SIAM J. IMAGING SCIENCES
Vol. 6, No. 1, pp. 263–284

Symmetrizing Smoothing Filters*

Peyman Milanfar[†]

Abstract. We study a general class of nonlinear and shift-varying smoothing filters that operate based on averaging. This important class of filters includes many well-known examples such as the bilateral filter, nonlocal means, general adaptive moving average filters, and more. (Many linear filters such as linear minimum mean-squared error smoothing filters, Savitzky–Golay filters, smoothing splines, and wavelet smoothers can be considered special cases.) They are frequently used in both signal and image processing as they are elegant, computationally simple, and high performing. The operators that implement such filters, however, are not symmetric in general. The main contribution of this paper is to provide a provably stable method for symmetrizing the smoothing operators. Specifically, we propose a novel approximation of smoothing operators by symmetric *doubly* stochastic matrices and show that this approximation is stable and accurate, even more so in higher dimensions. We demonstrate that there are several important advantages to this symmetrization, particularly in image processing/filtering applications such as denoising. In particular, (1) doubly stochastic filters generally lead to improved performance over the baseline smoothing procedure; (2) when the filters are applied iteratively, the symmetric ones can be guaranteed to lead to stable algorithms; and (3) symmetric smoothers allow an orthonormal eigendecomposition which enables us to peer into the complex behavior of such nonlinear and shift-varying filters in a locally adapted basis using principal components. Finally, a doubly stochastic filter has a simple and intuitive interpretation. Namely, it implies the very natural property that every pixel in the given input image has the same sum total contribution to the output image.

Key words. nonparametric regression, data smoothing, filtering, stochastic matrices, applications of Markov chains, positive matrices, Laplacian operator, applications of graph theory

AMS subject classifications. 62G08, 93E14, 93E11, 15B51, 60J20, 15B48, 35J05, 05C90

DOI. 10.1137/120875843

1. Introduction.

Given an $n \times 1$ data vector $\mathbf{y}$, a smoothing filter replaces each element of $\mathbf{y}$ by a normalized weighted combination of its elements. That is,

$$\widehat{\mathbf{y}} = \mathbf{A}\mathbf{y}, \tag{1.1}$$

where $\mathbf{A}$ is an $n \times n$ nonnegative matrix. While the analysis that follows can be cast for general nonnegative $\mathbf{A}$, we focus on the cases where $\mathbf{A}$ is constructed so that its rows sum to one. The corresponding matrices are called (row-)*stochastic* matrices. These filters are commonly used in signal processing applications because they keep the mean value of the signal unchanged. In particular, moving least squares averaging filters [38], the bilateral filter [51], and the nonlocal means filter [7] are all special cases. For their part, stochastic matrices find numerous applications in statistical signal processing, including in classical optimal filtering,

*Received by the editors May 3, 2012; accepted for publication (in revised form) October 2, 2012; published electronically February 12, 2013. This work was supported in part by AFOSR grant FA9550-07-1-0365 and NSF grant CCF-1016018.

http://www.siam.org/journals/siims/6-1/87584.html

[†]Google, Inc., Mountain View, CA, and University of California, Santa Cruz, CA 95064 (milanfar@ucsc.edu).

image denoising [11], Markov chain theory [44, 49], distributed processing [20], and many others.

While the smoothing operator in (1.1) has a linear appearance, the $\mathbf{A}$ we consider can in fact depend on the given data samples $\mathbf{y}$ and the locations $\mathbf{x}$ of these samples. Therefore, these filters are generally neither linear nor shift-invariant. As such, the standard Fourier transform results we are accustomed to for spectral analysis in an orthogonal basis do not apply, and our understanding of these filters has consequently been limited to their behavior in only their original sample space of definition (time domain, pixel domain, etc.). Understanding the spectral behavior of these filters in an orthogonal basis is important not only for better intuition about their properties, but also for analyzing their statistical performance [39]. This latter issue has become of great practical importance recently since many competing state-of-the-art smoothing algorithms invented in the last few years appear to display comparable performance, prompting many to wonder whether we have reached a limit on the performance of such filters[1] for the denoising application [11, 12, 37].

The fundamental technical roadblock in the spectral analysis of smoothing filters is that in general $\mathbf{A}$ is not symmetric or circulant. With a symmetric $\mathbf{A}$, its eigendecomposition would reveal the structure of the filter in the spectral sense, whereas in the latter case, the Fourier basis would diagonalize $\mathbf{A}$ and reveal the frequency domain filter behavior. Unfortunately, neither of these tools is directly applicable here.

The general construction of smoothing filters begins by specifying a (symmetric positive semidefinite) kernel $k_{ij} = K(y_i, y_j) \geq 0$ from which $\mathbf{A}$ is constructed.[2] More specifically,

$$a_{ij} = \frac{k_{ij}}{\sum_{i=1}^{n} k_{ij}}.$$

Each element of the smoothed signal $\widehat{\mathbf{y}}$ is then given by

$$\widehat{y}_j = \sum_{i=1}^{n} a_{ij}\; y_i,$$

where $[a_{1j},\ \ldots,\ a_{nj}]$ is the jth row of $\mathbf{A}$ whose elements sum to one:

$$\sum_{i=1}^{n} a_{ij} = 1.$$

It should be apparent that regardless of whether k_{ij} are symmetric or not, a_{ij} will generally not be so because the normalizing coefficients are not uniform. In matrix language, $\mathbf{A}$ can be written as a product

$$\mathbf{A} = \mathbf{D}^{-1}\mathbf{K}, \tag{1.2}$$

[1] The answer turns out to be no!

[2] In practice, the kernels we consider vary smoothly with the underlying (clean) signal, and furthermore it is commonplace to compute the kernel not on the original noisy $\mathbf{y}$ but on a "prefiltered" version of it with the intent to weaken the dependence of $\mathbf{A}$ on noise. More details on this point are provided in Appendix A.

where $\mathbf{D}$ is a nontrivial diagonal matrix with diagonal elements $[\mathbf{D}]_{jj} = \sum_{i=1}^{n} k_{ij}$. We again observe that even if $\mathbf{K}$ is symmetric, the resulting (row-)stochastic matrix $\mathbf{A}$ will generally not be symmetric due to the multiplication on the left by $\mathbf{D}^{-1}$. But is it the case that $\mathbf{A}$ must in general be "close" to symmetric? This paper answers this question in the affirmative and provides a constructive method for approximating a smoothing matrix $\mathbf{A}$ by a symmetric doubly stochastic matrix $\widehat{\mathbf{A}}$.

It is worth noting that the process of symmetrization can in general be carried out directly on any nonnegative smoothing matrix regardless of whether it is row-stochastic or not.[3] In the particular case of (1.2), the process will yield [39] the very same result regardless of whether we symmetrize $\mathbf{K}$ or $\mathbf{A} = \mathbf{D}^{-1}\mathbf{K}$.

We summarize the main goals of this paper:

- We propose a novel approximation of nonlinear smoothing operators by *doubly* stochastic matrices and show that this approximation is stable and accurate.
- We demonstrate the advantages to this symmetrization; namely,
  - we show that symmetrization leads to improved performance of the baseline smoother;
  - we use the symmetrization to derive an orthogonal basis of principal components that allows us to peer into the complex nature of nonlinear and shift-varying filters and their performance.

1.1. Some background.

Before we move to the details, here is a brief summary of relevant earlier work. In the context of expressing a nonlinear filtering scheme in an orthogonal basis, Coifman et al. [14] proposed the construction of diffusion maps and used eigenfunctions of the Laplacian on the manifold of patches derived from an image. Peyré provided an interesting spectral analysis of the graph Laplacian for nonlocal means and bilateral kernels in [43]. This paper also discussed symmetrization of the operator, but rather a different one carried out elementwise that does not preserve stochasticity. Furthermore, Peyré used a nonlinear thresholding procedure on the eigenvalues for denoising, and analyzed numerically the performance on some example by looking at the nonlinear approximation error in the eigenbases. We note that both of the above methods worked with a graph structure and therefore its Laplacian, whereas we work directly with the smoothing matrix. The relationship between the two has been clarified in several places, including recently in [39]. Namely, the Laplacian $\mathcal{L} = \mathbf{D}^{1/2}\,\mathbf{A}\,\mathbf{D}^{-1/2} - \mathbf{I}$. Therefore, the analysis we present here is directly relevant to the study of the spectrum of the Laplacian operator as well. Meanwhile, consistent with our analysis, Kindermann, Osher, and Jones [35] have proposed directly symmetrizing the nonlocal means or bilateral kernel matrices. But they too do not insist on maintaining the stochastic nature of the smoothing operator. Hence, our approach to making the smoothing operator doubly stochastic is new and different from the previous similar attempts. Finally, we note that the type of normalization we promote would likely have some impact in other areas of work well beyond the current filtering context, such as scale-space meshing in computer graphics [19] and in machine learning [3].

[3] How to carry out the symmetrization and whether it is useful in the case where $\mathbf{A}$ contains negative elements remain an interesting open problem.

As we mentioned earlier, many popular filters are contained in the class of smoothing operators we consider. To be more specific, we highlight a few such kernels which lead to smoothing matrices **A** which are *not* symmetric. These are commonly used in the signal and image processing, computer vision, and graphics literature for many purposes.

1.1.1. Classical Gaussian filters [53, 26, 55].

Measuring the Euclidean (spatial) distance between samples, the classical Gaussian kernel is

$$k_{ij} = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{h^2}\right).$$

Such kernels lead to the classical and well-worn Gaussian filters (including shift-varying versions [18]).

1.1.2. The bilateral filter [51, 21].

This filter takes into account both the spatial *and* datawise distances between two samples, in separable fashion, as follows:

$$k_{ij} = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{h_x^2}\right) \exp\left(\frac{-(y_i - y_j)^2}{h_y^2}\right) = \exp\left\{\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{h_x^2} + \frac{-(y_i - y_j)^2}{h_y^2}\right\}. \tag{1.3}$$

As can be observed in the exponent on the right-hand side, the similarity metric here is a weighted Euclidean distance between the vectors $(\mathbf{x}_i, y_i)$ and $(\mathbf{x}_j, y_j)$. This approach has several advantages. Namely, while the kernel is easy to construct, and computationally simple to calculate, it yields useful local adaptivity to the given data.

1.1.3. Nonlocal means [7, 33, 2].

The nonlocal means algorithm, originally proposed in [7] and [2], is a generalization of the bilateral filter in which the data-dependent distance term (1.3) is measured blockwise instead of pointwise:

$$k_{ij} = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{h_x^2}\right) \exp\left(\frac{-\|\mathbf{y}_i - \mathbf{y}_j\|^2}{h_y^2}\right), \tag{1.4}$$

where $\mathbf{y}_i$ and $\mathbf{y}_j$ refer now to *subsets* of samples (patches) in $\mathbf{y}$.

1.1.4. Locally adaptive regression kernel (LARK) [50].

The key idea behind this kernel is to robustly measure the local structure of data by making use of an estimate of the local *geodesic* distance between nearby samples:

$$k_{ij} = \exp\left\{-(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{Q}_{ij} (\mathbf{x}_i - \mathbf{x}_j)\right\}, \tag{1.5}$$

where $\mathbf{Q}_{ij} = \mathbf{Q}(y_i, y_j)$ is the covariance matrix of the gradient of sample values estimated from the given data [50], yielding an approximation of local geodesic distance in the exponent of the kernel. The dependence of $\mathbf{Q}_{ij}$ on the given data means that the smoothing matrix $\mathbf{A} = \mathbf{D}^{-1}\mathbf{K}$ is therefore nonlinear and shift-varying. This kernel is closely related but somewhat more general than the Beltrami kernel of [48] and the coherence enhancing diffusion approach of [54].

2. Nearness of stochastic and doubly stochastic matrices.

Our interest in this paper is to convert a smoothing operator $\mathbf{A}$ which is generically not symmetric into a symmetric doubly stochastic one. As we shall see, this is done quite easily using an iterative process. When a (row-)stochastic matrix is made symmetric, it must therefore have columns that sum to one as well. The class of nonnegative matrices whose rows *and* columns both sum to one are called *doubly* stochastic. Our aim is to show that when applied to a (row-)stochastic smoother $\mathbf{A}$, this process yields a nearby matrix $\widehat{\mathbf{A}}$ that has both its elements and its eigenvalues close to the original. This is the subject of this section. We note that classical results in this direction have been available since the 1970s. Of these, the work of Darroch and Ratcliff [17] and Csiszar [15] involving relative entropy is particularly noteworthy. Here, we prove that the set of $n \times n$ (row-)stochastic matrices and the corresponding set of doubly stochastic matrices are asymptotically close in the mean-squared error (MSE) sense.

Let $\mathcal{S}_n$ denote the set of $n \times n$ stochastic matrices with nonnegative entries, and define $\mathbf{1}$ as the $n \times 1$ vector of ones. By definition, any $\mathbf{A} \in \mathcal{S}_n$ satisfies

$$\mathbf{A}\,\mathbf{1} = \mathbf{1}. \tag{2.1}$$

The Perron–Frobenius theory of nonnegative matrices (cf. [44], [30, page 498]) provides a comprehensive characterization of their spectrum. Denoting the eigenvalues $\{\lambda_i\}_{i=1}^{n}$ in descending order, we have the following:[4]

1. $\lambda_1 = 1$ is the unique eigenvalue of $\mathbf{A}$ with maximum modulus.
2. λ_1 corresponds to positive right and left eigenvectors $\mathbf{v}_1$ and $\mathbf{u}_1$, where

$$\mathbf{v}_1 = \frac{1}{\sqrt{n}}\mathbf{1}, \tag{2.2}$$

$$\mathbf{A}^T\mathbf{u}_1 = \mathbf{u}_1, \tag{2.3}$$

$$(\mathit{ergodicity}) \quad \lim_{k\to\infty} \mathbf{A}^k = \mathbf{1}\,\mathbf{u}_1^T, \tag{2.4}$$

$$\|\mathbf{u}_1\|_1 = 1. \tag{2.5}$$

3. The *subdominant* eigenvalue λ_2 determines the ergodic rate of convergence. In particular, we have, *elementwise*,

$$\mathbf{A}^k = \mathbf{1}\,\mathbf{u}_1^T + \mathrm{O}(\lambda_2^k). \tag{2.6}$$

Using these properties, we prove a rather general result in the following lemma. For the proof, we refer the reader to Appendix B.

Lemma 2.1. *Denote the set of $n \times n$* doubly *stochastic matrices by $\mathcal{D}_n$. Any two matrices $\mathbf{A} \in \mathcal{S}_n$ and $\widehat{\mathbf{A}} \in \mathcal{D}_n$ satisfy*

$$\frac{1}{n}\|\mathbf{A}^k - \widehat{\mathbf{A}}^k\|_F \le c\,\lambda_2^k + \widehat{c}\,\widehat{\lambda}_2^k + \mathrm{O}(n^{-1/2}) \tag{2.7}$$

for all nonnegative integers k.

[4] Since we consider only positive semidefinite kernels k_{ij}, the eigenvalues are nonnegative and real throughout.

This result is a bound on the root mean-squared (RMS) difference between the elements of the respective matrices. It is quite general in the sense that the matrices are (powers of) *any* pair of nonnegative $n \times n$ stochastic and doubly stochastic matrices. This bound becomes tighter with n, and we note that since the subdominant eigenvalues of both $\mathbf{A}$ and $\widehat{\mathbf{A}}$ are strictly less than 1, the first two terms on the right-hand side of (2.7) also collapse to zero with increasing k. This result by itself shows that the set of *doubly* stochastic matrices $\mathcal{D}_n$ and the set of ordinary (row-)stochastic matrices $\mathcal{S}_n$ are close. But even more compelling is what happens when $\mathbf{A}$ is random [10, 5, 24, 25, 31]. In particular, it is known [25] that if the entries are drawn at random from a distribution on $[0, 1]$ such that $\mathbf{E}(\mathbf{A}_{i,j}) = 1/n$ and $\mathbf{Var}(\mathbf{A}_{i,j}) \leq c_1/n^2$, then the subdominant eigenvalue tends, in probability, to zero as $n \to \infty$ at a rate of $1/\sqrt{n}$. In fact, the same behavior occurs when only the rows are independent[5] [24].

We are less interested, however, in arbitrary elements of $\mathcal{D}_n$ and $\mathcal{S}_n$. Instead, it is more relevant for our purposes to consider a stochastic matrix $\mathbf{A}$ and to seek the nearest doubly stochastic matrix to it. One would expect that the general bound above would be even more informative when $\widehat{\mathbf{A}}$ is an explicit symmetric approximation of $\mathbf{A}$, and this is indeed the case.

The interesting practical question of how to find this nearest element was addressed by Sinkhorn and colleagues (see [36, 46, 47]). Specifically, a stochastic matrix (indeed any nonnegative matrix with positive diagonals) can be *scaled* to a doubly stochastic matrix via a procedure sometimes known as *iterative proportional scaling*, or *Sinkhorn balancing*. Sinkhorn and Knopp proved the following general result.

Theorem 2.2 (see [46, 47]). *Let $\mathbf{A}$ be a nonnegative matrix with total support. Then, there exist positive diagonal matrices $\mathbf{R} = \mathrm{diag}(\mathbf{r})$ and $\mathbf{C} = \mathrm{diag}(\mathbf{c})$ such that*

$$\widehat{\mathbf{A}} = \mathbf{R}\,\mathbf{A}\,\mathbf{C} \tag{2.8}$$

is doubly stochastic. The matrix $\widehat{\mathbf{A}}$ is unique, and the vectors $\mathbf{r}$ and $\mathbf{c}$ are unique up to a scalar factor, as in $\mu\mathbf{r}$ and $\mathbf{c}/\mu$ if and only if $\mathbf{A}$ is fully indecomposable.[6]

It is worth noting that for our purpose, the smoothing matrix $\mathbf{A} = \mathbf{D}^{-1}\mathbf{K}$ has strictly positive diagonal elements $a_{ii} = k_{ii}/d_{ii} \;\; > 0$, and generically satisfies the conditions of the above theorem. With this in mind, the actual algorithm for computing $\widehat{\mathbf{A}}$ is quite simple and involves only repeated normalization of the rows and columns of $\mathbf{A}$. We state the procedure in Algorithm 1 using MATLAB notation. The convergence of this iterative algorithm is known to be linear [36], with the rate given by the subdominant eigenvalue λ_2.

How good is $\widehat{\mathbf{A}}$ as an approximation to $\mathbf{A}$? Somewhat surprisingly, the matrices $\mathbf{A}$ and $\widehat{\mathbf{A}}$ related as in (2.8) are indeed optimally close in the relative entropy sense, as made precise by Darroch and Ratcliff [17] and Csiszar [15]. Namely, $\widehat{\mathbf{A}}$ minimizes the cross-entropy or Kullback–Leibler (KL) measure,

$$\sum_{i,j} \widehat{\mathbf{A}}_{ij} \log \frac{\widehat{\mathbf{A}}_{ij}}{\mathbf{A}_{ij}},$$

[5] This holds so long as the covariance also satisfies $\mathbf{Cov}(\mathbf{A}_{i,k}, \mathbf{A}_{j,k}) \leq c_2/n^3$.

[6] A matrix $\mathbf{A}$ is said to have total support if every positive entry in $\mathbf{A}$ can be (column) permuted into a positive diagonal matrix. A nonnegative matrix $\mathbf{A}$ is said to be fully indecomposable if there do not exist permutation matrices $\mathbf{P}$ and $\mathbf{Q}$ such that $\mathbf{PAQ}$ is of the form $\left[\begin{smallmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \bigcirc & \mathbf{A}_{22} \end{smallmatrix}\right]$ with $\mathbf{A}_{11}$ and $\mathbf{A}_{22}$ being square matrices. A fully indecomposable matrix has total support [6].

Algorithm 1. Algorithm for scaling a nonnegative matrix $\mathbf{A}$ to a nearby doubly stochastic matrix $\widehat{\mathbf{A}}$.

```
Given $\mathbf{A}$, let $(n, n) = \text{size}(\mathbf{A})$ and initialize $\mathbf{r} = \text{ones}(n, 1)$;
for $k = 1 : iter$;
    $\mathbf{c} = 1./(\mathbf{A}^T\,\mathbf{r})$;
    $\mathbf{r} = 1./(\mathbf{A}\,\mathbf{c})$;
end
$\mathbf{C} = \text{diag}(\mathbf{c})$; $\mathbf{R} = \text{diag}(\mathbf{r})$;
$\widehat{\mathbf{A}} = \mathbf{R}\,\mathbf{A}\,\mathbf{C}$
```

over all $\widehat{\mathbf{A}} \in \mathcal{D}_n$.

When the starting kernel k_{ij} is positive definite, the scaling procedure, which involves left and right multiplication by positive diagonal matrices, preserves this property and results in a positive definite, symmetric, and doubly stochastic $\widehat{\mathbf{A}}$. It is reasonable to ask why KL is useful as a measure of closeness to get a good nearby normalization, particularly for signal processing applications. One answer is that it is of course a natural metric to impose nonnegativity in the approximation, or, more precisely, to maintain the connectivity of the graph structure implied by the data. But could other distances or divergences replace the KL measure used here to do a similar or even better job? The evidence says no. In fact, other norms such as L_1 and L_2 for this approximation are more common in the machine learning literature (e.g., [56]). Conceptually, the L_2 projection would not seem to be a very good choice as it would likely push many entries to zero, which may not be desirable. We have observed experimentally that the use of either of these norms leads to quite severe perturbations of the eigenvalues of the smoothing matrix. Hence, we believe that the KL distance is indeed the most appropriate for the task.

Next, we study how the proposed diagonal scalings will perturb the eigenvalues of $\mathbf{A}$. It is not necessarily the case that a small perturbation of a matrix will give a small perturbation of its eigenvalues. The stability of eigenvalues of a matrix is in fact a strong function of the condition number of the matrix of its eigenvectors (cf. the Bauer–Fike theorem [4]). In our filtering framework, it is important to verify that the eigenvalues of the symmetrized matrix $\widehat{\mathbf{A}}$ are very near those of $\mathbf{A}$, because the spectrum determines the effect of the filter on the data and ultimately influences its statistical performance. The following result is applicable to bound the perturbation of the spectra.

Theorem 2.3 (see [4, Chap. 12]). *Let $\mathbf{A}$ and $\widehat{\mathbf{A}}$ be $n \times n$ matrices with ordered eigenvalues $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$ and $|\widehat{\lambda}_1| \geq |\widehat{\lambda}_2| \geq \cdots \geq |\widehat{\lambda}_n|$, respectively. If $\widehat{\mathbf{A}} = \mathbf{RAC}$ is symmetric, then*

$$\sum_{i=1}^{n} |\lambda_i - \widehat{\lambda}_i|^2 \leq 2\,\|\mathbf{A} - \widehat{\mathbf{A}}\|_F^2. \tag{2.9}$$

Now we can combine this result with Lemma 2.1 and arrive at the following.

Theorem 2.4. *Let* $\mathbf{A} \in \mathcal{S}_n$ *and the corresponding scaled matrix* $\widehat{\mathbf{A}} = \mathbf{RAC} \in \mathcal{D}_n$. *Then,*

$$\frac{1}{\sqrt{2}n}\left(\sum_{i=1}^{n} |\lambda_i - \widehat{\lambda}_i|^2\right)^{1/2} \leq \frac{1}{n}\|\mathbf{A} - \widehat{\mathbf{A}}\|_F \leq c_0\lambda_2 + c_1\widehat{\lambda}_2 + \mathrm{O}(n^{-1/2}). \tag{2.10}$$

The general conclusion here is that the scaled perturbation of the eigenvalues and the RMS variation in the elements of the smoothing matrix $\mathbf{A}$ are bounded from above.[7] We observe that the bound is composed of two terms that decay with dimension. Namely, as shown in [24], with random rows[8] with elements selected from some density on $[0, 1]$, the term $c_0\lambda_2 + c_1\widehat{\lambda}_2$ decays as $1/\sqrt{n}$. To assess how large this upper bound is for a given n, we would need to have an estimate of the coefficients c_0 and c_1. At this time, we do not have such an estimate in analytical form. But as we illustrate in the next section, experimental evidence suggests that they are well below 1.

3. The benefits of symmetrization.

Symmetrizing the smoothing operator is not just a mathematical nicety; it can have interesting practical advantages as well. In particular, three such advantages are that (1) given a smoother, its symmetrized version generally results in improved performance; (2) symmetrizing guarantees the stability of iterative filters based on the smoother; and (3) symmetrization enables us to peer into the complex behavior of smoothing filters in the transform domain using principal components. In what follows, we analyze these aspects theoretically, and while the results are valid for the general class of kernels described earlier, we choose to illustrate the practical effects of symmetrization using the LARK smoother [50] introduced earlier.

3.1. Performance improvement.

First, it is worth recalling an important result about the optimality of smoothers. In [13], Cohen proved that *asymmetric* smoothers are *inadmissible* with respect to the MSE measure. This means that for any linear smoother $\mathbf{A}$ there always exists a symmetric smoother $\widehat{\mathbf{A}}$ which outperforms it in the MSE sense. This result does not directly imply the same conclusion for *nonlinear* smoothers, but, considering at least the oracle filter where $\mathbf{A}$ depends only on the clean signal, it is an indication that improvement can be expected. More realistically, in the practical nonlinear case where (1) $\mathbf{A}$ is based on a sufficiently smooth kernel, and (2) the noise is weak,[9] improvement can be expected. This is in fact the case in practical scenarios because (1) the general class of kernels we employ, at least in signal and image processing, use the Gaussian form exemplified in section 1.1 which is a smooth function of its argument; and (2) the kernels are typically computed on some "prefiltered" version of the measured noisy data so that as far as the calculation of the kernel is concerned, the variance of the noise can be considered small. Meanwhile, we can show that from a purely algebraic point of view, the improvement results from the particular way in

[7] These results on the surface seem to contradict the earlier statement regarding the use of KL distance as a measure of distance between matrices. What is happening is that even though the Frobenius norm gives a bound on the eigenvalue perturbation, this bound is not very tight. Indeed, we observe that in practice, the eigenvalue error is actually much smaller than what the bound would imply. Unfortunately, we could not prove a stronger result at this point.

[8] Given that the weights are computed on pixels corrupted by some noise, this applies to our case.

[9] See Appendix A for details.

which Sinkhorn's diagonal scaling perturbs the eigenvalues. The following result is the first step in establishing this fact.

Theorem 3.1 (see [32]). *If* $\mathbf{A}$ *is row-stochastic and* $\widehat{\mathbf{A}} = \mathbf{RAC}$ *is doubly stochastic, then*

$$\mathbf{det}(\mathbf{RC}) \geq 1.$$

Furthermore, equality holds if and only if $\mathbf{A}$ *is actually doubly stochastic.*

It follows as a corollary that

$$\mathbf{det}(\mathbf{A}) \leq \mathbf{det}(\widehat{\mathbf{A}}), \tag{3.1}$$

or said another way, there exists a constant $0 < \alpha \leq 1$ such that $\mathbf{det}(\mathbf{A}) = \alpha\ \mathbf{det}(\widehat{\mathbf{A}})$. How is this related to the question of performance for the symmetrized versus unsymmetrized smoothers? The size of α is an indicator of how much difference there is between the two smoothers, and we use the above insight about the determinants to make a more direct observation about the bias-variance tradeoff of the respective filters. To do this, we first establish the following relationship between the trace and the determinant of the respective matrices. The proof is again given in Appendix B.

Theorem 3.2. *For any two matrices* $\mathbf{A}$ *and* $\widehat{\mathbf{A}}$ *with real eigenvalues in* $(0, 1]$*, if* $\mathbf{det}(\mathbf{A}) \leq \mathbf{det}(\widehat{\mathbf{A}})$*, then* $\mathbf{tr}(\mathbf{A}) \leq \mathbf{tr}(\widehat{\mathbf{A}})$*.*

The trace of a smoother is related to its *effective degrees of freedom*. The degrees of freedom of an estimator $\widehat{\mathbf{y}}$ of $\mathbf{y}$ measure the overall sensitivity of the estimated values with respect to the measured values [29] as follows:

$$\mathbf{df} = \sum_{i=1}^{n} \frac{\partial \widehat{y}_i}{\partial y_i}. \tag{3.2}$$

The quantity $\mathbf{df}$ is an indicator of how the smoother trades bias against variance. Recalling that the mean-squared error can be written as $\mathbf{MSE} = \|\text{bias}\|^2 + \mathbf{var}(\widehat{\mathbf{y}})$, larger $\mathbf{df}$ implies a "rougher" estimate, i.e., one with higher variance but smaller bias. In our case, the process of symmetrization produces an estimate that is indeed rougher, in proportion to how far the row-stochastic $\mathbf{A}$ is from being doubly stochastic. As we shall see, with nearly all high-performance denoising algorithms, particularly at moderate noise levels, the major problem is that they produce artifacts which are due to the bias of the smoother. As a result of symmetrization, this bias is reduced, though at the expense of a modest increase in variance, but ultimately leading to improved MSE performance.

Consider our smoothing framework $\widehat{\mathbf{y}} = \mathbf{Ay}$ again, and take the "oracle" scenario where $\mathbf{A}$ can be a function of $\mathbf{z}$, but we assume that it is not disturbed by noise. We denote the degrees of freedom of each smoother by $\mathbf{df} = \mathbf{tr}(\mathbf{A})$ and $\mathbf{df}_s = \mathbf{tr}(\widehat{\mathbf{A}})$, respectively. Taken together, Theorem 3.2 and the fact that symmetrizing the smoother increases the determinant of the smoothing operator imply an increase in the degrees of freedom; that is, $\mathbf{df} \leq \mathbf{df}_s$. We can estimate the size of this increase in relation to the size of $\alpha = \mathbf{det}(\mathbf{RC})^{-1}$. Let us write $\mathbf{df} = \beta\ \mathbf{df}_s$ with $0 < \beta \leq 1$ so that a small β indicates a significant increase in the degrees of freedom as a result of symmetrization. Now consider the ratio of the geometric to arithmetic means of the eigenvalues

$$\frac{\left(\prod_{i=1}^{n}\lambda_i\right)^{1/n}}{\left(\frac{1}{n}\sum_{i=1}^{n}\lambda_i\right)} = \left(\frac{\alpha^{1/n}}{\beta}\right)\frac{\left(\prod_{i=1}^{n}\widehat{\lambda}_i\right)^{1/n}}{\frac{1}{n}\sum_{i=1}^{n}\widehat{\lambda}_i}. \tag{3.3}$$

It is an interesting, and little known, fact that asymptotically with large n, the ratio of the geometric to the arithmetic mean for *any* random sequence of numbers in $(0,1]$ converges to the same constant[10] with high probability [1, 23]. Therefore asymptotically, $\beta = \mathrm{O}(\alpha^{1/n})$. This indicates that when the row-stochastic $\mathbf{A}$ is nearly symmetric already (i.e., $\alpha \approx 1$ and therefore $\beta \approx 1$), the gain is small. Next we provide some examples of the observed improvement.

Consider the image patches (each of size 21×21) shown in Figure 1 (top), and denote each, scanned columnwise into vector form, by $\mathbf{z}$. We corrupt these images with white Gaussian noise of variance $\sigma^2 = 100$ to get the noisy images $\mathbf{y}$ shown in the second row. For each patch, we compute the (oracle) LARK kernel [50], leading to the smoothing matrix $\mathbf{A}$. Next, we compute the doubly stochastic matrix $\widehat{\mathbf{A}}$ using Algorithm 1 described earlier. The respective smoothed estimates $\widehat{\mathbf{y}} = \mathbf{A}\mathbf{y}$ and $\widehat{\mathbf{y}}_s = \widehat{\mathbf{A}}\mathbf{y}$ are calculated, leading to the MSE values $\|\mathbf{z}-\widehat{\mathbf{y}}\|^2/n$ and $\|\mathbf{z}-\widehat{\mathbf{y}}_s\|^2/n$. These values, along with the percent improvement in the MSE, the effective degrees of freedom $\mathbf{df}_s$ of the symmetric smoother, and the corresponding values of β defined earlier, are shown[11] in Table 1. As expected, the values of β farther away from one result in the largest improvements in the MSE.

3.2. Stability of iterated smoothing.

The general class of smoothers for which $\|\mathbf{A}\mathbf{y}\| \leq \|\mathbf{y}\|$ are called *shrinking*[12] smoothers [9, 29]. This happens when all the singular values of $\mathbf{A}$ are bounded above by 1. This may seem like a minor issue at first, but it turns out to have important consequences when it comes to something we do routinely to improve the performance of some smoothers: iteration. Indeed, in some cases, iterated application of smoothers depends on whether the procedure is shrinking. In general, before symmetrization, the kernel-based smoothing filters are *not* shrinking. As an example, the largest singular values of $\mathbf{A}$ for the LARK filters are shown in Table 2.

With symmetrization, since the eigenvalues and singular values of $\widehat{\mathbf{A}}$ now coincide, the largest singular value must be equal to 1. Hence $\widehat{\mathbf{A}}$ is, in fact, guaranteed to be a shrinking smoother.

There are numerous ways in which iterative application of smoothers comes into play. One of the most useful and widely studied is *boosting*, also known as twicing [52], L_2-boosting [8], reaction-diffusion [40], and Bregman iteration [42]. We studied this approach in detail in [39]. The iteration is given by

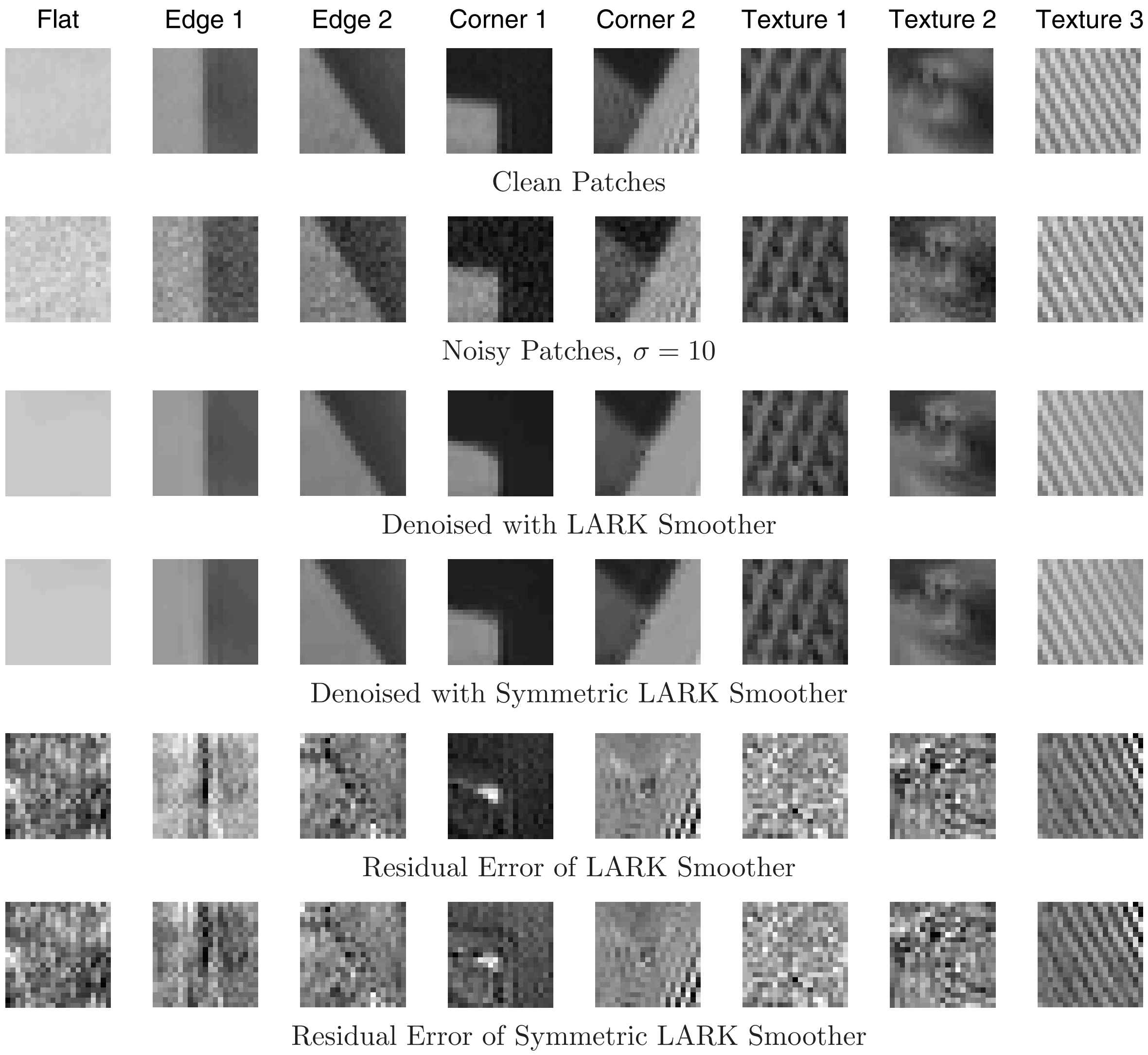
$$\widehat{\mathbf{y}}_k = \widehat{\mathbf{y}}_{k-1} + \mathbf{A}(\mathbf{y} - \widehat{\mathbf{y}}_{k-1}) = \sum_{l=0}^{k}\mathbf{A}(\mathbf{I}-\mathbf{A})^l\,\mathbf{y}. \tag{3.4}$$

This iteration will be stable so long as the largest singular value $(\mathbf{I}-\mathbf{A})$ is bounded by 1. We observe in Table 3 that this is in general not the case before the symmetrization.

[10] This constant is $e^{-\gamma}$, where $\gamma = 0.577215665$ is Euler's constant.

[11] The degrees of freedom of the unsymmetrized smoother $\mathbf{A}$ are given by $\mathbf{df} = \beta\,\mathbf{df}_s$.

[12] Other norms can be used for the definition, but we use the L_2 norm.



Clean Patches

Noisy Patches, $\sigma = 10$

Denoised with LARK Smoother

Denoised with Symmetric LARK Smoother

Residual Error of LARK Smoother

Residual Error of Symmetric LARK Smoother

Figure 1. *Denoising performance comparison using LARK smoother and its symmetrized version.*

Table 1

MSE comparisons for original versus symmetrized LARK smoother on various images shown in Figure 1 (top).

	Flat	Edge 1	Edge 2	Corner 1	Corner 2	Texture 1	Texture 2	Texture 3
MSE($\widehat{\mathbf{y}}$)	4.78	14.29	18.25	57.73	102.39	72.90	37.72	96.14
MSE($\widehat{\mathbf{y}}_s$)	4.79	10.10	18.20	20.02	99.22	72.99	37.63	94.30
(Improvement)	(-0.24%)	(29.34%)	(0.28%)	(65.32%)	(3.10%)	(-0.12%)	(0.26%)	(1.92%)
β	0.9335	0.8283	0.9211	0.7676	0.8305	0.9992	0.9747	0.9744
$\mathbf{df}_s$	12.94	29.88	75.01	54.47	88.59	371.72	217.16	115.47

Table 2

Top singular value of original (unsymmetrized) and symmetrized LARK smoothing matrices.

	Flat	Edge 1	Edge 2	Corner 1	Corner 2	Texture 1	Texture 2	Texture 3
Top sing. val. of $\mathbf{A}$	1.0182	1.0583	1.0374	1.0668	1.1361	1.0005	1.0689	1.0085
Top sing. val. of $\widehat{\mathbf{A}}$	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Table 3

Top singular values of (unsymmetrized and symmetrized) boosted LARK smoothing matrices.

	Flat	Edge 1	Edge 2	Corner 1	Corner 2	Texture 1	Texture 2	Texture 3
Top sing. val. of $\mathbf{I} - \mathbf{A}$	1.0040	1.0593	1.0154	1.0632	1.1081	0.7981	1.0520	1.0021
Top sing. val. of $\mathbf{I} - \widehat{\mathbf{A}}$	1.0	1.0	1.0	1.0	1.0	0.7999	1.0	1.0

Related iterative procedures, such as the *backfitting* method of [9], also depend strongly on this shrinking property. It was noted in [9] that backfitting, and the related Gauss–Seidel process, is stable and consistent only for symmetric smoothers. For this and all such iterative algorithms, the normalization proposed here allows this constraint to be relaxed.

3.3. Spectral decomposition. Given their data-adaptive nature, understanding the behavior of nonlinear smoothers is not easy. But considering the symmetric approximation we promote, this becomes tractable. Consider the eigendecompositions $\mathbf{A} = \mathbf{U}\widehat{\mathbf{S}}\mathbf{U}^{-1}$ and $\widehat{\mathbf{A}} = \mathbf{V}\mathbf{S}\mathbf{V}^T$. The latter, of course, is advantageous because the eigenvectors in the columns of $\mathbf{V}$ (the principal components) form an orthonormal basis which allows us to clearly see the local effect of this filter. In Figures 2 and 3 we visualize, as images, the four most dominant columns of $\mathbf{U}$ and $\mathbf{V}$, respectively. As is apparent, the orthonormal basis corresponding to the symmetrized matrix captures the local geometry of the patches very efficiently. With either a fixed basis such as Fourier or discrete cosine transform (DCT), or the nonorthogonal basis given by the columns of $\mathbf{U}$, many more terms are required to capture the local geometry, particularly at discontinuities. The important advantage we gain here is that the symmetrization allows for a sparse and compact representation of the edge in terms of a few principal components, therefore allowing high performance denoising [11].

Next, let us visually illustrate the weights for the patches shown in Figure 1 (top). Note that for a 21 × 21 image, each of these matrices is of size 441 × 441. As an example, the middle row of each of the smoothing matrices (corresponding to the center pixel of the patch) is illustrated[13] in Figure 4. The corresponding eigenvalues of $\mathbf{A}$ and $\widehat{\mathbf{A}}$ for these patches are also shown in Figure 5. We compute the squared difference between the elements a_{ij} and $\widehat{a}_{ij}$ for all patches, and these are shown in Table 4. The eigenvalue perturbation $\sum_{i=1}^{n}(\lambda_i - \widehat{\lambda}_i)^2$ is

[13] For convenience of illustration, the rows are reshaped to a 21×21 picture of weights. Also, a brief note on implementation: Each element of $\mathbf{A}$ uses a different covariance $\mathbf{Q}_{ij}$, where, in each row, the first of these indices is fixed. To compute the LARK features over these small patches, we used a region slightly larger than the ones shown, extracted from a bigger image. Each pixel i inside the patch has a kernel value $\mathbf{K}_{ij}$ corresponding to any given pixel j (either inside or outside the patch region). However, when actually forming the matrix $\mathbf{K}$, in its ith row only j inside the patch is used. After that, Sinkhorn balancing is implemented to generate the doubly stochastic W.

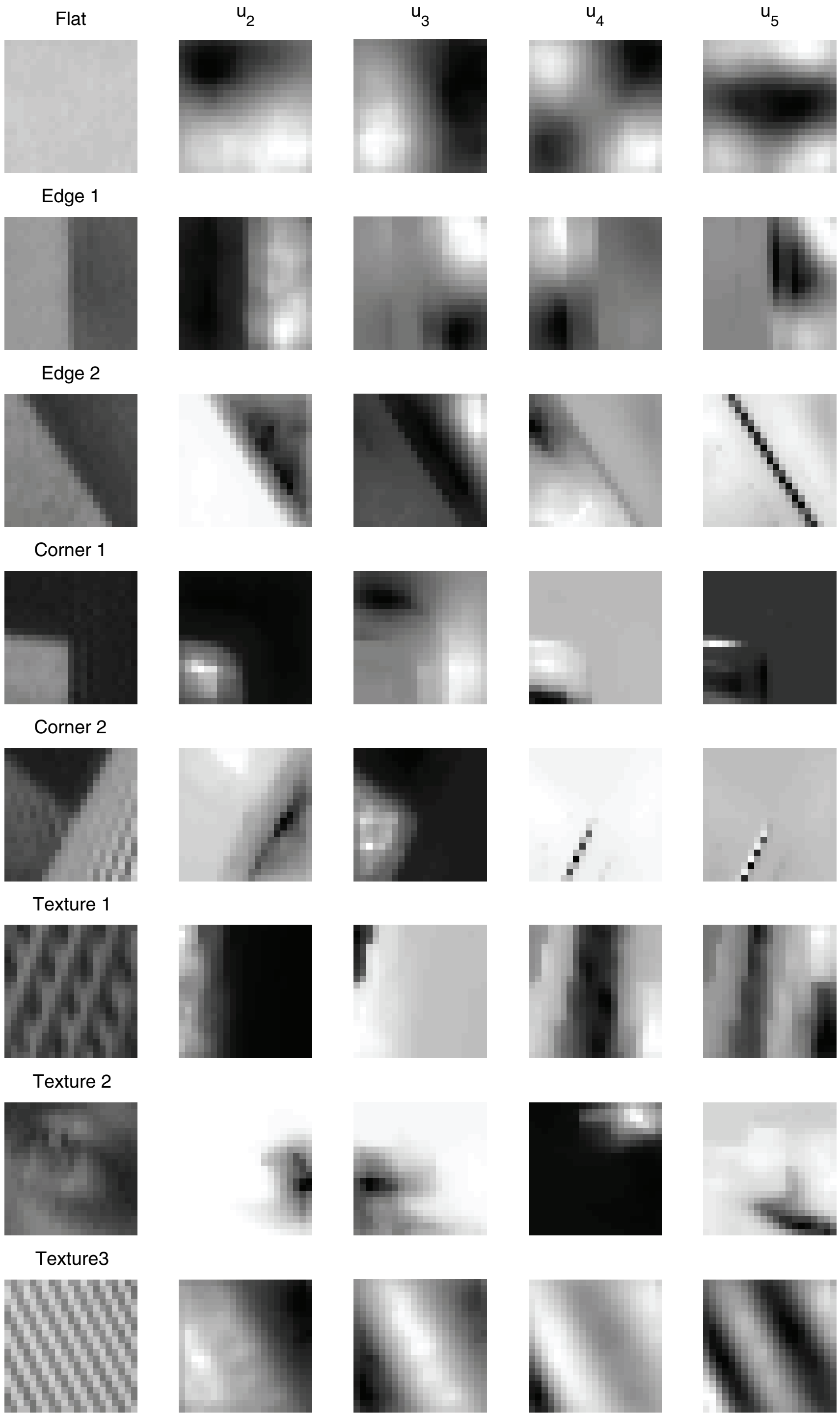


Figure 2. *Eigenvectors* $\mathbf{u}_2$ *through* $\mathbf{u}_5$ *of the* original *LARK smoother* $\mathbf{A}$ *for the patches shown in the first column. The dominant eigenvector* $\mathbf{u}_1$ *is constant, and is therefore not shown.*

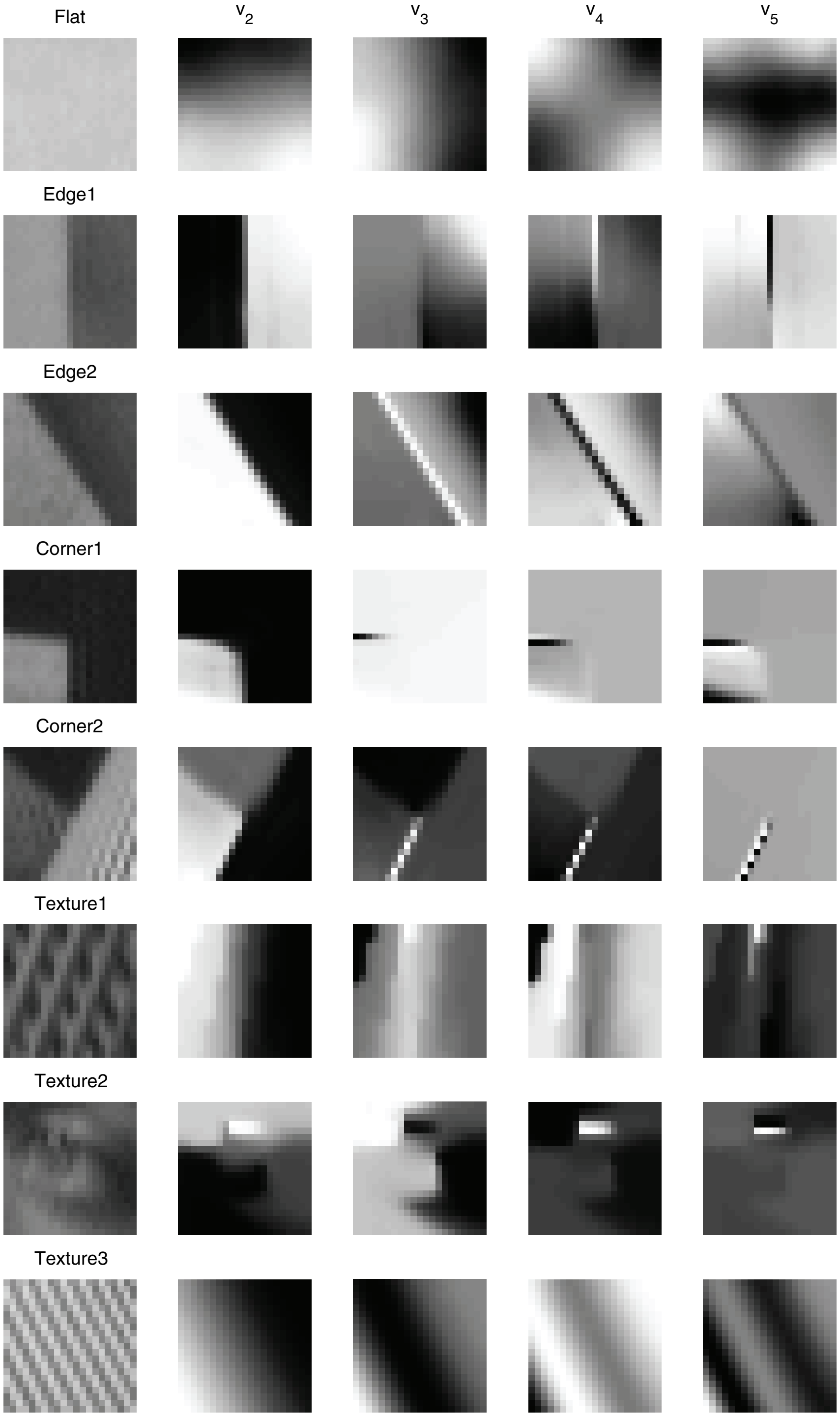


Figure 3. *Dominant eigenvectors* $\mathbf{v}_2$ *through* $\mathbf{v}_5$ *of the* symmetrized *LARK smoother* $\widehat{\mathbf{A}}$ *for the patches shown in the first column. The dominant eigenvector* $\mathbf{v}_1$ *is constant, and is therefore not shown.*

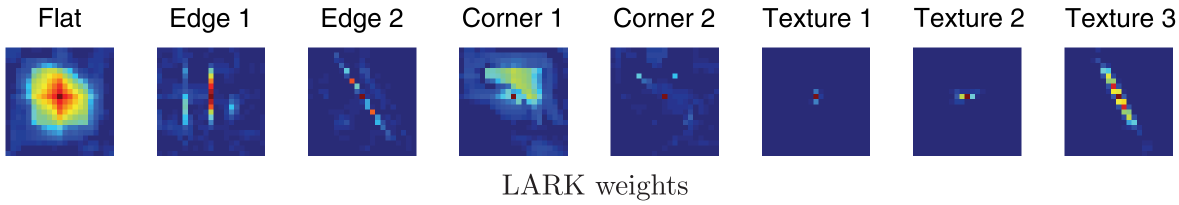


Figure 4. *(Square root of) LARK weights for the center pixel position.*

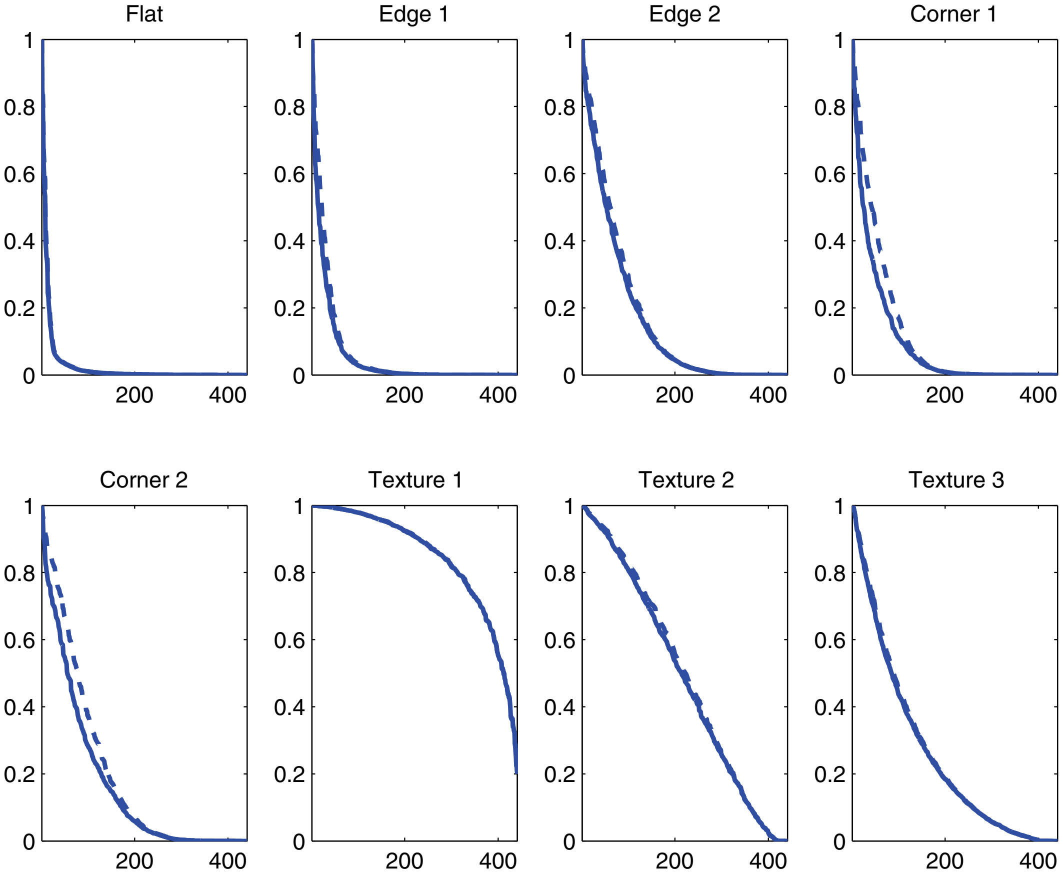


Figure 5. *Spectra of the smoothing matrices: the original* $\mathbf{A}$ *(solid), the symmetrized* $\widehat{\mathbf{A}}$ *(dashed).*

also shown in the same table. We note that these perturbations are smaller for highly textured images, which, as a practical matter, is encouraging.

Another important consequence of the spectral analysis is that we can study the "oracle" performance of the filter analytically. Namely, let us assume that the smoothing matrix $\mathbf{A}$ is given exactly (from the clean data.) We can then ask how good the performance of this filter can be in the MSE sense. First, consider the measured data as given by a simple additive

Table 4

Perturbation values for original versus symmetrized LARK smoother on various images shown in Figure 1 (top).

	Flat	Edge 1	Edge 2	Corner 1	Corner 2	Texture 1	Texture 2	Texture 3
$\|\mathbf{A}-\widehat{\mathbf{A}}\|_F^2$ (Difference)	0.22 (4.04%)	1.24 (10.48%)	1.00 (2.72%)	3.07 (15.23%)	3.3 (8.37%)	0.05 (0.01%)	1.02 (0.68%)	0.68 (1.10%)
$\sum_{i=1}^n(\lambda_i-\widehat{\lambda}_i)^2$ (Difference)	0.03 (0.51%)	0.47 (4.21%)	0.25 (0.70%)	1.49 (7.79%)	1.46 (3.87%)	0.00 (0.00%)	0.12 (0.08%)	0.05 (0.08%)

noise model:

$$\mathbf{y} = \mathbf{z} + \mathbf{e}, \tag{3.5}$$

where $\mathbf{z}$ is the latent image, and $\mathbf{e}$ is noise with $\mathbf{E}(\mathbf{e}) = 0$ and $\mathbf{Cov}(\mathbf{e}) = \sigma^2\mathbf{I}$.

We can compute the statistics of the smoother $\widehat{\mathbf{y}} = \mathbf{A}\mathbf{y}$. The bias in the estimate is

$$\text{bias} = \mathbf{E}(\widehat{\mathbf{y}}) - \mathbf{z} = \mathbf{E}(\mathbf{A}\mathbf{y}) - \mathbf{z} \approx \widehat{\mathbf{A}}\mathbf{z} - \mathbf{z} \ = (\widehat{\mathbf{A}} - \mathbf{I})\,\mathbf{z},$$

where $\widehat{\mathbf{A}}$ is the doubly stochastic, symmetric approximation. The squared magnitude of the bias is therefore

$$\|\text{bias}\|^2 = \|(\widehat{\mathbf{A}} - \mathbf{I})\mathbf{z}\|^2. \tag{3.6}$$

Writing the latent image $\mathbf{z}$ as a linear combination of the orthogonal principal components of $\widehat{\mathbf{A}}$ (that is, its eigenvectors) as $\mathbf{z} = \mathbf{V}\mathbf{b}$, we can rewrite the squared bias magnitude as

$$\|\text{bias}\|^2 = \|(\widehat{\mathbf{A}} - \mathbf{I})\mathbf{z}\|^2 = \|\mathbf{V}(\mathbf{S} - \mathbf{I})\mathbf{b}\|^2 = \|(\mathbf{S} - \mathbf{I})\mathbf{b}\|^2 = \sum_{i=1}^{n}(\widehat{\lambda}_i - 1)^2 b_i^2. \tag{3.7}$$

We also have

$$\mathbf{cov}(\widehat{\mathbf{y}}) = \mathbf{cov}(\mathbf{A}\,\mathbf{y}) \approx \mathbf{cov}(\widehat{\mathbf{A}}\,\mathbf{e}) = \sigma^2\widehat{\mathbf{A}}\,\widehat{\mathbf{A}}^T \qquad \Longrightarrow \qquad \mathbf{var}(\widehat{\mathbf{y}}) = \mathbf{tr}(\mathbf{cov}(\widehat{\mathbf{y}})) = \sigma^2\sum_{i=1}^{n}\widehat{\lambda}_i^2.$$

Overall, the mean-squared error is therefore given by

$$\mathbf{MSE} = \|\text{bias}\|^2 + \mathbf{var}(\widehat{\mathbf{y}}) \approx \sum_{i=1}^{n}(\widehat{\lambda}_i - 1)^2 b_i^2 + \sigma^2\widehat{\lambda}_i^2. \tag{3.8}$$

For a given latent image $\mathbf{z}$, or equivalently, given coefficients $\mathbf{b}$, the above MSE expression gives the ideal (lowest) error that an “oracle” version of the smoother could realize. This insight can be used to study the performance of existing smoothing filters for comparison, much in the same spirit as was done in [11]. Furthermore, we can also ask what the spectrum of an ideal smoothing filter for denoising a given image would look like. The answer is mercifully simple, given our formulation. Namely, by differentiating the MSE expression (3.8) with respect to

λ_i and setting the result to zero, we find that the best eigenvalues are given by the Wiener filter condition:

$$\lambda_i^* = \frac{b_i^2}{b_i^2 + \sigma^2} = \frac{1}{1 + \mathbf{snr}_i^{-1}}, \tag{3.9}$$

where $\mathbf{snr}_i = \frac{b_i^2}{\sigma^2}$ denotes the signal-to-noise ratio at each i. With this ideal spectrum, the smallest possible MSE value is obtained by replacing the eigenvalues in (3.8) with λ_i^* from (3.9), which after some algebra gives

$$\mathbf{MSE}_{min} = \sigma^2 \sum_{i=1}^{n} \lambda_i^*. \tag{3.10}$$

Interestingly, in patches that are relatively flat, the bias component of this minimum MSE is dominant. The fact that bias in flat regions is a problem in practice is a well-known phenomenon [11] for high performance algorithms such as BM3D [16].

Using the MSE expression in (3.10), we can also ask what class of images (that is, which sequences of b_i) will result in the worst or largest $\mathbf{MSE}_{min}$. This question must, of course, be asked subject to an energy constraint on the coefficients b_i. Recalling that $\mathbf{snr}_i = \frac{b_i^2}{\sigma^2}$, we can pose this problem as

$$\max_{\mathbf{b}} \sum_{i=1}^{n} \frac{b_i^2}{\sigma^2 + b_i^2} \qquad \text{subject to} \quad \mathbf{b}^T\mathbf{b} = 1.$$

This is a simple constrained optimization problem whose solution is readily found to be $b_i^2 = 1/n$. Generating images using the local basis with these coefficients yields completely unstructured patches. This is because a constant representation given by $b_i^2 = 1/n$ essentially corresponds to white noise. Such patches indeed visually appear as flat patches corrupted by noise. Since there is no redundancy in such patches, the estimator's bias becomes very large. Again, it has been noted that the best performing algorithms such as BM3D [16] in fact produce their largest errors in relatively flat but noisy areas, where visible artifacts appear. This is also consistent with what we know from the performance bound analysis provided by [11] and [37], namely, that the largest improvements we can expect to realize in future denoising algorithms are to be had in these types of regions.

4. Remarks and conclusions.

For the reader interested in applying and extending the results presented here, we make a few observations.

Remark 1. By nature, any smoothing filter with nonnegative coefficients can have relatively strong bias components. One well-known way to improve them is to use smoothers with negative coefficients, or equivalently, higher order regression [29]. Another is to simply normalize them as we have suggested here. The mechanism we have proposed for symmetrizing smoothers is general enough to be applied to *any* smoothing filter with nonnegative coefficients.

Remark 2. It is not possible to apply Sinkhorn balancing to matrices that have negative elements, as this can result in nonconvergence of the iteration in Algorithm 1. It is in fact unclear whether application of such a normalization scheme would have a performance advantage in such cases. Yet, it is certainly of interest to study mechanisms for symmetrizing general (not necessarily positive-valued) smoothing matrices, as this would facilitate their spectral analysis in orthonormal bases. As we hinted in [39], here we would be dealing with the class of *generalized* stochastic matrices [34].

Remark 3. It is well known [28] that when the smoother is symmetric, the estimator always has a Bayesian interpretation with a well-defined posterior density. By approximating a given smoother with a symmetric one, we have enabled such an interpretation. In particular, when the smoother is data-dependent, the interpretation is more appropriately defined as an empirical Bayesian procedure as described in [39].

Remark 4. It is possible, and in some cases desirable, to apply several smoothers to the data and to aggregate the result for improvement—a procedure known as boosting in the machine learning literature. The smoothers can be related (such as powers of a given smoothing matrix [39]) or chosen to provide intentionally different characteristics (one oversmoothing, and another undersmoothing). The results in this paper can be applied to all such procedures.

Remark 5. The normalization provided by the Sinkhorn algorithm can also be applied to scale the Laplacian matrix. Recalling that $\mathcal{L} = \mathbf{D}^{-1/2}\mathbf{K}\mathbf{D}^{-1/2} - \mathbf{I}$, we can apply Sinkhorn's scaling to the first term to obtain a newly scaled, doubly stochastic version of the kernel $\widehat{\mathbf{K}} = \mathbf{M}\mathbf{D}^{-1/2}\mathbf{K}\mathbf{D}^{-1/2}\mathbf{M}$, where $\mathbf{M}$ is a unique positive diagonal matrix. The scaled Laplacian can then be defined as $\widehat{\mathcal{L}} = \widehat{\mathbf{K}} - \mathbf{I}$. This scaled Laplacian now enjoys the interesting property that it has both row and column sums equal to *zero*. We speculate that this result may in fact yield improvements in various areas of application such as dimensionality reduction, data representation [3], clustering [56], segmentation [45], and others.

Remark 6. As observed by a reviewer, it is interesting to consider whether we can design cost functions (or, equivalently, PDEs) that lead naturally to symmetrized kernels. Similarly, it would be very useful to be able to design and compute approximations of existing kernel filters (such as bilateral, nonlocal means, etc.), or new kernels, from first principles, such that the coefficients are automatically and naturally symmetric, hence not requiring a symmetrization step.

Remark 7. Ideally, we wish to avoid altogether the calculation of the large matrix $\mathbf{W}$, followed by Sinkhorn balancing, in a sequential fashion. We have noted that the process of computing the spectrum of $\mathbf{W}$ can be made significantly more computationally efficient by making use of a sampling method [41] (more recently employed in [22] and elsewhere).

To summarize, we studied a class of smoothing filters which operate based on nonlinear, shift-variant averaging which are frequently used in both signal and image processing. We provided a matrix approximation that converts a given smoother to one that is symmetric and doubly stochastic. This enables us to not only improve performance of the base procedure, but also to peer into the complex behavior of such filters in the transform domain using principal components.

Appendix A. Approximation of nonlinear smoothers. In the course of the paper, we make the observation that the nonlinear smoothers we have considered here can be treated as if the smoothing matrix $\mathbf{A}$ is nearly decoupled from the noisy observation $\mathbf{y}$. We justify this approach here. For convenience, consider the filter for a single value at a time. Define the vector $\mathbf{a}(\mathbf{y}) = [a_{1j}, \ldots, a_{nj}]$, so that the jth element of the smoothed output vector $\mathbf{y}$ is given by

$$\widehat{y}_j = \mathbf{a}(\mathbf{y})^T \mathbf{y},$$

where, to simplify the notation, we have suppressed the dependence of the right-hand side on the index j. For the purpose of computing the smoothing operator from the data, in practice we always compute a "prefiltered" or "pilot" estimate first, whose intent is not to yield a final result on its own, but to suppress the sensitivity of the weight calculations to noise. Let this pilot estimate be $\tilde{\mathbf{y}} = \mathbf{z} + \boldsymbol{\epsilon}$, where we assume $\boldsymbol{\epsilon}$ is small. As such we can make the following first order Taylor approximation to the (practical) smoother which uses the pilot estimate:

$$\mathbf{a}(\tilde{\mathbf{y}})^T\mathbf{y} = \mathbf{a}(\mathbf{z}+\boldsymbol{\epsilon})^T\mathbf{y} \approx \left(\mathbf{a}(\mathbf{z}) + \nabla\mathbf{a}(\mathbf{z})^T\boldsymbol{\epsilon}\right)^T\mathbf{y} = \mathbf{a}(\mathbf{z})^T\mathbf{y} + \boldsymbol{\epsilon}^T\nabla\mathbf{a}(\mathbf{z})\mathbf{y},$$

where $\nabla\mathbf{a}(\mathbf{z})$ is the gradient of the vector $\mathbf{a}$ evaluated at the latent image. The first term $\mathbf{a}(\mathbf{z})^T\mathbf{y}$ on the right-hand side is the *oracle* smoother which we have used as a benchmark throughout the paper. The second term is the error between the practical smoother and the oracle:

$$\Delta = \mathbf{a}(\tilde{\mathbf{y}})^T\mathbf{y} - \mathbf{a}(\mathbf{z})^T\mathbf{y} \approx \boldsymbol{\epsilon}^T\nabla\mathbf{a}(\mathbf{z})\,\mathbf{y}.$$

We observe that when $\boldsymbol{\epsilon}$ and the gradient $\nabla\mathbf{a}$ are small, the approximation error can remain small. The first is a consequence of the quality of the chosen prefilter, which must be good, whereas the second is a result of the smoothness of the kernel—specifically, the magnitude of its gradient.[14] With an appropriate prefilter, and with a choice of a smooth kernel such as the Gaussian, we can be assured that the approximation is faithful for the analysis described here and further detailed in [39].

Appendix B. Proofs.

Proof of Lemma 2.1. Let

$$\begin{aligned}\mathbf{A}_1 &= \lim_{k\to\infty}\mathbf{A}^k = \mathbf{1}\,\mathbf{u}_1^T,\\ \widehat{\mathbf{A}}_1 &= \lim_{k\to\infty}\widehat{\mathbf{A}}^k = \frac{1}{n}\,\mathbf{1}\,\mathbf{1}^T.\end{aligned}$$

For all positive integers k, we have

$$\begin{aligned}\|\mathbf{A}^k - \widehat{\mathbf{A}}^k\|_F &= \|\mathbf{A}^k - \mathbf{A}_1 + \mathbf{A}_1 - \widehat{\mathbf{A}}^k\|_F\\ &\le \|\mathbf{A}^k - \mathbf{A}_1\|_F + \|\widehat{\mathbf{A}}^k - \mathbf{A}_1\|_F\\ &= \|\mathbf{A}^k - \mathbf{A}_1\|_F + \|\widehat{\mathbf{A}}^k - \mathbf{A}_1 + \widehat{\mathbf{A}}_1 - \widehat{\mathbf{A}}_1\|_F\\ &\le \|\mathbf{A}^k - \mathbf{A}_1\|_F + \|\widehat{\mathbf{A}}^k - \widehat{\mathbf{A}}_1\|_F + \|\widehat{\mathbf{A}}_1 - \mathbf{A}_1\|_F\\ &\le c\,n\,|\lambda_2|^k + \widehat{c}\,n\,|\widehat{\lambda}_2|^k + \left\|\mathbf{1}\,\mathbf{u}_1^T - \frac{1}{n}\,\mathbf{1}\,\mathbf{1}^T\right\|_F,\end{aligned} \tag{B.1}$$

where the last inequality follows from (2.6). The last term on the right-hand side can be

[14] When we speak of the smoothness of the kernel, we are not referring to whether the underlying signal is smooth. We are referring only to the way in which the kernel depends on its argument.

estimated as

$$\begin{aligned}\|\mathbf{1}\,\mathbf{u}_1^T - \frac{1}{n}\,\mathbf{1}\mathbf{1}^T\|_F^2 &= n\,\sum_{i=1}^n \left(u_{1i} - \frac{1}{n}\right)^2\\ &= n\sum_{i=1}^n \left(u_{1i}^2 - \frac{2}{n}u_{1i} + \frac{1}{n^2}\right)\\ &= n\left(\sum_{i=1}^n u_{1i}^2 - \frac{2}{n}\sum_{i=1}^n u_{1i} + \frac{1}{n}\right)\\ &\le n\left(1 - \frac{1}{n}\right) = n-1,\end{aligned}$$

where the last inequality follows since $\|\mathbf{u}_1\|_1 = 1$ and $\mathbf{u}_1 \ge 0$. Taking square roots and replacing this in (B.1), we have

$$\|\mathbf{A}^k - \widehat{\mathbf{A}}^k\|_F \le c\,n\,|\lambda_2|^k + \widehat{c}\,n\,|\widehat{\lambda}_2|^k + (n-1)^{1/2}\,. \tag{B.2}$$

Dividing by n yields the result. ■

Proof of Theorem 3.2. The determinant inequality implies that there exists a constant $0 < \alpha \le 1$ such that

$$\prod_{i=1}^n \lambda_i = \alpha\,\prod_{i=1}^n \widehat{\lambda}_i.$$

Assume that the trace inequality is *not* true. That is, we suppose

$$\sum_{i=1}^n \widehat{\lambda}_i < \sum_{i=1}^n \lambda_i. \tag{B.3}$$

As we shall see, this assumption leads to a contradiction. Invoking the geometric-arithmetic inequality [27], we write

$$g_n = \left(\prod_{i=1}^n \lambda_i\right)^{1/n} = \alpha^{1/n}\left(\prod_{i=1}^n \widehat{\lambda}_i\right)^{1/n} \le \alpha^{1/n}\left(\frac{1}{n}\sum_{i=1}^n \widehat{\lambda}_i\right) < \alpha^{1/n}\,a_n,$$

where g_n is the geometric mean and $a_n = n^{-1}\sum_{i=1}^n \lambda_i$ is the arithmetic mean. The last inequality follows by invoking (B.3). The above implies that for every $n \times n$ ($n \ge 2$) matrix $\mathbf{A}$ satisfying the conditions of the theorem, it must be the case that

$$\frac{g_n}{a_n} < \alpha^{1/n}. \tag{B.4}$$

Now consider the two matrices $\mathbf{A} = \mathrm{diag}\,[1,\ 1,\ \ldots,\ \alpha]$ and $\widehat{\mathbf{A}} = \mathrm{diag}\,[1, 1, \ldots, 1]$, which give

$$\frac{g_n}{a_n} = \frac{\alpha^{1/n}}{\frac{1}{n}(n-1+\alpha)} < \alpha^{1/n}.$$

Simplifying, this yields $\alpha > 1$, which is a contradiction. ■

Acknowledgments. I would like to thank my colleagues Gabriel Peyré, Michael Elad, and Jean-Michel Morel for their insightful comments and feedback.

REFERENCES

[1] J. M. Aldaz, *Concentration of the ratio between the geometric and arithmetic means*, J. Theoret. Probab., 23 (2010), pp. 498–508.

[2] S. P. Awate and R. T. Whitaker, *Unsupervised, information-theoretic, adaptive image filtering for image restoration*, IEEE Trans. Pattern Anal. Mach. Intell., 28 (2006), pp. 364–376.

[3] M. Belkin and P. Niyogi, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural Comput., 15 (2003), pp. 1373–1396.

[4] R. Bhatia, *Perturbation Bounds for Matrix Eigenvalues*, Classics Appl. Math. 53, SIAM, Philadelphia, 2007.

[5] C. Bordenave, P. Caputo, and D. Chafai, *Circular law theorem for random Markov matrices*, Probab. Theory Related Fields, 152 (2012), pp. 751–779.

[6] R. A. Brualdi, *Matrices of* 0*'s and* 1*'s with total support*, J. Combin. Theory Ser. A, 28 (1980), pp. 249–256.

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

[8] P. Buhlmann and B. Yu, *Boosting with the L_2 loss: Regression and classification*, J. Amer. Statist. Assoc., 98 (2003), pp. 324–339.

[9] A. Buja, T. Hastie, and R. Tibshirani, *Linear smoothers and additive models*, Ann. Statist., 17 (1989), pp. 453–510.

[10] D. Chafai, *Aspects of large random Markov kernels*, Stochastics, 81 (2009), pp. 415–429.

[11] P. Chatterjee and P. Milanfar, *Is denoising dead?*, IEEE Trans. Image Process., 19 (2010), pp. 895–911.

[12] P. Chatterjee and P. Milanfar, *Patch-based near-optimal denoising*, IEEE Trans. Image Process., 21 (2012), pp. 1635–1649.

[13] A. Cohen, *All admissible linear estimates of the mean vector*, Ann. Math. Statist., 37 (1966), pp. 458–463.

[14] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, *Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps*, Proc. Natl. Acad. Sci. USA, 102 (2005), pp. 7426–7431.

[15] I. Csiszar, *I-divergence geometry of probability distributions and minimization problems*, Ann. Probab., 3 (1975), pp. 146–158.

[16] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, *Image denoising by sparse* 3*-D transform-domain collaborative filtering*, IEEE Trans. Image Process., 16 (2007), pp. 2080–2095.

[17] J. Darroch and D. Ratcliff, *Generalized iterative scaling for log-linear models*, Ann. Math. Statist., 43 (1972), pp. 1470–1480.

[18] G. Deng and L. Cahill, *An adaptive Gaussian filter for noise reduction and edge detection*, in Nuclear Science Symposium and Medical Imaging Conference, IEEE Conference Record, Vol. 3, 1993, pp. 1615–1619.

[19] J. Digne, J.-M. Morel, C.-M. Souzani, and C. Lartigue, *Scale space meshing of raw data point sets*, Comput. Graph. Forum, 30 (2011), pp. 1630–1642.

[20] A. Dimakis, S. Kar, J. Moura, M. Rabbat, and A. Scaglione, *Gossip algorithms for distributed signal processing*, Proc. IEEE, 98 (2010), pp. 1847–1864.

[21] M. Elad, *On the origin of the bilateral filter and ways to improve it*, IEEE Trans. Image Process., 11 (2002), pp. 1141–1150.

[22] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, *Spectral grouping using the Nyström method*, IEEE Trans. Pattern Anal. Mach. Intell., 26 (2004), pp. 214–225.

[23] E. Gluskin and V. Milman, *Note on the geometric-arithmetic mean inequality*, in Geometric Aspects of Functional Analysis, Lecture Notes in Math. 1807, Springer, Berlin, 2003, pp. 130–135.

[24] G. Goldberg and M. Neumann, *Distribution of subdominant eigenvalues of matrices with random rows*, SIAM J. Matrix Anal. Appl., 24 (2003), pp. 747–761.

[25] G. Goldberg, P. Okunev, M. Neumann, and H. Schneider, *Distribution of subdominant eigenvalues of random matrices*, Methodol. Comput. Appl. Probab., 2 (2000), pp. 137–151.

[26] W. Härdle, *Applied Nonparametric Regression*, Cambridge University Press, Cambridge, UK, 1990.

[27] G. Hardy, J. E. Littlewood, and G. Pólya, *Inequalities*, 2nd ed., Cambridge University Press, Cambridge, UK, 1988.

[28] T. Hastie and R. Tibshirani, *Bayesian backfitting*, Statist. Sci., 15 (2000), pp. 196–223.

[29] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, New York, 2009.

[30] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1990.

[31] M. Horvat, *The ensemble of random Markov matrices*, J. Stat. Mech. Theory Exp., No. 7, (2009), P07005.

[32] C. R. Johnson and R. B. Kellogg, *An inequality for doubly stochastic matrices*, J. Res. Nat. Bur. Standards. Sect. B, 80 (1976), pp. 433–436.

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

[34] R. Khoury, *Closest matrices in the space of generalized doubly stochastic matrices*, J. Math. Anal. Appl., 222 (1998), pp. 562–568.

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

[36] P. A. Knight, *The Sinkhorn–Knopp algorithm: Convergence and applications*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 261–275.

[37] A. Levin and B. Nadler, *Natural image denoising: Optimality and inherent bounds*, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 2833–2840.

[38] D. Levin, *The approximation power of moving least-squares*, Math. Comp., 67 (1998), pp. 1517–1531.

[39] P. Milanfar, *A tour of modern image filtering*, IEEE Signal Processing Mag., 30 (2013), pp. 106–128.

[40] N. Nordstrom, *Biased anisotropic diffusion—a unified regularization and diffusion approach to edge detection*, Image Vision Comput., 8 (1990), pp. 318–327.

[41] E. J. Nyström, *Über die praktische Auflösung von linearen Integralgleichungen mit Anwendungen auf Randwertaufgaben der Potentialtheorie*, Comment. Phys.-Math., 4 (1928), pp. 1–52.

[42] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, *An iterative regularization method for total variation-based image restoration*, Multiscale Model. Simul., 4 (2005), pp. 460–489.

[43] G. Peyré, *Image processing with nonlocal spectral bases*, Multiscale Model. Simul., 7 (2008), pp. 703–730.

[44] E. Seneta, *Non-Negative Matrices and Markov Chains*, Springer Ser. Statist., Springer, NewYork, 1981.

[45] J. Shi and J. Malik, *Normalized cuts and image segmentation*, IEEE Trans. Pattern Anal. Mach. Intell., 22 (2000), pp. 888–905.

[46] R. Sinkhorn, *A relationship between arbitrary positive matrices and doubly stochastic matrices*, Ann. Math. Statist., 35 (1964), pp. 876–879.

[47] R. Sinkhorn and P. Knopp, *Concerning nonnegative matrices and doubly stochastic matrices*, Pacific J. Math., 21 (1967), pp. 343–348.

[48] A. Spira, R. Kimmel, and N. Sochen, *A short time Beltrami kernel for smoothing images and manifolds*, IEEE Trans. Image Process., 16 (2007), pp. 1628–1636.

[49] W. J. Stewart, *Introduction to the Numerical Solution of Markov Chains*, Princeton University Press, Princeton, NJ, 1994.

[50] H. Takeda, S. Farsiu, and P. Milanfar, *Kernel regression for image processing and reconstruction*, IEEE Trans. Image Process., 16 (2007), pp. 349–366.

[51] C. Tomasi and R. Manduchi, *Bilateral filtering for gray and color images*, in Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India, 1998, pp. 836–846.

[52] J. W. Tukey, *Exploratory Data Analysis*, Addison-Wesley, Reading, MA, 1977.

[53] M. P. Wand and M. C. Jones, *Kernel Smoothing*, Monogr. Statist. Appl. Probab., Chapman and Hall, London, 1995.

[54] J. Weickert, *Coherence-enhancing diffusion*, Int. J. Comput. Vision, 31 (1999), p. 111–127.

[55] L. P. Yaroslavsky, *Digital Picture Processing*, Springer-Verlag, Berlin, 1985.

[56] R. Zass and A. Shashua, *Doubly stochastic normalization for spectral clustering*, in Advances in Neural Information Processing Systems (NIPS), MIT Press, Cambridge, MA, 2006, pp. 1569–1576.