SPATIALLY ADAPTIVE SUPPORT AS A LEADING MODEL-SELECTION TOOL FOR IMAGE FILTERING

Vladimir Katkovnik, Alessandro Foi, Kostadin Dabov, and Karen Egiazarian

Department of Signal Processing, Tampere University of Technology
P.O. Box 553, 33101, Tampere, Finland
web: www.cs.tut.fi/~lasip email: firstname.lastname@tut.fi

ABSTRACT

One of the promising recent directions in nonparametric regression concerns the spatially adaptive estimation, which can be treated as an extended model selection problem where the basis as well as the basis supports are selected simultaneously. Our research group at the Department of Signal Processing, Tampere University of Technology, has been active in this novel field starting from 2002. The results obtained with application to different image and video processing problems are very positive and completely support the optimism following from general speculations about adaptive nonparametric modelling. In this paper we review and discuss some theoretical and practical aspects of estimation with varying spatially adaptive supports. Specifically, we consider the evolution from the LPA-ICI (local polynomial approximation - intersection of confidence intervals) algorithms to the most recent transform-based nonlocal methods.

1. INTRODUCTION

Let us refer to a couple of popular approaches exploited in image processing.
(1) *Parametric models,* usually linear, can be given explicitly as a series over basis functions ϕ_k in the form

$$y(x) = \sum_k c_k \phi_k(x). \tag{1}$$

While the standard polynomials do not fit for modelling "in large" of the so-called "natural" images, the wavelets and frames enabling localization of the estimates are quite efficient tools [41]. It was found in further developments that in many applications this sort of methods suffer from typical artifacts causing serious degradation of imaging and cannot compete with advanced modern techniques. A valuable alternative appeared in the form of the redundant representations where multiple estimates are generated for each image segment and the final estimate is calculated by aggregation of these partial estimates in to the final one (e.g. [8], [9]).
(2) *Variational models* formalize image reconstruction as an optimization problem

$$\hat{y} = \underset{y}{\operatorname{argmin}} \left(\frac{1}{\sigma^2} ||z - y||^2 + \mu \cdot \operatorname{pen}(y) \right), \tag{2}$$

where the first item of the criterion is a fidelity term corresponding to the Gaussian noise hypothesis and the second one is a penalty modelling the smoothness of y, with μ being a regularization parameter. Usually it is a functional of the derivatives of y. There are many versions of this sort of methods. One of the most popular is the total-variation minimization where the penalty is calculated as the total variation of y [44], [43]. Accurate mathematics developed for these methods make them attractive. However, a limitation of this sort of formulation follows immediately from (2), where a constant regularization parameter μ is used. It can be illustrated that by using a varying regularization parameter $\mu(x)$ we can dramatically improve denoising [48].

The nonparametric (local parametric) modelling can be addressed to the limitations of both above mentioned approaches. First of all, it works locally, as opposed to the localization in (2) achieved through a regularization parameter which is invariant and the same for all image points. Secondly, the model localization relaxes dependence of estimation on the basis function in (1) and using the adaptive neighborhood allows to get the parametric modelling for the area where this modelling accurately corresponds to $y(x)$. In its modern development, all nonparametric regression methods are spatially adaptive using varying spatially adaptive supports. We treat this sort of estimation as an extended model selection problem where the support selection is considered as an embedded element of the estimation model.

One of the promising recent directions in nonparametric regression modelling concerns the model selection treated as an extended problem, where the basis as well as the basis supports are selected simultaneously. In an adaptive version of this approach, the selected basis as well as the supports can be varying and even pointwise varying. In this formulation, we arrive to the novel class of adaptive estimates which are able to demonstrate improved and sometimes extraordinary performance.

In this paper we follow the evolution of our ideas and techniques from the straightforward local polynomial approximations with adaptive varying size symmetrical supports to shape-adaptive anisotropic support estimates. Further development goes to the transform-domain estimates producing multiple neighborhoodwise estimates aggregated into the final estimate by a special weighting. Our best results are achieved in the field of nonlocal-means [2] style algorithms, using grouping and the so-called collaborative filtering. The quality demonstrated by this type of the algorithms is beyond ability of most alternative techniques. We illustrate also our most recent development: a nonlocal algorithm with the grouping based on pointwise-adaptive anisotropic neighborhoods. Throughout the paper, our focus is on the development of the adaptivity in terms of data-driven support selection as the leading feature of the presented algorithms.

1.1 Observation model and main notation

In what follows we consider noisy observations z of the form

$$z(x) = y(x) + \eta(x), \quad x \in X, \tag{3}$$

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 to the image domain X, which can be a subset of either $\mathbb{Z}^2$ (discrete domain) or $\mathbb{R}^2$ (continuous domain). 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 its elements) of U. The symbol "$\circledast$" stands for the convolution operation.

2. LOCAL POLYNOMIAL APPROXIMATION WITH ADAPTIVE SCALE SUPPORTS

The presented approach is based on the Intersection of Confidence Intervals (ICI) rule for pointwise-adaptive estimation. Originally, the method has been developed for 1D signals [25, 29]. The idea was generalized for 2D image processing, where adaptive-size quadrant windows have been used [31].

The main intention of these techniques is to obtain, in a data-driven way, the largest local neighborhood of the estimation point in which the underlying parametric model fits the data.

2.1 Local Polynomial Approximation (LPA)

The Local Polynomial Approximation (LPA) (e.g., [11],[40]) is a technique which is applied for nonparametric estimation using a polynomial data fit in a sliding window. The polynomial order m and the window function w characterize the LPA. Specifically, for a point x, the LPA estimate $\hat{y}(x)$ of $y(x)$ given the noisy signal z is defined as

$$\begin{aligned} \hat{y}(x) &= \hat{p}(x), \\ \hat{p} &= \operatorname{argmin}_{p \in \mathcal{P}_m} \int_X w(x-v)\,(z(v) - p(v))^2\, dv, \end{aligned}$$

where $\mathcal{P}_m$ are the polynomials of order m. In other words, at every point x, the LPA provides the value $\hat{p}(x)$ of the best fitting polynomial $\hat{p}$ of order m, with the window $w(x-\cdot)$ determining the localization of this fit.

The LPA estimates are easily calculated by convolution against a kernel $g = \mathbf{w}\boldsymbol{\phi}\boldsymbol{\Phi}^{-1}[1\ 0 \cdots 0]^T$, where $\mathbf{w} = \operatorname{diag} w$ is the diagonal matrix composed by the weights w, $\boldsymbol{\phi}$ is a vector of $m+1$ polynomial functions (basis) $\phi_n = \frac{v^n}{n!}$, $n = 0, \ldots, m$, and $\boldsymbol{\Phi} = \boldsymbol{\phi}^T \mathbf{w} \boldsymbol{\phi}$ is the Gramian matrix (formed by the inner products of the basis elements against each other).

Starting from a *basic* window function w, one can obtain LPA's of different bandwidths/scales using scaled windows $w_h = w(\cdot/h)$, where $h \in \mathbb{R}^+ \setminus \{0\}$ is the *scale* parameter. The corresponding kernels are denoted as g_h. The support of the window $w(x-\cdot)$, or equivalently of the kernel $g_h(x-\cdot)$, is the estimator's support. It is common practice to use compactly supported window functions. In this case, by using a basic window w of unit length, we obtain that h coincides with the length of the window w_h. Hence, window length (size), scale, and bandwidth become interchangeable concepts.

The choice of the scale parameter is crucial when dealing with noisy data, because it controls the amount of smoothing introduced by the local approximation. A large h corresponds to a larger window and therefore to smoother estimates, with lower variance and typically increased estimation bias. A small h corresponds to noisier estimates, less biased, and with higher variance. Thus, the scale parameter h controls the trade-off between bias and variance in the LPA estimates.

In practice, the use of a fixed order m can be relaxed, and polynomial order mixtures [4],[12] are most often used, defining LPA kernels of the form $g_h = \sum_m \alpha_{h,m} g_h^{[m]}$, where $g_h^{[m]}$ is an LPA kernel of order m and the coefficients $\alpha_{h,m} \in \mathbb{R}$ are such that $\sum_m \alpha_{h,m} = 1$.

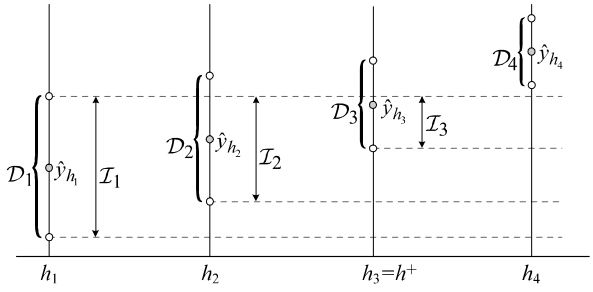

Figure 1: The Intersection of Confidence Intervals (ICI) rule.

2.2 Intersection of Confidence Intervals (ICI) rule

The Intersection of Confidence Intervals (ICI) rule [25, 29] is a criterion used for the adaptive selection of the size (length/scale) of the LPA window. The aim is to achieve a balance between the bias and the variance such that the pointwise mean square error (MSE) is minimized.

Let x be a fixed estimation point/pixel. The LPA estimates $\hat{y}_{h_j}(x) = \left(z \circledast g_{h_j}\right)(x)$ are calculated for a set $H = \{h_j\}_{j=1}^J$ of increasing scales $h_1 < \cdots < h_J$. The goal of the ICI is to select among these given estimates $\left\{\hat{y}_{h_j}(x)\right\}_{j=1}^J$ an adaptive estimate $\hat{y}_{h^+(x)}(x)$, $h^+(x) \in H$, such that $\hat{y}_{h^+(x)}(x)$ is close to an "ideal" estimate $\hat{y}_{h^*(x)}(x)$ which minimizes the MSE with respect to the variation of the scale h (note that $h^*(x)$ does not necessarily belong to H). Roughly speaking, the estimate $\hat{y}_{h^+(x)}(x)$ is the "best" among the given ones.

The ICI rule is as follows:

Consider the intersection of confidence intervals $\mathcal{I}_j = \bigcap_{i=1}^j \mathcal{D}_i$, *where*

$$\mathcal{D}_i = \left[\hat{y}_{h_i}(x) - \Gamma\sigma_{\hat{y}_{h_i}(x)}, \hat{y}_{h_i}(x) + \Gamma\sigma_{\hat{y}_{h_i}(x)}\right],$$

$\sigma_{\hat{y}_{h_i}(x)} = \operatorname{std}\{\hat{y}_{h_i}(x)\}$ *is the standard deviation of* $\hat{y}_{h_i}(x)$, *and* $\Gamma > 0$ *is a threshold parameter. Let* j^+ *be the largest of the indexes* j *for which* $\mathcal{I}_j$ *is non-empty,* $\mathcal{I}_{j^+} \neq \varnothing$ *and* $\mathcal{I}_{j^+ + 1} = \varnothing$. *The adaptive scale* $h^+(x)$ *is defined as* $h^+(x) = h_{j^+}$ *and the adaptive estimate is thus* $\hat{y}_{h^+(x)}(x)$.

An illustration of the ICI is given in Figure 1. The standard-deviations of the LPA estimates can be easily calculated from the ℓ^2-norm of the corresponding kernel as $\sigma_{\hat{y}_{h_j}(x)} = \operatorname{std}\left\{\hat{y}_{h_j}(x)\right\} = \sigma \left\|g_{h_j}\right\|_2$. Since the scales are increasing, the standard-deviations are decreasing and the confidence intervals shrink as j increases. Therefore, in the intersections we are testing estimates with progressively lower variance. The rationale behind the ICI is that the estimation bias is not too large as long as the intersections are non-empty. In practice this means that the ICI adaptively allows the maximum level of smoothing, stopping before oversmoothing begins. Asymptotically, the LPA-ICI adaptive estimator allows to get a near-optimal quality of signal recovery [25].

Overall this pointwise-adaptive algorithm searches for a largest local vicinity of the point of estimation where the estimate fits well to the data. The estimates $\hat{y}_{h_j}(x)$ are calculated for a set of window sizes (scales) and compared. The adaptive scale is defined as the largest of those for which estimate does not differ significantly from the estimators corresponding to the smaller window sizes. The above intersection of confidence intervals (ICI) rule is one of the versions of the general result known known as the Lepski's approach (e.g. [39], [42]). Several algorithms are developed, based on

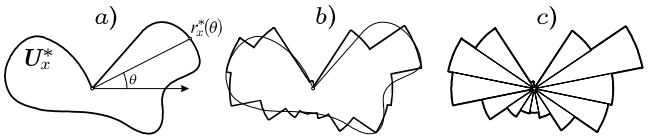

Figure 2: Approximation of an ideal starshaped anisotropic neighborhood using adaptive sectors.

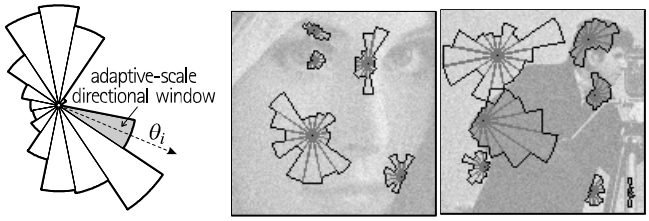

Figure 3: Anisotropic local approximations achieved by combining a number of adaptive-scale directional windows. The examples show some of these windows selected by the directional LPA-ICI for the noisy *Lena* and *Cameraman* images.

this sort of adaptive estimators: denoising [31, 29, 30] is the main, and most natural application, but also deconvolution and derivative estimation are problems where the adaptation can play a significant role in order to achieve an improved restoration performance.

3. LOCAL POLYNOMIAL APPROXIMATION WITH ANISOTROPIC SUPPORTS

3.1 Idea

Our main assumption for the design of the anisotropic estimator [23, 32] is that the optimal vicinity of the estimation point in which the model fits the data is a starshaped neighborhood which can be approximated by some sectorial decomposition with, say, K non-overlapping sectors. Such a sectorial approximation is shown in Figures 2 and 3. This irregular shape of these neighborhoods and their sectorial approximation is a direct manifestation of the *anisotropy* of the underlying signal or, roughly speaking, that the signal smoothness is different at different points and along different directions. To replicate this behavior in our estimator, we use special directional kernels defined on a sectorial support. Anisotropy is enabled by allowing different adaptive scales for different directions. Thus, the ICI rule is exploited K times, once for each sector. In this way, we reduce a complex multidimensional shape adaptation problem, to a number of scalar optimizations.

The directional estimates corresponding to the adaptive-scale sectors are then combined into the final *anisotropic* estimate. The resulting estimator is truly anisotropic, and its support can have quite an exotic shape. It is highly sensitive with respect to change-points, and allow to reveal fine elements of images from noisy observations, thus showing a remarkable advantage in the proposed strategy.

3.2 Implementation (Anisotropic LPA-ICI)

A collection of directional-LPA kernels $\{g_{h_j,\theta_k}\}_{h_j\in H,k=1,\dots,K}$ supported on such sectors is designed. Each kernel is characterized by a direction θ_k and a scale parameter h. The corresponding estimate of y is given by the convolution

$$\hat{y}_{h_j,\theta_k} = g_{h_j,\theta_k} \circledast z. \tag{4}$$

For a fixed x, we obtain a collection $\left\{\hat{y}_{h_j,\theta_k}(x)\right\}_{h_j\in H,k=1,\dots,K}$ which is multi-scale and multi-directional. For each specified direction θ_k, the ICI rule is used to select a pointwise-adaptive scale $h^+(x,\theta_k) \simeq r_x^*(\theta_k)$ that approximates the radius of the ideal neighborhood U_x^*. Let $\hat{y}_{h^+(x,\theta_k),\theta_k}$ be the directional adaptive-scale estimate,

$$\hat{y}_{h^+(x,\theta_k),\theta_k}(x) \triangleq \left(g_{h^+(x,\theta_k),\theta_k} \circledast z\right)(x) \qquad \forall x \tag{5}$$

and let

$$\sigma_k^2(x) \triangleq \sigma^2_{\hat{y}_{h^+(x,\theta_k),\theta_k}(x)} = var\left\{\hat{y}_{h^+(x,\theta_k),\theta_k}(x)\right\} \qquad \forall x \tag{6}$$

be its variance[1]. All these estimates can be fused in the final *anisotropic estimate* $\hat{y}$ as follows:

$$\begin{aligned} \hat{y}(x) &= \sum_k \lambda(x,\theta_k)\hat{y}_{h^+(x,\theta_k),\theta_k}(x), \\ \lambda(x,\theta_k) &= \frac{\sigma_k^{-2}(x)}{\sum_i \sigma_i^{-2}(x),} \end{aligned} \qquad \forall x\ . \tag{7}$$

The weights $\lambda(x,\theta_k)$ in the above convex combination are data-driven adaptive, as $\sigma_k^{-2}(x)$ depends on the adaptive $h^+(x,\theta_k)$. Formula (7) embeds and makes clear our basic intentions. We introduce the directional estimates $\hat{y}_{h_j,\theta_k}(x)$, optimize the scale parameter for each of the directions (sectors), and fuse the resulting directional adaptive estimates into the final one $\hat{y}(x)$ using the weights $\lambda(x,\theta_k)$. We call this approach the *anisotropic LPA-ICI technique*. The union U_x^+ of the supports of the kernels $g_{h^+(x,\theta_k),\theta_k}$,

$$U_x^+ = \bigcup_k \operatorname{supp} g_{h^+(x,\theta_k),\theta_k}, \tag{8}$$

is regarded as an approximation of the optimal U_x^*. Moreover, we note that (7) corresponds to a maximum-likelihood estimate of $y(x)$ given that the directional adaptive estimates $\hat{y}_{h^+(x,\theta_k),\theta_k}(x)$, $k = 1,\dots,K$, are all unbiased and independent[2].

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

$$\tilde{U}_x^+ = \left\{v \in X : (x-v) \in U_x^+\right\}, \tag{9}$$

in other words $\tilde{U}_x^+$ (with tilde) is obtained by translation and mirroring of U_x^+ (without tilde). The optimal $\tilde{U}_x^*$ is defined analogously.

Figure 4 shows an ideal example for a simple geometric image with four neighborhoods $\tilde{U}_x^*$. Figure 5 shows the corresponding adaptive anisotropic neighborhoods $\tilde{U}_x^+$ resulting from the anisotropic LPA-ICI approach for this noisy image. A comparison between the two figures shows the similarity between the ideal and the concrete case.

Figure 6 shows the noisy observation of the *Cameraman* image, ($\sigma = 25$). Figure 7 presents fragments of the original (for comparison) and of three restored images. One is

[1] In the equation (6), we treat h^+ as a purely deterministic variable. This simplification is however quite reasonable, as in practice the adaptive h^+ does not exhibit a significant variability.

[2] Here the independence follows from the sectors being non-overlapping. It can be shown [12] that, for a large number of sectors, the error in the directional estimates $\hat{y}_{h^+(x,\theta_k),\theta_k}$ is mostly due to variance and not to bias, as much as that these estimates can be practically treated as unbiased ones.

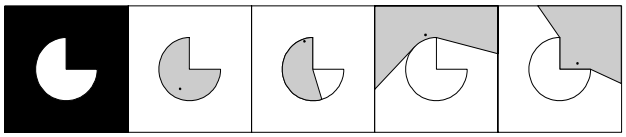

Figure 4: Examples of the ideal starshaped neighborhoods $\tilde{U}_x^*$. The first subimage to the left is the true signal y, followed by the illustrations of four different ideal neighborhoods $\tilde{U}_x^*$ corresponding to four different points $x \in X$.

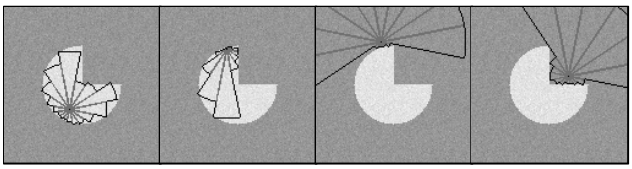

Figure 5: *Cheese*: adaptive anisotropic neighborhoods $\tilde{U}_x^+$ obtained through ICI using sectorial kernels. Compare with the ideal example shown in Figure 4.

obtained using the anisotropic LPA-ICI, the other two are the results of translation-invariant wavelet thresholding [5]. Wavelet thresholding is performed using the best-found (oracle) value of the threshold parameter. Although the PSNR values are not too different, the image reconstructed by the anisotropic algorithm is visually much better, presenting well defined edges, faithfully reconstructed details, and no noticeable artifacts (such as the unpleasant ringing visible in the Daubechies-wavelets estimate).

Further results and modifications of this algorithm for different applications (including gradient estimation, deconvolution, inverse-halftoning, video denoising, and signal-dependent noise removal) can be seen in [32, 23, 15, 33, 14, 24, 10, 14] and, in particular, in the thesis [12] and in the book [30].

4. RECURSIVELY ENLARGING ANISOTROPIC SUPPORTS

4.1 Idea

The idea behind this procedure [23] is to apply recursively the anisotropic LPA-ICI algorithm, filtering the final output $\hat{y}$ (7) once or many times over again.

Denoting by $\mathcal{LI}$ the overall anisotropic LPA-ICI filter, this recursion is expressed as follows:

$$\begin{cases} z^{[1]} = z, \\ \hat{y}^{[l]} = \mathcal{LI}(z^{[l]}), \qquad l = 1, 2, \dots \quad . \\ z^{[l+1]} = \hat{y}^{[l]}, \end{cases} \tag{10}$$

The square brackets [ ] indicate the iteration number.

Even if the ideal neighborhood U_x^* is always the same for all l, the support of the resulting kernel that is used for integration against $z(v^{\{l\}})$ in (10) may grow at every iteration. For example, assuming that the ideal neighborhoods do not change with l, at the second iteration the estimation support with respect to the initial observations z is $\mathrm{supp} \int 1_{\tilde{U}_x^*}(v) 1_{\tilde{U}_v^*}(\cdot) dv = \bigcup_{v \in \tilde{U}_x^*} \tilde{U}_v^*$. This is illustrated in Figure 8, with (a) some ideal starshaped neighborhoods $\tilde{U}_v^*$ corresponding to points v belonging to, (b) the ideal neighborhood $\tilde{U}_x^*$ of the estimation point x, and (c) the resulting *enlarged* neighborhood of x, $\bigcup_{v \in \tilde{U}_x^*} \tilde{U}_v^*$, obtained by the second iteration of the adaptive algorithm. Such sets are not necessarily starshaped with respect to x.

Figure 6: The noisy observation of the *Cameraman* image, $\sigma = 25$ (PSNR=19.97dB).

Figure 7: Denoising of the *Cameraman* image, $\sigma = 25$. Clockwise from top-left: original image, anisotropic LPA-ICI estimate, PSNR=28.1dB, translation-invariant Daubechies wavelets (DB4), PSNR=27.4dB, and translation-invariant Haar wavelets, PSNR=27.8dB.

4.2 Implementation (Recursive Anisotropic LPA-ICI)

After setting the initial conditions $y^{[0]} = z$ and $\hat{\sigma}_y^{[0]} \equiv \sigma$, the l-th recursive step of the modified recursive algorithm is

$$\hat{y}^{[l]} = \mathcal{LI}(\hat{y}^{[l-1]}), \quad \hat{\sigma}_{\hat{y}^{[l]}} = \left(\sum\nolimits_k \left(\hat{\sigma}_k^{[l]} \right)^{-2} \right)^{-1/2}, \quad l = 1, 2, \dots,$$

where $\hat{\sigma}_k^{[l]} = \hat{\sigma}_{\hat{y}^{[l]}_{h^+(x,\theta_k),\theta_k}}$, $\hat{\sigma}_{\hat{y}^{[l]}_{h,\theta_k}} = \alpha (g_{h,\theta_k}^2 \circledast \hat{\sigma}^2_{\hat{y}^{[l-1]}})^{1/2}$, and $\alpha < 1$ being the fixed correcting factor, which accounts for the correlation of the noise in the filtered samples. Few iterations are usually sufficient for the algorithm to reach a

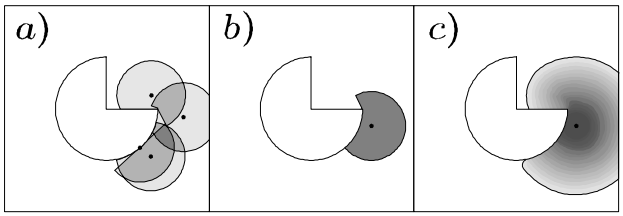

Figure 8: Some ideal starshaped neighborhoods $\tilde{U}_v^*$ (a) corresponding to points v belonging to the ideal neighborhood $\tilde{U}_x^*$ of the estimation point x (b) and the resulting *fattened* neighborhood of x, $\bigcup_{v\in\tilde{U}_x^*}\tilde{U}_v^*$ (c), obtained by the second iteration of the adaptive algorithm.

Figure 9: Restored *Cameraman* image after three iterations of the recursive anisotropic algorithm (PSNR=28.5dB).

numerical steady-state.

For the denoising of the noisy *Cameraman* ($\sigma = 25$), we obtain a PSNR of 28.44dB at the third iteration, shown in Figure 9). A similar result cannot be achieved by the non-recursive algorithm.

5. TRANSFORM METHODS WITH ADAPTIVE-SHAPE SUPPORTS

5.1 Idea

Despite the outstanding success demonstrated by these adaptive techniques, it must be observed that, at a local level, such estimators are characterized by a tightly restricted number of parameters, namely the polynomial order. While the intrinsic stability of these simple models makes their spatial adaptation effective and robust, it also poses a significant limit to their fitting ability. Low-order polynomial models appear often too rigid for satisfactorily approximating the finer local behaviors of natural images.

Improved approximation can be achieved by means of higher-order models. Unfortunately, direct application of adaptive scale-selection procedures for higher-order models fails, because of the very large variance inherent of the higher-order estimates [28]. A number of compromises have been proposed in the literature: for example, exploiting a lower-order (usually zero-order) model in the spatial-adaptation step and a higher-order one (first- or second-order) for the approximation and estimation; or separately producing few spatially-adaptive approximations of different fixed orders (e.g. zero, first, and second order) and then combining them, with some (adaptive or non-adaptive) weights (see, e.g., [4], [25]). However, the choice of the final (i.e. maximum) order remains crucial. It should be noted that if this order is unrestricted there is no smoothing, since the model attains a perfect fit to the noisy observations. It turns out that it is necessary not only to adapt with respect to the spatial features in the image, but also with respect to the different orders which are required to model such various features. We solve this dilemma by decomposing the higher-order models in several orthogonal complements and adaptively compose the most appropriate model from the corresponding subspaces using conventional estimators such as thresholding. This is done at a local level, on an adaptive starshaped neighborhood defined via a reliable zero-/first-order multidirectional estimator, namely the anisotropic LPA-ICI [23, 32]. In this way, we come to pointwise spatially- and order-adaptive anisotropic polynomial estimators.

We make one step further, and relax the procedure by replacing polynomial models with more general transforms. In particular, we concentrate on the shape-adaptive variants of the discrete cosine transform (DCT) because of their near-optimal decorrelation ability for natural images. Low-complexity versions of such transforms exist and can be exploited to develop fast algorithms based on the above paradigm.

This class of algorithms differ from the above anisotropic LPA-ICI in a number of fundamental aspects.

Firstly, the *pointwise* approximations of the signal are replaced by *local* approximations over neighborhoods (we remind that in LPA-ICI algorithms the estimates are always pointwise, meaning that each LPA is exploited to estimate a unique pixel only).

Further, the *fixed* polynomial order is replaced by an *adaptive* order (which is restricted only by the size of the neighborhood where the local model is fitted).

Finally, we exploit *aggregation*, in order to compensate to the variability of these local estimates and thus to improve the quality of the final estimate.

The following Pointwise Shape-Adaptive DCT algorithm [19] has been introduced through a number of publications, targeting various image filtering problems [16]-[22]. Its most comprehensive presentation can be found in the thesis [13].

5.2 Implementation (Pointwise Shape-Adaptive DCT algorithm)

Our approach to estimation for a point x_0 can be roughly described as the following four stage procedure.

Stage I (*spatial adaptation*)**:** For every $x \in X$, define a neighborhood $\tilde{U}_x^+$ of x where a simple low-order polynomial model fits the data;

Stage II (*order selection*)**:** apply some localized transform (parametric series model) to the data on the set $\tilde{U}_x^+$, use thresholding operator (model selection procedure) in order to identify the significant (i.e. nonzero) elements of the transform (and thus the order of the parametric model).

Stage III (*approximation*)**:** Calculate, by inverse-transformation of the significant elements only, the corresponding estimates $\hat{y}_{\tilde{U}_x^+}(v)$ of the signal for all $v \in \tilde{U}_x^+$. These $\hat{y}_{\tilde{U}_x}$ are calculated for all $x \in X$.

Stage IV (*aggregation*)**:** Let $x_0 \in X$ and $I_{x_0} =$

$\left\{x \in X : x_0 \in \tilde{U}_x^+\right\}$ be the set of the centers of the neighborhoods which have x_0 as a common point. The final estimate $\hat{y}(x_0)$ is calculated as an aggregate of $\left\{\hat{y}_{\tilde{U}_x^+}(x_0)\right\}_{x \in I_{x_0}}$.

One key aspect in this procedure is that by demanding the local fit of a low-order polynomial model, we are able to avoid the presence of singularities, discontinuities, or sharp transitions within the transform support $\tilde{U}_x^+$. In this way, we increase the sparsity in the transform domain, improving the effectiveness of thresholding.

Let us present in detail these various stages, with particular attention to the spatial adaptation.

5.2.1 Spatial adaptation: adaptive anisotropic neighborhoods

We exploit a simplified structure for defining the anisotropic neighborhood U_x^+, where sectorial kernels are replaced by 1-D kernels and the union (8) is replaced by a hull.

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 1-D "linewise" directional-LPA 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 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.

Shape-adaptive neighborhood structure

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 10. 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 $x \in X$.

We emphasize again the distinction (9) between the neighborhoods U_x^+ and $\tilde{U}_x^+$. 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.

5.2.2 Order selection and approximation: local estimates

For every point $x \in X$, we construct a *local* estimate $\hat{y}_{\tilde{U}_x^+} : \tilde{U}_x^+ \to \mathbb{R}$ of the signal y by thresholding in some transform domain,

$$\hat{y}_{\tilde{U}_x^+} = T_{\tilde{U}_x^+}^{-1}\left(\mathtt{shrink}\left(T_{\tilde{U}_x^+}\left(z_{|\tilde{U}_x^+}\right)\right)\right), \tag{11}$$

where $T_{\tilde{U}_x^+}$ is a shape-adaptive transform (e.g., the shape-adaptive DCT) and **shrink** is a some shrinkage operator (e.g., hard-thresholding or Wiener filtering). In hard-thresholding, only the coefficients whose amplitude is larger than a threshold are kept, all other smaller coefficients are discarded and replaced by zeros. The sparsity (low-complexity model) achieved thanks to the adaptive selection of the transform support ensures that most of the energy of the original signal is carried by only few noisy coefficients, which are kept after thresholding, and that the many discarded coefficients contain mostly noise.

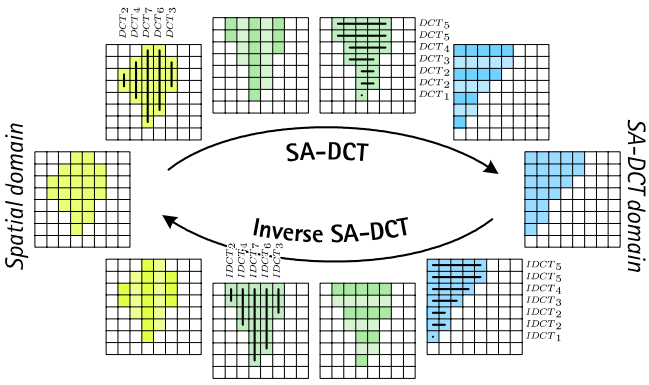

Figure 11: Illustration of the shape-adaptive DCT transform and its inverse. Transformation is computed by cascaded application of one-dimensional varying-length DCT transforms, along the columns and along the rows.

All the basis elements (or generators) corresponding to the transform $T_{\tilde{U}_x^+}$ need to be supported on the adaptive neighborhood $\tilde{U}_x^+$.Furthermore, in case of an orthonormal transform, orthonormality is considered with respect to the norm of $L^2\left(\tilde{U}_x^+\right)$. Thus, to each differently-shaped neighborhood corresponds a different basis and hence a different transform. The practical design of transforms for arbitrarily-shaped domains has been subject of extensive research, with the shape-adaptive DCT (SA-DCT) [45, 46, 34] appearing as the most successful option. This low-complexity transform (implemented in a separable-like manner, as shown in Figure 11) is our transform of choice for the examples shown in this paper.

Since the anisotropic neighborhoods corresponding to nearby points are usually overlapping, for a point x_0 in the image we have a number of estimates $\hat{y}_{\tilde{U}_x^+}(x_0)$, $x \in I_{x_0}$.

5.2.3 Aggregation: global estimate

In order to obtain a single *global* estimate $\hat{y} : X \to \mathbb{R}$ defined on the whole image domain, all the local estimates (11) are averaged together using adaptive weights $w_x \in \mathbb{R}$ in the following convex combination:

$$\hat{y} = \frac{\sum_{x \in X} w_x \hat{y}_{\tilde{U}_x^+}{}^{|X}}{\sum_{x \in X} w_x \chi_{\tilde{U}_x^+}}. \tag{12}$$

Here, the weights w_x that are inversely proportional to the variance of $\hat{y}_{\tilde{U}_x^+}$, typically expressed by the norm of the coefficients used for the reconstruction after shrinkage ([13],[26]).

The aggregation (12) implies that for a point $x_0 \in X$, the final estimate $\hat{y}(x_0)$ depends on data given in the union of the neighborhoods $\bigcup_{x \in I_{x_0}} \tilde{U}_x^+$, as opposed to the anisotropic LPA-ICI, where the final estimate for a point depends on the data given in the corresponding neighborhood only. In this sense, in our aggregation the initial local estimate $\hat{y}_{\tilde{U}_{x_0}^+}$ is supplanted by one with much larger areas of the data involved.

5.2.4 Two-stage algorithm: Hard-thresholding + Wiener filtering

This kind of algorithms is nearly always implemented using two stages. A first stage, as described above, uses hard-thresholding as the shrinkage operator. After aggregation (12), the estimate $\hat{y}$ is used as a reference image for a second stage. There, the various steps of the first stage are

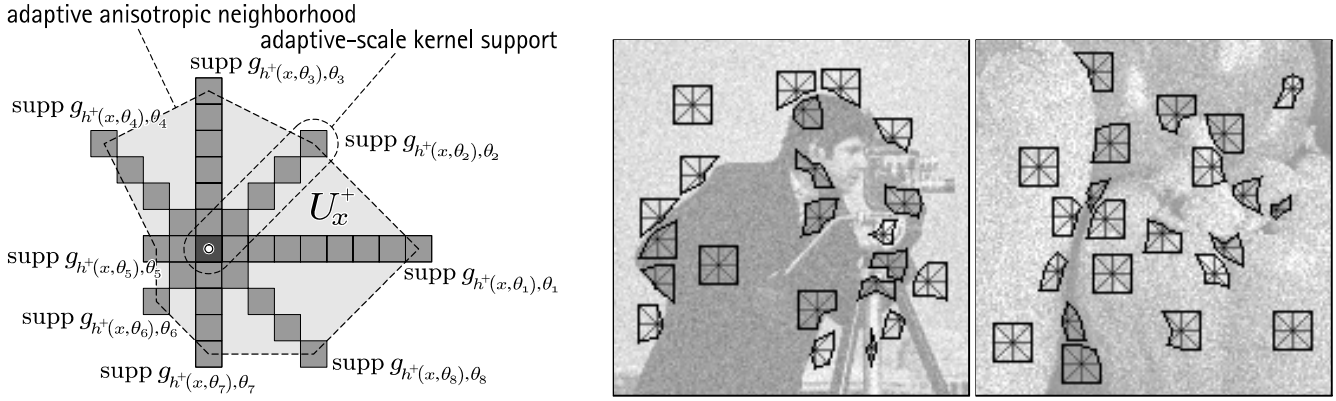

Figure 10: Simplified implementation of the LPA-ICI anisotropic neighborhoods. "Linewise"one-dimensional directional LPA kernels are used for 8 directions. The anisotropic neighborhood U_x^+ is constructed as the polygonal hull of the adaptive-scale kernels' supports (left). Thus, only the adaptive scales 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* and *Peppers* images (right). Here we use $h \in H = \{1, 2, 3, 5, 7, 9\}$.

Figure 12: Denoised *Cameraman* by the Pointwise Shape-Adaptive DCT algorithm (PSNR=29.10dB).

repeated; however, hard-thresholding is now replaced by empirical Wiener filtering, with the spectra from the estimate $T_{\tilde{U}_x^+}\left(\hat{y}_{|\tilde{U}_x^+}\right)$ used for determining the attenuation of the noisy spectra $T_{\tilde{U}_x^+}\left(z_{|\tilde{U}_x^+}\right)$.

Overall, these various modifications with respect to the anisotropic LPA-ICI denoising of Section 2 lead to a dramatic improvement in quality, as can be seen by comparing Figure 12 with Figures 7 and 9.

6. BLOCKWISE NONLOCAL SUPPORTS

6.1 Idea

The blockwise *nonlocal* estimation means that the data are windowed/segmented into overlapping blocks and one looks

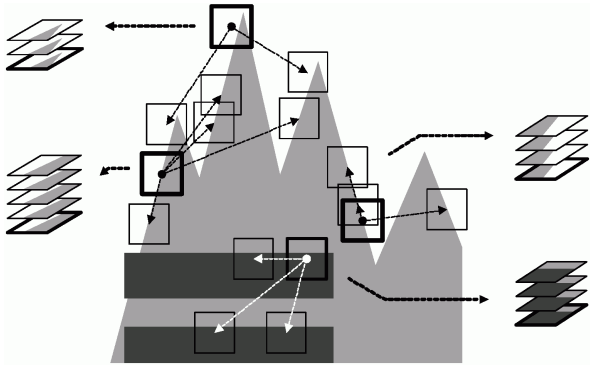

Figure 13: A simple example of grouping in an artificial image, where for each reference block (with thick borders) there exist perfectly similar ones.

for mutually similar blocks, which are collected into groups, so that the data in these groups can be processed jointly. In this way we arrive to the nonlocal varying adaptive support estimator where the data used in the estimation can be located quite far from each other. This estimation can be treated as a kind of non-local means [2],[3],[37],[35],[36].

To clarify the idea of grouping, let us consider an illustrative example of blockwise nonlocal estimation of the image in Figure 13 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) an optimal estimator. In this way, we achieve an accuracy that cannot be obtained by processing the separate blocks independently.

However, perfectly identical blocks are unlikely in natural images. If non-identical fragments are allowed within the same group, averaging is no longer optimal. Therefore, a filtering strategy more effective than averaging should be employed.

In this and in the following section, we consider two specific nonlocal algorithms. First, an algorithm with the square supports and, second, one with pointwise-adaptive anisotropic supports, similar to those used in Section 5.2. What makes these two algorithm very different from other

non-local estimators is the use of full-rank complete transforms for the modeling both the blocks and their mutual difference. In this way, the developed model-selection includes both adaptive support and adaptive order.

A detailed description of the first algorithm was published in [7]; the second one is very recent and its preliminary version is presented in [6].

6.2 Implementation (Block-matching and 3D filtering algorithm)

6.2.1 Block-matching

Let $x \in X$ and denote by $\tilde{B}_x \subset \mathbb{Z}^2$ be the square block of size $l \times l$ "centered"[3] at x. Let $\mathbb{B}$ be the collection of all such blocks which are entirely contained in X, $\mathbb{B} = \left\{\tilde{B}_x : x \in X,\ \tilde{B}_x \subset X\right\}$. Equivalently, define $X_{\mathbb{B}} = \left\{x \in X : \tilde{B}_x \in \mathbb{B}\right\} = \left\{x \in X : \tilde{B}_x \subset X\right\} \subset X$.

For each block $\tilde{B}_x \in \mathbb{B}$, (i.e. for each point $x \in X_{\mathbb{B}}$), we look for "similar" blocks $\tilde{B}_{x'}$ whose range distance $d_z(x, x')$ with respect to $\tilde{B}_x$,

$$d_z(x, x') = \left\| z_{|\tilde{B}_x} - z_{|\tilde{B}_{x'}} \right\|_2, \tag{13}$$

is smaller than a fixed threshold $\tau_{\text{match}} \geq 0$. Thus, we construct the set S_x that contains the central points of the found blocks:

$$S_x = \left\{x' \in X_{\mathbb{B}} : d_z(x, x') \leq \tau_{\text{match}}\right\}. \tag{14}$$

The threshold τ_{match} is the maximum d_z-distance for which two blocks are considered similar. In case of heavy noise, we embed a coarse prefiltering within d_z (e.g., ℓ^2-distance of thresholded spectra). Otherwise, we can increase l.

To a fixed "reference" block $\tilde{B}_{x_R} \in \mathbb{B}$ associate a collection (disjoint union) $\widetilde{\mathbb{B}}_{x_R}$ of neighborhoods[4]:

$$\widetilde{\mathbb{B}}_{x_R} = \coprod_{x \in S_{x_R}} \tilde{B}_x. \tag{15}$$

6.2.2 Group

Given $\widetilde{\mathbb{B}}_{x_R}$, we build a *group* by stacking together the noisy patches $z_{|\tilde{B}_x}$, $\tilde{B}_x \in \widetilde{\mathbb{B}}_{x_R}$. This group is a 3-D data array defined on the square prism $B \times \{1, \ldots, |S_{x_R}|\}$. In compact form, the group is denoted $\mathbf{Z}_{x_R} : \widetilde{\mathbb{B}}_{x_R} \to \mathbb{R}$.

Groups are characterized by both:

- ◇ *intra*-block correlation between the pixels of each grouped block (natural images);
- ○ *inter*-block correlation between the corresponding pixels of different blocks (grouped block are similar).

However, special care is needed, because:

- ◇ blocks are not necessary flat or smooth but can be anything;
- ○ "similar" does not mean "identical".

Thus, when approaching groups, we have the following goals:

- ◇ exploit intra-block correlation whenever possible, without smoothing away details;
- ○ exploit similarity in the forms in which it exists, without forcing dissimilar blocks to become identical.

6.2.3 Collaborative filtering

With the above mentioned goals in mind, we process the group by the so-called *collaborative filtering* approach. This term is explained as follows:

- *each grouped block collaborates for the filtering of all others, and vice versa.*
- *the filtering provides individual estimates for all grouped blocks (not necessarily equal).*

We realize the collaborative filtering as shrinkage in a 3-D transform domain of the form

$$\widehat{\mathbf{Y}}_{x_R} = T^{3\text{D}\,-1}\left(\mathtt{shrink}\left(T^{3\text{D}}\left(\mathbf{Z}_{x_R}\right)\right)\right),$$

where $T^{3\text{D}}$ is a 3-D transform. In practice, $T^{3\text{D}}$ is a separable transform obtained by composing a 2-D transform $T^{2\text{D}}$ (providing *intra*-block decorrelation) with a 1-D transform $T^{1\text{D}}$ (for the *inter*-block decorrelation of the $T^{2\text{D}}$-spectra): $T^{3\text{D}} = T^{2\text{D}} \circ T^{1\text{D}}$.

The group estimate $\widehat{\mathbf{Y}}_{x_R} : \widetilde{\mathbb{B}}_{x_R} \to \mathbb{R}$ is composed of slices with local block estimates $\hat{y}_{x,x_R} : \tilde{B}_x \to \mathbb{R}$ for each $\tilde{B}_x \in \widetilde{\mathbb{B}}_{x_R}$.

6.2.4 Aggregation

For each reference point $x_R \in X$, grouping and collaborative filtering generate a group $\widehat{\mathbf{Y}}_{x_R}$ of $|S_{x_R}|$ distinct *local* estimates of y. Overall, we have a highly redundant and rich representation of the original image y composed of the estimates

$$\coprod_{x_R \in X,\ x \in S_{x_R}} \hat{y}_{x,x_R}, \quad \text{where} \quad \hat{y}_{x,x_R} : \tilde{B}_x \to \mathbb{R}.$$

It is important to emphasize that different groups $\mathbf{Z}_{x_R}$ and $\mathbf{Z}_{x'_R}$ can lead to different estimates $\hat{y}_{x,x_R}$ and $\hat{y}_{x,x'_R}$ even when these estimates are defined on the same block $\tilde{B}_x$. Similar to Section 5.2.3, in order to obtain a single *global* estimate $\hat{y} : X \to \mathbb{R}$ 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:

$$\hat{y} = \frac{\sum_{x_R \in X} \sum_{x \in S_{x_R}} w_{x_R} \hat{y}_{x,x_R}{}_{|X}}{\sum_{x_R \in X} \sum_{x \in S_{x_R}} w_{x_R} \chi_{\tilde{B}_x}}.$$

As for (12), the weights used for the aggregation are inversely proportional to the norm of the $T^{3\text{D}}$-coefficients used for the reconstruction after shrinkage.

6.2.5 Two-stage algorithm: Hard-thresholding + Wiener filtering

Also this algorithm exploits two similar stages, one based on hard-thresholding and another based on empirical Wiener filtering. Here, in particular, during the second stage, we can use the estimate $\hat{y}$ from the first stage to improve the accuracy of the block-matching, i.e. by replacing z with $\hat{y}$ in (13).

---

[3] When l is even the center of the block can be taken as, say, the $(l/2, l/2)$ pixel within the block.

[4] The set (15) must not be interpreted as the mere union of the blocks $\tilde{B}_x$ such that $x \in S_{x_R}$. While, for simplicity, we may write

$$\coprod_{x \in S_{x_R}} \tilde{B}_x = \left\{\tilde{B}_x : x \in S_{x_R}\right\},$$

a more proper notation is actually

$$\coprod_{x \in S_{x_R}} \tilde{B}_x = \left\{\left(\tilde{B}_x, x\right) : x \in S_{x_R}\right\} \subset X \times S_{x_R} \subset X \times X,$$

because we need to distinguish between different blocks coming from different elements of S_{x_R}.

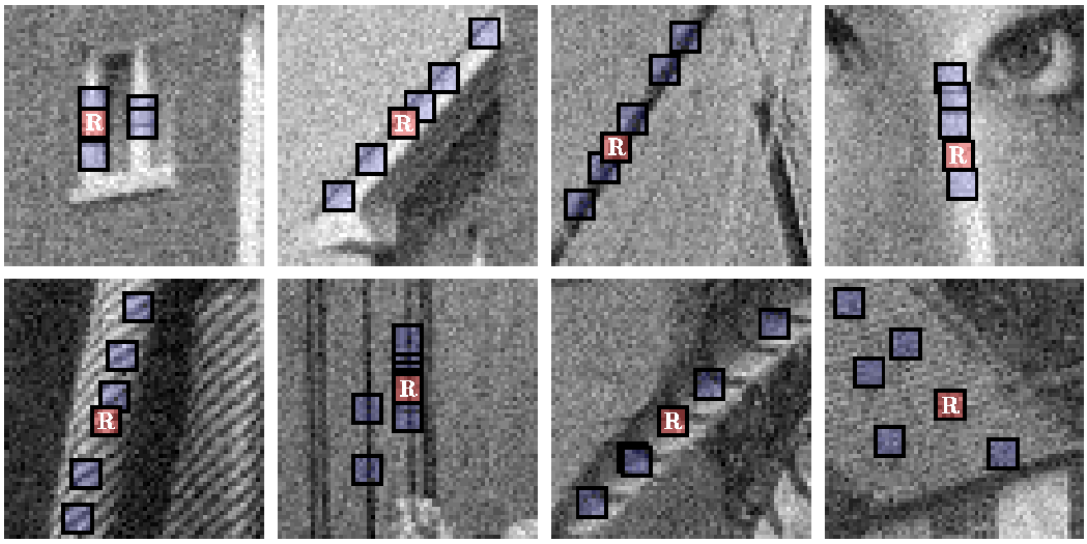

Figure 14: Illustration of grouping blocks from noisy natural images corrupted by white Gaussian noise with standard deviation 15 and zero mean. Each fragment shows a reference block marked with "R"and a few of the blocks matched to it.

Figure 15: Denoised *Cameraman* by the Block-Matching and 3D filtering (BM3D) algorithm (PSNR=29.48dB).

The strength of the nonlocal blockwise modeling is largely confirmed by numerous experiments [7],[47],[38]. Here, because of length limitation, we present only the result of denoising of *Cameraman* ($\sigma = 25$), shown in Figure 15. As can be seen by a comparing this figure with Figure 7, Figure 9, and Figure 12, the BM3D estimate is better that the estimates obtained by the other algorithms presented in the previous sections.

7. SHAPE-ADAPTIVE NONLOCAL SUPPORTS

7.1 Idea

The existence of mutually-similar patches is, as illustrated in Figure 14, a characteristic feature of natural images. Due to this, the above BM3D algorithm can generally achieve an excellent denoising accuracy. However, the assumption that the block should be square is very artificial. By replacing fixed-size block transforms with adaptive-shape transforms, we obtain a more flexible tool with a potential for better denoising performance.

7.2 Implementation (Block-matching 3D shape-adaptive filtering algorithm)

The main modifications with respect to the above blockwise algorithm concern with the matching and with the grouping.

7.2.1 Block-matching

The adaptive neighborhoods $\tilde{U}_x^+$ can be *too small* for reliable matching, especially when filtering image details. Therefore, the matching for $\tilde{U}_x^+$ needs to be carried out for a superset. In particular, we use square blocks as supersets and perform a block-matching procedure much like the one in the blockwise algorithm.

Let $\tilde{B}_x \subset \mathbb{Z}^2$ be the square block of size $(2h_{\text{max}} - 1) \times (2h_{\text{max}} - 1)$ centered at x and consider the collection $\mathbb{B}$ of all such blocks which are entirely contained in X, $\mathbb{B} = \left\{\tilde{B}_x : x \in X,\ \tilde{B}_x \subset X\right\}$. We indicate by $X_{\mathbb{B}} \subset X$ the set of points for which we can construct a block belonging to $\mathbb{B}$, $X_{\mathbb{B}} = \left\{x \in X : \tilde{B}_x \in \mathbb{B}\right\}$. 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. Note that $\delta_{\mathbb{B}}(x)$ is univocally defined and non-zero only for x sufficiently close to the boundary ∂X of X.

For each point $x \in X_{\mathbb{B}}$, we construct the set S_x of the central points of blocks found similar to $\tilde{B}_x$, as by (14).

7.2.2 Adaptive-shape matching

Let now $x_R \in X$ be a "reference" point and define $x_R^\delta = x_R + \delta_{\mathbb{B}}(x_R)$. Using the result $S_{x_R^\delta}$ 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)

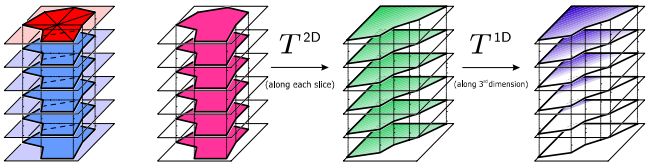

Figure 16: Shape-adaptive grouping and the forward shape-adaptive transform used in the collaborative filtering of the group.

$\widetilde{\mathbb{U}}_{x_R}$ of neighborhoods having the same shape and defined as

$$\widetilde{\mathbb{U}}_{x_R} = \coprod_{x+\delta_{\mathbb{B}}(x_R)\in S_{x_R^\delta}} \tilde{U}^+_{x,x_R}, \tag{16}$$

where $\tilde{U}^+_{x,x_R}$ are mirrored translates of U^+_x of the form

$$\begin{aligned}\tilde{U}^+_{x,x_R} &= \left\{v \in X : (x-v) \in U^+_{x_R}\right\} = \\ &= \left\{v \in X : (x_R - x + v) \in \tilde{U}^+_{x_R}\right\}.\end{aligned}$$

In plain words, $\tilde{U}^+_{x,x_R}$ 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 "reference point" x_R and not those corresponding to x itself. Obviously, $\tilde{U}^+_{x,x} = \tilde{U}^+_x$.

Because of this construction, all neighborhoods in $\widetilde{\mathbb{U}}_{x_R}$ have the same shape, which is completely determined by the adaptive scales $\left\{h^+(x_R,\theta_k)\right\}_{k=1}^8$ at x_R.

7.2.3 Shape-adaptive group

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 $\tilde{U}^+_{x_R} \times \left\{1,\ldots,\left|S_{x_R^\delta}\right|\right\}$, as illustrated in Figure 16(left). In compact form, the group is denoted as $\mathbf{Z}_{x_R} : \widetilde{\mathbb{U}}_{x_R} \to \mathbb{R}$.

7.2.4 Collaborative hard-thresholding

Given a group $\mathbf{Z}_{x_R}$, collaborative filtering is realized as shrinkage in a 3-D transform, T^{3D}, domain. Here, T^{3D} is a composition of 2-D shape-adaptive transform (which is applied on each slice $z_{|\tilde{U}^+_{x,x_R}}$ of the group), with an orthonormal 1-D transform T^{1D} (applied along the third dimension of the group), as shown in Figure 16.

The remainder of the algorithm follows the general lines of the blockwise algorithm. The reader interested in the specific technicalities can refer to [6].

This algorithm represent the latest development for what concerns image denoising. In terms of spatial adaptivity, the shape-adaptive nonlocal supports used here give the algorithm unprecedented flexibility. By comparing Figure 17 with Figure 15, we can see that not only the PSNR of the shape-adaptive nonlocal algorithm is higher than its blockwise counterpart, but also that edges and details are now sharper and better reconstructed.

Figure 17: Denoised *Cameraman* by the Block-Matching and 3D Shape-Adaptive DCT filtering (BM3D-SADCT) algorithm (PSNR=29.56dB).

8. CONCLUSIONS

In this paper we reviewed our recent developments with demonstration of evolution of the ideas and techniques from simple pointwise to more complex nonlocal blockwise and anisotropic estimators. The state-of-the-art performance of the developed algorithms is confirmed by multiple experiments, including comprehensive comparative evaluations by Ghent University [47] and Stanford University [38].

All developed algorithms are available from our website `http://ww.cs.tut.fi/~lasip/`, thus the algorithms can be tested.

Here we would like to mention some of the problems of our special concern for further research.

(1) Generalized model selection problem.

According to the standard setting in mathematical statistics, the model-selection problem concerns with the model comparison, selection, aggregation, etc., with a given support which is the same for all models (e.g., [1],[27]).

Unsolved problem: consider the support as a varying element of the model and consider simultaneous selection of the local parametric model and its support.

(2) Image modelling for nonlocal estimation.

In nonlocal estimation typically we deal with a multistep processing: search for mutually similar blocks, grouping, groupwise processing, aggregation of the multiple estimates. While reasonable models and statistical procedures are developed for each one of these steps, overall the techniques are heuristic, because composed of parts which are not completely fitted to each other. However, the outstanding performance of the developed algorithms gives clear evidence that there is a model, implicitly embedded inside the algorithms, which is meaningful for modeling natural images.

Unsolved problem: development of a statistical observation model leading to a technique where all steps such as grouping, groupwise processing and aggregation appear as a result of some standard statistical technique (ML, EM, etc.). This sought model is interesting not only because it can formally justify the algorithm as a whole, but especially because it can be treated as a good model for natural images.

9. ACKNOWLEDGMENTS

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) and by the Tampere Graduate School in Information Science and Engineering (TISE).

REFERENCES

[1] Barron, A., Rissanen, J., Yu, B., "The Minimum Description Length Principle in Coding and Modeling", (invited paper), *Information Theory 50 Years of Discovery*, (Sergio Verdu, Editor, Steven McLaughlin, Coeditor), IEEE Press, 1998.

[2] Buades, A., B. Coll, and J.M. Morel, "A review of image denoising algorithms, with a new one," *SIAM Multiscale Modeling and Simulation*, vol. **4**, pp. 490–530, 2005.

[3] Buades, A., B. Coll, and J.M. Morel, "Nonlocal image and movie denoising", *Int. J. Computer Vision*, July 2007.

[4] Cleveland, W.S., and C. Loader, "Smoothing by local regression: principles and methods", *Statistical theory and computational aspects of smoothing*, Springer, New York, pp. 10-49, 1996.

[5] Coifman, R.R., and D. Donoho, "Translation-invariant de-noising", in *Wavelets and Statistics* (editors. A. Antoniadis and G. Oppenheim), *Lecture Notes in Statistics*, Springer-Verlag, pp.125-150, 1995.

[6] Dabov, K., A. Foi, V. Katkovnik, and K. Egiazarian, "A Non-Local and Shape-Adaptive Transform-Domain Collaborative Filtering", to appear in *Proc. 2008 Int. Workshop on Local and Non-Local Approximation in Image Processing, LNLA 2008*, Lausanne, Switzerland, August 2008.

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

[8] Elad, M., "Why shrinkage is still relevant for redundant representations?", *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5559-5569, 2006.

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

[10] Ercole, C., A. Foi, V. Katkovnik, and K. Egiazarian, "Spatio-temporal pointwise adaptive denoising of video: 3D non-parametric approach", *Proc. of the 1st International Workshop on Video Processing and Quality Metrics for Consumer Electronics, VPQM2005*, Scottsdale, AZ, January 2005.

[11] Fan, J., and I. Gijbels, *Local polynomial modelling and its application*, Chapman and Hall, London, 1996.

[12] Foi, A., *Anisotropic nonparametric image processing: theory, algorithms and applications,* Ph.D. Thesis, Dip. di Matematica, Politecnico di Milano, ERLTDD-D01290, April 2005.

[13] Foi, A., "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, December 2007.

[14] Foi, A., S. Alenius, M. Trimeche, V. Katkovnik, and K. Egiazarian, "A spatially adaptive Poissonian image deblurring", *IEEE 2005 Int. Conf. Image Processing, ICIP 2005,* September 2005.

[15] Foi, A., R. Bilcu, V. Katkovnik, and K. Egiazarian, "Anisotropic local approximations for pointwise adaptive signal-dependent noise removal", (accepted) *XIII European Signal Proc. Conf., EUSIPCO 2005*, September 2005.

[16] Foi, A., K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-adaptive DCT for denoising and image reconstruction", *Proc. SPIE El. Imaging 2006, Image Process.: Algorithms and Systems V*, 6064A-18, San Jose, California, USA, Jan. 2006.

[17] Foi, A., and V. Katkovnik, "From local polynomial approximation to pointwise shape-adaptive transforms: an evolutionary nonparametric regression perspective", *Proc. 2006 Int. TICSP Workshop Spectral Meth. Multirate Signal Process., SMMSP 2006*, Florence, Sep. 2006.

[18] Foi, A., V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT as an overcomplete denoising tool", *Proc. 2005 Int. TICSP Workshop Spectral Meth. Multirate Signal Process., SMMSP 2005*, pp. 164-170, Riga, June 2005.

[19] Foi, A., 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.

[20] Foi, A., V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT denoising with structure preservation in luminance-chrominance space", *Proc. 2nd Int. Workshop Video Process. Quality Metrics Consum. Electron., VPQM2006*, Scottsdale, AZ, Jan. 2006.

[21] Foi, A., V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT for high-quality deblocking of compressed color images", *Proc. 14th Eur. Signal Process. Conf., EUSIPCO 2006*, Florence, Sep. 2006.

[22] Foi, A., V. Katkovnik, and K. Egiazarian, "Signal-dependent noise removal in Pointwise Shape-Adaptive DCT domain with locally adaptive variance", *Proc. 15th Eur. Signal Process. Conf., EUSIPCO 2007*, Poznań, Sep. 2007.

[23] Foi, A., V. Katkovnik, K. Egiazarian, and J. Astola, "A novel anisotropic local polynomial estimator based on directional multiscale optimizations", *Proc. 6th IMA Int. Conf. Math. in Signal Processing*, Cirencester (UK), pp. 79-82, 2004.

[24] Foi, A., V. Katkovnik, K. Egiazarian, and J. Astola, "Inverse halftoning based on the anisotropic LPA-ICI deconvolution", *Proc. Int. TICSP Workshop Spectral Meth. Multirate Signal Proc., SMMSP 2004,* Vienna, pp. 49-56, Sep. 2004.

[25] Goldenshluger, A., and A. Nemirovski, "On spatial adaptive estimation of nonparametric regression", *Math. Meth. Statistics*, vol. 6, pp. 135-170, 1997.

[26] Guleryuz, O., "Weighted Averaging for Denoising with Overcomplete Dictionaries", *IEEE Trans. Image Processing*, vol. 16, no. 12, pp. 3020-3034, 2007.

[27] Hansen, A.J., and B. Yu, "Model Selection and the Principle of Minimum Description Length", *J. of the American Statistical Association*, vol. 96, pp. 746-774, 2001.

[28] Hart, J.D., *Nonparametric smoothing and lack-of-fit tests*, Springer-Verlag, New York, 1997.

[29] Katkovnik, V., "A new method for varying adaptive bandwidth selection", *IEEE Trans. on Signal Proc.,* vol. 47, no. 9, pp. 2567-2571, 1999.

[30] Katkovnik, V., K. Egiazarian, J. Astola, *Local Approximation Techniques in Signal and Image Processing*, SPIE PRESS, Bellingham, Washington, 2006.

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

[32] Katkovnik, V., A. Foi, K. Egiazarian, and J. Astola, "Directional varying scale approximations for anisotropic signal processing", *Proc. XII European Signal Proc. Conf., EUSIPCO 2004,* Vienna, pp. 101-104, September 2004.

[33] Katkovnik, V., A. Foi, K. Egiazarian, and J. Astola, "Anisotropic local likelihood approximations", *Proc. of Electronic Imaging 2005*, 5672-19, January 2005.

[34] Kauff, P., and K. Schuur, "Shape-adaptive DCT with block-based DC separation and ΔDC correction", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 3, pp. 237-242, 1998.

[35] Kervrann, C., and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation", *Research Report INRIA*, RR-5624, July 2005.

[36] Kervrann, C., and J. Boulanger, "Unsupervised Patch-Based Image Regularization and Representation", ECCV 2006, Part IV, LNCS 3954, pp. 555-567, 2006.

[37] Kervrann, C., and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image regularization and representation", *Int. J. Computer Vision*, vol. 79, no. 1, pp. 45-69, August 2008.

[38] Lansel, S., D. Donoho, and T. Weissman, "DenoiseLab: a standard test set and evaluation method to compare denoising algorithms", `http://www.stanford.edu/~slansel/DenoiseLab/`.

[39] Lepski, O., E. Mammen and V. Spokoiny, "Ideal spatial adaptation to inhomogeneous smoothness: an approach based on kernel estimates with variable bandwidth selection". *Annals of Statistics*, vol. 25, no. 3, 929–947, 1997.

[40] Loader, C., *Local regression and likelihood*, Series Statistics and Computing, Springer-Verlag, New York, 1999.

[41] Mallat, S., *A wavelet tour of signal processing*, Academic Press, 1999.

[42] Polzehl, J., and V. Spokoiny, "Image denoising: pointwise adaptive approach," *The Annals of Statistics*, vol. 31, no. 1, pp. 30-57, 2003.

[43] Rudin, L, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D,* 60 2, pp. 259-268, 1993.

[44] Rudin, L, and S. Osher, "Total variation based image restoration with free local constraints", in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, pp. 31-35, Austin, TX, USA, 1994.

[45] Sikora, T., "Low complexity shape-adaptive DCT for coding of arbitrarily shaped image segments", *Signal Processing: Image Communication*, vol. 7, pp. 381-395, 1995.

[46] Sikora, T., and B. Makai, "Shape-adaptive DCT for generic coding of video", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 5, no. 1, pp. 59-62, 1995.

[47] Vansteenkiste, E., D. Van der Weken, W. Philips, and E. Kerre, "Perceived image quality measurement of state-of-the-art noise reduction schemes", *LNCS 4179 - ACIVS 2006*, pp. 114-124, Springer, Sept. 2006.

[48] Vanzella, W., F.A. Pellegrino, V. Torre, "Self-adaptive regularization", *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol. 26, no. 6, pp. 804-809, 2004.