2D Orthogonal Locality Preserving Projection for Image Denoising

Gitam Shikkenawis, *Student Member, IEEE*, and Suman K. Mitra, *Senior Member, IEEE*

Abstract—Sparse representations using transform-domain techniques are widely used for better interpretation of the raw data. Orthogonal locality preserving projection (OLPP) is a linear technique that tries to preserve local structure of data in the transform domain as well. Vectorized nature of OLPP requires high-dimensional data to be converted to vector format, hence may lose spatial neighborhood information of raw data. On the other hand, processing 2D data directly, not only preserves spatial information, but also improves the computational efficiency considerably. The 2D OLPP is expected to learn the transformation from 2D data itself. This paper derives mathematical foundation for 2D OLPP. The proposed technique is used for image denoising task. Recent state-of-the-art approaches for image denoising work on two major hypotheses, i.e., non-local self-similarity and sparse linear approximations of the data. Locality preserving nature of the proposed approach automatically takes care of self-similarity present in the image while inferring sparse basis. A global basis is adequate for the entire image. The proposed approach outperforms several state-of-the-art image denoising approaches for gray-scale, color, and texture images.

Index Terms—Sparse data representations, dictionary learning, image denoising, two dimensional orthogonal locality preserving projection.

I. Introduction

SPARSE representations using domain transformations have become immensely popular in last decade. Applications such as object recognition, data dimensionality reduction [5], [7], [16], [18], [20], [32], image restoration like denoising, deblurring, inpainting, compressive sensing etc. [9]–[11], [22], [23] use sparse representations of data. Some of the domain transformation techniques learn basis from given data itself. Principal Component Analysis (PCA) [20], Neighborhood Preserving Embedding (NPE) [15], Locality Preserving Projection (LPP) [16] are some of the widely used transformation domain techniques. LPP aims at preserving local structure of data in the transformed domain as well. As the basis learnt using LPP are not orthogonal, which may be desirable in many applications, Orthogonal Locality Preserving Projection (OLPP) having orthogonal basis vectors was proposed [5]. All the techniques discussed so far

The authors are with the Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar 382007, India (e-mail: 201221004@daiict.ac.in; suman_mitra@daiict.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

process only one dimensional data i.e. vectors. Two or more dimensional data such as images first have to be transformed into vector format. In order to process two dimensional data directly, Two Dimensional PCA (2D-PCA) [32], Two Dimensional LPP (2D-LPP) [7], Two Dimensional LPP based on Maximum Scatter Difference (2D-LPP/MSD) [29], Two Dimensional NPP (2D-NPP) [34] etc. have been proposed.

In this article, we formulate Two Dimensional Orthogonal Locality Preserving Projection (2D-OLPP). The approach directly processes images in two dimensional (2D) format i.e. matrix format, hence the overhead of transforming them in vectors gets reduced and the spatial neighborhood information remains intact. Due to 2D data processing, the basis matrix turns out to be much more compact than the one obtained by OLPP, reducing the time and space complexities of the algorithm considerably. Orthogonal nature makes the approach easier to be used in many applications that require orthogonal basis. So far, LPP and its variants have been mostly used for object recognition and dimensionality reduction related applications [5], [7], [16], [27]. In this article, we use the proposed 2D-OLPP approach for image denoising.

Noise in images gets introduced during acquisition process and hence denoising becomes an integral part of image generation process if clean images are desired. High correlation among the neighboring pixels of the image is one of the first properties that was used for image denoising tasks with spatial domain filters. These filtering techniques used pixel based local correlations. Concept of 'non-local self similarity' showed structural similarities between fixed sized patches from different spatial locations of the image [3]. To model the redundancies of the image, linear transformations such as Fourier Transform, Discrete Cosine Transform (DCT), Wavelet Transform, Block DCT [9] have been widely used.

Recent state of the art techniques for image denoising rely on two statistics of natural images: (1) There exists self-similarity between the patches from different locations of the same image [3], (2) Image patches can be sparsely represented by linear combinations of the basis vectors. Collection of these basis vectors of same or different linear transforms is known as 'dictionary'. It can comprise of the fixed universal basis stated above or can be adaptively learnt using patches from the image itself. Principal Component Analysis (PCA) [23] and Independent Component Analysis (ICA) [17] have been used to adaptively learn dictionaries from noisy input images. Local pixel grouping (LPG) based denoising techniques proposed

in [23] and [35] learn a set of orthonormal PCA basis for each patch of the image from a group of patches similar to the reference patch. Clustering based Sparse Representations (CSR) [10] learn a single dictionary per cluster after clustering all the overlapping patches of the image. Block Matching 3D (BM3D) [9] groups structurally similar patches to form a 3D stack of patches and projects them on a 3D dictionary i.e. 2D-DCT and 1D-Haar. Collaborative filtering is performed using hard thresholding on the transformed coefficients followed by inverse transform and aggregation of the patches. Global dictionary for sub-image or a large portion of the image is learnt using Orthogonal Locality Preserving Projection (OLPP) [28] from all the overlapping patches present in that portion and image denoising is performed in the OLPP domain.

Expected Patch Log Likelihood (EPLL) [36] is maximized to find a reconstructed image in which every patch is likely under the prior while keeping the image still close to the corrupted image. EPLL and method proposed in [33] use Gaussian Mixture Model (GMM) for learning the dictionary and noisy coefficients are denoised by structured sparse estimation. K-Singular Value Decomposition (K-SVD) ('K' denotes number of clusters) [13] and its variants learn an over-complete dictionary for entire noisy image and sparse linear combinations of the dictionary vectors are selected for each patch. Non-local sparse models are generated in [22] by grouping together similar patches in a matrix and using the concept of low rank matrix completion. The matrix norm is penalized subject to a hard constraint imposing coupled sparsity on the coefficients.

In most of the approaches discussed so far, a dictionary is obtained for each patch or a cluster of patches after grouping together structurally similar patches. Also, many of the non-local self similarity based approaches define a smaller search neighborhood for grouping similar patches or fix the maximum number of patches to be grouped. A global dictionary learning scheme (2D-OLPP) is proposed, which takes care of both, i.e. the basis are learned while taking care of the structural similarity between patches. The OLPP based image denoising approach also uses the principle of similarity preservation maintains structural similarity while learning the basis, but it requires the patches to be converted in vector format and works on a large window of the image at a time, instead of the whole image. In 2D-OLPP, patches from the entire input image are considered as it is (in matrix format) and weighed according to their structural similarity during the basis learning process. Hence, a *global dictionary* is learnt from the noisy input image, which remarkably reduces the computational complexity of the approach, and at the same time implements the underlying idea of non-local self similarity in dictionary learning process. A modified Wiener filter update rule to eliminate noise from two dimensional patches in the transform domain is also suggested.

Organization of the article is as follows: Similarity preserving projections and mathematical derivation of 2D-OLPP are discussed in Section 2. Suitability of 2D-OLPP for image denoising along with analysis of various parameters are explained in Section 3. Extensive experiments on some of the benchmark gray-scale and color image data-sets are reported in Section 4 followed by Conclusions in Section 5.

II. PROJECTIONS THAT PRESERVE SIMILARITY

In this section, we introduce dictionary learning scheme with the property of similarity preservation. Locality Preserving Projection (LPP) is a linear transformation approach that preserves the local structure (neighborhood information of the data) in the transformed domain as well [16]. Non-orthogonality of LPP makes it difficult to reconstruct the data. Also, having orthonormal basis is advantageous in many applications especially in image denoising as there is a need to revert back from the transform domain and project the data in the original space (*i.e.* spatial domain). In case of orthogonal basis $\mathbf{V}$, $\mathbf{V}^{-1} = \mathbf{V}^T$, hence computation of the inverse of the basis matrix is no longer required. Orthogonal Locality Preserving Projection (OLPP) [5] was introduced to orthogonalize the basis of LPP.

Let us consider we have N data points $\mathbf{X_1}, \mathbf{X_2}, \cdots, \mathbf{X_N}$ of size $m \times n$. In case of OLPP, higher dimensional data points are first converted to vector format $\mathbf{X_i^v}$s of size $mn \times 1$. Data matrix $\mathbf{X^v}$ is formed by arranging these data points in columns i.e. $\mathbf{X^v} = [\mathbf{X_1^v}, \mathbf{X_2^v}, \cdots, \mathbf{X_N^v}]$ where $\mathbf{X_i^v}$s represent vector form of the corresponding 2-dimensional data point $\mathbf{X_i}$. The data matrix thus turns out to be of $mn \times N$ dimensions. Objective function of OLPP is as follows:

$$min \sum_{ij} (\mathbf{Y_i^v} - \mathbf{Y_j^v})^2 W_{ij} \qquad (1)$$

here, $\mathbf{Y_i^v}$ is the projection of $\mathbf{X_i^v}$ in the transformed domain using the basis vectors $\mathbf{V}$ i.e. $\mathbf{Y_i^v} = \mathbf{V}^T\mathbf{X_i^v}$. After simplifying the objective function and adding normalization and orthogonalization constraints, the optimization problem turns out to be:

$$min \ \mathbf{V}^T\mathbf{X}^v\mathbf{L}\mathbf{X}^{v^T}\mathbf{V}$$

subject to the constraints

$$\mathbf{V}^T\mathbf{X}^v\mathbf{S}\mathbf{X}^{v^T}\mathbf{V} = \mathbf{1}; \mathbf{v_m^T}\mathbf{v_k} = 0, \quad \forall m = \{1, \cdots, k-1\} \quad (2)$$

here, $\mathbf{L} = \mathbf{S} - \mathbf{W}$ is the Laplacian matrix, $W_{ij} = e^{\frac{-\left\|\mathbf{X_i^v} - \mathbf{X_j^v}\right\|^2}{t}}$ is the weight matrix and $S_{ii} = \sum_j W_{ij}$ [16]. Basis vectors $\mathbf{v_k}$ are found in an iterative manner, assuring that the new vector learnt is orthonormal to all the previously found basis vectors. $\mathbf{v_1}$ is the eigenvector associated with the smallest non-zero eigenvalue of $\mathbf{Z^{(1)}} = \left(\mathbf{X}^v\mathbf{S}\mathbf{X}^{v^T}\right)^{-1}\mathbf{X}^v\mathbf{L}\mathbf{X}^{v^T}$. Similarly, $\mathbf{v_k}, \forall k$ is the eigenvector associated with the smallest non-zero eigenvalue of $\mathbf{Z^{(k)}}$.

$$\mathbf{Z}^{(k)} = \left\{\mathbf{I} - \left(\mathbf{X}^v\mathbf{S}\mathbf{X}^{v^T}\right)^{-1}\mathbf{V}^{(k-1)}\left[\mathbf{U}^{(k-1)}\right]^{-1}\left[\mathbf{V}^{(k-1)}\right]^T\right\}\mathbf{Z}^{(1)}$$

$$\mathbf{V}^{(k-1)} = \left[\mathbf{v_1}, \ldots, \mathbf{v_{k-1}}\right]$$

$$\mathbf{U}^{(k-1)} = \left[\mathbf{V}^{(k-1)}\right]^T\left(\mathbf{X}^v\mathbf{S}\mathbf{X}^{v^T}\right)^{-1}\mathbf{V}^{(k-1)} \qquad (3)$$

The basis matrix $\mathbf{V}$ thus found is orthogonal. These domain transformation techniques (PCA, LPP and OLPP) deal with one dimensional data, hence each higher dimensional data

point needs to be vectorized. Also, spatial neighborhood information is lost due to the vectorization.

A. Two Dimensional Orthogonal Locality Preserving Projection (2D-OLPP): The Proposal

As discussed in Section I, it is advantageous to process two dimensional data directly. Two dimensional extensions of PCA, LPP and NPP proposed in [7], [32], and [34] respectively, directly process the data matrix and have shown improvement in the performance over their one dimensional (1D) counterparts. 2D-LPP/MSD [29], an extension of LPP motivated by the idea of maximum scatter difference, maximizes the difference of between-class scatter and within-class scatter. The approach is supervised in nature and the basis vectors obtained are also non-orthogonal, hence it is not suitable for image denoising. In case of 2D-LPP and 2D-NPP also, the basis found are not orthogonal. Orthogonal basis in both the cases can be directly obtained by ignoring the normalization constraint of the data [27], [34]. However, the results show drastic decrease in the performance of these algorithms, which demonstrates a strong need for the normalization constraint ($\mathbf{V^T X^v S X^{v^T} V = 1}$), especially in case of two dimensional data processing [34]. Hence, it is important to retain this constraint while obtaining the orthogonal basis. With this aim in mind, two dimensional orthogonal locality preserving projection (2D-OLPP) is formulated and derived mathematically in this section.

In the proposed formulation, the data matrix X of dimensions $(mN \times n)$ is formed by arranging all the data matrices in row format $X = [X_1; X_2; \cdots; X_N]$, thus keeping the data points intact. In this article, we are processing images, hence referring data points and images interchangeably. As per the establishment of LPP, the aim is to preserve local structure of the data. Hence, the objective function remains same as that of LPP and OLPP:

$$min \sum_{ij} (\mathbf{Y_i} - \mathbf{Y_j})^2 W_{ij} \qquad (4)$$

here, $\mathbf{Y_i} = \mathbf{X_i V}$ is the projection of $\mathbf{X_i}$ using the transformation matrix $\mathbf{V} = [\mathbf{v_1}, \mathbf{v_2}, \cdots, \mathbf{v_k}]$ where $\mathbf{v_i}$s are orthogonal basis vectors. Here, weight matrix $\mathbf{W}$, Laplacian matrix $\mathbf{L}$ and $\mathbf{S}$ are found in the same manner as suggested in Extended LPP [26]. Weights in the matrix $\mathbf{W}$ are assigned in a monotonically decreasing z-shaped fashion. Based on the range of values given as input, weights are assigned over the complete scale as per Equation 5. As the distance between data points increases, weight at that point decreases.

$$W_{ij} = \begin{cases} 1; & \text{if } x \le a \\ 1 - 2\left(\dfrac{x-a}{b-a}\right)^2; & \text{if } a \le x \le \dfrac{a+b}{2} \\ 2\left(b - \dfrac{x}{b-a}\right)^2; & \text{if } \dfrac{a+b}{2} \le x \le b \\ 0; & \text{otherwise} \end{cases} \qquad (5)$$

a and b specify the range of values along which the function changes its value and can be controlled. Upto a, the value of function is kept constant 1 while after b, the value is

set to 0. Between a and b two functions are used, making the final output function a z-shaped one. Slope of the function is dependent on parameters a and b. Here, x represents Euclidean distance between two data points i.e. $x = \|\mathbf{X_i} - \mathbf{X_j}\|$. As opposed to conventional LPP where only few neighbors are considered, here data points that are at a moderate distance from the point of interest are also taken into consideration and weighed accordingly. After simplifying, the objective function turns out to be:

$$arg\,min \sum_{ij} (\mathbf{Y_i} - \mathbf{Y_j})^2 W_{ij} = arg\,min\ \mathbf{V^T X^T}\left(L \otimes I_m\right)\mathbf{XV}$$

$$(6)$$

subject to the constraint,

$$\mathbf{V^T X^T}\left(\mathbf{S} \otimes \mathbf{I_m}\right)\mathbf{XV} = 1 \qquad (7)$$

where, $\mathbf{X}$ is the data matrix (having data points in 2D format), $\mathbf{L} = \mathbf{S} - \mathbf{W}$ is the Laplacian matrix and $S_{ii} = \sum_j W_{ij}$. I_m is the identity matrix of size $m \times m$ and $\otimes$ is the Kronecker product operator. To obtain orthogonal basis, one more constraint is added.

$$\mathbf{v_i^T v_k} = 0, \quad \forall i = \{1, 2, \cdots, k-1\} \qquad (8)$$

For the first eigenvector $\mathbf{v_1}$, the problem reduces to minimizing Equation 6 subject to Equation 7. Hence $\mathbf{v_1}$ is the eigenvector corresponding to the smallest non-zero eigenvalue of the following generalized eigenvalue solution:

$$\mathbf{X^T}\left(\mathbf{L} \otimes \mathbf{I_m}\right)\mathbf{XV} = \lambda \mathbf{X^T}\left(\mathbf{S} \otimes \mathbf{I_m}\right)\mathbf{XV} \qquad (9)$$

For rest of the basis vectors $\mathbf{v_i}$, the problem now formulates to minimizing Equation 6 subject to constraints in Equations 7 and 8. Using Lagrange's multiplier approach to solve the minimization problem,

$$\begin{aligned} C^{(K)} &= \mathbf{v_k^T X^T}\left(\mathbf{L} \otimes \mathbf{I_m}\right)\mathbf{X v_k} \\ &\quad - \lambda \left[\mathbf{v_k^T X^T}\left(\mathbf{S} \otimes \mathbf{I_m}\right)\mathbf{X v_k} - 1\right] \\ &\quad - \mu_1 \mathbf{v_k^T v_1} - \mu_2 \mathbf{v_k^T v_2} - \cdots - \mu_{k-1}\mathbf{v_k^T v_{k-1}} \end{aligned}$$

For optimizing the minimization problem, partial derivatives of $C^{(K)}$ with respect to λ, μ_i and $\mathbf{v_k}$ are equated to zero as per the Lagrange's multiplier optimization technique.

$$\frac{\partial C^{(K)}}{\partial \lambda} = 0 \Rightarrow \mathbf{V^T X^T}\left(\mathbf{S} \otimes \mathbf{I_m}\right)\mathbf{XV} = 1$$

$$\frac{\partial C^{(K)}}{\partial \mu_i} = 0 \Rightarrow \mathbf{v_k^T v_i} = 0$$

$$\frac{\partial C^{(K)}}{\partial \mathbf{v_k}} = 0 \Rightarrow 2\mathbf{X^T}\left(\mathbf{L} \otimes \mathbf{I_m}\right)\mathbf{X v_k}$$

$$- 2\lambda \mathbf{X^T}\left(\mathbf{S} \otimes \mathbf{I_m}\right)\mathbf{X v_k}$$

$$- \mu_1 \mathbf{v_1} - \mu_2 \mathbf{v_2} - \cdots - \mu_{k-1}\mathbf{v_{k-1}} = 0 \qquad (10)$$

Multiplying Equation 10 by $\mathbf{v_k^T}$, we can obtain λ:

$$\lambda = \frac{\mathbf{v_k^T X^T}\left(\mathbf{L} \otimes \mathbf{I_m}\right)\mathbf{X v_k}}{\mathbf{v_k^T X^T}\left(\mathbf{S} \otimes \mathbf{I_m}\right)\mathbf{X v_k}} \qquad (11)$$

From the above value of λ and Equation 9, it can be observed that λ is the equation to be minimized.

Patch #	Reference Patch (150)	1780	3191	4059	1021	692	8291
Patch Image							
Euclidean Distance	0	109.749	128.768	139.309	188.155	164.213	173.465
Weight	0	0.931	0.905	0.889	0.798	0.846	0.828

Since, $\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]$ is positive definite and non-singular [16], its inverse exists. Now, multiplying Equation 10 successively by $\mathbf{v_i^T}\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}$; $\forall i = 1, 2, \cdots, k-1$, we get a set of $(k-1)$ equations:

$$\mu_1\mathbf{v_i^T}\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\mathbf{v_1}$$
$$+\cdots+\mu_{k-1}\mathbf{v_i^T}\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\mathbf{v_{k-1}}$$
$$=2\mathbf{v_i^T}\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\left[\mathbf{X^T}\left(\mathbf{L}\otimes\mathbf{I_m}\right)\mathbf{X}\right]\mathbf{v_k} \quad (12)$$

Converting this set of (k-1) equations stated in Equation 12 to matrix format,

$$\mathbf{U^{(k-1)}}\mu^{(k-1)} = 2\left[\mathbf{V^{(k-1)}}\right]^T\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}$$
$$\times\left[\mathbf{X^T}\left(\mathbf{L}\otimes\mathbf{I_m}\right)\mathbf{X}\right]\mathbf{v_k} \quad (13)$$

where,

$$\mathbf{V^{(k-1)}} = \left[\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_{(k-1)}}\right]$$
$$\mu^{(k-1)} = \left[\mu_1, \mu_2, \ldots, \mu_{(k-1)}\right]$$
$$\mathbf{U^{(k-1)}} = \left[\mathbf{V^{(k-1)}}\right]^T\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\mathbf{V^{(k-1)}}$$

$\mu^{(k-1)}$ can be obtained by multiplying Equation 13 with $\left[\mathbf{U^{(k-1)}}\right]^{-1}$:

$$\mu^{(k-1)} = 2\left[\mathbf{U^{(k-1)}}\right]^{-1}\left[\mathbf{V^{(k-1)}}\right]^T$$
$$\times\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\left[\mathbf{X^T}\left(\mathbf{L}\otimes\mathbf{I_m}\right)\mathbf{X}\right]\mathbf{v_k} \quad (14)$$

Multiplying Equation 10 by $\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}$ and replacing $\mu^{(k-1)}$ with the value obtained in Equation 14,

$$\{\mathbf{I}-\mathbf{A}\}\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}$$
$$\times\left[\mathbf{X^T}\left(\mathbf{L}\otimes\mathbf{I_m}\right)\mathbf{X}\right]\mathbf{v_k} = \lambda\mathbf{v_k}$$
$$\mathbf{A} = \left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\mathbf{V^{(k-1)}}\left[\mathbf{U^{(k-1)}}\right]^{-1}\left[\mathbf{V^{(k-1)}}\right]^T$$

As stated before, λ is the criterion to be minimized and hence, $\mathbf{v_k}$ turn out to be eigenvectors of the following:

$$\mathbf{Z^{(k)}} = \{\mathbf{I}-A\}\left[\mathbf{X^T}\left(\mathbf{S}\otimes\mathbf{I_m}\right)\mathbf{X}\right]^{-1}\left[\mathbf{X^T}\left(\mathbf{L}\otimes\mathbf{I_m}\right)\mathbf{X}\right] \quad (15)$$

This is an iterative procedure, each iteration of which computes $\mathbf{v_k}$, the eigenvector associated with the smallest non-zero eigenvalue of $Z^{(k)}$.
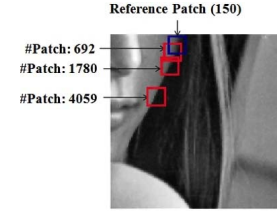

Fig. 1. Reference patch from 'Lena' image (blue boundary) and some of its similar patches (red boundary) from the whole image.

The main advantage of performing 2D-OLPP instead of conventional 1D-OLPP is computational efficiency. In case of OLPP (1D), the basis matrix turns out to be of $mn \times mn$ dimensions i.e. it takes mn iterations to build the complete basis matrix V, whereas for 2D-OLPP, dimensions of $\mathbf{V}$ are $n \times n$. Hence, it takes only n iterations to compute $\mathbf{V}$ which makes 2D-OLPP computationally more efficient. Due to compact data arrangement, space complexity also reduces. The time complexity analysis is discussed in detail in Section III-C.

III. IMAGE DENOISING USING 2D-OLPP

In this article, the idea of non-local transformed domain image denoising is explored with use of the proposed two-dimensional orthogonal locality preserving projection (2D-OLPP). Suitability of 2D-OLPP for image denoising, parameter selection and noise removal process in the transformed domain are discussed in this section.

A. Incorporating Non-Local Self Similarity

The basis learning procedure of 2D-OLPP automatically takes care of non-local self similarity i.e. the patches that share more structural similarity are assigned higher weights using the z-shaped weighing function while constructing weight matrix $\mathbf{W}$. Similarity between patches is measured by Euclidean distance. The reference patch, some of its structurally similar patches from the entire image, their euclidean distances from the reference patch and respective weights are shown in Table I. In Figure 1, the reference patch is shown using blue boundary in the original image. Three of its similar patches i.e. patch #1780, patch #4059 and patch #692 are shown with red boundary. Though patch #692 is local neighbor of the reference patch, patches #1780 and #4059 are structurally more similar to it and hence are assigned higher weights.
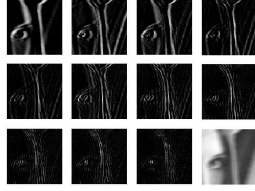
Fig. 2. Coefficient-wise plot for all the 12×12 overlapping patches of a 128×128 portion of 'Lena' image using 2D-OLPP.
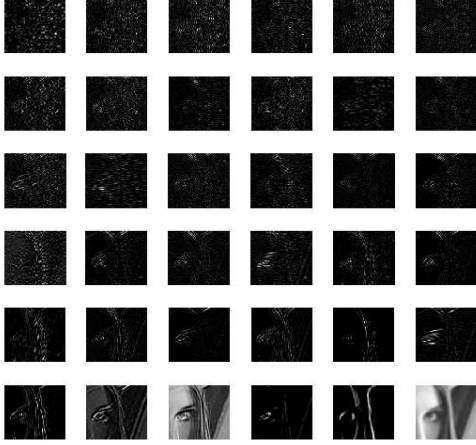

Fig. 3. Coefficient-wise plot for all the 6×6 overlapping patches of a 128×128 portion of 'Lena' image using OLPP.

B. Sparsity

Although, the number of basis vectors in case of 2D-OLPP for an $m \times n$ image reduces from mn to n, the coefficients in the 2D-OLPP domain are still sparse. To demonstrate the sparseness, coefficient plots using each of the n basis vectors are shown in Figure 2 for a 120×120 portion of 'Lena' image. All 12×12 overlapping patches are used for learning the basis. Sparsity in the coefficients can still be observed in case of 2D-OLPP with only 12 basis vectors as compared to 144 for OLPP or PCA (all the approaches that work in vector format). As it is not possible to show all 144 OLPP basis vectors for 12×12 patches, coefficient plots considering all 6×6 patches from the same image are presented in Figure 3.

C. Time Complexity Analysis

The cost, as far as time complexity is concerned, appears to be much less in case of 2D-OLPP as compared to state of the art methods. Assuming total number of overlapping patches in the image is N, average number of patches similar to the reference patch is by P, average time required to search similar patches for each reference patch is T_s, time taken to generate the Laplacian matrix L for N patches is NT_L and size of each patch is $l \times l$. In approaches that require the patch (in matrix format) to be transformed into vector format, size of each representative vector turns out to be $l^2 \times 1$.

In PCA based approaches [23], [35], for each reference patch, similar patches are grouped together and covariance matrix of dimensions $l^2 \times l^2$ is computed, eigenvectors of which form the transformation matrix. Thus, the time

TABLE II
SUMMARY OF TIME COMPLEXITIES OF DENOISING APPROACHES. N: NUMBER OF PATCHES, $l \times l$: SIZE OF A PATCH, T_s: AVERAGE TIME REQUIRED TO SEARCH SIMILAR PATCHES OF THE REFERENCE PATCH, P: AVERAGE NUMBER OF SIMILAR PATCHES, NT_L: TIME TAKEN TO GENERATE THE LAPLACIAN MATRIX L FOR N PATCHES

Technique	Time Complexity
LPG-PCA	$\mathcal{O}(N[T_s + Pl^4 + l^6])$
BM3D	$\mathcal{O}(N[T_s + Pl^3 + P^2l^2])$
OLPP	$\mathcal{O}(NT_L + l^6)$
2D-OLPP	$\mathcal{O}(NT_L + l^3)$

complexity becomes $\mathcal{O}(N[T_s + Pl^4 + l^6])$. Note that computation of eigenvectors of an $l \times l$ matrix takes $\mathcal{O}(l^3)$ time. BM3D [9] stacks similar patches of the reference patch and performs 2D and 1D transforms on patches of that stack. The 2D transforms lead to $\mathcal{O}(Pl^3)$ complexity whereas for 1D it is $\mathcal{O}(P^2l^2)$. As the procedure is repeated for all overlapping patches, overall time complexity turns out to be $\mathcal{O}(N[T_s + Pl^3 + P^2l^2])$.

In case of OLPP [28] where patches are considered in vector format, $\mathcal{O}(NT_L)$ time is required to compute the Laplacian matrix and $\mathcal{O}(l^6)$ for the transformation matrix i.e. basis matrix, as the global basis is computed only once for the whole image. Thus, overall time complexity is $\mathcal{O}(NT_L + l^6)$. Because of 2D patch processing in the proposed approach, the eigen decomposition problem reduces to $l \times l$ matrix instead of $l^2 \times l^2$ and hence requires $\mathcal{O}(l^3)$ time which leads to $\mathcal{O}(NT_L + l^3)$ complexity. Thus, the proposed approach is faster than other state of the art image denoising approaches. Time complexities of the approaches discussed above are summarized in Table II.

D. Selection of σ and Patch Size

Two important parameters for transformed domain image denoising procedure are standard deviation of noise (σ) and patch size. In this article, we are denoising images corrupted with i.i.d. Gaussian noise with zero mean and σ standard deviation i.e. $\mathcal{N}(0, \sigma)$. All the methods compared in this article assume that standard deviation of noise (σ) i.e. the noise level, is known. In practical scenarios, value of σ might not be available, hence methods for estimating σ can be used prior to denoising. Donoho and Johnstone [12] derived value of σ from the finest scale empirical wavelet coefficient of the given input data. Highest sub-band of a Daubechies 2 wavelet transform was used in [19] to estimate noise variance. This approach is also used in [6] and [23] to estimate the noise level. Another set of approaches use homogeneous regions of image to estimate noise variance/standard deviation [1]. Methods proposed in [14] used local statistics of image to estimate noise level. Noise level estimation based on principal component analysis has been recently proposed in [24]. Several other approaches have also been proposed in literature to estimate noise level from given noisy image.

Another important parameter in the denoising procedure is patch size. All the experiments reported in this article use fixed patch size i.e. 12×12. Denoising experiments with different patch sizes ranging from 6×6 to 22×22 were carried out on the Lansel database of natural images.

TABLE III

AVERAGE DENOISING RESULTS FOR ALL THE IMAGES FROM
LANSEL DATABASE ($\sigma = 20$) USING THE PROPOSED
APPROACH WITH VARIOUS PATCH SIZES

Patch size	6	8	10	12	15	20	22
PSNR	31.19	31.68	31.80	**31.93**	31.77	31.68	31.58
SSIM	0.822	0.859	0.867	**0.873**	0.872	0.871	0.868

Average denoising results in terms of PSNR and SSIM values for various patch sizes are reported in Table III. Though the performances of some of the patch sizes are not significantly different, patches with 12×12 size produce best PSNR and SSIM results. This performance has remained consistent across other databases, hence we have used patch-size 12×12 for the experiments reported in Section IV.

E. Noise Removal

Experiments carried out in the previous subsections suggest that the proposed 2D-OLPP can be effectively applied for image denoising tasks. Noisy image formation model in spatial domain is represented as $\mathbf{A}_\eta = \mathbf{A} + \eta$ and in transformed domain, using basis matrix $\mathbf{V}$, the model turns out to be $\mathbf{A}_\eta \mathbf{V} = \mathbf{A}\mathbf{V} + \eta_V$. Here, $\eta \in \mathcal{N}(0, \sigma)$ i.e. i.i.d. Gaussian noise with zero mean and σ standard deviation. A data matrix $\mathbf{X}$ consisting of all the patches $\mathbf{X_i}$ from the noisy input image $\mathbf{A}_\eta$, is constructed as discussed in Section II-A. The basis matrix $\mathbf{V}$ is learnt following the iterative procedure and patch $\mathbf{X_i}$ in spatial domain is transformed to $\mathbf{Y_i}$ in the 2D-OLPP domain. Coefficients of patches in transformed domain are altered to eliminate the noise.

Given that $\mathbf{V}$ is an orthonormal basis, some of the standard forms of eliminating noise generally known as shrinkage rules can be directly used. Some of the most popular shrinkage rules are hard thresholding, soft thresholding [9], [11] and Wiener filter update rule (1D) [9], [23], [35]. The rule to update the coefficients in the transformed domain using Wiener filter is given as $\frac{\sigma_{A_V}^2}{\sigma_{A_V}^2 + \sigma^2} \mathbf{A}_\eta \mathbf{V}$. Here, σ is the standard deviation of noise. For a particular coefficient position i, σ_{i, A_V} is computed by considering the ith coefficients from all the similar patches of the reference patch.

In this article, the Wiener filter update rule is modified for two dimensional (2D) patches. Instead of using only one coefficient across patches for estimating variance at a particular coefficient position, coefficients falling in 3×3 window around the coefficient of interest are used. Median of the variance of the particular window is used to compute the update rule *i.e.* $\sigma_{A_{Vnew}}^2 = median\left[\sigma_{i, A_V}^2 \in N_i\right]$ where N_i defines the neighboring window of the ith coefficient position. Note that, similar concept of median is the simplest form of filter for data denoising. Experiments have been performed to compare the proposed modified Wiener filter (2D-Wie) update rule with the other shrinkage rules such as hard thresholding (HardT) and conventional Wiener filter update (1D-Wie) in the 2D-OLPP denoising framework. Average PSNR and SSIM results for Lansel database reported in Table IV show improvement over the other shrinkage rules.

TABLE IV

AVERAGE DENOISING RESULTS FOR LANSEL DATABASE FOR
DIFFERENT NOISE REMOVAL (FILTERING) TECHNIQUES

| Filter	$\sigma = 20$		$\sigma = 30$		$\sigma = 40$	
Type	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
HardT	29.47	0.801	28.02	0.737	26.66	0.673
1D-Wie	31.67	0.855	29.89	0.795	28.63	0.732
2D-Wie	**31.69**	**0.872**	**30.65**	**0.821**	**29.43**	**0.762**

Denoising experiments on a large set of natural and texture databases on both gray-scale and color images are included in the next section.

IV. EXPERIMENTS

Algorithmic comparison of the proposed 2D-OLPP denoising with general framework of non-local self similarity based transformed domain image denoising approaches is shown in Figure 4. Basis vectors obtained using PCA are shown in the general framework in Figure 4. As it is 1D processing, patches are converted to vector format because of which size of the basis turns out to be $l^2 \times l^2$ where $l \times l$ is the patch size. First step involves grouping together similar patches and learning the local dictionary. Generally grouping is carried out based on the Euclidean distance between patches and defining a threshold either based on distance or number of patches. For LPG-PCA, this process is repeated for all the overlapping patches whereas for CSR, new set of PCA basis are generated for each cluster of patches. BM3D uses fixed 2D-DCT and 1D-Haar basis on the grouped patches. As it uses 2D transform, patch to vector conversion is not required. Coefficients of the patches in the respective transform domain are modified by applying one of the shrinkage rules discussed in III-E to eliminate noise. Denoised patch is transformed back to the spatial domain and is placed back at its original position after aggregation.

OLPP based denoising approach [28], being a global one, learns the basis only once for a large portion of image. Due to its one dimensional processing, the patches are required to be converted into vector format making the size of the dictionary $l^2 \times l^2$. The proposed approach directly uses image patches in matrix format for dictionary learning. For patches of size $l \times l$, basis i.e. 2D-OLPP dictionary turns out to be of size $l \times l$ only as shown in Figure 4. Also, the basis learning process itself takes care of similarity between patches as per the objective of 2D-OLPP, *i.e.* patches that are similar in spatial domain should remain close and have similar coefficients in the transformed domain as well. Hence, there is no need to explicitly group similar patches and a global dictionary is sufficient. Coefficients of patches in the 2D-OLPP domain are altered using modified Wiener filter update rule for 2D patch processing and inverse transform is performed to map denoised patches back to the spatial domain. Final restored image is obtained by averaging multiple candidate values for a particular pixel position.

A global 2D-OLPP basis for the entire image is learned using all the overlapping patches of size 12×12. For a given reference patch $\mathbf{Y}_r$, any patch $\mathbf{Y}_i$ for which $\|\mathbf{Y}_r - \mathbf{Y}_i\| \leq 1.7\sigma l$ is considered structurally similar and hence neighbor [25],
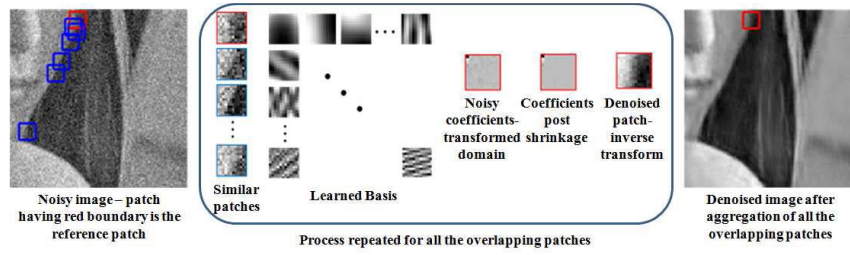
(a) Framework for non-local self similarity based transformed domain image denoising.
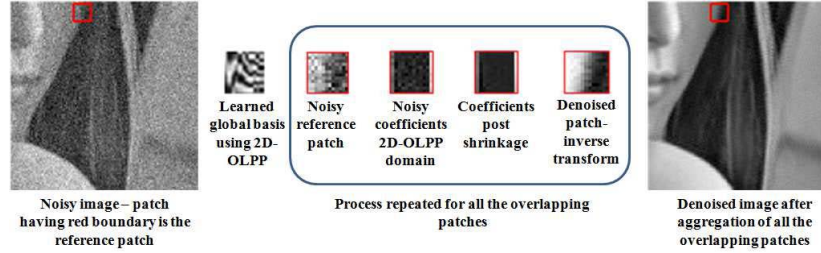
(b) Framework for the proposed 2D-OLPP based image denoising.

Fig. 4.   Algorithmic comparison of the denoising approaches. All the patches are assumed to be of fixed size $l \times l$. (a) First step is to group together similar patches for a given reference patch. In case of 1D processing, the patches are converted to the vector format before learning the dictionary which also increases the size of dictionary i.e. number of basis vectors to be learned. In case of 1D data processing, $l \times l$ 2D patch is required to be converted to $l^2 \times 1$ sized vector, thus l^2 basis vectors are to be learned. The noisy reference patch is then projected on these basis vectors, coefficients in the transform are filtered to eliminate noise and inverse transformation is performed to get back to the spatial domain. This process is repeated for all overlapping patches as separate dictionary for all/cluster of patches is learned. (b) In case of the proposed denoising approach, only one global dictionary is sufficient for the entire image. Due to 2D data processing, the basis learnt are much more compact i.e. l as explained in Section II-A. All the noisy patches are then projected on the same global basis, coefficients are modified using the modified Wiener filter update rule (Section III-E), patches are transformed back to the spatial domain and aggregated.

where l is the patch size. Weight, according to Equation 5, is assigned to patches that are similar to the reference patch while constructing the 2D-OLPP basis. Parameter 'b' of Equation 5 is assigned value $1.7\sigma l$ for constructing weight matrix $\mathbf{W}$. This automatically takes care of similar patches, hence explicit search for similar patches of each patch is not required.

The proposed approach is compared with some of the state of the art approaches. Publicly available codes with the author specified parameter values are used for all the competing approaches to ensure unbiased comparison. All the experiments are performed in MATLAB R2012b environment on Lenovo Z510 with Intel core i7 windows 8 laptop. Restored image quality evaluation measures Peak Signal to Noise Ratio (PSNR) in terms of decibel (db) scale and Mean Structural Similarity Index Measure (MSSIM) [30] are used.

Apart from the existing approaches, to have fair comparison between the dictionary learning schemes and to establish the claim that locality preserving property plays an important role for the global basis to be suffice for the entire image, experiments using PCA in the global dictionary learning framework have also been performed. Instead of grouping together similar patches (as in some of the PCA based state of the art approaches), only one global PCA basis is learnt from all possible overlapping patches of the image to be denoised. Denoising procedure as explained for the proposed approach is followed.

Average denoising results for various databases using the Global PCA based denoising approach along with the other

state of the art approaches are reported. Also, to show quantitative range of the results using 2D-OLPP based denoising approach, best and worst case performances in terms of PSNR and SSIM values have been reported.

Image denoising experiments on some of the benchmark gray-scale and color image databases are reported in Sections IV-A and IV-B respectively. Section IV-C shows results of noise to noise test and method noise comparison for various approaches.

A. Gray-Scale Image Denoising

Experiments on both natural and textured gray-scale images are performed. The proposed approach is compared with some of the state of the art denoising approaches i.e. principal component analysis with local pixel grouping (LPG-PCA) [35], block matching 3D (BM3D) [9], expected patch log-likelihood (EPLL) [36], clustering based sparse representation (CSR) [10] and OLPP based denoising [28].

For natural images, the Lansel database containing 13 widely used images for denoising (airplane, barbara, boats, couple, elaine, fingerprint, goldhill, house, man, mandrill, peppers, stream, zelda) is used. Noisy images with additive white Gaussian noise i.e. $\mathcal{N}(0, \sigma)$ having three noise levels $\sigma \in \{20, 30, 40\}$ are considered for experimentation. A portion from 'Barabara' image containing face part of the image, noisy image with $\sigma = 30$ and denoising results using various approaches are shown in Figure 5. It can be observed that features of the face are nicely preserved using

Fig. 5. Denoising experiment on a part of natural image ('Barbara'). Left to right, top to bottom: Clean Image, Noisy image ($\sigma = 30$), LPG-PCA(28.39,0.843), BM3D(28.93,0.857), EPLL(27.44,0.816), CSR(28.78,0.850), OLPP(27.45,0.821), 2D-OLPP(30.32,0.875). (*Zoom in pdf for better view*).

Fig. 6. Denoising experiment on portion of the 'Lena' image. Left to right, top to bottom: Clean Image, Noisy image ($\sigma = 30$), LPG-PCA(31.73,0.847), BM3D(32.38,0.851), EPLL(31.78,0.831), CSR(32.36,0.850), OLPP(31.42,0.820), 2D-OLPP(33.49,0.842). (*Zoom in pdf for better view*).

the proposed approach. Another denoising experiment on a portion of 'Lena' image with similar set of results is shown in Figure 6. LPG-PCA performs well in homogeneous regions of the image but oversmooths finer textural details. BM3D tends to over sharpen some of the edges (lips and shoulder portions of Lena image). BM3D also smooths out finer textures as it uses a tensor product of DCT and Haar wavelet basis, the latter of which is akin to performing diffusion filtering which can erase subtle textures [31]. EPLL restores textural details but sometimes produces spiky artifacts in the denoised image which can be clearly observed by zooming in. CSR based denoising approach produces small blocky artifacts specially in smooth (homogeneous) regions which are not present in outputs produced by the proposed approach. It is observed that the proposed approach outperforms all other approaches

visually as well as quantitatively. It is to be noted that the proposed approach is able to preserve the finer textural details even at higher noise levels.

2D-OLPP is a two-dimensional extension of OLPP. It is not only computationally more efficient as compared to OLPP (as discussed in Section III-C) but also processing two dimensional patches directly improves the denoising performance. In the process of matrix to vector conversion, neighborhood information of a pixel is not retained. This may not play any role in maintaining the similarity criterion, but while learning the basis, neighbors of a pixel turn out to be informative. It seems that finer textural details are restored well by the OLPP based approach but it undersmooths the homogeneous regions, hence performance of OLPP degrades for natural images. Artifacts are visible specially on the skin portions of both the 'Barbara' and 'Lena' images (denoised) using OLPP based method. 2D-OLPP overcomes these limitations and produces superior results both quantitatively and visually.

The PSNR and SSIM values for some of the images are reported in Figure 7 for 3 noise levels $\sigma \in \{20, 30, 40\}$ using various denoising approaches. The results are shown in graphical format for better visual comprehension. For each noise level, the resultant value is shown using a different symbol. Note that the results are not continuous, results of all the methods are joined by a line for each noise level for ease of comparison.

In order to show fine structure preservation property, experiments are performed on the UIUC Texture database (http://www-cvr.ai.uiuc.edu/ponce_grp/data/index. html#texture) containing 100 images of 17 different textures such as bark, brick, wood, carpet, wall etc. Average PSNR and SSIM values for some of the textures (by averaging all the images having same texture) are reported in Figure 8 for 3 different noise levels $\sigma \in \{20, 30, 40\}$. It can be observed that for both the gray-scale image databases, the proposed approach outperforms all the approaches both in terms of PSNR and SSIM values. Average results for all 100 images from UIUC texture database are reported in Table V. For both natural and texture databases, it can be observed that both the similarity preserving transformed domain approaches, i.e. OLPP and 2D-OLPP outperform the denoising results obtained using global PCA in the same framework, which shows the impact of locality preserving property while learning the basis.

B. Color Image Denoising

The proposed denoising approach is also applied for color image denoising. As the codes for EPLL and CSR approaches were not available for color images, comparison of the proposed approach with global PCA, LPG-PCA, BM3D and OLPP is reported. BM3D uses decorrelated color space YCbCr for the denoising purpose. Noise removal is performed only in the Y-channel of the image while Cb and Cr channels are kept as it is and after denoising the Y-channel, the image is transformed back to the original RGB color space. Thus, the information in Cb and Cr channels is ignored completely. On the other hand LPG-PCA denoises R, G and B channels independently, treating each of them as a separate
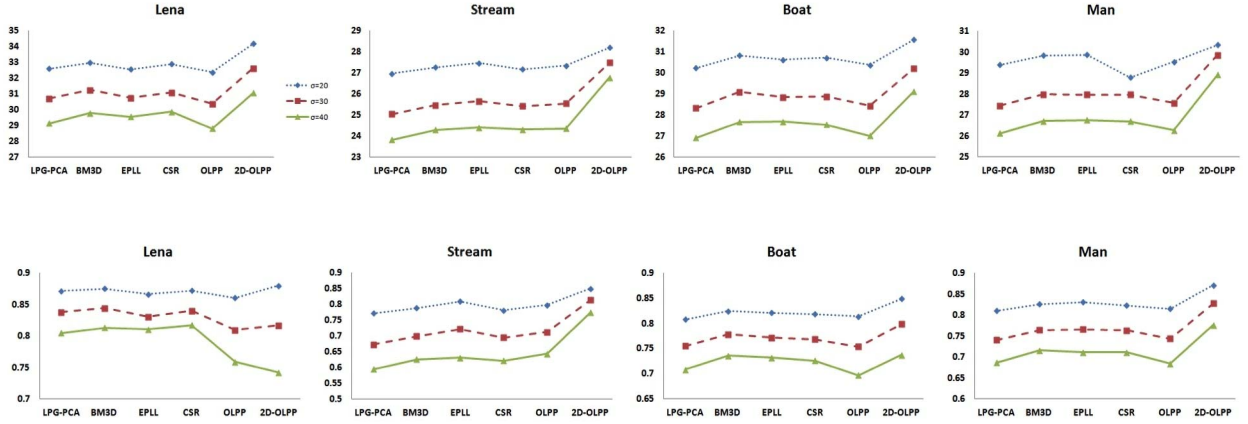
Fig. 7.   Denoising results of some of the images from the Lansel database of gray-scale natural images. For each image, results in terms of PSNR and SSIM are shown. Horizontal axis represents various methods of denoising and vertical axis represents the PSNR and SSIM values (first and second row respectively) for all the images. Results are shown for three noise levels: $\sigma = 20$ (dotted), $\sigma = 30$ (dashed), $\sigma = 40$ (solid) as indicated in the left most graph of the first row.
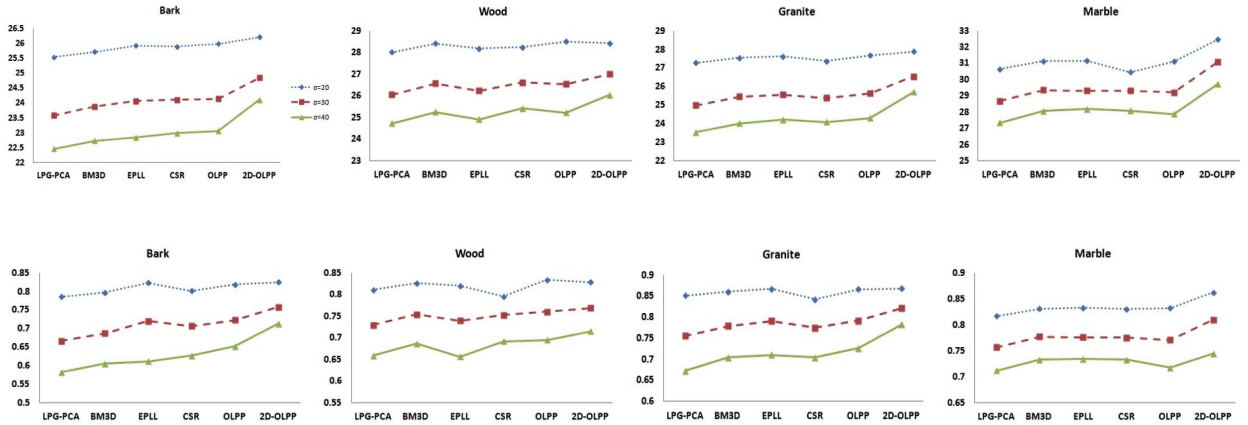

Fig. 8.   Denoising results of some of the images from the UIUC Texture database (gray-scale). For each image, results in terms of PSNR and SSIM are shown. Horizontal axis represents various methods of denoising and vertical axis represents the PSNR and SSIM values (first and second row respectively) for all the images. Results are shown for three noise levels: $\sigma = 20$ (dotted), $\sigma = 30$ (dashed), $\sigma = 40$ (solid) as indicated in the left most graph of the first row.

Fig. 9.   Denoising experiment on a color image from the Kodak database. Left to right, top to bottom: Clean Image, Noisy Image ($\sigma = 50$), LPG-PCA (26.96, 0.487), BM3D (29.91, 0.749), OLPP (28.80, 0.699), 2D-OLPP (31.67, 0.765). (*Zoom in pdf for better view*).

gray-scale image. In this case, the dependency or correlation between the R, G, B channels is lost. In case of global PCA and OLPP, denoising is performed by converting the 3D patch in vector format.

In order to preserve the dependencies between R, G, B channels as well as to process the 3D block/patch altogether, matricization, also known as unfolding of the 3-dimensional block, formally called 3D tensor [21], is carried out. As we have R, G and B- 3 channels, the 3-dimensional array is reordered into 3-mode matrices using the unfolding process. Each mode is denoised using the proposed technique and

folded back to form the 3D block. Due to the 3-mode unfolding, resultant 3 folds are averaged to produce the final denoised block. The process of unfolding is described in Appendix.

Two color image data sets namely the Kodak image database (http://r0k.us/graphics/kodak/) containing 24 natural images and first 28 images of the Brodatz color texture database (http://multibandtexture.recherche.usherbrooke.ca/colored_brodatz_more.html) are used for experimentation. $12 \times 12 \times 3$ image patches are used for denoising process. To establish the unfolding based approach for color image denoising, experiments have been conducted using both
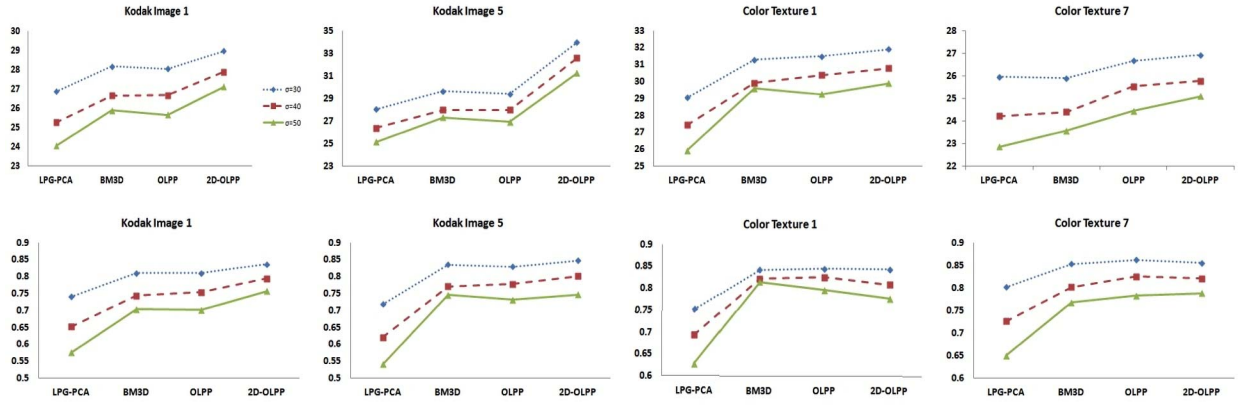
Fig. 10. Denoising results of some of the images from the Kodak database of color natural images (column 1 and 2) and the Brodatz color texture database (column 3 and 4). Horizontal axis represents various methods of denoising and vertical axis represents the PSNR and SSIM values (first and second row respectively) for all the images. Results are shown for three noise levels: $\sigma = 30$ (dotted), $\sigma = 40$ (dashed), $\sigma = 50$ (solid) as indicated in the left most graph of the first row.

TABLE V

AVERAGE DENOISING RESULTS FOR ALL THE IMAGES FROM GRAY-SCALE DATABASES ALONG WITH BEST AND WORST PERFORMANCES FOR THE PROPOSED 2D-OLPP BASED DENOISING APPROACH: THE LANSEL DATABASE AND THE UIUC TEXTURE DATABASE ($\sigma \in \{20, 30, 40\}$)

Database	Denoising Approach	$\sigma = 20$		$\sigma = 30$		$\sigma = 40$	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Global PCA	30.31	0.829	28.32	0.767	26.92	0.710
	LPG-PCA	30.42	0.834	28.48	0.78	27.08	0.73
	BM3D	30.78	0.84	29.03	0.796	27.66	0.754
Lansel	EPLL	30.46	0.839	28.60	0.788	27.48	0.747
	CSR	30.69	0.841	28.90	0.793	27.76	0.755
	OLPP	30.45	0.838	28.52	0.787	27.20	0.743
	2D-OLPP	**31.68**	**0.872**	**30.65**	**0.821**	**29.43**	**0.762**
Lansel	2D-OLPP (Best)	36.08	0.954	34.14	0.945	32.09	0.934
	2D-OLPP (Worst)	27.40	0.816	26.00	0.762	25.08	0.696
	Global PCA	28.44	0.829	26.59	0.756	25.27	0.693
	LPG-PCA	28.25	0.807	26.24	0.722	24.96	0.654
	BM3D	28.58	0.819	26.74	0.747	25.49	0.686
UIUC Texture	EPLL	28.46	0.824	26.58	0.746	25.37	0.669
	CSR	28.54	0.817	26.75	0.748	25.57	0.683
	OLPP	28.67	0.829	26.79	0.757	25.56	0.696
	2D-OLPP	**29.20**	**0.843**	**27.91**	**0.789**	**26.92**	**0.739**
UIUC Texture	2D-OLPP (Best)	32.47	0.945	31.08	0.926	29.71	0.903
	2D-OLPP (Worst)	26.20	0.773	24.83	0.710	24.11	0.642

YCbCr and separate RGB channels based denoising in the 2D-OLPP based framework. Average results using YCbCR, separate RGB and proposed unfolding for some of the images from Kodak database with noise level ($\sigma = 20$) in terms of (PSNR, SSIM) are (29.59, 0.420), (29.39, 0.753) and (34.84, 0.828) respectively. It can be observed that, in the proposed 2D-OLPP denoising framework, unfolding based processing of color images is surpassing the other schemes, and hence all the further experiments reported in this article have been performed using the unfolding based procedure.

Denoising experiment on a color image from the Kodak database is shown in Figure 9. The resultant PSNR and SSIM values clearly indicate superiority of the proposed approach over LPG-PCA, BM3D and OLPP. Smoothness achieved in the denoised image using BM3D in homogeneous regions is better but it also smooths edges in those regions. Some undesired textures get generated while restoring the image in case of both LPG-PCA and BM3D. As mentioned earlier, restored image
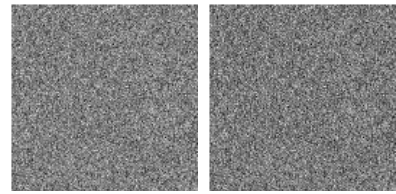

Fig. 11. Noise-to-noise criterion (Left) Original white noise image ($\mathbf{N}(0, \sigma)$, $\sigma = 40$) (Right) Output of the proposed 2D-OLPP approach.

using OLPP is under-smoothed and contains some blob-like artifacts.

PSNR and SSIM values for some of the images from the Kodak database and Brodatz color texture database are reported in graphical format in Figure 10 for 3 different noise levels $\sigma \in \{30, 40, 50\}$, along with the average results for all the images in Table VI. Proposed denoising scheme beats the other approaches with a margin of almost $1db$ (PSNR) for all the three noise levels in case of the Kodak database.

TABLE VI

AVERAGE DENOISING RESULTS FOR COLOR IMAGE DATABASES ALONG WITH BEST AND WORST PERFORMANCES FOR THE PROPOSED 2D-OLPP
BASED DENOISING APPROACH: THE KODAK DATABASE AND THE BRODATZ TEXTURE DATABASE ($\sigma \in \{30, 40, 50\}$)

| Database	Denoising Approach	$\sigma = 30$		$\sigma = 40$		$\sigma = 50$	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Kodak	Global PCA	28.56	0.757	26.89	0.700	25.62	0.649
	LPG-PCA	28.32	0.72	26.63	0.628	25.36	0.556
	BM3D	30.20	0.839	28.51	0.783	27.90	0.761
	OLPP	29.51	0.825	28.16	0.776	27.10	0.732
	2D-OLPP	**31.34**	**0.842**	**29.84**	**0.806**	**28.85**	**0.790**
Kodak	2D-OLPP (Best)	33.98	0.873	32.84	0.842	31.67	0.838
	2D-OLPP (Worst)	28.36	0.797	27.17	0.751	26.08	0.729
Brodatz color texture	Global PCA	24.37	0.766	22.82	0.720	21.80	0.690
	LPG-PCA	26.62	0.816	24.92	0.750	23.52	0.680
	BM3D	27.28	0.866	25.67	0.817	25.18	0.795
	OLPP	**27.82**	**0.876**	**26.53**	**0.841**	25.40	0.803
	2D-OLPP	27.55	0.868	26.37	0.834	**25.61**	**0.808**
Brodatz color texture	2D-OLPP (Best)	30.78	0.922	30.78	0.895	29.898	0.873
	2D-OLPP (Worst)	24.46	0.817	22.742	0.779	21.540	0.727
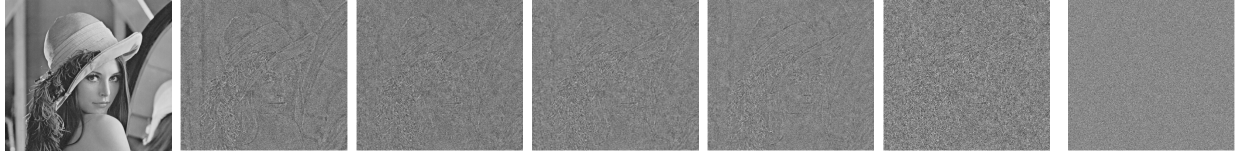
Fig. 12.   Method noise comparison on a natural image ('Lena'). Left to right: Clean image, LPG-PCA, BM3D, EPLL, CSR, OLPP, 2D-OLPP. (*Zoom in pdf for better view*).

C. The Noise Test

As suggested in [4], one of the way to characterize artifact free approaches is to check how the approach behaves on white noise image. Given a white noise image as input of the denoising approach, it should get transformed into white noise as only white noise is free from structure [2]. This is called the noise-to-noise criterion. Hence, if the output still remains white noise with same or lower variance, it will not have any artifacts in the denoised image. To test the noise-to-noise criterion, on the proposed 2D-OLPP based approach, a white noise image ($\mathcal{N}(o, \sigma), \sigma = 40$) was given as input. The resultant image along with the input image is shown in Figure 11. The proposed approach transforms white noise into white noise and hence satisfies the noise-to-noise criterion.

Method noise, also known as the residual image, is defined as the difference between a noisy image and its denoised version [3], [4]. It is one of the most powerful visual evaluation measures for denoising approaches. Ideally it should look like pure noise (white noise). Many denoising approaches remove image information (mainly the edges and fine textures) while eliminating noise. Method noise shows what part of noisy image has been removed by denoising process. Hence, the more method noise resembles white noise, lesser information content has been removed while denoising and the better is the denoising approach. Method noise for 'Lena' image using different denoising approaches is shown in Figure 12. All the prominent edges/features are clearly visible in method noise produced by the LPG-PCA approach. BM3D and EPLL both have tendency to smooth out sharp edges, which is evident from the residual images as most prominent edges of the respective images are present. Still, method noise is reduced considerably in both the approaches. In comparison

with LPG-PCA, method noise is lesser in case of CSR, but structure of the original image is present in the residual image. Residual image produced from the denoised image using OLPP is close to noise, but the structural details can be observed by zooming in. We request the readers to zoom in the pdf to have better view. For 2D-OLPP, method noise for 'Lena' image completely resembles white noise and no structural details are present.

V. CONCLUSIONS

Formulation of Two Dimensional Orthogonal Locality Preserving Projection (2D-OLPP) along with its application to image denoising is presented in this article. For denoising, 2D-OLPP processes two-dimensional image patches directly that preserves the spatial information. Computational complexity of 2D-OLPP is derived and it has been shown that it is significantly less than that of the other denoising algorithms. In contrast to the state of the art algorithms for denoising where basis are computed for each image patch, a global basis is sufficient for the entire image in 2D-OLPP. The approach is tested extensively on several benchmark data sets. The results obtained are very encouraging and appeared to be comparable with the forerunner approaches of image denoising. Finer textural details are well preserved even at higher noise levels. The proposed approach can further be extended for image deblurring and inpainting tasks.

APPENDIX
UNFOLDING THEORY

A tensor is a multidimensional array and order/mode of a tensor is the number of dimensions [8], [21]. For an n-order tensor $\mathbf{X} \in \mathbf{R}^{\mathbf{I_1} \times \mathbf{I_2} \times \cdots \times \mathbf{I_N}}$, mode-n unfolding is denoted

by $\mathbf{X_{(n)}}$. The tensor element $(i_1, i_2, \ldots, i_N)$ gets mapped to the matrix element (i_n, j). Value of j is computed as follows (refer [21] for details):

$$j = 1 + \sum_{k=1, k \neq n}^{N} (i_k - 1) J_k; \quad J_k = \prod_{m=1, m \neq n}^{k-1} I_m. \quad (16)$$

REFERENCES

[1] S. K. Abramov, V. V. Lukin, B. Vozel, K. Chehdi, and J. T. Astola, "Segmentation-based method for blind evaluation of noise variance in images," *J. Appl. Remote Sens.*, vol. 2, no. 1, p. 023533, 2008.

[2] F. Attneave, "Some informational aspects of visual perception," *Psychol. Rev.*, vol. 61, no. 3, pp. 183–193, 1954.

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

[4] A. Buades, B. Coll, and J. M. Morel, "Image denoising methods. A new nonlocal principle," *SIAM Rev.*, vol. 52, no. 1, pp. 113–147, 2010.

[5] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal Laplacianfaces for face recognition," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3608–3614, Nov. 2006.

[6] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Trans. Image Process.*, vol. 9, no. 9, pp. 1522–1531, Sep. 2000.

[7] S. Chen, H. Zhao, M. Kong, and B. Luo, "2D-LPP: A two-dimensional extension of locality preserving projections," *Neurocomputing*, vol. 70, nos. 4–6, pp. 912–921, 2007.

[8] R. Costantini, L. Sbaiz, and S. Süsstrunk, "Higher order SVD analysis for dynamic texture synthesis," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 42–52, Jan. 2008.

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

[10] W. Dong, X. Li, D. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *Proc. IEEE CVPR*, Jun. 2011, pp. 457–464.

[11] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[12] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Statist. Assoc.*, vol. 90, no. 432, pp. 1200–1224, 1995.

[13] M. Elad and M. Aharon, "Image denoising via learned dictionaries and sparse representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2006, pp. 895–900.

[14] M. Ghazal and A. Amer, "Homogeneity localization using particle filters with application to noise estimation," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1788–1796, Jul. 2011.

[15] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Proc. 10th IEEE ICCV*, vol. 2. Oct. 2005, pp. 1208–1213.

[16] X. He and P. Niyogi, "Locality preserving projections," in *Proc. Adv. Neural Inf. Process. Syst.*, Mar. 2003, pp. 153–160.

[17] P. Hoyer, "Independent component analysis in image denoising," Ph.D. dissertation, Dept. Comput. Sci. Eng., Helsinki Univ. Technol., Helsinki, Finland, 1999.

[18] A. Hyvärinen, "Survey on independent component analysis," *Neural Comput. Surv.*, vol. 2, no. 4, pp. 94–128, 1999.

[19] I. M. Johnstone and B. W. Silverman, "Wavelet threshold estimators for data with correlated noise," *J. Roy. Statist. Soc.*, vol. 59, no. 2, pp. 319–351, 1997.

[20] I. Jolliffe, *Principal Component Analysis*. Berlin, Germany: Springer-Verlag, 2002.

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

[22] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 2272–2279.

[23] D. D. Muresan and T. W. Parks, "Adaptive principal components and image denoising," in *Proc. ICIP*, Sep. 2003, pp. 101–104.

[24] S. Pyatykh, J. Hesser, and L. Zheng, "Image noise level estimation by principal component analysis," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 687–699, Feb. 2013.

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

[26] G. Shikkenawis and S. K. Mitra, "Improving the locality preserving projection for dimensionality reduction," in *Proc. IEEE 3rd Int. Conf. Emerg. Appl. Inf. Technol.*, Nov./Dec. 2012, pp. 161–164.

[27] G. Shikkenawis, S. K. Mitra, and A. Rajwade, "A new orthogonalization of locality preserving projection and applications," in *Pattern Recognition and Machine Intelligence*. Berlin, Germany: Springer-Verlag, 2013, pp. 277–283.

[28] G. Shikkenawis, S. K. Mitra, and A. Rajwade, "Image denoising using orthogonal locality preserving projections," *J. Electron. Imag.*, vol. 24, no. 4, p. 043018, 2015.

[29] S. Wang, N. Zhang, X. Peng, and C. Zhou, "Two-dimensional locality preserving projection based on maximum scatter difference," *J. Inf. Comput. Sci.*, vol. 8, no. 3, pp. 484–494, 2011.

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[31] M. Welk, G. Steidl, and J. Weickert, "Locally analytic schemes: A link between diffusion filtering and wavelet shrinkage," *Appl. Comput. Harmon. Anal.*, vol. 24, no. 2, p. 195–224, 2008.

[32] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004.

[33] 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.

[34] H. Zhang, Q. M. J. Wu, T. W. S. Chow, and M. Zhao, "A two-dimensional neighborhood preserving projection for appearance-based face recognition," *Pattern Recognit.*, vol. 45, no. 5, pp. 1866–1876, 2012.

[35] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognit.*, vol. 43, no. 4, pp. 1531–1549, Apr. 2010.

[36] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. IEEE ICCV*, Nov. 2011, pp. 479–486.

Gitam Shikkenawis received the bachelor's degree in computer science from Sardar Patel University, India, in 2010, and the M.Tech. degree in machine intelligence from the Dhirubhai Ambani Institute of Information and Communication Technology (DAIICT), India, in 2012. She is currently a Ph.D. Scholar with DAIICT. Her research interests include pattern recognition, image processing, and machine learning. She was a recipient of the Google India Women in Engineering Award in 2011.

Suman K. Mitra received the Ph.D. degree from the Indian Statistical Institute, Calcutta. He was with the Institute of Neural Computation, University of California at San Diego, USA, and the Department of Mathematics, IIT Bombay. He is currently a Professor and the Dean of Academic Program with the Dhirubhai Ambani Institute of Information and Communication Technology, India. His research interests include image processing, pattern recognition, Bayesian networks, and digital image processing.