] V. Katkovnik, A. Foi, K. Dabov, and K. Egiazarian, "Spatially adaptive support as a leading model-selection tool for image filtering," in *Proc. 2008 Workshop Inf. Theoretic Methods in Science and Engineering, WITMSE 2008*, Tampere, Finland, August 2008.

[10] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "Directional varying scale approximations for anisotropic signal processing," in *Proc. European Signal Process. Conf.*, Vienna, Austria, September 2004, pp. 101–104.

[11] V. Katkovnik, K. Egiazarian, and J. Astola, *Local Approximation Techniques in Signal and Image Processing*. SPIE Press, 2006, vol. PM157.

[12] P. Kauff and K. Schüür, "Shape-adaptive DCT with block-based DC separation and ΔDC correction," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 8, no. 3, pp. 237–242, March 2002.

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

[14] S. Lyu and E. Simoncelli, "Modeling multiscale subbands of photographic images with fields of Gaussian scale mixtures," *to appear in IEEE Trans. Patt. Analysis and Machine Intell.*, 2008.

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

[16] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, April 2004.

---

[4] A discussion about the subjective perceptual quality of the P.SA-DCT, with particular emphasis on the restoration of textures, can be found in [4].

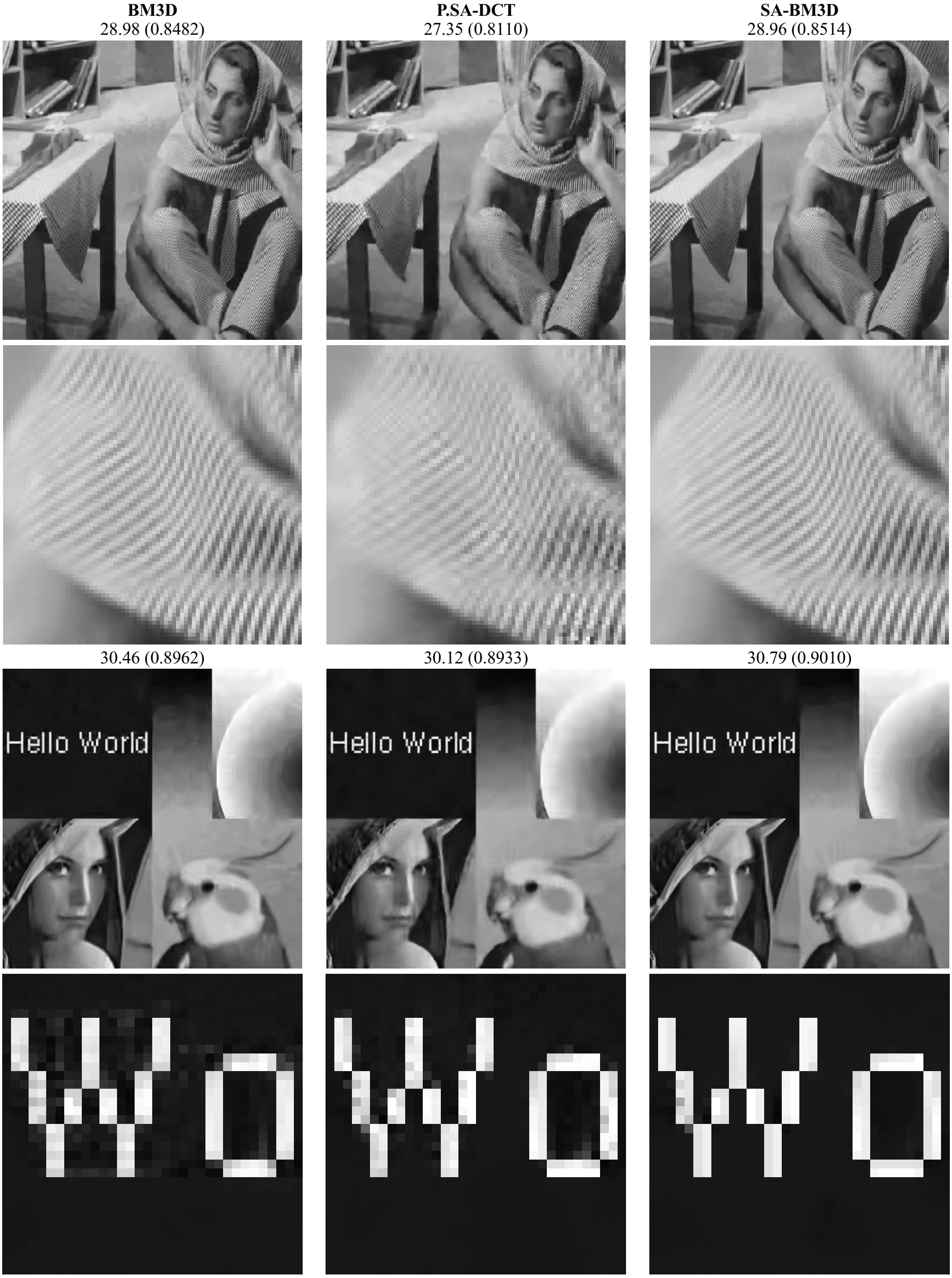


Figure 5: Comparison between denoised *Barbara* and *Montage* by the BM3D (left), the SA-DCT (center), and the SA-BM3D (right) filters. The numbers above the images are the corresponding PSNR and SSIM (in parentheses) values (calculated on the whole image). The standard deviation of the noise in the input images is $\sigma = 35$.

BM3D
27.93 (0.8218)

P.SA-DCT
27.51 (0.8147)

SA-BM3D
27.95 (0.8232)

Figure 6: Comparison between denoised *Cameraman* by the BM3D (left), the SA-DCT (senter), and the SA-BM3D (right) filters. The numbers above the images are the corresponding PSNR and SSIM (in parentheses) values (calculated on the whole image). The standard deviation of the noise in the input image is $\sigma = 35$.

Publication [P8]

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D Image Denoising with Shape-Adaptive Principal Component Analysis", *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations*, Saint-Malo, France, April 2009.

BM3D Image Denoising with Shape-Adaptive Principal Component Analysis

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian

Department of Signal Processing, Tampere University of Technology
P.O. Box 553, 33101 Tampere, Finland. E-mail: firstname.lastname@tut.fi

Abstract—**We propose an image denoising method that exploits nonlocal image modeling, principal component analysis (PCA), and local shape-adaptive anisotropic estimation. The nonlocal modeling is exploited by grouping similar image patches in 3-D groups. The denoising is performed by shrinkage of the spectrum of a 3-D transform applied on such groups. The effectiveness of the shrinkage depends on the ability of the transform to sparsely represent the true-image data, thus separating it from the noise. We propose to improve the sparsity in two aspects. First, we employ image patches (neighborhoods) which can have data-adaptive shape. Second, we propose PCA on these adaptive-shape neighborhoods as part of the employed 3-D transform. The PCA bases are obtained by eigenvalue decomposition of empirical second-moment matrices that are estimated from groups of similar adaptive-shape neighborhoods. We show that the proposed method is competitive and outperforms some of the current best denoising methods, especially in preserving image details and introducing very few artifacts.**

I. INTRODUCTION

Image denoising is a vivid research subject in signal processing because of its fundamental role in many applications. Some of the most recent and successful advances are based on: Gaussian scale mixtures (GSM) modeling in overcomplete multiscale transform domain [9], [15], [8]; learned dictionaries of atoms to filter small square neighborhoods [6]; steering kernel regression [16] (also combined with learned dictionaries [2]); shape-adaptive DCT (SA-DCT) on neighborhoods whose shapes are adaptive to the image structures [7]; nonlocal filtering based on the assumption that there exist mutually similar blocks in natural images [1], [12], [3], [4]. Among these different strategies, the nonlocal filtering, which originates from the work of Buades et al. [1], is the one that demonstrates the biggest potential. In particular, the transform-based BM3D filter [3] can be considered state-of-the-art in image denoising [13].

The BM3D filter exploits a specific nonlocal image modeling [5] through a procedure termed *grouping and collaborative filtering*. Grouping finds mutually similar 2-D image blocks and stacks them together in 3-D arrays. Collaborative filtering produces individual estimates of all grouped blocks by filtering them jointly, through transform-domain shrinkage of the 3-D arrays (groups). In doing so, BM3D relies both on *nonlocal* and *local* characteristics of natural images, namely the abundance of mutually similar patches and that the image data is locally highly correlated. If these characteristics are verified, the group enjoys correlation in all three dimensions and a sparse representation of the true signal is obtained by applying a decorrelating 3-D transform on the group. The effectiveness of the subsequent shrinkage depends on the sparsity of the true signal; i.e. the true signal can be better separated from the noise when its energy is compactly represented in the 3-D transform domain. We have shown [3] that even if square image patches and a fixed 3-D transform are used, the denoising performance is very high and the obtained MSE results are still beyond the capabilities of most of the more recent and advanced algorithms. However, square image blocks containing fine image details, singularities, or sharp and curved edges are examples where a non-adaptive transform is not able to deliver a sparse representation. Thus, for these blocks, the BM3D filter may introduce certain artifacts and the denoising is not very effective. Unfortunately, these are often the very parts of the image where the visual attention is mainly focused.

In order to further increase the sparsity of the true signal in the 3-D spectra, in [4] we proposed a generalization of the BM3D filter, which uses *grouping* of mutually similar *adaptive-shape neighborhoods*. The employed 3-D transform there is a separable composition of the (2-D) SA-DCT and a 1-D orthonormal transform. This transform is applied on 3-D groups that are generalized cylinders with adaptive-shape cross sections (as opposed to square prisms in BM3D). The adaptive-shape neighborhoods enable local adaptivity to image features so that the true signal in such a neighborhood is mostly homogeneous. Thus, the spatial correlation improves the sparsity as compared with the BM3D. However, even though the neighborhoods have adaptive shapes, the SA-DCT basis are fixed for any given shape; i.e. the basis elements do not adapt to the signal within the grouped neighborhoods.

In this paper, to enable adaptivity of the applied shape-adaptive transform basis to the input data, we propose principal component analysis (PCA) as part of the 3-D transform used for collaborative filtering. For a 3-D group of adaptive-shape image patches, we obtain a PCA basis by eigenvalue decomposition of an empirical second-moment matrix computed from these patches. As principal components (PC), we select only the eigenvectors whose corresponding eigenvalues are greater than a threshold that is proportional to noise variance. Hence, the overall 3-D transform is a separable composition of the PCA (applied on each image patch) and a fixed orthogonal 1-D transform in the third dimension.

In the sequel we present the developed denoising method and show that it is competitive and outperforms some of the current best denoising methods, particularly in preserving image details and producing very few artifacts.

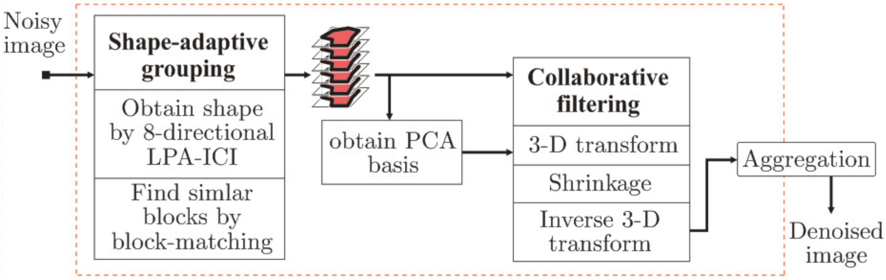


Fig. 1. Flowchart of the proposed BM3D-SAPCA image denoising method. Operations surrounded by dashed lines are repeated for each processed coordinate of the input image.

II. BM3D-SAPCA ALGORITHM

A. Algorithm outline

Following is an outline of the proposed algorithm, which we denominate BM3D-SAPCA, and which is illustrated in Figure 1. The input noisy image is assumed to be corrupted by an additive white Gaussian noise with variance σ^2 and zero mean.

- The input image is processed in raster scan where at each processed pixel the following operations are performed:

1) Obtain adaptive-shape neighborhood centered at the current pixel using the 8-directional LPA-ICI exactly as in [7], [4]. The neighborhood is enclosed within a fixed-size and non-adaptive square block, which we term *reference block*. We denote the number of pixels in the neighborhood by N_{el}.
2) Find blocks that are similar to the reference one using block-matching and extract an adaptive-shape neighborhood from each of these matched blocks using the shape obtained in Step 1. The number of matched blocks is denoted by N_{gr}.
3) Determine the transform to be applied on the adaptive-shape neighborhoods. We have two cases, depending on whether $\frac{N_{gr}}{N_{el}}$ is larger or smaller than a fixed threshold τ.
    a) If $\frac{N_{gr}}{N_{el}} \geq \tau$, we consider that we have found a sufficient number of mutually similar neighborhoods to reliably estimate a second-moment matrix. The eigenvectors of this matrix constitute the shape-adaptive PCA basis. Subsequently, we retain only those eigenvectors whose corresponding eigenvalues are greater than a predefined threshold, thus obtaining a *trimmed* shape-adaptive PCA transform.
    b) If $\frac{N_{gr}}{N_{el}} < \tau$, we deem there are not enough similar neighborhoods to use as training data and we resort to the fixed (i.e. non data-adaptive) SA-DCT, exactly as in [4].
4) Form a 3-D array (called *group*) by stacking together the $\min(N_{gr}, N_2)$ adaptive-shape neighborhoods with highest similarity to the reference one, where N_2 is a fixed parameter that restricts the number of filtered neighborhoods.
5) Apply the transform obtained in Step 3 on each of the grouped adaptive-shape neighborhoods. Subsequently, apply a 1-D orthogonal transform (e.g., Haar wavelet decomposition) along the third dimension of the 3-D group.
6) Perform shrinkage (hard-thresholding or empirical Wiener filtering) on the 3-D spectrum.
7) Invert the 3-D transform from Step 5 to obtain estimates for all of the grouped adaptive-shape neighborhoods.
8) Return the obtained estimates to their original locations using weighted averaging in case of overlapping.

B. Trimmed PCA

Since the main contribution of the proposed method is the application of the shape- and data-adaptive PCA transform on groups of adaptive-shape neighborhoods, we explain in detail what is done in Step 3a, while we refer the reader to our previous works [3], [7], [4] for details on the other steps of the algorithm. The input for Step 3a is a group of N_{gr} adaptive-shape neighborhoods that are found to be mutually similar. We represent each of these 2-D neighborhoods as a 1-D column vector $\overrightarrow{v}_i$ of length N_{el}, $i = 1, \ldots, N_{gr}$. An $N_{el} \times N_{el}$ sample second-moment matrix is then computed by matrix multiplication,

$$C = \left[\overrightarrow{v}_1 \;\; \overrightarrow{v}_2 \;\; \ldots \;\; \overrightarrow{v}_{N_{gr}} \right] \left[\overrightarrow{v}_1 \;\; \overrightarrow{v}_2 \;\; \ldots \;\; \overrightarrow{v}_{N_{gr}} \right]^T, \quad (1)$$

and subsequently its eigenvalue decomposition yields

$$U^T C U = S = \operatorname{diag}\left(s_1, s_2, \ldots, s_{N_{el}}\right),$$

where U is orthonormal matrix and S is a diagonal matrix containing eigenvalues ordered by magnitude, $s_1 > s_2 > \cdots > s_{N_{el}}$. Finally, the PCs used for the decomposition of the adaptive–shape neighborhoods are the first N_{trim} columns of U, where N_{trim} is the number of eigenvalues greater than $\lambda\sigma^2$, λ being a fixed threshold. An illustration of trimmed PCs is given in Figure 2.

C. Iterative refinement

Similarly to [3], [7], [4], the above algorithm can be applied in more than one iteration. In this work, we propose a three-iteration approach. In the first iteration, the shrinkage is performed by hard-thresholding and the search for similar blocks is performed in the noisy image, since this is the only

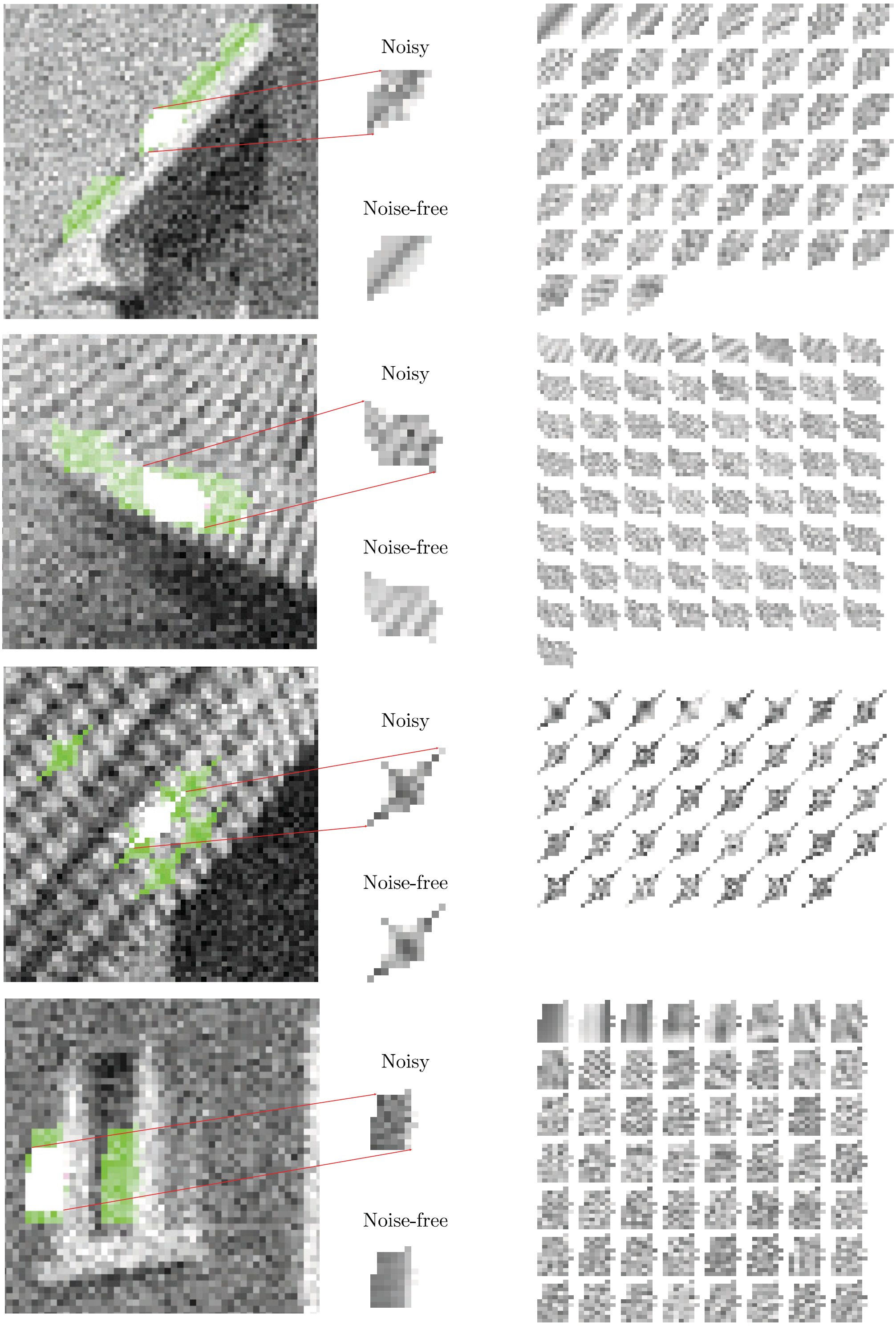


Fig. 2. Illustration of the PCs (shown on the right side) after trimming for four particular adaptive-shape neighborhoods. The green overlay is used to show the found similar neighborhoods used to form a 3-D group. The PCs are listed in decreasing magnitude of their corresponding eigenvalues. One can observe that the first few PCs have the strongest similarity with the noise-free signal in the neighborhood.

TABLE I

PSNR (dB) AND MSSIM RESULTS OF THE PROPOSED BM3D-SAPCA IMAGE DENOISING ALGORITHM. (THE MSSIM RESULTS ARE IN THE LOWER ROW OF EACH TABLE CELL.)

σ/ PSNR	*C.man* 256^2	*House* 256^2	*Peppers* 256^2	*Montage* 256^2	*Lena* 512^2	*Barbara* 512^2	*Boats* 512^2	*Man* 512^2	*Couple* 512^2	*Hill* 512^2
5/ 34.16	38.53 0.9622	39.99 0.9589	38.30 0.9563	41.40 0.9818	38.82 0.9454	38.34 0.9655	37.47 0.9434	38.03 0.9565	37.63 0.9529	37.30 0.9455
15/ 24.61	32.36 0.9089	35.15 0.8987	32.94 0.9063	35.72 0.9555	34.42 0.8976	33.30 0.9253	32.29 0.8568	32.20 0.8740	32.24 0.8784	32.05 0.8455
25/ 20.18	29.81 0.8635	32.95 0.8602	30.43 0.8685	32.96 0.9291	32.22 0.8643	30.99 0.8938	30.03 0.8037	29.81 0.8111	29.82 0.8214	29.96 0.7788
35/ 17.25	28.17 0.8269	31.37 0.8374	28.74 0.8357	30.88 0.9030	30.72 0.8367	29.35 0.8605	28.51 0.7613	28.29 0.7633	28.23 0.7730	28.62 0.7299

available input. In the following two iterations, we utilize the estimate image from each previous iteration in the following manner:

- the search for similar blocks is done in the estimate image,
- the sample second-moment matrix (1) is computed from neighborhoods extracted from the estimate image,
- for the third iteration, the shrinkage is performed by empirical Wiener filtering.

The improvement contributed by the second and by the third iteration can be justified as follows. Because noise has already been attenuated in the estimate images, both the block-matching and the estimation of the second moments are more accurate when these operations are carried out on an estimate image. It results in sparser 3-D group spectra, as both the mutual similarity within each group and decorrelation due to PCA are enhanced. In addition, the empirical Wiener filtering is more effective than hard-thresholding when the estimate image from the second iteration is used for providing a reliable estimate of the power spectrum of the 3-D groups.

III. RESULTS

We present results obtained with the proposed three-iteration BM3D-SAPCA. We use different values for τ and λ in each iteration: $\tau = 1.3$, $\lambda = 49$ in the first iteration, $\tau = 1$, $\lambda = 13$ in the second iteration, and $\tau = 0.7$, $\lambda = 13$ in the third iteration. The complexity of the algorithm is linear with respect to the number of pixels in the input image and the computation time of our Matlab-only implementation (M-file) takes approximately 4 minutes on a 2.0-GHz Pentium machine for a 256×256 image.

The PSNR and the mean structural similarity index map (MSSIM) [17] results of the proposed method are provided in Table I. In Figure 3 we compare denoised images of the proposed method and the methods that it extends: BM3D [3], P.SA-DCT [7], and SA-BM3D [4]. Relatively high noise standard deviations ($\sigma = 35$ and $\sigma = 25$) were used in order to emphasize the differences in the results by each method. From the figure, we observe that the proposed method effectively reconstructs fine details and at the same time introduces less artifacts than the other methods. This observation is also supported by the improvement in terms of MSSIM and PSNR over the other methods.

IV. DISCUSSION AND CONCLUSIONS

Regardless of the shape of the adaptive-shape neighborhood obtained in Step 2 of our algorithm, we use block-matching of *square blocks* to find similar neighborhoods, just as it is done in the original BM3D algorithm [3]. There arises the question why the matching is not done between adaptive-shape neighborhoods. In the presence of noise, the employed distance measure (ℓ^2-norm of the difference between two neighborhoods) used for the matching can be reliably computed only if the number of elements in the neighborhood is greater than a number that depends on the signal-to-noise ratio (SNR) in these blocks (an illustration of such “breakdown point” can be seen in [3]). Thus, considering that a shape-adaptive neighborhood may consist only of a single pixel and that the SNR can be relatively low, we conclude that the matching cannot be done among the adaptive-shape neighborhoods with sufficient reliability.

It is worth discussing the condition for using the PCA instead of the fixed SA-DCT, i.e. $\frac{N_{gr}}{N_{el}} \geq \tau$. Let us remark that the effectiveness of the PCA crucially depends on the sample second-moment matrix S (1). The ideal case would be to compute the second moments from the noise-free image (i.e., from an oracle), but in reality we can only estimate them either from the noisy data (as in the first iteration of the algorithm) or from otherwise distorted data (for the second and third iterations, the matrix is computed from the image estimate produced by the first and second iteration, respectively). A reasonable assumption is that greater N_{gr} results in better estimation of the second-moment matrix. When N_{gr} is small (say $N_{gr} < \tau N_{el}$), the second moments cannot be reliably estimated due to limited training data and we resort to the fixed SA-DCT transform. We recall that in the three iterations we use $\tau = 1.3, 1, 0.7$, respectively. Note that already in the case $N_{gr} < N_{el}$ (when $\tau < 1$), we have $N_{el} - N_{gr}$ null eigenvalues whose corresponding eigenvectors do not convey meaningful information. However, these eigenvectors will be trimmed since we basically keep only those eigenvectors corresponding to the larger eigenvalues. In fact, this trimming can be considered as part of the shrinkage as it is equivalent to unconditional truncation to zero of the transform coefficients that would have corresponded to the discarded eigenvectors. We note that the PC trimming approach, though a standard procedure in the practical application of the PCA (e.g., Section 2.8.5. of [10], Chapter 6 of [11]), is quite coarse and better ways could be devised to incorporate the eigenvalue magnitudes in the shrinkage itself.

Let us briefly compare our method with the K-SVD [6] and the local PCA denoising [14], both of which employ data-adaptive transform bases. The K-SVD performs training of a global (overcomplete, in general) dictionary of basis elements on small square patches; the training can be done on a set of noise-free images or directly on the noisy image. The efficacy of the subsequent denoising (performed on noisy image patches) depends on the ability of the dictionary elements to sparsely represent true-image data and thus separate it from the noise. The local PCA denoising [14] exploits PCA on square image blocks, where the covariance matrix used by the PCA is estimated from all blocks in a given neighborhood. The proposed approach exploits adaptive-shape neighborhoods, which allows for further adaptation to image details and it estimates the second-moment matrix for the PCA using *similar* adaptive-shape neighborhoods as training data — and not just any local neighborhoods. Furthermore, the sparsity of the true data is further increased by applying a transform along the third dimension of the grouped adaptive-shape neighborhoods.

The experimental results shown in Section III are very promising and demonstrate that by employing shape-adaptive PCA we can further improve the state-of-the-art denoising performance of the BM3D algorithm. Future work shall address novel shrinkage criteria, which are adaptive with respect to the utilized transforms, and the use of adaptive transforms for the third-dimension of the group.

ACKNOWLEDGMENT

This work was supported by the Academy of Finland (project no. 213462, Finnish Programme for Centres of Excellence in Research 2006-2011, project no. 118312, Finland Distinguished Professor Programme 2007-2010, and project no. 129118, Postdoctoral Researcher's Project 2009-2011) and by Tampere Graduate School in Information Science and Engineering (TISE).

REFERENCES

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

[2] P. Chatterjee and P. Milanfar, "Clustering-based denoising with locally learned dictionaries," *IEEE Trans. Image Process.*, 2008, accepted for publication.

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

[4] ——, "A nonlocal and shape-adaptive transform-domain collaborative filtering," in *Proc. Local and Nonlocal Approx. in Image Process.*, Lausanne, Switzerland, September 2008.

[5] ——, "Spatially adaptive support as a leading model-selection tool for image filtering," in *Proc. WITMSE*, Tampere, Finland, August 2008.

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

[7] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise Shape-Adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411, May 2007.

[8] J. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, Genova, Italy, September 2005.

[9] J. A. Guerrero-Colón, E. P. Simoncelli, and J. Portilla, "Image denoising using mixtures of Gaussian scale mixtures," in *Proc. IEEE Int. Conf. Image Process.*, October 2008.

[10] J. E. Jackson, *A User's Guide to Principal Components*. Wiley, 1991.

[11] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, 2002.

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

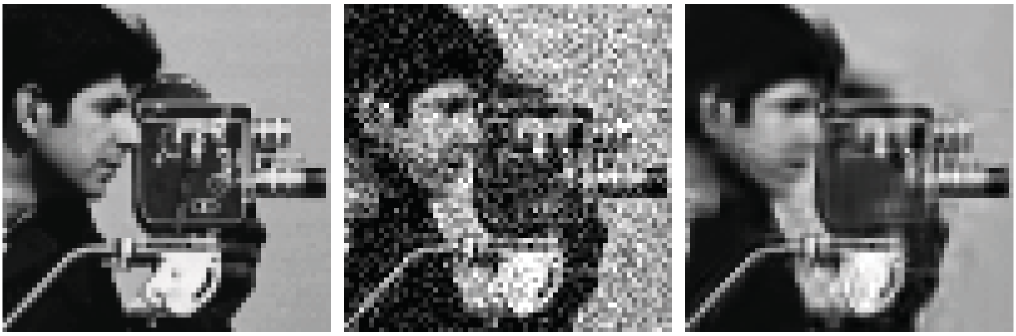
[13] S. Lansel, "DenoiseLab," http://www.stanford.edu/~slansel/DenoiseLab.

[14] D. Muresan and T. Parks, "Adaptive principal components and image denoising," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, September 2003.

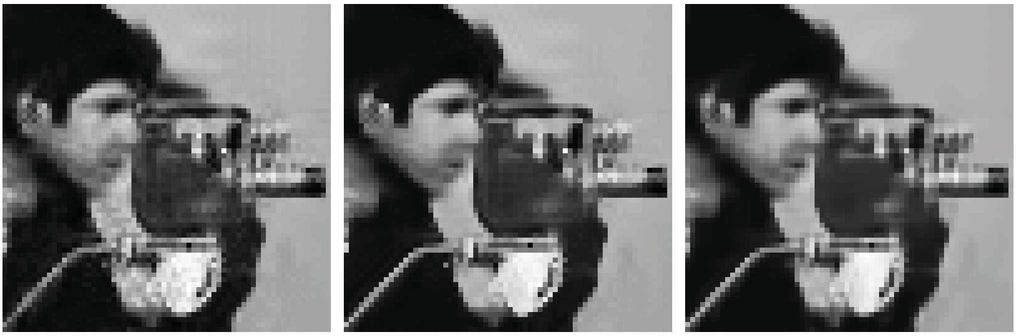
[15] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli, "Image denoising using a scale mixture of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, November 2003.

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

[17] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error measurement to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, April 2004.



Original Noisy, $\sigma = 35$ BM3D (27.82, 0.8207)



P.SADCT (27.51, 0.8143) SA-BM3D (28.02, 0.8228) BM3D-SAPCA (28.16, 0.8269)

Original Noisy, $\sigma = 25$ BM3D (29.90, 0.8010)

P.SADCT (29.47, 0.7882) SA-BM3D (29.83, 0.7994) BM3D-SAPCA (30.02, 0.8037)

Fig. 3. Zoomed fragments of *Cameraman* and *Boats* images filtered with the methods that we compare with. The numbers listed in brackets are PSNR [dB] and MSSIM results, respectively.