Adaptive Image Denoising by Targeted Databases

Enming Luo, *Student Member, IEEE*, Stanley H. Chan, *Member, IEEE*, and Truong Q. Nguyen, *Fellow, IEEE*

***Abstract*—We propose a data-dependent denoising procedure to restore noisy images. Different from existing denoising algorithms which search for patches from either the noisy image or a generic database, the new algorithm finds patches from a database that contains relevant patches. We formulate the denoising problem as an optimal filter design problem and make two contributions. First, we determine the basis function of the denoising filter by solving a group sparsity minimization problem. The optimization formulation generalizes existing denoising algorithms and offers systematic analysis of the performance. Improvement methods are proposed to enhance the patch search process. Second, we determine the spectral coefficients of the denoising filter by considering a localized Bayesian prior. The localized prior leverages the similarity of the targeted database, alleviates the intensive Bayesian computation, and links the new method to the classical linear minimum mean squared error estimation. We demonstrate applications of the proposed method in a variety of scenarios, including text images, multiview images and face images. Experimental results show the superiority of the new algorithm over existing methods.**

***Index Terms*—Patch-based filtering, image denoising, external database, optimal filter, non-local means, BM3D, group sparsity, Bayesian estimation**

I. Introduction

A. Patch-based Denoising

Image denoising is a classical signal recovery problem where the goal is to restore a clean image from its observations. Although image denoising has been studied for decades, the problem remains a fundamental one as it is the test bed for a variety of image processing tasks.

Among the numerous contributions in image denoising in the literature, the most highly-regarded class of methods, to date, is the class of *patch-based image denoising* algorithms [1–9]. Interested readers can refer to [10] for a comprehensive overview of some recent classical and learning-based methods. The idea of a patch-based denoising algorithm is simple: Given a $\sqrt{d} \times \sqrt{d}$ patch $\boldsymbol{q} \in \mathbb{R}^d$ from the noisy image, the algorithm finds a set of reference patches $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k \in \mathbb{R}^d$ and applies some linear (or non-linear) function Φ to obtain an estimate $\widehat{\boldsymbol{p}}$ of the unknown clean patch $\boldsymbol{p}$ as

$$\widehat{\boldsymbol{p}} = \Phi(\boldsymbol{q}; \boldsymbol{p}_1, \ldots, \boldsymbol{p}_k). \tag{1}$$

For example, in non-local means (NLM) [1], Φ is a weighted average of the reference patches, whereas in BM3D [3], Φ is a transform-shrinkage operation.

B. Internal vs External Denoising

For any patch-based denoising algorithm, the denoising performance is intimately related to the reference patches $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k$. Typically, there are two sources of these patches: the noisy image itself and an external database of patches. Denoising using the former is known as *internal denoising* [11], whereas the latter is known as *external denoising* [12, 13].

Internal denoising is more popular than external denoising because it is computationally less expensive. Moreover, internal denoising does not require a training stage, hence making it free of training bias. Furthermore, Glasner et al. [14] showed that patches tend to recur within an image at a different location, orientation, or scale. Thus searching for patches within the noisy image is often a plausible approach. However, on the downside, internal denoising often fails for rare patches — patches that seldom recur in an image. This phenomenon is known as the "rare patch effect", and is widely regarded as a bottleneck of internal denoising [15, 16]. There are some works [17, 18] attempting to alleviate the rare patch problem. However, the extent to which these methods can achieve is still limited.

External denoising [6, 19–21] is an alternative solution to internal denoising. Levin et al. [16, 22] showed that in the limit, the theoretical minimum mean squared error of denoising is achievable by using an infinitely large external database. Recently, Chan et al. [21] developed a computationally efficient sampling scheme to reduce the complexity and demonstrated practical usage of large databases. However, for most of these works the databases are *generic*. These databases, although large in volume, do not necessarily contain useful information to denoise the noisy image of interest. For example, it is clear that a database of natural images is not useful to denoise a noisy portrait image.

C. Adaptive Image Denoising

In this paper, we propose an adaptive image denoising algorithm using a *targeted* external database instead of a *generic* database. Here, a targeted database refers to a database that contains images *relevant* to the noisy image only. As will be illustrated in later parts of this paper, targeted external databases could be obtained in many practical scenarios, such

E. Luo and T. Nguyen are with Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093, USA. Emails: eluo@ucsd.edu and nguyent@ece.ucsd.edu

S. Chan is with School of Electrical and Computer Engineering, and Department of Statistics, Purdue University, West Lafayette, IN 47907, USA. Email: stanleychan@purdue.edu

This work was supported in part by a Croucher Foundation Post-doctoral Research Fellowship, and in part by the National Science Foundation under grant CCF-1160832. Preliminary material in this paper was presented at the 39th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, May 2014.

This paper follows the concept of reproducible research. All the results and examples presented in the paper are reproducible using the code and images available online at http://videoprocessing.ucsd.edu/~eluo as text images (*e.g.*, newspapers and documents), human faces (under certain conditions), and images captured by multiview camera systems. Other possible scenarios include images of license plates, medical CT and MRI images, and images of landmarks.

The concept of using targeted external databases has been proposed in various occasions, *e.g.*, [23–26]. However, none of these methods are tailored for image denoising problems. The objective of this paper is to bridge the gap by addressing the following question:

(Q): Suppose we are *given* a targeted external database, how should we design a denoising algorithm which can *maximally* utilize the database?

Here, we assume that the reference patches $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k$ are *given*. We emphasize that this assumption is application specific — for the examples we mentioned earlier (*e.g.*, text, multiview, face, etc), the assumption is typically true because these images have relatively less variety in content.

At a first glance, question (Q) may look trivial because we can extend existing internal denoising algorithms in a brute-force way to handle external databases. For example, one can modify existing algorithms, *e.g.*, [1, 3, 5, 27, 28], so that the patches are searched from a database instead of the noisy image. Likewise, one can also treat an external database as a "video" and feed the data to multi-image denoising algorithms, *e.g.*, [29–32]. However, the problem of these approaches is that the brute force modifications are heuristic. There is no theoretical guarantee of performance. This suggests that a straight-forward modification of existing methods does *not* solve question (Q), as the database is not maximally utilized.

An alternative response to question (Q) is to train a statistical prior of the targeted database, *e.g.*, [6, 19, 20, 33–36]. The merit of this approach is that the performance often has theoretical guarantee because the denoising problem can now be formulated as a maximum a posteriori (MAP) estimation. However, the drawback is that many of these methods require a large number of training samples which is not always available in practice.

D. Contributions and Organization

In view of the above seemingly easy yet challenging question, we introduced a new denoising algorithm using targeted external databases in [37]. Compared to existing methods, the method proposed in [37] achieves better performance and only requires a small number of external images. In this paper, we extend [37] by offering the following new contributions:

1) Generalization of Existing Methods. We propose a generalized framework which encapsulates a number of denoising algorithms. In particular, we show (in Section III-B) that the proposed group sparsity minimization generalizes both fixed basis and PCA methods. We also show (in Section IV-B) that the proposed local Bayesian MSE solution is a generalization of many spectral operations in existing methods.
2) Improvement Strategies. We propose two improvement strategies for the generalized denoising framework. In Section III-D, we present a patch selection optimization to improve the patch search process. In Section IV-D, we present a soft-thresholding and a hard-thresholding method to improve the spectral coefficients learned by the algorithm.
3) Detailed Proofs. Proofs of the results in this paper and [37] are presented in the Appendix.

The rest of the paper is organized as follows. After outlining the design framework in Section II, we present the above contributions in Section III and IV. Experimental results are discussed in Section V, and concluding remarks are given in Section VI.

II. Optimal Linear Denoising Filter

The foundation of our proposed method is the classical optimal linear denoising filter design problem [38]. In this section, we give a brief review of the design framework and highlight its limitations.

A. Optimal Filter

The design of an optimal denoising filter can be posed as follows: Given a noisy patch $\boldsymbol{q} \in \mathbb{R}^d$, and assuming that the noise is i.i.d. Gaussian with zero mean and variance σ^2, we want to find a linear operator $\boldsymbol{A} \in \mathbb{R}^{d\times d}$ such that the estimate $\widehat{\boldsymbol{p}} = \boldsymbol{A}\boldsymbol{q}$ has the minimum mean squared error (MSE) compared to the ground truth $\boldsymbol{p} \in \mathbb{R}^d$. That is, we want to solve the optimization

$$\boldsymbol{A} = \underset{\boldsymbol{A}}{\arg\min}\ \mathbb{E}\left[\|\boldsymbol{A}\boldsymbol{q} - \boldsymbol{p}\|_2^2\right]. \tag{2}$$

Here, we assume that $\boldsymbol{A}$ is symmetric, or otherwise the Sinkhorn-Knopp iteration [39] can be used to symmetrize $\boldsymbol{A}$, provided that entries of $\boldsymbol{A}$ are non-negative. Given a symmetric $\boldsymbol{A}$, one can apply the eigen-decomposition, $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, where $\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d] \in \mathbb{R}^{d\times d}$ is the basis matrix and $\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, \ldots, \lambda_d\} \in \mathbb{R}^{d\times d}$ is the diagonal matrix containing the spectral coefficients. With $\boldsymbol{U}$ and $\boldsymbol{\Lambda}$, the optimization problem in (2) becomes

$$(\boldsymbol{U}, \boldsymbol{\Lambda}) = \underset{\boldsymbol{U},\boldsymbol{\Lambda}}{\arg\min}\ \mathbb{E}\left[\left\|\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q} - \boldsymbol{p}\right\|_2^2\right], \tag{3}$$

subject to the constraint that $\boldsymbol{U}$ is an orthonormal matrix.

The joint optimization (3) can be solved by noting the following Lemma.

Lemma 1: Let $\boldsymbol{u}_i$ be the ith column of the matrix $\boldsymbol{U}$, and λ_i be the (i,i)th entry of the diagonal matrix $\boldsymbol{\Lambda}$. If $\boldsymbol{q} = \boldsymbol{p} + \boldsymbol{\eta}$, where $\boldsymbol{\eta} \overset{\text{iid}}{\sim} \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{I})$, then

$$\mathbb{E}\left[\left\|\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q} - \boldsymbol{p}\right\|_2^2\right] = \sum_{i=1}^{d}\left[(1-\lambda_i)^2(\boldsymbol{u}_i^T\boldsymbol{p})^2 + \sigma^2\lambda_i^2\right]. \tag{4}$$

The proof of Lemma 1 is given in [40]. With Lemma 1, the denoised patch can be derived from (3) as follows.

Lemma 2: The denoised patch $\widehat{\boldsymbol{p}}$ using the optimal $\boldsymbol{U}$ and $\boldsymbol{\Lambda}$ of (3) is

$$\widehat{\boldsymbol{p}} = \boldsymbol{U}\left(\mathrm{diag}\left\{\frac{\|\boldsymbol{p}\|^2}{\|\boldsymbol{p}\|^2+\sigma^2}, 0, \ldots, 0\right\}\right)\boldsymbol{U}^T\boldsymbol{q},$$

where $\boldsymbol{U}$ is any orthonormal matrix with the first column $\boldsymbol{u}_1 = \boldsymbol{p}/\|\boldsymbol{p}\|_2$.

Proof: See Appendix A. ■

Lemma 2 states that if hypothetically we are given the ground truth $\boldsymbol{p}$, the optimal denoising process is to first project the noisy observation $\boldsymbol{q}$ onto the subspace spanned by $\boldsymbol{p}$, then perform a Wiener shrinkage $\|\boldsymbol{p}\|^2/(\|\boldsymbol{p}\|^2+\sigma^2)$, and re-project the shrinkage coefficients to obtain the denoised estimate. However, since in reality we never have access to the ground truth $\boldsymbol{p}$, this optimal result is not achievable.

B. Problem Statement

Since the oracle optimal filter is not achievable in practice, the question becomes whether it is possible to find a surrogate solution that does not require the ground truth $\boldsymbol{p}$.

To answer this question, it is helpful to separate the joint optimization (3) by first fixing $\boldsymbol{U}$ and minimize the MSE with respect to $\boldsymbol{\Lambda}$. In this case, one can show that (4) achieves the minimum when

$$\lambda_i = \frac{(\boldsymbol{u}_i^T\boldsymbol{p})^2}{(\boldsymbol{u}_i^T\boldsymbol{p})^2+\sigma^2}, \tag{5}$$

in which the minimum MSE estimator is given by

$$\widehat{\boldsymbol{p}} = \boldsymbol{U}\left(\mathrm{diag}\left\{\frac{(\boldsymbol{u}_1^T\boldsymbol{p})^2}{(\boldsymbol{u}_1^T\boldsymbol{p})^2+\sigma^2}, \ldots, \frac{(\boldsymbol{u}_d^T\boldsymbol{p})^2}{(\boldsymbol{u}_d^T\boldsymbol{p})^2+\sigma^2}\right\}\right)\boldsymbol{U}^T\boldsymbol{q}, \tag{6}$$

where $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d\}$ are the columns of $\boldsymbol{U}$.

Inspecting (6), we identify two parts of the problem:

1) Determine $\boldsymbol{U}$. The choice of $\boldsymbol{U}$ plays a critical role in the denoising performance. In literature, $\boldsymbol{U}$ is typically chosen as the FFT or the DCT basis [3, 4]. Other basis, such as the PCA basis (and its variations) [5, 7, 8], can also be used. However, the optimality of these bases is not fully understood.
2) Determine $\boldsymbol{\Lambda}$. Even if $\boldsymbol{U}$ is fixed, the optimal $\boldsymbol{\Lambda}$ in (5) still depends on the unknown ground truth $\boldsymbol{p}$. In [3], $\boldsymbol{\Lambda}$ is determined by hard-thresholding a stack of DCT coefficients or applying an empirical Wiener filter constructed from a first-pass estimate. In [7], $\boldsymbol{\Lambda}$ is formed by the PCA coefficients of a set of relevant noisy patches. Again, it is unclear which of these is optimal.

Motivated by the problems about $\boldsymbol{U}$ and $\boldsymbol{\Lambda}$, in the following two sections we present our proposed method for each of these problems. We discuss its relationship to prior works, and present ways to further improve it.

III. Determine $\boldsymbol{U}$

In this section, we present our proposed method to determine the basis matrix $\boldsymbol{U}$ and show that it is a generalization of a number of existing denoising algorithms. We also discuss ways to improve $\boldsymbol{U}$.

A. Patch Selection via k Nearest Neighbors

Given a noisy patch $\boldsymbol{q}$ and a targeted database $\{\boldsymbol{p}_j\}_{j=1}^n$, our first task is to fetch the k most "relevant" patches. The patch selection is performed by measuring the similarity between $\boldsymbol{q}$ and each of $\{\boldsymbol{p}_j\}_{j=1}^n$, defined as

$$d(\boldsymbol{q}, \boldsymbol{p}_j) = \|\boldsymbol{q} - \boldsymbol{p}_j\|_2, \quad \text{for } j = 1, \ldots, n. \tag{7}$$

We note that (7) is equivalent to the standard k nearest neighbors (kNN) search.

kNN has a drawback that under the ℓ_2 distance, some of the k selected patches may not be truly relevant to the denoising task, because the query patch $\boldsymbol{q}$ is noisy. We will come back to this issue in Section III-D by discussing methods to improve the robustness of the kNN.

B. Group Sparsity

Without loss of generality, we assume that the kNN returned by the above procedure are the first k patches of the data, *i.e.*, $\{\boldsymbol{p}_j\}_{j=1}^k$. Our goal now is to construct $\boldsymbol{U}$ from $\{\boldsymbol{p}_j\}_{j=1}^k$.

We postulate that a good $\boldsymbol{U}$ should have two properties. First, $\boldsymbol{U}$ should make the projected vectors $\{\boldsymbol{U}^T\boldsymbol{p}_j\}_{j=1}^k$ similar in both *magnitude* and *location*. This hypothesis follows from the observation that since $\{\boldsymbol{p}_j\}_{j=1}^k$ have small ℓ_2 distances from $\boldsymbol{q}$, it must hold that any $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ (hence $\boldsymbol{U}^T\boldsymbol{p}_i$ and $\boldsymbol{U}^T\boldsymbol{p}_j$) in the set should also be similar. Second, we require that each projected vector $\boldsymbol{U}^T\boldsymbol{p}_j$ contains as few non-zeros as possible, *i.e.*, *sparse*. The reason is related to the shrinkage step to be discussed in Section IV, because a vector of few non-zero coefficients has higher energy concentration and hence is more effective for denoising.

In order to satisfy these two criteria, we propose to consider the idea of *group sparsity*[1], which is characterized by the matrix $\ell_{1,2}$ norm, defined as [2]

$$\|\boldsymbol{X}\|_{1,2} \overset{\text{def}}{=} \sum_{i=1}^{d} \|\boldsymbol{x}_i\|_2, \tag{8}$$

for any matrix $\boldsymbol{X} \in \mathbb{R}^{d\times k}$, where $\boldsymbol{x}_i \in \mathbb{R}^k$ is the ith row of a matrix $\boldsymbol{X}$. In words, a small $\|\boldsymbol{X}\|_{1,2}$ makes sure that $\boldsymbol{X}$ has few non-zero entries, and the non-zero entries are located similarly in each column [6, 41]. A pictorial illustration is shown in Figure 1.

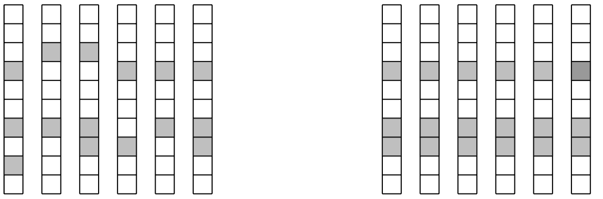
Going back to our problem, we propose to minimize the $\ell_{1,2}$-norm of the matrix $\boldsymbol{U}^T\boldsymbol{P}$:

$$\begin{array}{ll} \underset{\boldsymbol{U}}{\text{minimize}} & \|\boldsymbol{U}^T\boldsymbol{P}\|_{1,2} \\ \text{subject to} & \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}, \end{array} \tag{9}$$

where $\boldsymbol{P} \overset{\text{def}}{=} [\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k]$. The equality constraint in (9) ensures that $\boldsymbol{U}$ is orthonormal. Thus, the solution of (9) is an

[1] Group sparsity was first proposed by Cotter et al. for group sparse reconstruction [41] and later used by Mairal et al. for denoising [6], but towards a different end from the method presented in this paper.

[2] In general one can define $\|\boldsymbol{X}\|_{p,q} = \sum_{i=1}^d \|\boldsymbol{x}_i\|_q^p$ [6].



(a) sparse (b) group sparse

Fig. 1: Comparison between sparsity (where columns are sparse, but do not coordinate) and group sparsity (where all columns are sparse with similar locations).

orthonormal matrix $\boldsymbol{U}$ which maximizes the group sparsity of the data $\boldsymbol{P}$.

Interestingly, and surprisingly, the solution of (9) is indeed *identical* to the classical principal component analysis (PCA). The following lemma summarizes the observation.

Lemma 3: The solution to (9) is that

$$[\boldsymbol{U}, \boldsymbol{S}] = \text{eig}(\boldsymbol{P}\boldsymbol{P}^T), \tag{10}$$

where $\boldsymbol{S}$ is the corresponding eigenvalue matrix.

Proof: See Appendix B. ■

Remark 1: In practice, it is possible to improve the fidelity of the data matrix $\boldsymbol{P}$ by introducing a diagonal weight matrix

$$\boldsymbol{W} = \frac{1}{Z}\,\text{diag}\left\{e^{-\|\boldsymbol{q}-\boldsymbol{p}_1\|^2/h^2}, \ldots, e^{-\|\boldsymbol{q}-\boldsymbol{p}_k\|^2/h^2}\right\}, \tag{11}$$

for some user tunable parameter h and a normalization constant $Z \overset{\text{def}}{=} \boldsymbol{1}^T\boldsymbol{W}\boldsymbol{1}$. Consequently, we can define

$$\overline{\boldsymbol{P}} = \boldsymbol{P}\boldsymbol{W}^{1/2}. \tag{12}$$

Hence (10) becomes $[\boldsymbol{U}, \boldsymbol{S}] = \text{eig}(\boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T)$.

C. Relationship to Prior Works

The fact that (10) is the solution to a group sparsity minimization problem allows us to understand the performance of a number of existing denoising algorithms to some extent.

1) BM3D [3]: It is perhaps a misconception that the underlying principle of BM3D is to enforce sparsity of the 3-dimensional data volume (which we shall call it a 3-way tensor). However, what BM3D enforces is the *group sparsity* of the slices of the tensor, not the sparsity of the tensor.

To see this, we note that the 3-dimensional transforms in BM3D are separable (*e.g.*, DCT2 + Haar in its default setting). If the patches $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k$ are sufficiently similar, the DCT2 coefficients will be similar in *both* magnitude and frequency. Therefore, by fixing the frequency and tracing the DCT2 coefficients along the third axis, the output signal will be almost flat. Hence, the final Haar transform will return a sparse vector. Clearly, such sparsity is based on the stationarity of the DCT2 coefficients along the third axis. In essence, this is group sparsity.

2) HOSVD [9]: The true tensor sparsity can only be utilized by the high order singular value decomposition (HOSVD), which is recently studied in [9]. Let $\mathcal{P} \in \mathbb{R}^{\sqrt{d}\times\sqrt{d}\times k}$ be the tensor by stacking the patches $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k$ into a 3-dimensional array, HOSVD seeks three orthonormal matrices $\boldsymbol{U}^{(1)} \in \mathbb{R}^{\sqrt{d}\times\sqrt{d}}$, $\boldsymbol{U}^{(2)} \in \mathbb{R}^{\sqrt{d}\times\sqrt{d}}$, $\boldsymbol{U}^{(3)} \in \mathbb{R}^{k\times k}$ and an array $\mathcal{S} \in \mathbb{R}^{\sqrt{d}\times\sqrt{d}\times k}$, such that

$$\mathcal{S} = \mathcal{P} \times_1 \boldsymbol{U}^{(1)^T} \times_2 \boldsymbol{U}^{(2)^T} \times_3 \boldsymbol{U}^{(3)^T},$$

where $\times_k$ denotes a tensor mode-k multiplication [42].

As reported in [9], the performance of HOSVD is indeed worse than BM3D. This phenomenon can now be explained, because HOSVD ignores the fact that image patches tend to be group sparse instead of being tensor sparse.

3) Shape-adaptive BM3D [4]: As a variation of BM3D, SA-BM3D groups similar patches according to a shape-adaptive mask. Under our proposed framework, this shape-adaptive mask can be modeled as a spatial weight matrix $\boldsymbol{W}_s \in \mathbb{R}^{d\times d}$ (where the subscript s denotes *spatial*). Adding $\boldsymbol{W}_s$ to (12), we define

$$\overline{\boldsymbol{P}} = \boldsymbol{W}_s^{1/2}\boldsymbol{P}\boldsymbol{W}^{1/2}. \tag{13}$$

Consequently, the PCA of $\overline{\boldsymbol{P}}$ is equivalent to SA-BM3D. Here, the matrix $\boldsymbol{W}_s$ is used to control the relative emphasis of each pixel in the spatial coordinate.

4) BM3D-PCA [5] and LPG-PCA [7]: The idea of both BM3D-PCA and LPG-PCA is that given $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k$, $\boldsymbol{U}$ is determined as the principal components of $\boldsymbol{P} = [\boldsymbol{p}_1, \ldots, \boldsymbol{p}_k]$. Incidentally, such approaches arrive at the same result as (10), *i.e.*, the principal components are indeed the solution of a group sparse minimization. However, the motivation of using the group sparsity is not noticed in [5] and [7]. This provides additional theoretical justifications for both methods.

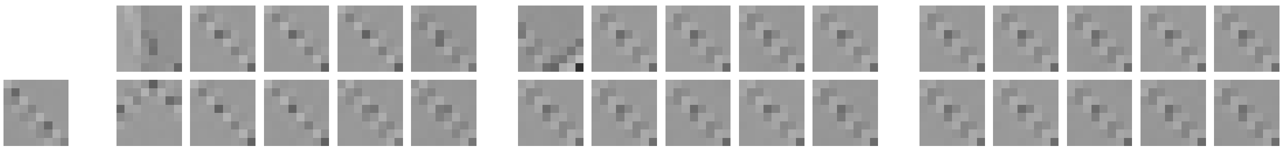
5) KSVD [19]: In KSVD, the dictionary plays the same role as $\boldsymbol{U}$. The dictionary can be trained either from the single noisy image, or from an external (generic or targeted) database. However, the training is performed once for *all* patches of the image. In other words, the noisy patches share a *common* dictionary. In our proposed method, *each* noisy patch has an individually trained basis matrix. Clearly, the latter approach, while computationally more expensive, is significantly more data adaptive than KSVD.

D. Improvement: Patch Selection Refinement

The optimization problem (9) suggests that the $\boldsymbol{U}$ computed from (10) is the optimal basis with respect to the reference patches $\{\boldsymbol{p}_j\}_{j=1}^k$. However, one issue that remains is how to improve the selection of k patches from the original n patches. Our proposed approach is to formulate the patch selection as an optimization problem

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{minimize}} & \boldsymbol{c}^T\boldsymbol{x} + \tau\varphi(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{x}^T\boldsymbol{1} = k, \quad 0 \le \boldsymbol{x} \le 1, \end{array} \tag{14}$$

where $\boldsymbol{c} = [c_1, \cdots, c_n]^T$ with $c_j \overset{\text{def}}{=} \|\boldsymbol{q} - \boldsymbol{p}_j\|_2$, $\varphi(\boldsymbol{x})$ is a penalty function and $\tau > 0$ is a parameter. In (14), each c_j



(a) $\boldsymbol{p}$ (b) $\varphi(\boldsymbol{x}) = 0$ (c) $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$ (d) $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$

Fig. 2: Refined patch matching results: (a) ground truth, (b) 10 best reference patches using $\boldsymbol{q}$ ($\sigma = 50$), (c) 10 best reference patches using $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$ (where $\tau = 1/(2n)$), (d) 10 best reference patches using $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$ (where $\tau = 1$).

is the distance $\|\boldsymbol{q} - \boldsymbol{p}_j\|_2$, and x_j is a weight indicating the emphasis of $\|\boldsymbol{q} - \boldsymbol{p}_j\|_2$. Therefore, the minimizer of (14) is a sequence of weights that minimize the overall distance.

To gain more insight into (14), we first consider the special case where the penalty term $\varphi(\boldsymbol{x}) = 0$. We claim that, under this special condition, the solution of (14) is equivalent to the original kNN solution in (7). This result is important, because kNN is a fundamental building block of all patch-based denoising algorithms. By linking kNN to the optimization formulation in (14) we provide a systematic strategy to improve the kNN.

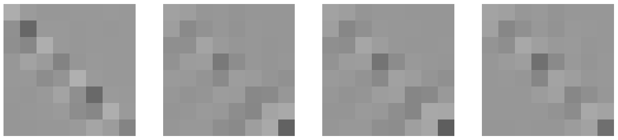
The proof of the equivalence between kNN and (14) can be understood via the following case study where $n = 2$ and $k = 1$. In this case, the constraints $\boldsymbol{x}^T\boldsymbol{1} = 1$ and $0 \leq \boldsymbol{x} \leq 1$ form a closed line segment in the positive quadrant. Since the objective function $\boldsymbol{c}^T\boldsymbol{x}$ is linear, the optimal point must be at one of the vertices of the line segment, which is either $\boldsymbol{x} = [0, 1]^T$, or $\boldsymbol{x} = [1, 0]^T$. Thus, by checking which of c_1 or c_2 is smaller, we can determine the optimal solution by setting $x_1 = 1$ if c_1 is smaller (and vice versa). Correspondingly, if $x_1 = 1$, then the first patch $\boldsymbol{p}_1$ should be selected. Clearly, the solution returned by the optimization is exactly the kNN solution. A similar argument holds for higher dimensions, hence justifying our claim.

Knowing that kNN can be formulated as (14), our next task is to choose an appropriate penalty term. The following are two possible choices.

1) Regularization by Cross Similarity: The first choice of $\varphi(\boldsymbol{x})$ is to consider $\varphi(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{B}\boldsymbol{x}$, where $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ is a symmetric matrix with $B_{ij} \overset{\text{def}}{=} \|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2$. Writing (14) explicitly, we see that (14) becomes

$$\underset{0\leq\boldsymbol{x}\leq 1,\, \boldsymbol{x}^T\boldsymbol{1}=k}{\text{minimize}} \quad \sum_j x_j\|\boldsymbol{q}-\boldsymbol{p}_j\|_2 + \tau\sum_{i,j} x_i x_j \|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2. \quad (15)$$

The penalized problem (15) suggests that the optimal k reference patches should not be determined merely from $\|\boldsymbol{q} - \boldsymbol{p}_j\|_2$ (which could be problematic due to the noise present in $\boldsymbol{q}$). Instead, a good reference patch should also be similar to all other patches that are selected. The cross similarity term $x_i x_j\|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2$ provides a way for such measure. This shares some similarities to the patch ordering concept proposed by Cohen and Elad [27]. The difference is that the patch ordering proposed in [27] is a shortest path problem that tries to organize the noisy patches, whereas ours is to solve a regularized optimization.



$\boldsymbol{p}$ Ground Truth | $\varphi(\boldsymbol{x}) = 0$ 28.29 dB | $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$ 28.50 dB | $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$ 29.30 dB

Fig. 3: Denoising results: A ground truth patch cropped from an image, and the denoised patches of using different improvement schemes. Noise standard deviation is $\sigma = 50$. $\tau = 1/(2n)$ for $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$ and $\tau = 1$ for $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$.

Problem (15) is in general not convex because the matrix $\boldsymbol{B}$ is not positive semidefinite. One way to relax the formulation is to consider $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$. Geometrically, the solution of using $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$ tends to identify patches that are close to the *sum* of all other patches in the set. In many cases, this is similar to $\varphi(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{B}\boldsymbol{x}$ which finds patches that are similar to every *individual* patch in the set. In practice, we find that the difference between $\varphi(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{B}\boldsymbol{x}$ and $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$ in the final denoising result (PSNR of the entire image) is marginal. Thus, for computational efficiency we choose $\varphi(\boldsymbol{x}) = \boldsymbol{1}^T\boldsymbol{B}\boldsymbol{x}$.

2) Regularization by First-pass Estimate: The second choice of $\varphi(\boldsymbol{x})$ is based on a *first-pass estimate* $\overline{\boldsymbol{p}}$ using some denoising algorithms, for example, BM3D or the proposed method without this patch selection step. In this case, by defining $e_j \overset{\text{def}}{=} \|\overline{\boldsymbol{p}} - \boldsymbol{p}_j\|_2$ we consider the penalty function $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$, where $\boldsymbol{e} = [e_1, \cdots, e_n]^T$. This implies the following optimization problem

$$\underset{0\leq\boldsymbol{x}\leq 1,\, \boldsymbol{x}^T\boldsymbol{1}=k}{\text{minimize}} \quad \textstyle\sum_j x_j\|\boldsymbol{q} - \boldsymbol{p}_j\|_2 + \tau\sum_j x_j\|\overline{\boldsymbol{p}} - \boldsymbol{p}_j\|_2. \quad (16)$$

By identifying the objective of (16) as $(\boldsymbol{c}+\tau\boldsymbol{e})^T\boldsymbol{x}$, we observe that (16) can be solved in closed form by locating the k smallest entries of the vector $\boldsymbol{c} + \tau\boldsymbol{e}$.

The interpretation of (16) is straight-forward: The linear combination of $\|\boldsymbol{q}-\boldsymbol{p}_j\|_2$ and $\|\overline{\boldsymbol{p}}-\boldsymbol{p}_j\|_2$ shows a competition between the noisy patch $\boldsymbol{q}$ and the first-pass estimate $\overline{\boldsymbol{p}}$. In most of the common scenarios, $\|\boldsymbol{q} - \boldsymbol{p}_j\|_2$ is preferred when noise level is low, whereas $\overline{\boldsymbol{p}}$ is preferred when noise level is high. This in turn requires a good choice of τ. Empirically, we find that $\tau = 0.01$ when $\sigma < 30$ and $\tau = 1$ when $\sigma > 30$ is a good balance between the performance and generality.

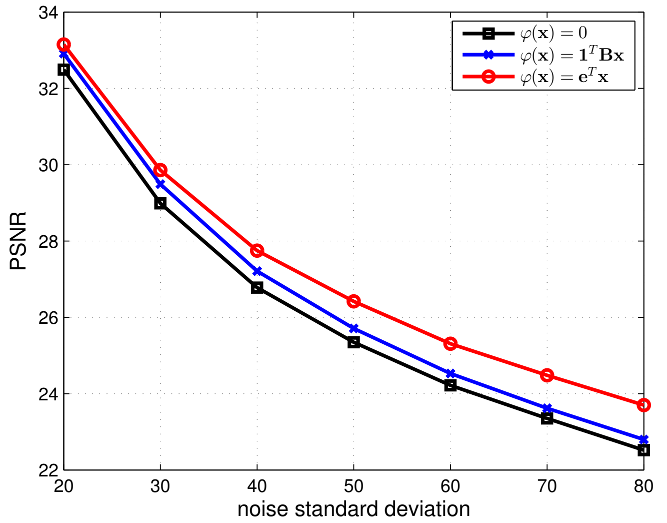


Fig. 4: Denoising results of three patch selection improvement schemes. The PSNR value is computed from a 432×381 image.

3) Comparisons: To demonstrate the effectiveness of the two proposed patch selection steps, we consider a ground truth (clean) patch shown in Figure 2 (a). From a pool of $n = 200$ reference patches, we apply an exhaustive search algorithm to choose $k = 40$ patches that best match with the noisy observation $\boldsymbol{q}$, where the first 10 patches are shown in Figure 2 (b). The results of the two selection refinement methods are shown in Figure 2 (c)-(d), where in both cases the parameter τ is adjusted for the best performance. For the case of $\varphi(\boldsymbol{x}) = \mathbf{1}^T\boldsymbol{B}\boldsymbol{x}$, we set $\tau = 1/(200n)$ when $\sigma < 30$ and $\tau = 1/(2n)$ when $\sigma > 30$. For the case of $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$, we use the denoised result of BM3D as the first-pass estimate $\overline{\boldsymbol{p}}$, and set $\tau = 0.01$ when $\sigma < 30$ and $\tau = 1$ when $\sigma > 30$. The results in Figure 3 show that the PSNR increases from 28.29 dB to 28.50 dB if we use $\varphi(\boldsymbol{x}) = \mathbf{1}^T\boldsymbol{B}\boldsymbol{x}$, and further increases to 29.30 dB if we use $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$. The full performance comparison is shown in Figure 4, where we show the PSNR curve for a range of noise levels of an image. Since the performance of $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$ is consistently better than $\varphi(\boldsymbol{x}) = \mathbf{1}^T\boldsymbol{B}\boldsymbol{x}$, in the rest of the paper we focus on $\varphi(\boldsymbol{x}) = \boldsymbol{e}^T\boldsymbol{x}$.

IV. Determine $\boldsymbol{\Lambda}$

In this section, we present our proposed method to determine $\boldsymbol{\Lambda}$ for a fixed $\boldsymbol{U}$. Our proposed method is based on the concept of a Bayesian MSE estimator.

A. Bayesian MSE Estimator

Recall that the noisy patch is related to the latent clean patch as $\boldsymbol{q} = \boldsymbol{p} + \boldsymbol{\eta}$, where $\boldsymbol{\eta} \overset{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \sigma^2\boldsymbol{I})$ denotes the noise. Therefore, the conditional distribution of $\boldsymbol{q}$ given $\boldsymbol{p}$ is

$$f(\boldsymbol{q}\,|\,\boldsymbol{p}) = \mathcal{N}(\boldsymbol{p},\, \sigma^2\boldsymbol{I}). \tag{17}$$

Assuming that the prior distribution $f(\boldsymbol{p})$ is known, it is natural to consider the Bayesian mean squared error (BMSE) between the estimate $\widehat{\boldsymbol{p}} \overset{\text{def}}{=} \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q}$ and the ground truth $\boldsymbol{p}$:

$$\text{BMSE} \overset{\text{def}}{=} \mathbb{E}_{\boldsymbol{p}}\left[\mathbb{E}_{\boldsymbol{q}|\boldsymbol{p}}\left[\|\widehat{\boldsymbol{p}} - \boldsymbol{p}\|_2^2 \;\middle|\; \boldsymbol{p}\right]\right]. \tag{18}$$

Here, the subscripts remark the distributions under which the expectations are taken.

The BMSE defined in (18) suggests that the optimal $\boldsymbol{\Lambda}$ should be the minimizer of the optimization problem

$$\boldsymbol{\Lambda} = \underset{\boldsymbol{\Lambda}}{\arg\min}\;\; \mathbb{E}_{\boldsymbol{p}}\left[\mathbb{E}_{\boldsymbol{q}|\boldsymbol{p}}\left[\left\|\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q} - \boldsymbol{p}\right\|_2^2 \;\middle|\; \boldsymbol{p}\right]\right]. \tag{19}$$

In the next subsection we discuss how to solve (19).

B. Localized Prior from the Targeted Database

Minimizing BMSE over $\boldsymbol{\Lambda}$ involves knowing the prior distribution $f(\boldsymbol{p})$. However, in general, the exact form of $f(\boldsymbol{p})$ is never known. This leads to many popular models in the literature, *e.g.*, Gaussian mixture model [35], the field of expert model [34, 43], and the expected patch log-likelihood model (EPLL) [20, 44].

One common issue of all these models is that the prior $f(\boldsymbol{p})$ is built from a generic database of patches. In other words, $f(\boldsymbol{p})$ models *all* patches in the database. As a result, $f(\boldsymbol{p})$ is often a high dimensional distribution with complicated shapes.

In our problem, the difficult prior modeling becomes a much simpler task when a targeted database is available. The reason is that while the shape of the distribution $f(\boldsymbol{p})$ is still unknown, the subsampled reference patches (which are few but highly representative) could be well approximated as samples drawn from a single Gaussian centered around some mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. Therefore, by appropriately estimating $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ of this *localized* prior, we can derive the optimal $\boldsymbol{\Lambda}$ as given by the following Lemma:

Lemma 4: Let $f(\boldsymbol{q}\,|\,\boldsymbol{p}) = \mathcal{N}(\boldsymbol{p}, \sigma^2\boldsymbol{I})$, and let $f(\boldsymbol{p}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ for any vector $\boldsymbol{\mu}$ and matrix $\boldsymbol{\Sigma}$, then the optimal $\boldsymbol{\Lambda}$ that minimizes (18) is

$$\boldsymbol{\Lambda} = \left(\text{diag}(\boldsymbol{G} + \sigma^2\boldsymbol{I})\right)^{-1}\text{diag}(\boldsymbol{G}), \tag{20}$$

where $\boldsymbol{G} \overset{\text{def}}{=} \boldsymbol{U}^T\boldsymbol{\mu}\boldsymbol{\mu}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\Sigma}\boldsymbol{U}$.

Proof: See Appendix C. ■

To specify $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, we let

$$\boldsymbol{\mu} = \sum_{j=1}^{k} w_j\boldsymbol{p}_j, \quad \boldsymbol{\Sigma} = \sum_{j=1}^{k} w_j(\boldsymbol{p}_j - \boldsymbol{\mu})(\boldsymbol{p}_j - \boldsymbol{\mu})^T, \tag{21}$$

where w_j is the jth diagonal entry of $\boldsymbol{W}$ defined in (11). Intuitively, an interpretation of (21) is that $\boldsymbol{\mu}$ is the non-local mean of the reference patches. However, the more important part of (21) is $\boldsymbol{\Sigma}$, which measures the *uncertainty* of the reference patches with respect to $\boldsymbol{\mu}$. This uncertainty measure makes some fundamental improvements to existing methods which will be discussed in Section IV-C.

We note that Lemma 4 holds even if $f(\boldsymbol{p})$ is not Gaussian. In fact, for any distribution $f(\boldsymbol{p})$ with the first cumulant $\boldsymbol{\mu}$ and the second cumulant $\boldsymbol{\Sigma}$, the optimal solution in (41) still holds. This result is equivalent to the classical linear minimum MSE (LMMSE) estimation [45].

From a computational perspective, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ defined in (21) lead to a very efficient implementation as illustrated by the following lemma.

Lemma 5: Using $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ defined in (21), the optimal $\boldsymbol{\Lambda}$ is given by

$$\boldsymbol{\Lambda} = \left(\mathrm{diag}(\boldsymbol{S} + \sigma^2 \boldsymbol{I})\right)^{-1} \mathrm{diag}(\boldsymbol{S}), \tag{22}$$

where $\boldsymbol{S}$ is the eigenvalue matrix of $\boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T$.

Proof: See Appendix D. ■

Combining Lemma 5 with Lemma 3, we observe that for any set of reference patches $\{\boldsymbol{p}_j\}_{j=1}^k$, $\boldsymbol{U}$ and $\boldsymbol{\Lambda}$ can be determined *simultaneously* through the eigen-decomposition of $\boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T$. Therefore, we arrive at the overall algorithm shown in Algorithm 1.

Algorithm 1 Proposed Algorithm

```
Input: Noisy patch $q$, noise variance $\sigma^2$, and clean reference patches $p_1, \ldots, p_k$
Output: Estimate $\widehat{p}$
Learn $U$ and $\Lambda$
    • Form data matrix $P$ and weight matrix $W$
    • Compute eigen-decomposition $[U, S] = \mathrm{eig}(PWP^T)$
    • Compute $\Lambda = (\mathrm{diag}(S + \sigma^2 I))^{-1} \mathrm{diag}(S)$
Denoise: $\widehat{p} = U \Lambda U^T q$.
```

C. Relationship to Prior Works

It is interesting to note that many existing patch-based denoising algorithms assume some notions of prior, either explicitly or implicitly. In this subsection, we mention a few of the important ones. For notational simplicity, we will focus on the ith diagonal entry of $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_1, \ldots, \lambda_d\}$.

1) BM3D [3], Shape-Adaptive BM3D [4] and BM3D-PCA [5] : BM3D and its variants have two denoising steps. In the first step, the algorithm applies a basis matrix $\boldsymbol{U}$ (either a pre-defined basis such as DCT, or a basis learned from PCA). Then, it applies a hard-thresholding to the projected coefficients to obtain a filtered image $\overline{\boldsymbol{p}}$. In the second step, the filtered image $\overline{\boldsymbol{p}}$ is used as a pilot estimate to the desired spectral component

$$\lambda_i = \frac{(\boldsymbol{u}_i^T \overline{\boldsymbol{p}})^2}{(\boldsymbol{u}_i^T \overline{\boldsymbol{p}})^2 + \sigma^2}. \tag{23}$$

Following our proposed Bayesian framework, we observe that the role of using $\overline{\boldsymbol{p}}$ in (23) is equivalent to assuming a dirac delta prior

$$f(\boldsymbol{p}) = \delta(\boldsymbol{p} - \overline{\boldsymbol{p}}). \tag{24}$$

In other words, the prior that BM3D assumes is concentrated at one point, $\overline{\boldsymbol{p}}$, and there is no measure of uncertainty. As a result, the algorithm becomes highly sensitive to the first-pass estimate. In contrast, (21) suggests that the first-pass estimate can be defined as a non-local mean solution. Additionally, we incorporate a covariance matrix $\boldsymbol{\Sigma}$ to measure the uncertainty of observing $\boldsymbol{\mu}$. These provide a more robust estimate to the denoising algorithm which is absent from BM3D and its variants.

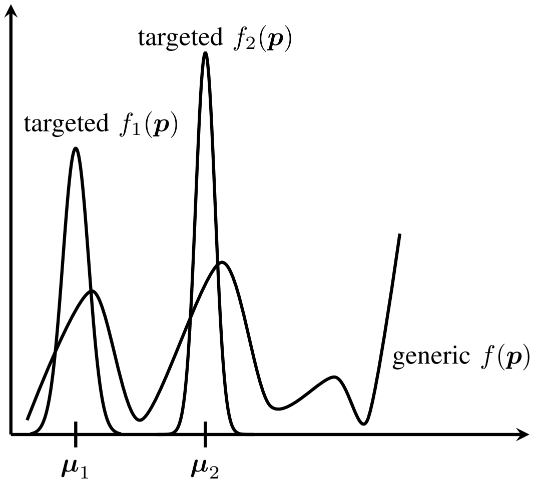

targeted $f_2(\boldsymbol{p})$
targeted $f_1(\boldsymbol{p})$
generic $f(\boldsymbol{p})$
$\boldsymbol{\mu}_1$
$\boldsymbol{\mu}_2$

Fig. 5: Generic prior vs targeted priors: Generic prior has an arbitrary shape spanned over the entire space; Targeted priors are concentrated at the means. In this figure, $f_1(\boldsymbol{p})$ and $f_2(\boldsymbol{p})$ illustrate two targeted priors which correspond to two patches of an image.

2) LPG-PCA [7]: In LPG-PCA, the ith spectral component λ_i is defined as

$$\lambda_i = \frac{(\boldsymbol{u}_i^T \boldsymbol{q})^2 - \sigma^2}{(\boldsymbol{u}_i^T \boldsymbol{q})^2}, \tag{25}$$

where $\boldsymbol{q}$ is the noisy patch. The (implicit) assumption in [7] is that $(\boldsymbol{u}_i^T \boldsymbol{q})^2 \approx (\boldsymbol{u}_i^T \boldsymbol{p})^2 + \sigma^2$, and so substituting $(\boldsymbol{u}_i^T \boldsymbol{p})^2 \approx (\boldsymbol{u}_i^T \boldsymbol{q})^2 - \sigma^2$ into (5) would yield (25). However, the assumption implies the existence of a perturbation $\Delta\boldsymbol{p}$ such that $(\boldsymbol{u}_i^T \boldsymbol{q})^2 = (\boldsymbol{u}_i^T(\boldsymbol{p}+\Delta\boldsymbol{p}))^2 + \sigma^2$. Letting $\overline{\boldsymbol{p}} = \boldsymbol{p} + \Delta\boldsymbol{p}$, we see that LPG-PCA implicitly assumes a dirac prior as in (23) and (24). The denoising result depends on the magnitude of $\Delta\boldsymbol{p}$.

3) Generic Global Prior [22]: As a comparison to methods using generic databases such as [22], we note that the key difference lies in the usage of a *global* prior versus a *local* prior. Figure 5 illustrates the concept pictorially. The generic (global) prior $f(\boldsymbol{p})$ covers the entire space, whereas the targeted (local) prior is concentrated at its mean. The advantage of the local prior is that it allows one to denoise an image with few reference patches. It saves us from the intractable computation of learning the global prior, which is a high-dimensional non-parametric function.

4) Generic Local Prior – EPLL [20], K-SVD [19, 33]: Compared to learning-based methods that use local priors, such as EPLL [20] and K-SVD [19, 33], the most important merit of the proposed method is that it requires significantly fewer training samples. A thorough justification will be discussed in Section V.

5) PLOW [46] : PLOW has a similar design process as ours by considering the optimal filter. The major difference is that in PLOW, the denoising filter is derived from the full covariance matrices of the data and noise. As we will see in the next subsection, the linear denoising filter of our work is a truncated SVD matrix computed from a set of similar patches. The merit of the truncation is that it often reduces MSE in the bias-variance trade off [40].

D. *Improving* $\boldsymbol{\Lambda}$

The Bayesian framework proposed above can be generalized to further improve the denoising performance. Referring to (19), we observe that the BMSE optimization can be reformulated to incorporate a penalty term in $\boldsymbol{\Lambda}$. Here, we consider the following ℓ_α penalized BMSE:

$$\mathrm{BMSE}_\alpha \overset{\mathrm{def}}{=} \mathbb{E}_{\boldsymbol{p}}\left[\mathbb{E}_{\boldsymbol{q}|\boldsymbol{p}}\left[\left\|\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q}-\boldsymbol{p}\right\|_2^2 \middle| \boldsymbol{p}\right]\right] + \gamma\|\boldsymbol{\Lambda}\mathbf{1}\|_\alpha, \tag{26}$$

where $\gamma > 0$ is the penalty parameter, and $\alpha \in \{0, 1\}$ controls which norm to be used. The solution to the minimization of (26) is given by the following lemma.

Lemma 6: Let s_i be the ith diagonal entry in $\boldsymbol{S}$, where $\boldsymbol{S}$ is the eigenvalue matrix of $\boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T$, then the optimal $\boldsymbol{\Lambda}$ that minimizes BMSE_α is $\mathrm{diag}\{\lambda_1, \cdots, \lambda_d\}$, where

$$\lambda_i = \max\left(\frac{s_i - \gamma/2}{s_i + \sigma^2}, 0\right), \qquad \text{for } \alpha = 1, \tag{27}$$

$$\lambda_i = \frac{s_i}{s_i + \sigma^2}\mathbb{1}\left(\frac{s_i^2}{s_i + \sigma^2} > \gamma\right), \qquad \text{for } \alpha = 0. \tag{28}$$

Proof: See Appendix E. ■

The motivation of introducing an ℓ_α-norm penalty in (26) is related the group sparsity used in defining $\boldsymbol{U}$. Recall from Section III that since $\boldsymbol{U}$ is the optimal solution to a group sparsity optimization, only few of the entries in the ideal projection $\boldsymbol{U}^T\boldsymbol{p}$ should be non-zero. Consequently, it is desired to require $\boldsymbol{\Lambda}$ to be sparse so that $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q}$ has similar spectral components as that of $\boldsymbol{p}$.

To demonstrate the effectiveness of the proposed ℓ_α formulation, we consider the example patch shown in Figure 3. For a refined database of $k = 40$ patches, we consider the original minimum BMSE solution ($\gamma = 0$), the ℓ_0 solution with $\gamma = 0.02$, and the ℓ_1 solution with $\gamma = 0.02$. The results in Figure 6 show that with the proposed penalty term, the new BMSE_α solution performs consistently better than the original BMSE solution.

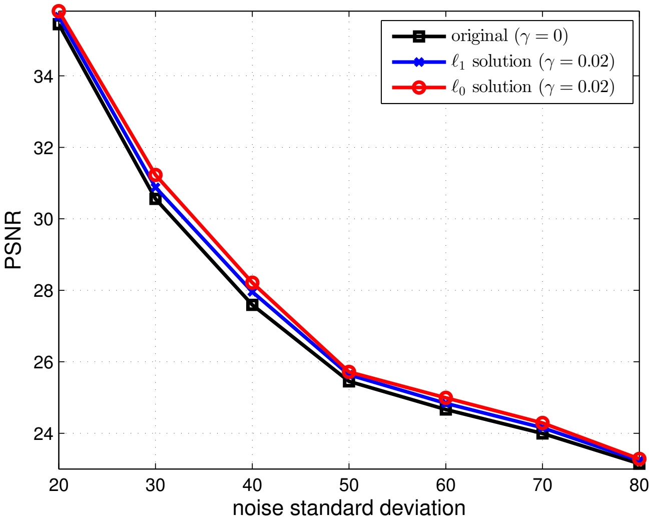


Fig. 6: Comparisons of the ℓ_1 and ℓ_0 adaptive solutions over the original solution with $\gamma = 0$. The PSNR value for each noise level is averaged over 100 independent trials to reduce the bias due to a particular noise realization.

V. Experimental Results

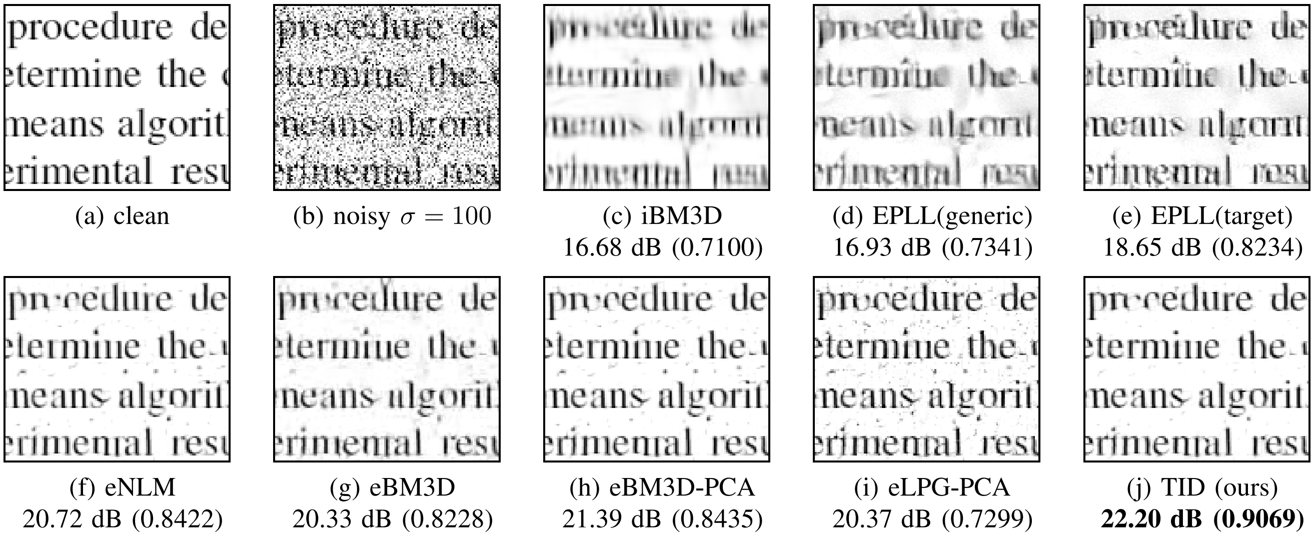
In this section, we present additional experimental results.

A. *Comparison Methods*

The methods we choose for comparison are BM3D [3], BM3D-PCA [5], LPG-PCA [7], NLM [1], EPLL [20] and KSVD [19]. We name our proposed method as Targeted Image Denoising (TID). Except for EPLL and KSVD, all other four methods are internal denoising methods. We re-implement and modify the internal methods so that patch search is performed over the targeted external databases. These methods are iterated for two times where the solution of the first step is used as a basic estimate for the second step. The specific settings of each algorithm are as follows:

1) BM3D [3]: As a benchmark of internal denoising, we run the original BM3D code provided by the author[3]. Default parameters are used in the experiments, *e.g.*, the search window is 39×39. We have included a discussion in Section V-B about the influence of different search window size to the denoising performance. As for external denoising, we implement an external version of BM3D. To ensure a fair comparison, we set the search window identical to other external denoising methods.
2) BM3D-PCA [5] and LPG-PCA [7]: $\boldsymbol{U}$ is learned from the best k external patches, which is the same as in our proposed method. $\boldsymbol{\Lambda}$ is computed following (23) for BM3D-PCA and (25) for LPG-PCA. In BM3D-PCA's first step, the threshold is set to 2.7σ.
3) NLM [1]: The weights in NLM are computed according to a Gaussian function of the ℓ_2 distance of two patches [47, 48]. However, instead of using all reference patches in the database, we use the best k patches following [2].
4) EPLL [20]: In EPLL, the default patch prior is learned from a generic database (200,000 8×8 patches). For a fair comparison, we train the prior distribution from our targeted databases using the same EM algorithm mentioned in [20].
5) KSVD [19]: In KSVD, two dictionaries are trained including a global dictionary and a targeted dictionary. The global dictionary is trained from a generic database of 100,000 8×8 patches by the KSVD authors. The targeted dictionary is trained from a targeted database of 100,000 8×8 patches containing similar content of

[3] http://www.cs.tut.fi/~foi/GCF-BM3D/



(a) clean

(b) noisy $\sigma = 100$

(c) iBM3D 16.68 dB (0.7100)

(d) EPLL(generic) 16.93 dB (0.7341)

(e) EPLL(target) 18.65 dB (0.8234)

(f) eNLM 20.72 dB (0.8422)

(g) eBM3D 20.33 dB (0.8228)

(h) eBM3D-PCA 21.39 dB (0.8435)

(i) eLPG-PCA 20.37 dB (0.7299)

(j) TID (ours) **22.20 dB (0.9069)**

Fig. 7: Denoising text images: Visual comparison and objective comparison (PSNR and SSIM in the parenthesis). The test image size is of 127×104. Prefix "i" stands for internal denoising (*i.e.*, single-image denoising), and prefix "e" stands for external denoising (*i.e.*, using external databases).

the noisy image. Both dictionaries are of size 64×256. To emphasize the difference between the original algorithms (which are single-image denoising algorithms) and the corresponding new implementations for external databases, we denote the original, (single-image) denoising algorithms with "i" (internal), and the corresponding new implementations for external databases with "e" (external).

We add zero-mean Gaussian noise with standard deviations from $\sigma = 20$ to $\sigma = 80$ to the test images. The patch size is set as 8×8 ($i.e., d = 64$). Two quality metrics, namely Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) are used to evaluate the objective quality of the denoised images.

B. Denoising Text and Documents

Our first experiment considers denoising a text image with the help of other similar but non-identical texts. This is a simplified setup for problems such as hand writing, bar codes and license plates. To prepare the experiment, we add noise to a randomly chosen document and use 9 other clean documents (of the same font size) for denoising.

1) Denoising Performance: Figure 7 shows the denoising results when we add excessive noise ($\sigma = 100$) to an image. Among all the methods, TID yields the highest PSNR and SSIM values. The PSNR is 5 dB better than the benchmark BM3D (internal) denoising algorithm. Some existing learning-based methods, such as EPLL, do not perform well due to the insufficient training samples from the targeted database. Compared to other external denoising methods, TID shows a better utilization of the targeted database.

Since the default search window size for internal BM3D is only 39×39, we conduct an experiment to explore the effect of different search window sizes for BM3D. The PSNR results are shown in Table 1. We see that a larger window size improves the BM3D denoising performance since more patch redundancy can be exploited. However, even if we extend the search to an external database (which is the case for eBM3D), the performance is still worse than the proposed method.

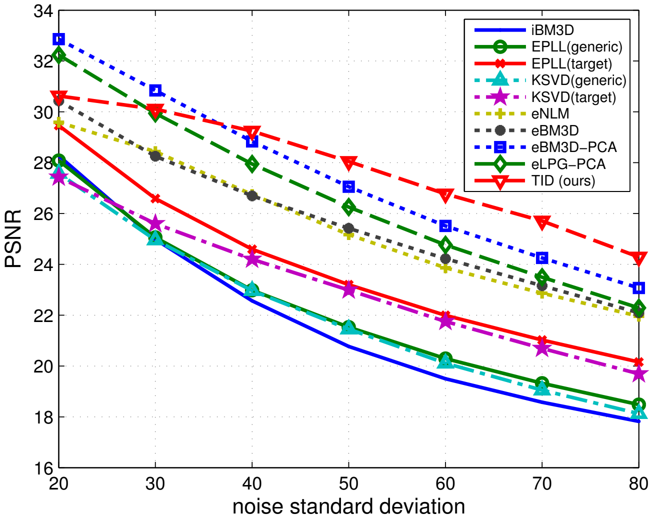


Fig. 8: Text image denoising: Average PSNR vs noise levels. In this plot, each PSNR value is averaged over 8 test images. The typical size of a test image is about 300×200.

In Figure 8, we plot and compare the average PSNR values on 8 test images over a range of noise levels. We observe that at low noise levels ($\sigma < 30$), TID performs worse than eBM3D-PCA and eLPG-PCA. One reason is that the pilot estimates $\overline{\boldsymbol{p}}$ for these two methods are more reliable at low noise, where the distinctive features of the text pattern could be utilized. However, as noise level increases, it becomes more problematic to identify text from the noise. Thus, the pilot

	search window size	$\sigma = 30$	$\sigma = 50$	$\sigma = 70$
	(39×39)	24.73	20.44	18.21
BM3D	(119×119)	26.91	21.24	19.01
	(199×199)	28.02	21.53	19.27
eBM3D	(external database)	28.48	25.49	23.09
TID (ours)	(external database)	**30.79**	**28.43**	**25.97**

Table 1: PSNR results using BM3D with different search window sizes and the proposed method. We test the performance for three different noise levels ($\sigma = 30, 50, 70$). The reported PSNR is computed on the entire image of size 301×218.

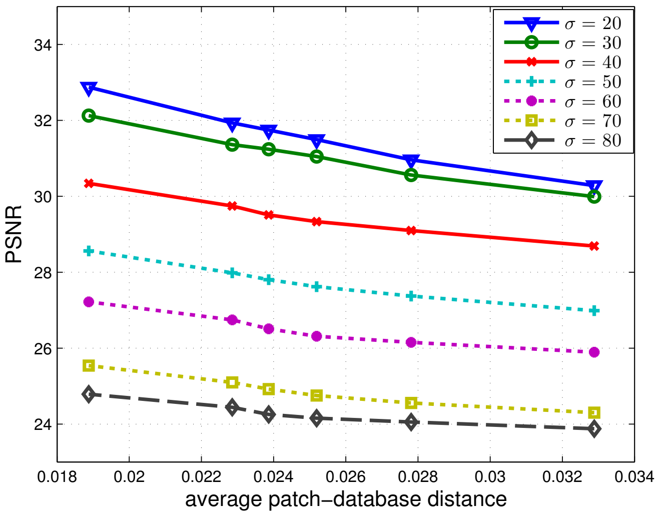


Fig. 9: Denoising performance in terms of the database quality. The average patch-to-database distance $\overline{d}(\mathcal{P})$ is a measure of the database quality.

estimates degrade and so these two methods perform worse. For example, for $\sigma = 60$, our average PSNR result is 1.26 dB better than the second best result by eBM3D-PCA.

For the two learning-based methods, *i.e.,* EPLL and KSVD, as can be seen, using a targeted database yields better results than using a generic database, which validates the usefulness of a targeted database. However, they perform worse than other non-learning methods. One reason is that a *large* number of training samples are needed for these learning-based methods – for EPLL, the large number of samples is needed to build the Gaussian mixtures, whereas for KSVD, the large number of samples is needed to train the over-complete dictionary. In contrast, TID is fully functional even if the database is small.

2) Database Quality: We are interested in knowing how the quality of a database would affect the denoising performance, as that could offer us important insights about the sensitivity of the algorithm. To this end, we compute the average distance from a given database to a clean image that we would like to obtain. Specifically, for each patch $\boldsymbol{p}_i \in \mathbb{R}^d$ in a clean image containing m patches and a database $\mathcal{P}$ of n patches, we compute its minimum distance

$$d(\boldsymbol{p}_i, \mathcal{P}) \overset{\text{def}}{=} \min_{\boldsymbol{p}_j \in \mathcal{P}} \|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2 / \sqrt{d}.$$

The average patch-database distance is then defined as $\overline{d}(\mathcal{P}) \overset{\text{def}}{=} (1/m)\sum_{i=1}^{m} d(\boldsymbol{p}_i, \mathcal{P})$. Therefore, a smaller $\overline{d}(\mathcal{P})$ indicates that the database is more relevant to the ground truth (clean) image.

Figure 9 shows the results of six databases $\mathcal{P}$, where each is a random subset of the original targeted database. For all noise levels ($\sigma = 20$ to 80), PSNR decreases linearly as the patch-to-database distance increase, Moreover, the decay rate is slower for higher noise levels. The result suggests that the quality of the database has a more significant impact under low noise conditions, and less under high noise conditions.

C. Denoising Multiview Images

Our second experiment considers the scenario of capturing images using a multiview camera system. The multiview images are captured at different viewing positions. Suppose that one or more cameras are not functioning properly so that some images are corrupted with noise. Our goal is to demonstrate that with the help of the other clean views, the noisy view could be restored.

To simulate the experiment, we download 4 multiview datasets from Middlebury Computer Vision Page[4]. Each set of images consists of 5 views. We add i.i.d. Gaussian noise to one view and then use the rest 4 views to assist in denoising.

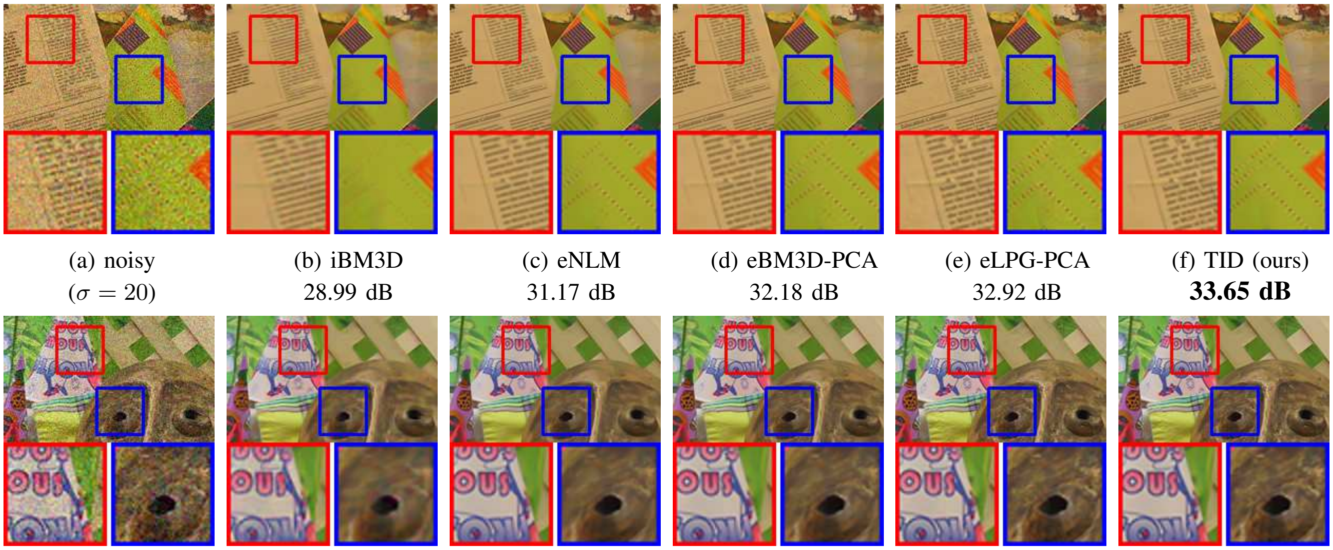
In Figure 10, we visually show the denoising results of the "Barn" and "Cone" multiview datasets. Compared to other methods, TID has the highest PSNR values. The magnified areas indicate that our proposed method removes the noise significantly and better reconstructs some fine details. In Figure 11, we plot and compare the average PSNR values on 4 test images over a range of noise levels. The results show that TID is consistently better than its competitors. For example, for $\sigma = 50$, TID is 1.06 dB better than eBM3D-PCA and 2.73 dB better than iBM3D. The superior performance confirms our hypothesis that even with a good database, not all denoising algorithms would perform equally well. In fact, in order to maximize the performance, one still has to carefully design an algorithm that can fully utilize the database.

D. Denoising Human Faces

Our third experiment considers denoising face images.

In the first part of this experiment, we use the Gore face dataset from [49]. The upper row of Figure 12 shows some examples, where each image has a size 60×80. We add noise to 8 randomly chosen images in this dataset and use the other images (29 images in our experiment) for denoising. The results of the experiments are shown in the bottom row of Figure 12. The result shows that even though the facial expressions are different and there are misalignments between images, TID still generates reasonable denoising results. In Figure 13, we plot the average PSNR curves over the 8 test images, where we see consistently better results compared to other methods.

[4] http://vision.middlebury.edu/stereo/



(a) noisy ($\sigma = 20$) (b) iBM3D 28.99 dB (c) eNLM 31.17 dB (d) eBM3D-PCA 32.18 dB (e) eLPG-PCA 32.92 dB (f) TID (ours) **33.65 dB**

(g) noisy ($\sigma = 20$) (h) iBM3D 28.77 dB (i) eNLM 29.97 dB (j) eBM3D-PCA 31.03 dB (k) eLPG-PCA 31.80 dB (l) TID (ours) **32.18 dB**

Fig. 10: Multiview image denoising: Visual comparison and objective comparison (PSNR). [Top] "Barn"; [Bottom] "Cone".

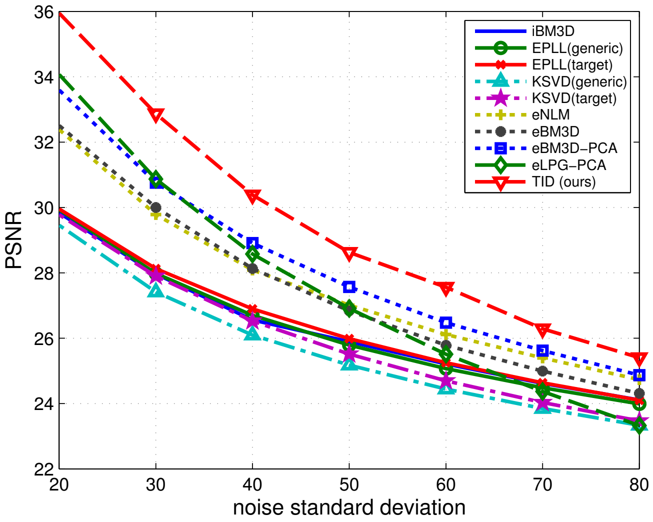
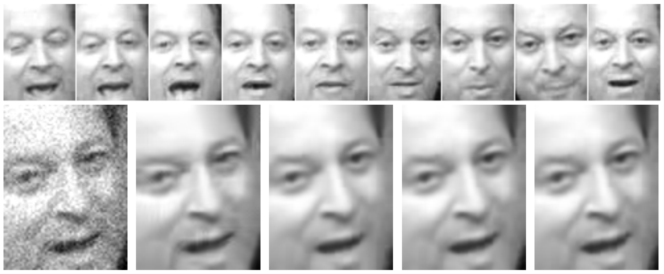


Fig. 11: Multiview image denoising: Average PSNR vs noise levels. In this plot, each PSNR value is averaged over 4 test images. The typical size of a test image is about 450×350.

As a second experiment, we use the FEI face dataset from [50]. The dataset consists of 100 aligned frontal face images. We randomly choose 8 of these images as the test images and use the remaining 92 images for denoising. Figure 14 shows an example of a noisy image and 4 database images. Unlike the Gore face dataset where the subject remains the same, the FEI dataset contains faces of different subjects. The results are shown in Figure 15, where we plot the average PSNR curves. From the plot, we observe that the proposed method still yields the highest PSNR values consistently, although the marginal difference with other methods is less significant as compared to the Gore dataset.



noisy ($\sigma = 20$) iBM3D 32.04 dB eNLM 32.74 dB eBM3D-PCA 33.29 dB TID (ours) **33.86 dB**

Fig. 12: Face denoising of Gore dataset [49]. [Top] Database images; [Bottom] Denoising results.

E. Runtime Comparison

Our current implementation is in MATLAB (single thread). The runtime is about 144s to denoise an image (301×218) with a targeted database consisting of 9 images of similar sizes. The code is run on an Intel Core i7-3770 CPU. In Table 2, we show a runtime comparison with other methods. We observe that the runtime of TID is indeed not significantly worse than other external methods. In particular, the runtime of the proposed method is in the same order of magnitude as eNLM, eBM3D, eBM3D-PCA and eLPG-PCA.

We remark that most of the runtime of the proposed method is spent on searching similar patches and computing SVD. Speed improvement for the proposed method is possible. First, we can apply techniques to enable fast patch search, e.g., patch match [51, 52], KD tree [53], or fast SVD [54]. Second, random sampling schemes can be applied to further reduce the computational complexity [21]. Third, since the denoising is independently performed on each patch, GPU can be used to parallelize the computation.

	iBM3D	EPLL(generic)	EPLL(target)	KSVD(generic)	KSVD(target)
runtime (sec)	0.97	35.17	10.21	0.32	0.13
	eNLM	eBM3D	eBM3DPCA	eLPGPCA	TID (ours)
runtime (sec)	95.68	99.17	102.21	102.14	144.33

Table 2: Runtime comparison for different denoising methods. The test image is of size 301×218. For EPLL and KSVD methods, the time to train a finite Gaussian mixture model and the time to learn a dictionary is not included in the above runtime. For other external denoising methods, the targeted database consists of 9 images of similar sizes of the test image.

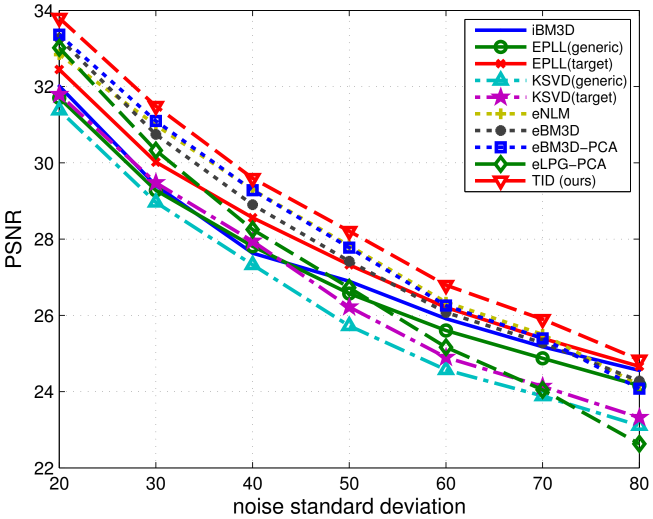


Fig. 13: Face denoising of Gore dataset [49]: Average PSNR vs noise levels. In this plot, each PSNR value is averaged over 8 test images. Each test image is of size 60×80.

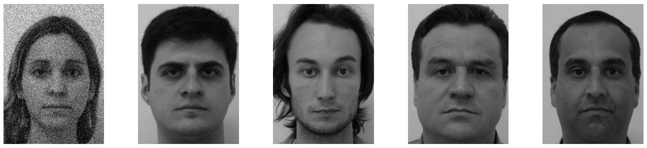


Fig. 14: FEI face dataset [50]: The first image is one (out of 8) noisy test images with $\sigma = 40$. The remaining are 4 (out of 92) clean database images. All images have size 260×360.

F. Discussion and Future Work

Two open questions shall be explored in a near future. First, assuming that there is a perturbation on the database, how much MSE will be changed? Answering the question will provide us information about the sensitivity of the algorithm when there are changes in font size (in the text example), view angle (in the multiview example), and facial expression (in the face example). Second, given a clean patch, how many patches do we need to put in the database in order to ensure that the clean patch is close to at least one of the patches in the database? The answer to this question will inform us about the size of the targeted database.

VI. Conclusion

Classical image denoising methods based on a single noisy input or generic databases are approaching their performance limits. We proposed an adaptive image denoising algorithm using targeted databases. The proposed method, called Targeted Image Denosing (TID), applies a group sparsity minimization and a localized prior to learn the optimal denoising filter. We show that TID generalizes a number of existing patch-based denoising algorithms such as BM3D, BM3D-PCA, Shape-adaptive BM3D, LPG-PCA, and EPLL. Based on the new framework, we proposed improvement schemes, namely an improved patch selection procedure for determining the basis matrix and a penalized minimization for determining the spectral coefficients. For a variety of scenarios including text, multiview images and faces, we demonstrated empirically that TID has superior performance over existing methods. With the increasing amount of image data online, data-dependent algorithms seem to be a plausible direction for future denoising research.

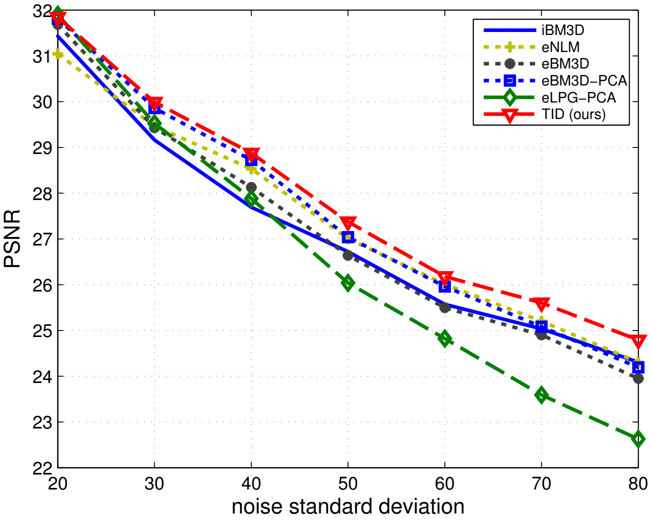


Fig. 15: Face denoising of FEI face dataset [50]: Average PSNR vs noise levels. In this plot, each PSNR value is averaged over 8 test images. Each test image is of size 260×360.

Appendix

A. Proof of Lemma 2

Proof: From (4), the optimization is

$$\begin{array}{cl} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_d,\lambda_1,\ldots,\lambda_d}{\text{minimize}} & \sum_{i=1}^{d} \left[(1-\lambda_i)^2(\boldsymbol{u}_i^T\boldsymbol{p})^2 + \sigma^2\lambda_i^2\right] \\ \text{subject to} & \boldsymbol{u}_i^T\boldsymbol{u}_i = 1, \quad \boldsymbol{u}_i^T\boldsymbol{u}_j = 0. \end{array}$$

Since each term in the sum of the objective function is non-negative, we can consider the minimization over each individual term separately. This gives

$$\begin{aligned} \underset{\boldsymbol{u}_i,\lambda_i}{\text{minimize}} \quad & (1-\lambda_i)^2(\boldsymbol{u}_i^T\boldsymbol{p})^2+\sigma^2\lambda_i^2 \\ \text{subject to} \quad & \boldsymbol{u}_i^T\boldsymbol{u}_i=1. \end{aligned} \tag{29}$$

In (29), we temporarily dropped the orthogonality constraint $\boldsymbol{u}_i^T\boldsymbol{u}_j = 0$, which will be taken into account later. The Lagrangian function of (29) is

$$\mathcal{L}(\boldsymbol{u}_i,\lambda_i,\beta)=(1-\lambda_i)^2(\boldsymbol{u}_i^T\boldsymbol{p})^2+\sigma^2\lambda_i^2+\beta(1-\boldsymbol{u}_i^T\boldsymbol{u}_i),$$

where β is the Lagrange multiplier. Differentiating $\mathcal{L}$ with respect to $\boldsymbol{u}_i$, λ_i and β yields

$$\frac{\partial\mathcal{L}}{\partial\lambda_i}=-2(1-\lambda_i)(\boldsymbol{u}_i^T\boldsymbol{p})^2+2\sigma^2\lambda_i \tag{30}$$

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{u}_i}=2(1-\lambda_i)^2(\boldsymbol{u}_i^T\boldsymbol{p})\boldsymbol{p}-2\beta\boldsymbol{u}_i \tag{31}$$

$$\frac{\partial\mathcal{L}}{\partial\beta}=1-\boldsymbol{u}_i^T\boldsymbol{u}_i. \tag{32}$$

Setting $\partial\mathcal{L}/\partial\lambda_i=0$ yields

$$\lambda_i=(\boldsymbol{u}_i^T\boldsymbol{p})^2/\left((\boldsymbol{u}_i^T\boldsymbol{p})^2+\sigma^2\right). \tag{33}$$

Substituting this λ_i into (31) and setting $\partial\mathcal{L}/\partial\boldsymbol{u}_i=0$ yields

$$\frac{2\sigma^4(\boldsymbol{u}_i^T\boldsymbol{p})\boldsymbol{p}}{\left((\boldsymbol{u}_i^T\boldsymbol{p})^2+\sigma^2\right)^2}-2\beta\boldsymbol{u}_i=0. \tag{34}$$

Therefore, the optimal pair ($\boldsymbol{u}_i$, β) of (29) must be the solution of (34). The corresponding λ_i can be calculated via (33).

Referring to (34), we observe two possible scenarios. First, if $\boldsymbol{u}_i$ is any unit vector orthogonal to $\boldsymbol{p}_i$, and $\beta=0$, then (34) can be satisfied. This is a trivial solution, because $\boldsymbol{u}_i\perp\boldsymbol{p}$ implies $\boldsymbol{u}_i^T\boldsymbol{p}=0$, and hence $\lambda_i=0$. The second case is that

$$\boldsymbol{u}_i=\boldsymbol{p}/\|\boldsymbol{p}\|_2, \quad \text{and} \quad \beta=\frac{\sigma^4\|\boldsymbol{p}\|^2}{(\|\boldsymbol{p}\|^2+\sigma^2)^2}. \tag{35}$$

Substituting (35) shows that (34) is satisfied. This is the non-trivial solution. The corresponding λ_i in this case is $\|\boldsymbol{p}\|^2/(\|\boldsymbol{p}\|^2+\sigma^2)$.

Finally, taking into account of the orthogonality constraint $\boldsymbol{u}_i^T\boldsymbol{u}_j=0$ if $i\neq j$, we can choose $\boldsymbol{u}_1=\boldsymbol{p}/\|\boldsymbol{p}\|_2$, and $\boldsymbol{u}_2\perp\boldsymbol{u}_1$, $\boldsymbol{u}_3\perp\{\boldsymbol{u}_1,\boldsymbol{u}_2\}$, $\ldots$, $\boldsymbol{u}_d\perp\{\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots\boldsymbol{u}_{d-1}\}$. Therefore, the denoising result is

$$\widehat{\boldsymbol{p}}=\boldsymbol{U}\left(\text{diag}\left\{\frac{\|\boldsymbol{p}\|^2}{\|\boldsymbol{p}\|^2+\sigma^2},0,\ldots,0\right\}\right)\boldsymbol{U}^T\boldsymbol{q},$$

where $\boldsymbol{U}$ is any orthonormal matrix with the first column $\boldsymbol{u}_1=\boldsymbol{p}/\|\boldsymbol{p}\|_2$. ■

B. Proof of Lemma 3

Proof: Let $\boldsymbol{u}_i$ be the ith column of $\boldsymbol{U}$. Then, (9) becomes

$$\begin{aligned} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_d}{\text{minimize}} \quad & \textstyle\sum_{i=1}^d\|\boldsymbol{u}_i^T\boldsymbol{P}\|_2 \\ \text{subject to} \quad & \boldsymbol{u}_i^T\boldsymbol{u}_i=1, \quad \boldsymbol{u}_i^T\boldsymbol{u}_j=0. \end{aligned} \tag{36}$$

Since each term in the sum of (36) is non-negative, we can consider each individual term

$$\begin{aligned} \underset{\boldsymbol{u}_i}{\text{minimize}} \quad & \|\boldsymbol{u}_i^T\boldsymbol{P}\|_2 \\ \text{subject to} \quad & \boldsymbol{u}_i^T\boldsymbol{u}_i=1, \end{aligned}$$

which is equivalent to

$$\begin{aligned} \underset{\boldsymbol{u}_i}{\text{minimize}} \quad & \|\boldsymbol{u}_i^T\boldsymbol{P}\|_2^2 \\ \text{subject to} \quad & \boldsymbol{u}_i^T\boldsymbol{u}_i=1. \end{aligned} \tag{37}$$

The constrained problem (37) can be solved by considering the Lagrange function,

$$\mathcal{L}(\boldsymbol{u}_i,\beta)=\|\boldsymbol{u}_i^T\boldsymbol{P}\|_2^2+\beta(1-\boldsymbol{u}_i^T\boldsymbol{u}_i). \tag{38}$$

Taking derivatives $\frac{\partial\mathcal{L}}{\partial\boldsymbol{u}_i}=0$ and $\frac{\partial\mathcal{L}}{\partial\beta}=0$ yield

$$\boldsymbol{P}\boldsymbol{P}^T\boldsymbol{u}_i=\beta\boldsymbol{u}_i, \quad \text{and} \quad \boldsymbol{u}_i^T\boldsymbol{u}_i=1.$$

Therefore, $\boldsymbol{u}_i$ is the eigenvector of $\boldsymbol{P}\boldsymbol{P}^T$, and β is the corresponding eigenvalue. Since the eigenvectors are orthonormal to each other, the solution automatically satisfies the orthogonality constraint that $\boldsymbol{u}_i^T\boldsymbol{u}_j=0$ if $i\neq j$. ■

C. Proof of Lemma 4

Proof: First, by plugging $\boldsymbol{q}=\boldsymbol{p}+\boldsymbol{\eta}$ into BMSE we get

$$\begin{aligned} \text{BMSE} &= \mathbb{E}_{\boldsymbol{p}}\left[\mathbb{E}_{\boldsymbol{q}|\boldsymbol{p}}\left[\left\|\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T(\boldsymbol{p}+\boldsymbol{\eta})-\boldsymbol{p}\right\|_2^2\middle|\boldsymbol{p}\right]\right] \\ &= \mathbb{E}_{\boldsymbol{p}}\left[\boldsymbol{p}^T\boldsymbol{U}\left(\boldsymbol{I}-\boldsymbol{\Lambda}\right)^2\boldsymbol{U}^T\boldsymbol{p}\right]+\sigma^2\text{Tr}\left(\boldsymbol{\Lambda}^2\right). \end{aligned}$$

Recall the fact that for any random variable $\boldsymbol{x}\sim\mathcal{N}(\boldsymbol{\mu}_x,\boldsymbol{\Sigma}_x)$ and any matrix $\boldsymbol{A}$, it holds that $\mathbb{E}\left[\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}\right]=\mathbb{E}[\boldsymbol{x}]^T\boldsymbol{A}\mathbb{E}[\boldsymbol{x}]+\text{Tr}\left(\boldsymbol{A}\boldsymbol{\Sigma}_x\right)$. Therefore, the above BMSE can be simplified as

$$\begin{aligned} &\boldsymbol{\mu}^T\boldsymbol{U}(\boldsymbol{I}-\boldsymbol{\Lambda})^2\boldsymbol{U}^T\boldsymbol{\mu}+\text{Tr}\left(\boldsymbol{U}(\boldsymbol{I}-\boldsymbol{\Lambda})^2\boldsymbol{U}^T\boldsymbol{\Sigma}\right)+\sigma^2\text{Tr}\left(\boldsymbol{\Lambda}^2\right) \\ &=\text{Tr}\left((\boldsymbol{I}-\boldsymbol{\Lambda})^2\boldsymbol{U}^T\boldsymbol{\mu}\boldsymbol{\mu}^T\boldsymbol{U}+(\boldsymbol{I}-\boldsymbol{\Lambda})^2\boldsymbol{U}^T\boldsymbol{\Sigma}\boldsymbol{U}\right)+\sigma^2\text{Tr}\left(\boldsymbol{\Lambda}^2\right) \\ &=\text{Tr}\left((\boldsymbol{I}-\boldsymbol{\Lambda})^2\boldsymbol{G}\right)+\sigma^2\text{Tr}(\boldsymbol{\Lambda}^2) \\ &=\sum_{i=1}^d\left[(1-\lambda_i)^2g_i+\sigma^2\lambda_i^2\right], \end{aligned} \tag{39}$$

where $\boldsymbol{G}\overset{\text{def}}{=}\boldsymbol{U}^T\boldsymbol{\mu}\boldsymbol{\mu}^T\boldsymbol{U}+\boldsymbol{U}^T\boldsymbol{\Sigma}\boldsymbol{U}$ and g_i is the ith diagonal entry in $\boldsymbol{G}$.

Setting $\partial\text{BMSE}/\partial\lambda_i=0$ yields

$$2(1-\lambda_i)g_i+2\sigma^2\lambda_i=0. \tag{40}$$

Therefore, the optimal λ_i is $g_i/(g_i+\sigma^2)$ and the optimal $\boldsymbol{\Lambda}$ is

$$\boldsymbol{\Lambda}=\text{diag}\left\{\frac{g_1}{g_1+\sigma^2},\cdots,\frac{g_d}{g_d+\sigma^2}\right\}, \tag{41}$$

which, by definition, is $\left(\text{diag}(\boldsymbol{G}+\sigma^2\boldsymbol{I})\right)^{-1}\text{diag}(\boldsymbol{G})$. ■

D. Proof of Lemma 5

Proof: First, we write $\boldsymbol{\Sigma}$ in (21) in the matrix form

$$\begin{aligned} \boldsymbol{\Sigma} &= \left(\boldsymbol{P}-\boldsymbol{\mu}\boldsymbol{1}^T\right)\boldsymbol{W}\left(\boldsymbol{P}-\boldsymbol{\mu}\boldsymbol{1}^T\right)^T \\ &= \boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T-\boldsymbol{\mu}\boldsymbol{1}^T\boldsymbol{W}\boldsymbol{P}^T-\boldsymbol{P}\boldsymbol{W}\boldsymbol{1}\boldsymbol{\mu}^T+\boldsymbol{\mu}\boldsymbol{1}^T\boldsymbol{W}\boldsymbol{1}\boldsymbol{\mu}^T. \end{aligned}$$

It is not difficult to see that $\boldsymbol{1}^T\boldsymbol{W}\boldsymbol{P}^T=\boldsymbol{\mu}^T, \boldsymbol{P}\boldsymbol{W}\boldsymbol{1}=\boldsymbol{\mu}$ and $\boldsymbol{1}^T\boldsymbol{W}\boldsymbol{1}=1$. Therefore,

$$\boldsymbol{\Sigma}= \boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T-\boldsymbol{\mu}\boldsymbol{\mu}^T-\boldsymbol{\mu}\boldsymbol{\mu}^T+\boldsymbol{\mu}\boldsymbol{\mu}^T= \boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T-\boldsymbol{\mu}\boldsymbol{\mu}^T,$$

which gives

$$\boldsymbol{\mu}\boldsymbol{\mu}^T + \boldsymbol{\Sigma} = \boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T. \tag{42}$$

Note that $\boldsymbol{G} = \boldsymbol{U}^T\boldsymbol{\mu}\boldsymbol{\mu}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\Sigma}\boldsymbol{U} = \boldsymbol{U}^T(\boldsymbol{\mu}\boldsymbol{\mu}^T + \boldsymbol{\Sigma})\boldsymbol{U}$. Substituting (42) into $\boldsymbol{G}$ and using equation (10), we have

$$\boldsymbol{G} = \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{W}\boldsymbol{P}^T\boldsymbol{U} = \boldsymbol{U}^T\boldsymbol{U}\boldsymbol{S}\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{S}.$$

Therefore, by Lemma 4,

$$\boldsymbol{\Lambda} = \left(\mathrm{diag}(\boldsymbol{S} + \sigma^2\boldsymbol{I})\right)^{-1}\mathrm{diag}(\boldsymbol{S}). \tag{43}$$

■

E. Proof of Lemma 6

By Lemma 5, it holds that

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{p}}\left[\mathbb{E}_{\boldsymbol{q}|\boldsymbol{p}}\left[\left\|\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{q} - \boldsymbol{p}\right\|_2^2 \middle| \boldsymbol{p}\right]\right]\\ &= \sum_{i=1}^{d}\left[(1-\lambda_i)^2 s_i + \sigma^2\lambda_i^2\right]\\ &= \sum_{i=1}^{d}\left[(s_i+\sigma^2)\left(\lambda_i - \frac{s_i}{s_i+\sigma^2}\right)^2 + \frac{s_i\sigma^2}{s_i+\sigma^2}\right].\end{aligned}$$

Therefore, the minimization of (26) becomes

$$\underset{\lambda_i}{\text{minimize}} \sum_{i=1}^{d}\left[(s_i+\sigma^2)\left(\lambda_i - \frac{s_i}{s_i+\sigma^2}\right)^2\right] + \gamma\|\boldsymbol{\Lambda}\mathbf{1}\|_\alpha, \tag{44}$$

where $\gamma\|\boldsymbol{\Lambda}\mathbf{1}\|_\alpha = \gamma\sum_{i=1}^{d}|\lambda_i|$ or $\gamma\sum_{i=1}^{d}\mathbb{1}(\lambda_i \neq 0)$ for $\alpha = 1$ or 0. We note that when $\alpha = 1$ or 0, (44) is the standard shrinkage problem [55], in which a closed form solution exists. The solutions are given by

$$\lambda_i = \max\left(\frac{s_i - \gamma/2}{s_i+\sigma^2}, 0\right), \qquad \text{for } \alpha = 1,$$

and

$$\lambda_i = \frac{s_i}{s_i+\sigma^2}\mathbb{1}\left(\frac{s_i^2}{s_i+\sigma^2} > \gamma\right), \qquad \text{for } \alpha = 0.$$

References

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

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

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

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "A nonlocal and shape-adaptive transform-domain collaborative filtering," in *Proc. Intl. Workshop on Local and Non-Local Approx. in Image Process. (LNLA'08)*, pp. 1–8, Aug. 2008.

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D image denoising with shape-adaptive principal component analysis," in *Signal Process. with Adaptive Sparse Structured Representations (SPARS'09)*, pp. 1–6, Apr. 2009.

[6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'09)*, pp. 2272–2279, Sep. 2009.

[7] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, pp. 1531–1549, Apr. 2010.

[8] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally centralized sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1620 – 1630, Apr. 2013.

[9] A. Rajwade, A. Rangarajan, and A. Banerjee, "Image denoising using the higher order singular value decomposition," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 35, no. 4, pp. 849 – 862, Apr. 2013.

[10] L. Shao, R. Yan, X. Li, and Y. Liu, "From heuristic optimization to dictionary learning: A review and comprehensive comparison of image denoising algorithms," *IEEE Trans. Cybernetics*, vol. 44, no. 7, pp. 1001–1013, Jul. 2014.

[11] M. Zontak and M. Irani, "Internal statistics of a single natural image," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'11)*, pp. 977–984, Jun. 2011.

[12] I. Mosseri, M. Zontak, and M. Irani, "Combining the power of internal and external denoising," in *Proc. Intl. Conf. Computational Photography (ICCP'13)*, pp. 1–9, Apr. 2013.

[13] H. C. Burger, C. J. Schuler, and S. Harmeling, "Learning how to combine internal and external denoising methods," *Pattern Recognition*, pp. 121–130, 2013.

[14] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. Intl. Conf. Computer Vision (ICCV'09)*, pp. 349–356, Sep. 2009.

[15] P. Chatterjee and P. Milanfar, "Is denoising dead?," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 895–911, Apr. 2010.

[16] A. Levin and B. Nadler, "Natural image denoising: Optimality and inherent bounds," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'11)*, pp. 2833–2840, Jun. 2011.

[17] R. Yan, L. Shao, S. D. Cvetkovic, and J. Klijn, "Improved nonlocal means based on pre-classification and invariant block matching," *Journal of Display Technology*, vol. 8, no. 4, pp. 212–218, Apr. 2012.

[18] Y. Lou, P. Favaro, S. Soatto, and A. Bertozzi, "Nonlocal similarity image filtering," in *Image Analysis and Processing*, pp. 62–71. Springer, 2009.

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

[20] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. IEEE Intl. Conf. Computer Vision (ICCV'11)*, pp. 479–486, Nov. 2011.

[21] S. H. Chan, T. Zickler, and Y. M. Lu, "Monte Carlo non-local means: Random sampling for large-scale image filtering," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3711–3725, Aug. 2014.

[22] A. Levin, B. Nadler, F. Durand, and W. T. Freeman, "Patch complexity, finite pixel correlations and optimal denoising," in *Proc. 12th Euro. Conf. Computer Vision (ECCV'12)*, vol. 7576, pp. 73–86. Oct. 2012.

[23] N. Joshi, W. Matusik, E. Adelson, and D. Kriegman, "Personal photo enhancement using example images," *ACM Trans. Graph*, vol. 29, no. 2, pp. 1–15, Apr. 2010.

[24] L. Sun and J. Hays, "Super-resolution from internet-scale scene matching," in *Proc. IEEE Intl. Conf. Computational Photography (ICCP'12)*, pp. 1–12, Apr. 2012.

[25] M. K. Johnson, K. Dale, S. Avidan, H. Pfister, W. T. Freeman, and W. Matusik, "CG2Real: Improving the realism of computer generated images using a large collection of photographs," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 9, pp. 1273–1285, Sep. 2011.

[26] M. Elad and D. Datsenko, "Example-based regularization deployed to super-resolution reconstruction of a single image," *The Computer Journal*, vol. 18, no. 2-3, pp. 103–121, Sep. 2007.

[27] I. Ram, M. Elad, and I. Cohen, "Image processing using smooth ordering of its patches," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2764–2774, Jul. 2013.

[28] L. Shao, H. Zhang, and G. de Haan, "An overview and performance evaluation of classification-based least squares trained filters," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1772–1782, Oct. 2008.

[29] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3D transform-domain collaborative filtering," in *Proc. 15th Euro. Signal Process. Conf.*, vol. 1, pp. 145–149, Sep. 2007.

[30] L. Zhang, S. Vaddadi, H. Jin, and S. Nayar, "Multiple view image denoising," in *Proc. IEEE Intl. Conf. Computer Vision and Pattern Recognition (CVPR'09)*, pp. 1542–1549, Jun. 2009.

[31] T. Buades, Y. Lou, J. Morel, and Z. Tang, "A note on multi-image denoising," in *Proc. IEEE Intl. Workshop on Local and Non-Local Approx. in Image Process. (LNLA'09)*, pp. 1–15, Aug. 2009.

[32] E. Luo, S. H. Chan, S. Pan, and T. Q. Nguyen, "Adaptive non-local means for multiview image denoising: Searching for the right patches via a statistical approach," in *Proc. IEEE Intl. Conf. Image Process. (ICIP'13)*, pp. 543–547, Sep. 2013.

[33] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: Design of dictionaries for sparse representation," *Proc. SPARS*, vol. 5, pp. 9–12, 2005.

[34] S. Roth and M.J. Black, "Fields of experts," *Intl. J. Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009.

[35] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2481–2499, May 2012.

[36] R. Yan, L. Shao, and Y. Liu, "Nonlocal hierarchical dictionary learning using wavelets for image denoising," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4689–4698, Dec. 2013.

[37] E. Luo, S. H. Chan, and T. Q. Nguyen, "Image denoising by targeted external databases," in *Proc. IEEE Intl. Conf. Acoustics, Speech and Signal Process. (ICASSP '14)*, pp. 2469–2473, May 2014.

[38] P. Milanfar, "A tour of modern image filtering," *IEEE Signal Process. Magazine*, vol. 30, pp. 106–128, Jan. 2013.

[39] P. Milanfar, "Symmetrizing smoothing filters," *SIAM J. Imaging Sci.*, vol. 6, no. 1, pp. 263–284, 2013.

[40] H. Talebi and P. Milanfar, "Global image denoising," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 755–768, Feb. 2014.

[41] S. Cotter, B. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2477–2488, Jul. 2005.

[42] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[43] S. Roth and M. Black, "Fields of experts: A framework for learning image priors," in *Proc. IEEE Computer Soc. Conf. Computer Vision Pattern Recognition, 2005*, vol. 2, pp. 860–867 vol. 2, Jun. 2005.

[44] D. Zoran and Y. Weiss, "Natural images, gaussian mixtures and dead leaves," *Advances in Neural Information Process. Systems (NIPS'12)*, vol. 25, pp. 1745–1753, 2012.

[45] S. M. Kay, *Fundamentals of statistical signal processing: Detection theory*, Prentice-hall, 1998.

[46] P. Chatterjee and P. Milanfar, "Patch-based near-optimal image denoising," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1635–1649, Apr. 2012.

[47] A. Buades, B. Coll, and J. M. Morel, "Non-local means denoising," [on line] http://www.ipol.im/pub/art/2011/bcm_nlm/, 2011.

[48] E. Luo, S. Pan, and T. Nguyen, "Generalized non-local means for iterative denoising," in *Proc. 20th Euro. Signal Process. Conf. (EUSIPCO'12)*, pp. 260–264, Aug. 2012.

[49] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 34, no. 11, pp. 2233–2246, Nov. 2012.

[50] C.E. Thomaz and G.A. Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," *Image and Vision Computing*, vol. 28, no. 6, pp. 902 – 913, 2010.

[51] M. Mahmoudi and G. Sapiro, "Fast image and video denoising via nonlocal means of similar neighborhoods," *IEEE Signal Process. Letters*, vol. 12, no. 12, pp. 839–842, Dec. 2005.

[52] R. Vignesh, B. T. Oh, and C.-C. J. Kuo, "Fast non-local means (NLM) computation with probabilistic early termination," *IEEE Signal Process. Letters*, vol. 17, no. 3, pp. 277–280, Mar. 2010.

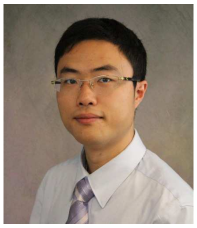
[53] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 36, no. 11, pp. 2227–2240, Nov. 2014.

[54] C. Boutsidis and M. Magdon-Ismail, "Faster SVD-truncated regularized least-squares," in *IEEE Intl. Symp. Information Theory (ISIT'14)*, pp. 1321–1325, Jun. 2014.

[55] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented Lagrangian method for total variation video restoration," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3097–3111, Nov. 2011.

Enming Luo (S'14) received the B.Eng. degree in Electrical Engineering from Jilin University, China, in 2007, and the M.Phil. degree in Electrical Engineering from Hong Kong University of Science and Technology in 2009. He is currently pursuing the Ph.D. degree in Electrical and Computer Engineering at the University of California, San Diego.

Mr. Luo was an engineer at ASTRI, Hong Kong, from 2009 to 2010, and was an intern at Cisco and InterDigital in 2011 and 2012, respectively. His research interests include image restoration (denoising, super-resolution and debluring), machine learning and computer vision.



Stanley H. Chan (S'06-M'12) received the B.Eng. degree in Electrical Engineering (with first class honor) from the University of Hong Kong in 2007, the M.A. degree in Mathematics and the Ph.D. degree in Electrical Engineering from the University of California at San Diego, La Jolla, CA, in 2009 and 2011, respectively.

Dr. Chan was a postdoctoral research fellow in the School of Engineering and Applied Sciences and the Department of Statistics at Harvard University, Cambridge, MA, from January 2012 to July 2014. He joined Purdue University, West Lafayette, IN, in August 2014, where he is currently an assistant professor of Electrical and Computer Engineering, and an assistant professor of Statistics. His research interests include statistical signal processing and graph theory, with applications to imaging and network analysis. He was a recipient of the Croucher Foundation Scholarship for Ph.D. Studies 2008-2010 and the Croucher Foundation Fellowship for Post-doctoral Research 2012-2013.

Truong Q. Nguyen (F'05) is currently a Professor at the ECE Dept., UCSD. His current research interests are 3D video processing and communications and their efficient implementation. He is the coauthor (with Prof. Gilbert Strang) of a popular textbook, Wavelets & Filter Banks, Wellesley-Cambridge Press, 1997, and the author of several matlab-based toolboxes on image compression, electrocardiogram compression and filter bank design.

Prof. Nguyen received the IEEE Transaction in Signal Processing Paper Award (Image and Multidimensional Processing area) for the paper he co-wrote with Prof. P. P. Vaidyanathan on linear-phase perfect-reconstruction filter banks (1992). He received the NSF Career Award in 1995 and is currently the Series Editor (Digital Signal Processing) for Academic Press. He served as Associate Editor for the IEEE Transaction on Signal Processing 1994-96, for the Signal Processing Letters 2001-2003, for the IEEE Transaction on Circuits & Systems from 1996-97, 2001-2004, and for the IEEE Transaction on Image Processing from 2004-2005.