Patch-based Image Denoising: Probability Distribution Estimation vs. Sparsity Prior

Dai-Viet Tran[1], Sébastien Li-Thiao-Té[1], Marie Luong[2], Thuong Le-Tien[3], Françoise Dibos[1]

[1]Laboratoire LAGA - Université Paris 13, France, [2]Laboratoire L2TI - Université Paris 13, France,
[3]Hochiminh University of Technology, Vietnam

Abstract—Patch-based image denoising can be interpreted under the Bayesian framework which incorporates the image formation model and a prior image distribution. In the sparsity approach, the prior is often assumed to obey an arbitrarily chosen distribution. Our motivation is to estimate the probability directly from the distribution of image patches extracted from good quality images, thanks to a given dictionary and the redundancy of information between many images. In this paper, we provide a scheme to estimate the probability distribution and also an optimized algorithm for denoising. We demonstrate that using the estimated probability distribution as the image prior is more efficient than the state-of-the-art sparsity models for noise removal.

Index Terms—Patch-based, denoising, sparse representation, probability distribution estimation.

I. INTRODUCTION

Image denoising is a well-known technique in low level computer vision that aims to recover a clean image $\mathbf{X}$ from its noise-corrupted observation $\mathbf{Y}$. We assume that the noisy image is generated by $\mathbf{Y} = \mathbf{X} + \mathbf{V}$, where $\mathbf{V}$ is zero-mean additive Gaussian noise with standard deviation σ. Many denoising approaches in the state-of-the-art have been developed, including filtering based [1], total variation based [2], transform based [3], learning based [4], sparse representation based [5, 6], or deep-learning based methods [7, 8], etc. Among various contributions, the class of patch-based representation methods [6, 9–12] have achieved success and attracted intensive research interests in the past decade. In these algorithms, a noisy image $\mathbf{Y}$ is divided into a set of overlapping image patches $\{\mathbf{y}\}$ ($\mathbf{y} \in R^m$), and the denoising problem is considered on each patch. The noisy patch is modeled as $\mathbf{y} = \mathbf{x} + \mathbf{v}$, where $\mathbf{x} \in R^m$ is the corresponding clean version of $\mathbf{y}$ and $\mathbf{v}$ is the additive Gaussian noise in each patch.

The idea of patch-based denoising is based on an interesting observation in which a clean image patch $\mathbf{x}$ can be represented as a linear combination of atoms in a given dictionary $\mathbf{D}$, $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}$, with $\mathbf{D} \in R^{m \times K}, m \leq K$, and $\boldsymbol{\alpha} \in R^K$ is a coefficient vector. The denoising task is equivalent to solving for the coefficient vector $\boldsymbol{\alpha}$ such that $\mathbf{y} = \mathbf{D}\boldsymbol{\alpha} + \mathbf{v}$. More precisely, our objective is to find the value of $\boldsymbol{\alpha}$ under a given dictionary $\mathbf{D}$ and a noisy observation $\mathbf{y}$, which maximizes the posterior probability distribution $p(\boldsymbol{\alpha}|\mathbf{D}, \mathbf{y})$. Using Bayes's theorem, we can formulate our problem as:

$$\boldsymbol{\alpha} = \arg\max_{\boldsymbol{\alpha}} p(\boldsymbol{\alpha}|\mathbf{D}, \mathbf{y}) = \arg\max_{\boldsymbol{\alpha}} p(\mathbf{y}|\mathbf{D}, \boldsymbol{\alpha})p(\boldsymbol{\alpha}) \quad (1)$$

By assuming that $\mathbf{y}$ is disturbed by additive Gaussian noise $\mathbf{v} \sim \mathcal{N}(0, \sigma^2)$, the likelihood term in (1) is determined as:

$$p(\mathbf{y}|\mathbf{D}, \boldsymbol{\alpha}) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2} \parallel \mathbf{y} - \mathbf{D}\boldsymbol{\alpha} \parallel_2^2\right) \quad (2)$$

Putting (2) into (1) and using the minus logarithm, we have:

$$\boldsymbol{\alpha} = \arg\min_{\boldsymbol{\alpha}} \left\{\parallel \mathbf{y} - \mathbf{D}\boldsymbol{\alpha} \parallel_2^2 - \lambda \log p(\boldsymbol{\alpha})\right\} \quad (3)$$

Where $\lambda = 2\sigma^2$. The problem in (3) cannot be solved without any prior knowledge of $p(\boldsymbol{\alpha})$. In the literature, many authors supposed that $p(\boldsymbol{\alpha})$ follows an i.i.d. (independent and identically distributed) distribution law (e.g. Laplacian [4, 13], etc). Mathematically, this leads to model the log-likelihood probability ($\log p(\boldsymbol{\alpha})$) by the ℓ_p-norm: $\log p(\boldsymbol{\alpha}) = \gamma ||\boldsymbol{\alpha}||_p = \gamma(\sum_i |\alpha_i|^p)^{1/p}$, where $\gamma > 0$ is a regularization parameter and $p \geq 0$. Typically, the sparsity of a vector $\boldsymbol{\alpha}$ is related to the so-called ℓ_0-norm, which counts the number of the nonzero entries of $\boldsymbol{\alpha}$. By that way, an image patch can be represented as a linear combination of few atoms in the dictionary [5]. In practice, denoising problems can be resolved in taking into account ℓ_p-norms with $0 \leq p \leq 2$. Fig. 1 shows the distribution of coefficients $\boldsymbol{\alpha}$ in R^2 with different type of sparsity priors. Many algorithms have been developed to solve the sparse problem with different hypotheses of sparsity such that ℓ_0-norm [14], ℓ_p-norm ($0 \leq p \leq 1$) [15], ℓ_1-norm [16], ℓ_2-norm [17].
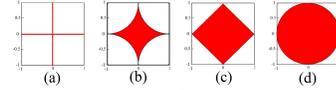

Fig. 1: Geometric interpretations of different norms in 2-D space. (a), (b), (c), (d) are the unit ball of the ℓ_0-norm, ℓ_p-norm ($0 \leq p \leq 1$), ℓ_1-norm and ℓ_2-norm in 2-D space, respectively

In the sparse denoising approach, the reconstruction of a noisy patch under a given dictionary is forced to an arbitrary distribution (e.g. Laplacian [4, 13], etc). But in reality, there are no guarantees that $p(\boldsymbol{\alpha})$ obeys this distribution. Our motivation is to develop a patch-based denoising method where $p(\boldsymbol{\alpha})$ is estimated from the distribution of coefficient vectors $\boldsymbol{\alpha}$ in a R^K space; this distribution is calculated with the help of standard (clean, good quality) images and a pre-trained dictionary. This idea exploits the redundancy between standard images and the denoised image, where visual contents tend to repeat in different images. Therefore, the patches in the noiseless image follow a similar distribution as the database constructed from standard images. Hence, we can use $p(\boldsymbol{\alpha})$ learned from the database to denoise an image patch. The idea of probability distribution estimation is described as follows. From the standard images, we randomly extract a set of N clean image patches to form a database of patches, where each patch is denoted as $\mathbf{x} = [x_1, \ldots, x_m]^T \in R^m$. Let $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_K] \in R^{m \times K}$ denote a given dictionary. A good quality patch $\mathbf{x}$ in the database can be represented in dictionary space $\Omega_\alpha \in R^K$ by its orthogonal projection:

$$\boldsymbol{\alpha} = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{x} = \mathbf{P}_D\boldsymbol{\alpha} \quad (4)$$

where $\mathbf{P}_D = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T$ is the projection matrix. Hence, with a database of patches in the image domain, we can obtain a distribution of decomposition coefficients in Ω_α. Fig. 2 shows the distribution of $\boldsymbol{\alpha}$ in 3D and 2D visualization, respectively, whose forms are different from sparse distribution in Fig. 1.

There are two main contributions in this paper. First, we propose using the probability distribution function $p(\boldsymbol{\alpha})$, which is estimated from image patches in the database, as an image prior to replace the sparsity models in the denoising problem. Second, we propose a denoising strategy to take into account the obtained $p(\boldsymbol{\alpha})$ in solving the optimization problem in (3). The remainder of this paper is organized as follows: Section II tackles the estimation of the probability distribution $p(\boldsymbol{\alpha})$ in which $p(\boldsymbol{\alpha})$ is a piece-wise constant function. Then, a scheme for solving the optimization problem indicated in (3) is proposed to find $\boldsymbol{\alpha}$. Experimental results presented in section III

prove the out-performance of proposed method over the sparsity ones. Section IV discusses the limitations, future developments and concluding remarks.
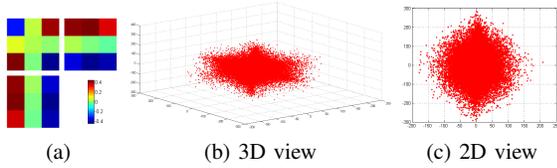

| (a) | (b) 3D view | (c) 2D view |

Fig. 2: Distribution of decomposition coefficients $\boldsymbol{\alpha}$ of 500000 patches in the database. (a) The given dictionary with 3 atoms. (b) 3D view of Ω_α. (c) 2D view of Ω_α.

II. DENOISING BY PROBABILITY DISTRIBUTION ESTIMATION

A. Estimation of probability distribution $p(\boldsymbol{\alpha})$ from the database

As estimation is a difficult problem in high dimensional spaces, we consider the $\boldsymbol{\alpha}$-space with dimensions up to three ($\Omega_\alpha \in R^3$). This is equivalent to choosing a dictionary with only three atoms ($K = 3$) $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3] \in R^{m \times 3}$. After that, we randomly extract a set of N image patches $\mathbf{x} \in R^m$ from the standard images to create a database of patches. In the next stage, all of patches in the database are projected into $\boldsymbol{\alpha}$-space Ω_α using the projection matrix $\mathbf{P}_D$. As a result, we obtain a distribution of N points in Ω_α. To demonstrate, we used the Kodak PhotoCD Dataset (Fig. 4) as standard images and randomly extracted a set of 500000 image patches. After the decomposition in (4), we obtained 500000 corresponding points distributed in Ω_α as shown in Fig. 2.

In the scope of this work, rather than forcing the probability distribution $p(\boldsymbol{\alpha})$ to a specific i.i.d. distribution law according to a sparsity model ($\log(p(\boldsymbol{\alpha})) \sim ||\boldsymbol{\alpha}||_p$), we prefer to estimate it directly from the distribution of patches in Ω_α. A simple and efficient approach is to discretize Ω_α into a set of disjoint subvolumes, $\Omega_\alpha = \{\Omega_\alpha^s\}$, and count the density of occurrence of points in each subvolume Ω_α^s. As a result, $p(\boldsymbol{\alpha})$ becomes a piece-wise constant value function. Precisely, we divide Ω_α into grid cells (each cell corresponds to a subvolume Ω_α^s), in which each dimension is split into a set of intervals along the range limited by the minimum and maximum values of $\boldsymbol{\alpha}$ in this dimension. The probability distribution $p(\boldsymbol{\alpha})$ is a cell-wise constant function whose values are calculated by the density of number of points falling in each cell as indicated in (5).

$$p(\boldsymbol{\alpha}) = \begin{cases} \dfrac{1}{V_\alpha^s} \dfrac{\sum point \in \Omega_\alpha^s}{\sum point \in \Omega_\alpha} = \dfrac{1}{V_\alpha^s} \dfrac{N_s}{N}, & \text{if } \boldsymbol{\alpha} \in \Omega_\alpha^s \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where N_s is the number of points inside cell Ω_α^s which contains $\boldsymbol{\alpha}$, V_α^s is the volume of Ω_α^s (as $\mathbf{D} \in R^{m \times 3}$), N is the total number of points in Ω_α that is equal to the number of image patches in the database.

An important notification in the estimation of $p(\boldsymbol{\alpha})$ is how to divide each dimension of Ω_α into the set of intervals to restrict the number of empty subvolumes, in which the solution of $\boldsymbol{\alpha}$ in (3) is undetermined. Our strategy is to split each axis of Ω_α such that the same number of points falls into each interval. By that way, the length of each interval can adapt to the density of points. In the experiments, each axis of Ω_α is separated into 2^n intervals by $2^n - 1$ median points using a recursive division (n times), so that each interval contains $N/2^n$ points. As a result, we obtain a grid of $(2^n - 1)^K$ quantiles $\mathcal{G}_\alpha = \{\boldsymbol{\alpha}^i | i = 1, \ldots, (2^n - 1)^K\}$ in the R^K space Ω_α. In Fig. 3, we demonstrate an example of distribution of points in Ω_α which is partitioned in $4 \times 4 \times 4$ ($(2^n)^K$ with $n = 2$, $K = 3$) rectangular cells.

B. Solving the optimization problem

In this section, we will present how to employ the estimated $p(\boldsymbol{\alpha})$ in image denoising task. Given the dictionary $\mathbf{D}$ in section II-A and the prior knowledge of probability distribution $p(\boldsymbol{\alpha})$ estimated from the database of patches, our aim is to recover a clean image patch $\mathbf{x}$ from its noisy version $\mathbf{y}$. The denoised patch can be generated as a linear combination of atoms in the dictionary, $\hat{\mathbf{x}} = \mathbf{D}\boldsymbol{\alpha}^*$. Our objective is equivalent to find the value of $\boldsymbol{\alpha}^*$ which satisfies the optimization problem in (3):

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha} \in \Omega_\alpha} \left\{ ||\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}||_2^2 - \lambda \log p(\boldsymbol{\alpha}) \right\} = \arg\min_{\boldsymbol{\alpha} \in \Omega_\alpha} J(\boldsymbol{\alpha}) \quad (6)$$
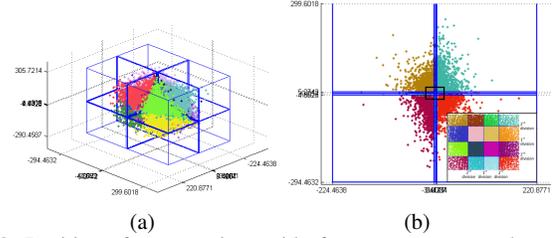

| (a) | (b) |

Fig. 3: Partition of $\boldsymbol{\alpha}$-space into grid of $4 \times 4 \times 4$ rectangular cells. (a) View in 3D. (b) A x-y slice at position $z = 0$ with a zoom in of center.

As indicated in previous section, we have divided the $\boldsymbol{\alpha}$-space Ω_α into disjoint rectangular cells Ω_α^s, and the value of $p(\boldsymbol{\alpha})$ is constant in each cell Ω_α^s. Consequently, the cost function $J(\boldsymbol{\alpha})$ in (6) becomes a cell-wise quadratic function. A simple way for finding the solution of $\boldsymbol{\alpha}$ is to minimize $J(\boldsymbol{\alpha})$ on each rectangular cell Ω_α^s and then select the best value $\boldsymbol{\alpha}^*$ from which we achieve the smallest cost. Therefore, we can rewrite (6) as:

$$\boldsymbol{\alpha}^* = \arg\min_{\Omega_\alpha^s \subset \Omega_\alpha} \left\{ \arg\min_{\boldsymbol{\alpha} \in \Omega_\alpha^s} J(\boldsymbol{\alpha}) \right\} = \arg\min_{\Omega_\alpha^s \subset \Omega_\alpha} J_s(\boldsymbol{\alpha}) \quad (7)$$

With $p(\boldsymbol{\alpha})$ is calculated from (5), the minimum cost function on Ω_α^s, denoted as $J_s(\boldsymbol{\alpha})$, becomes:

$$J_s(\boldsymbol{\alpha}) = \arg\min_{\boldsymbol{\alpha} \in \Omega_\alpha^s} J(\boldsymbol{\alpha}) = \arg\min_{\boldsymbol{\alpha} \in \Omega_\alpha^s} ||\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}||_2^2 - \lambda \log\left(\frac{N_s}{V_\alpha^s N}\right) \quad (8)$$

$J_s(\boldsymbol{\alpha})$ turns into the minimization of a convex quadratic function $J(\boldsymbol{\alpha})$ on a subvolume Ω_α^s. Because there exists only one cell $\Omega_\alpha^d \in \Omega_\alpha$ such that the derivative $\frac{\partial J(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} = 0$ at $\boldsymbol{\alpha}^d = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{y}$, and $\boldsymbol{\alpha}^d \in \Omega_\alpha^d$, it is easy to prove that for all other cells $\Omega_\alpha^s \neq \Omega_\alpha^d$, the optimized value of $J_s(\boldsymbol{\alpha})$ occurs at one of four corners of Ω_α^s. Moreover, we have partitioned the $\boldsymbol{\alpha}$-space Ω_α into 2^{nK} rectangular cells by grid of $(2^n - 1)^K$ quantiles $\mathcal{G}_\alpha$. Therefore, these points becomes the corners of rectangular cells Ω_α^s, and the optimal value of $J(\boldsymbol{\alpha})$ in (6) will take place either at $\boldsymbol{\alpha}^d$ or at one point in $\mathcal{G}_\alpha$. Let $L_\alpha = \{\boldsymbol{\alpha}^d, \mathcal{G}_\alpha\}$ denote the set of anchor points. For a noisy image $\mathbf{y}$, to minimize (8), we can easily compute the value of cost function $J(\boldsymbol{\alpha})$ at anchor points and choose the optimized point $\boldsymbol{\alpha}^*$ among L_α where $J(\boldsymbol{\alpha}^*)$ is minimum. The value of $p(\boldsymbol{\alpha})$ (second term in (8)) and the production $\mathbf{D}\boldsymbol{\alpha}$ at each anchor point $\boldsymbol{\alpha} \in L_\alpha$ can be computed offline to accelerate the speed of the optimization of (8). The complexity of solving (8) is $O(m^2 2^{nK})$ which depends on the patch size ($\mathbf{y} \in R^m$) and the number of anchor points ($(2^n - 1)^K$). To sum up, the denoising work flow involving the optimization process is described in algorithm (1).

As it can be seen in (6), the estimation of probability distribution $p(\boldsymbol{\alpha})$ has a direct influence on solving $\boldsymbol{\alpha}$. The two important parameters that affect the determination of $p(\boldsymbol{\alpha})$ are the number of dimensions (K) of $\boldsymbol{\alpha}$-space Ω_α, as well as the number of intervals along each dimension. The former, K is related to the number of

atoms in the dictionary (K), and the latter depends on the number of grid points we choose on each dimension. In our experiments, each coordinate axis of Ω_α is split by 32 intervals (2^n intervals with $n = 5$). Moreover, in high dimensional space Ω_α ($K > 3$), for a given distribution of patches (points) in Ω_α, the number of rectangular cells will increase exponentially, resulting in a large portion of empty cells, and thus invalid estimates. For the visualization purpose, we choose a dictionary composed of only three elements ($K = 3$) so that $p(\boldsymbol{\alpha})$ can be visually drawn in a three dimensional coordinates.

Algorithm 1: Denoising

Input : Set of standard images $\{\mathbf{X}\}$, Noisy image $\mathbf{Y}$, Dictionary $\mathbf{D}$, noise level σ, regularized parameter λ, patch size, overlap between two patches.

1 Estimate the probability distribution $p(\boldsymbol{\alpha})$ using (5) by first determining the grid of $(2^n - 1)^K$ median points $\mathcal{G}_\alpha = \{\boldsymbol{\alpha}^i | i = 1, \ldots, (2^n - 1)^K\}$ (divide each dimension into 2^n intervals, each one contains $N/2^n$ points) from standard images $\{\mathbf{X}\}$ and dictionary $\mathbf{D}$ as described in section (II-A).

2 Partition noisy image $\mathbf{Y}$ into overlapping patches $\mathbf{y}$

3 **for** *each image patch* $\mathbf{y} \in \mathbf{Y}$ **do**

4   Subtract its mean value: $\mathbf{y} = \mathbf{y} - \boldsymbol{\mu}_y$.

5   Calculate $\boldsymbol{\alpha}^d = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{y}$ and find its rectangular cell Ω_α^d such that $\boldsymbol{\alpha}^d \in \Omega_\alpha^d \subset \Omega_\alpha$.

6   Define a set of anchor points: $L_\alpha = \{\boldsymbol{\alpha}^d, \mathcal{G}_\alpha\}$

7   Calculate values of the cost function $J(\boldsymbol{\alpha})$ in (6) at all anchor points $\boldsymbol{\alpha} \in L_\alpha$.

8   Choose the optimized value $\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha} \in L_\alpha} J(\boldsymbol{\alpha})$.

9   Estimate the denoised image patch: $\hat{\mathbf{x}} = \mathbf{D}\boldsymbol{\alpha}^* + \boldsymbol{\mu}_y$

10 **end**

11 Average the overlapped regions of denoised patches $\hat{\mathbf{x}}$ to obtain the entire denoised image $\hat{\mathbf{X}}$.

Output: The denoised image $\hat{\mathbf{X}}$.

III. Denoising performance and evaluation

In this section, our aim is to demonstrate that for a given dictionary, using the estimated probability distribution as the image prior is more efficient than the sparsity models for noise removal. Given a dictionary, our method differs from the existing sparsity models in the way we determine the probability distribution function $p(\boldsymbol{\alpha})$. While the sparsity methods assume that the probability $p(\boldsymbol{\alpha})$ in Ω_α obeys a specific model, we propose to estimate $p(\boldsymbol{\alpha})$ directly from the distribution of $\boldsymbol{\alpha}$ in Ω_α. In the remaining of this section, we compare the denoising performance of our proposed method with two famous sparse models ℓ_0-norm (9) and ℓ_1-norm (10), which known as least absolute shrinkage and selection operator (LASSO). The comparison evaluates the influence of the chosen probability models of $p(\boldsymbol{\alpha})$ (sparsity or estimation) on the reconstruction of noisy image via solving the optimization problem (6). Therefore, we fix the dictionary $\mathbf{D}$, as well as the size of image patch $\mathbf{y}$ for the proposed and sparse methods to have a nondiscriminatory assessment. Comparison with other denoising methods such as BM3D [3], NLM [9], deep-learning [8] is out of the scope of the current manuscript.

$$\boldsymbol{\alpha} = \arg\min_{\boldsymbol{\alpha}} \left\{ \| \mathbf{y} - \mathbf{D}\boldsymbol{\alpha} \|_2^2 - \lambda ||\boldsymbol{\alpha}||_0 \right\} \quad (9)$$

$$\boldsymbol{\alpha} = \arg\min_{\boldsymbol{\alpha}} \left\{ \| \mathbf{y} - \mathbf{D}\boldsymbol{\alpha} \|_2^2 - \lambda ||\boldsymbol{\alpha}||_1 \right\} \quad (10)$$

The problem in (9) can be efficiently solved by the orthogonal matching pursuit (OMP) algorithm [14]. In our implementation, we use the LARS algorithm [13] to find the solution of (10). We refer to our proposed method as ProbaEst, and the others in (9) and (10) as OMP and LARS, respectively.

Our proposed estimation method is limited to a dictionary with three atoms ($K = 3$). For this reason, the method is a priori best suited for low complexity images. To explore its performance, we have constructed a test benchmark on synthetic images in section III-B. We also discuss the results obtained on binary and natural images in sections III-C and III-D.

A. Parameter setting

In all tests, the size of an image patch is set to 3×3 and the overlap between two adjacent patches is 2 pixels. In cases of binary and natural images, the Kodak PhotoCD Dataset (shown in Fig. 4) is used as standard images. For the stage of estimation of $p(\boldsymbol{\alpha})$ as described in section II-A, we use $N = 500000$ patches randomly extracted from standard images to form the database of patches.

In our empirical work, each axis in the $\boldsymbol{\alpha}$-space is divided into $2^5 = 32$ intervals. Moreover, the noisy images are generated from the corresponding noise-free versions by adding Gaussian noise with different levels $\sigma = 10, 20, 30$. In regard to binary tests, all images are converted into binary ones using a threshold method. To evaluate the objective quality of denoising of vertical structure and natural images, two widely used metrics peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) are employed. For binary images, we adopt two metrics: the Dice ratio and the Distance-Reciprocal Distortion Measure (DRDM [18]) which exploits mutual relations of pixels within image to measure the distortion between two images. An essential notice is that the small value of DRDM indicates that two binary images are close.

Fig. 4: Some of standard images in Kodak PhotoCD Dataset (http://r0k.us/graphics/kodak/)

B. Denoising of vertical structure images

We first evaluate the denoising performance of our proposed method ProbaEst with two sparse models OPM and LARS on simple vertical structure images. We generate 4 vertical structure test images made of constant-gray-value stripes placed on a dark background as show in Fig. 5(b)-(e), where both the width and values of stripes are chosen randomly. We also need an adaptive standard image with similar vertical structures to exploit the redundancy of informations between images. Therefore, we create an image with stripes of gray levels gradually increasing from 1 to 255 as shown on Fig. 5(a) and use it as standard image for all comparing methods.
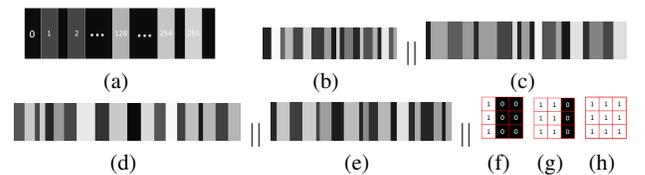

Fig. 5: (a) Vertical structure standard images with gradually increasing values of stripes. (b)-(e) Vertical structure test images with random width and values of stripes. (f)-(h) The patch-form of the three elements (atoms) of the dictionary $\mathbf{D}$

In our experiments, we observe that the results of denoising are highly dependent on the choice of dictionary $\mathbf{D}$. In regard to our task, where all images are composed of vertical stripes, a dictionary with vertical structural atoms can be more adapted to the image structure. Therefore, we select a dictionary with the vertical structure

Image		$\sigma = 10$			$\sigma = 20$			$\sigma = 30$		
		OMP	LARS	ProbaEst	OMP	LARS	ProbaEst	OMP	LARS	ProbaEst
PSNR	test1	35.96	**37.77**	35.73	31.23	31.95	**32.47**	28.16	28.47	**29.58**
	test2	36.06	**38.17**	35.70	31.23	32.14	**32.56**	28.19	28.87	**29.67**
	test3	35.95	**38.26**	35.63	31.38	32.39	**32.52**	28.42	28.86	**29.86**
	test4	35.99	**38.10**	35.69	31.36	32.03	**32.48**	28.32	28.61	**29.90**
	Average	35.99	**38.07**	35.68	31.30	32.13	**32.51**	28.27	28.70	**29.75**
SSIM	test1	0.887	0.951	**0.963**	0.773	0.849	**0.881**	0.670	0.737	**0.785**
	test2	0.876	0.947	**0.958**	0.740	0.827	**0.864**	0.630	0.711	**0.760**
	test3	0.868	0.943	**0.954**	0.737	0.827	**0.861**	0.626	0.706	**0.757**
	test4	0.884	0.950	**0.960**	0.764	0.844	**0.876**	0.657	0.729	**0.779**
	Average	0.879	0.948	**0.959**	0.735	0.837	**0.870**	0.645	0.721	**0.770**

TABLE I: The PSNR and SSIM measure of denoising of vertical stripes images in Fig. 5 with different methods.

as exposed in Fig. 5(f)-(h). Tables I provides the results of denoising, where the best values of PSNR and SSIM are in bold red numbers. We can observe that for small noise corruption ($\sigma = 10$), the LARS algorithm (10) yields better values of PSNR, but with heavier noise levels, the proposed method produces higher performance, with the improvements are $0.4 \sim 1.5$ dB of PSNR and $0.03 \sim 0.12$ of SSIM. For visual assessment, Fig. 6 demonstrates the subjective results of image test4 with noise levels $\sigma = 30$. It is evident that the OMP and LARS are likely to generate much more artifacts in the denoised images than the proposed method. More precisely, referring to the results from Fig. 6(c)-(e), with the chosen adaptive vertical dictionary, the sparsity methods tend to strengthen the vertical artifacts, while the proposed method ProbaEst is more robust against that kind of reconstruction distortion.
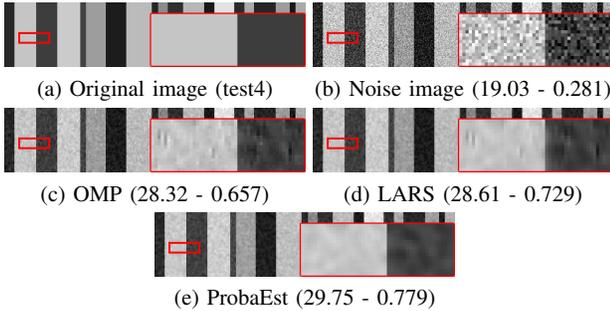

(a) Original image (test4)   (b) Noise image (19.03 - 0.281)

(c) OMP (28.32 - 0.657)   (d) LARS (28.61 - 0.729)

(e) ProbaEst (29.75 - 0.779)

Fig. 6: Results of denoising on image test4 with $\sigma = 30$. (a)-(e) are the original image, noisy image, result of OMP, LARS and ProbaEst, respectively, with the zoom-in of region-of-interest (R.O.I).

C. Denoising of binary images

In this part, we will not only compare the performance of proposed method with the sparse patch-based algorithms OMP and LARS, but also with others denoising methods developed for binary images, called Iterated Conditional Mode (ICM) [19] and Graph Cuts [20]. The ICM method is based on the maximization of local conditional probabilities, while the latter optimizes the flow through an associated network constructed from image. Please refer to [19] and [20] for more details.

In the experiment, all images are converted into binary form using a threshold method as a pre-processing step. We used all the images (24) in the Kodak PhotoCD Dataset (Fig. 4) as standard images and some other widely used test images (shown in Fig. 7) for the evaluation. The dictionary is chosen as the same as Fig. 5(f)-(h) for the ProbaEst, OMP and LARS methods.

Table II presents the results of denoising of binary images in both terms of Dice Ratio and DRDM, which demonstrates the efficiency of proposed method for binary image denoising. For visual evaluation, Fig. 8 shows the results of denoising on binary image of monarch. As can be observed, the ProbaEst is more efficient in noise reduction than the sparsity methods (OMP, LARS) and the ICM. More particularly, the OMP and LARS generate images with vertical artifacts, while the Graph Cuts tends to produce an over-expansion foreground image.

Visually, the proposed method achieves very competitive denoising performance, where its result is more similar to the original image.

(a)   (b)   (c)   (d)   (e)

Fig. 7: Some of widely used test images. (a) House. (b) Lena. (c) Monarch. (d) Peppers. (e) Baby.

(a) Original im. (b) Noise image   (c) OMP   (d) LARS
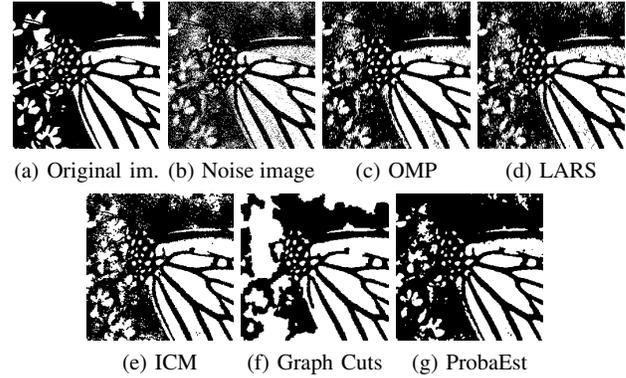
(e) ICM   (f) Graph Cuts   (g) ProbaEst

Fig. 8: Results of denoising on binary image of monarch with $\sigma = 30$.

D. Denoising of natural images

In the rest of our work, we compare the performance of the proposed method with two sparsity-based algorithms OMP and LARS for natural images. As in section III-C, we use all 24 images in the Kodak dataset (Fig. 4) as standard images. Moreover, we carry out the experiments by adopting the same dictionary for all methods, which is directly trained from each noisy image, using the K-SVD method [10]. In reality, with a simple dictionary of three atoms, the denoising results will not be good enough for visual quality requirement. But we have to emphasize again that the objective of our research is to demonstrate that using the estimated probability distribution as the image prior is more efficient than a sparsity model for noise removal. As illustrated in table III, the proposed method yields better values of PSNR and SSIM than the others in most of cases of high noise levels ($\sigma = 20, 30$). In particular, Fig. 9 shows the denoising results in which the proposed method ProbaEst and the LARS are more efficient in noise reduction than the OMP. The results of LARS and ProbaEst are very similar. But if we carefully observe the background and some homogeneous regions (R.O.I regions) in the images, the ProbaEst produces less artifacts than the LARS.

IV. DISCUSSION

In this paper, we proposed a novel idea for patch-based denoising where the prior of patches distribution is learned from the external database. More precisely, we proposed a method for the estimation of probability distribution function by spatial division and demonstrated how to employ it in an optimization-based denoising process. The experimental results indicate that the approach improves considerably the results of denoising for some simple models of image when compared to the classical sparsity methods. The objective of our work is to make in evidence that the idea of embedding the probability distribution learned from external database is efficient and is a promising approach for image denoising.

However, a drawback of the proposed method is the implementation for high dimensional ($K > 3$) $\boldsymbol{\alpha}$-space due to the curse of dimensionality. In that case, using the same framework of spatial

Images		σ = 20					σ = 30					σ = 40				
		OMP	LARS	ICM	Gr.Cuts	ProbaEst	OMP	LARS	ICM	Gr.Cuts	ProbaEst	OMP	LARS	ICM	Gr.Cuts	ProbaEst
Dice	House	0.875	0.877	**0.902**	0.819	0.900	0.856	0.856	**0.883**	0.797	**0.883**	0.838	0.838	0.871	0.788	**0.875**
	Lena	0.908	0.917	0.940	0.916	**0.941**	0.883	0.884	0.921	0.889	**0.923**	0.847	0.848	**0.902**	0.871	0.900
	Monarch	0.923	0.930	0.930	0.876	**0.948**	0.894	0.894	0.894	0.808	**0.931**	0.863	0.863	0.861	0.759	**0.923**
	Peppers	0.941	0.948	0.955	0.946	**0.963**	0.918	0.918	0.930	0.919	**0.951**	0.888	0.888	0.907	0.885	**0.939**
	Baby	0.971	0.974	0.980	0.976	**0.980**	0.961	0.961	0.974	0.969	**0.976**	0.945	0.945	0.967	0.962	**0.970**
	Average	0.923	0.929	0.941	0.906	**0.946**	0.902	0.902	0.920	0.876	**0.932**	0.876	0.876	0.901	0.853	**0.921**
DRDM	House	10.56	10.00	6.44	19.90	**6.24**	12.73	12.72	8.91	23.90	**8.48**	14.93	14.96	10.32	25.21	**9.51**
	Lena	9.43	8.24	4.96	8.57	**4.61**	12.23	12.21	7.04	12.52	**6.57**	16.58	16.53	9.43	15.50	**9.29**
	Monarch	4.49	3.93	4.25	8.84	**2.23**	6.51	6.50	7.13	15.99	**3.26**	9.00	9.01	10.27	21.97	**3.85**
	Peppers	6.19	5.22	4.31	5.17	**2.90**	8.99	8.99	7.73	9.11	**4.16**	12.93	12.94	10.98	14.71	**5.69**
	Baby	4.08	3.44	1.90	2.43	**1.72**	5.77	5.78	2.77	3.53	**2.37**	9.05	9.07	4.17	5.04	**3.40**
	Average	6.95	5.96	4.37	8.98	**3.54**	9.24	9.24	6.71	13.01	**4.96**	12.49	12.50	9.03	16.48	**6.34**

TABLE II: Comparison of denoising methods for binary images using Dice ratio and DRDM measurement.

Images		σ = 10			σ = 20			σ = 30		
		OMP	LARS	ProbaEst	OMP	LARS	ProbaEst	OMP	LARS	ProbaEst
PSNR	Monarch	32.29	**32.38**	31.69	28.42	28.96	**29.23**	25.86	26.55	**27.07**
	House	33.17	**33.65**	33.17	29.22	30.21	**30.29**	26.63	27.84	**28.17**
	Lena	33.04	**33.50**	32.73	28.90	29.81	**29.90**	26.38	27.52	**27.78**
	Peppers	33.11	**33.69**	32.85	29.04	30.07	**30.14**	26.48	27.64	**27.90**
	Barbara	**31.63**	31.28	30.66	28.07	28.51	**28.58**	25.72	26.54	**26.69**
	C. Man	**31.22**	30.53	29.96	27.94	**28.17**	28.16	25.61	26.04	**26.43**
	Baby	33.12	**33.59**	33.40	29.00	30.01	**30.14**	26.50	27.78	**27.98**
	Average	32.51	**32.66**	32.06	28.65	29.39	**29.49**	26.15	27.13	**27.57**
SSIM	Monarch	0.906	**0.937**	0.927	0.807	0.860	**0.867**	0.716	0.779	**0.800**
	House	0.833	**0.856**	0.854	0.689	0.751	**0.760**	0.575	0.653	**0.677**
	Lena	0.873	**0.910**	0.898	0.744	0.811	**0.818**	0.637	0.714	**0.734**
	Peppers	0.876	**0.917**	0.906	0.752	0.825	**0.832**	0.642	0.726	**0.748**
	Barbara	0.876	**0.880**	0.868	0.764	0.797	**0.799**	0.666	0.717	**0.728**
	C. Man	0.847	**0.879**	0.869	0.709	0.771	**0.777**	0.595	0.661	**0.685**
	Baby	0.876	**0.905**	0.902	0.747	0.804	**0.812**	0.637	0.709	**0.725**
	Average	0.869	**0.897**	0.889	0.744	0.802	**0.810**	0.638	0.708	**0.728**

TABLE III: Denoising performance on natural images with respect to PSNR and SSIM.

discretization is not efficient and inapplicable. Nevertheless, to improve the quality of denoising for real natural images, we have to select a dictionary with higher number of atoms ($K \gg 3$). To deal with this problem, we can use the Gaussian mixture model (GMM) to learn the probability distribution function $p(\boldsymbol{\alpha})$ as a mixture of a finite number of Gaussian distributions. We will expand proposed method with GMM in future work.

REFERENCES

[1] C. Tomasi and R. Manduchi. "Bilateral filtering for gray and color images". In: *International Conference on Computer Vision*. Jan. 1998, pp. 839–846.

[2] F. Dibos and G. Koepfler. "Global Total Variation Minimization". In: *SIAM Journal on Numerical Analysis* 37.2 (1999), pp. 646–664.

[3] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. "Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering". In: *IEEE Transactions on Image Processing* 16.8 (Aug. 2007), pp. 2080–2095.

[4] D. H. Trinh, M. Luong, F. Dibos, J. M. Rocchisani, C. D. Pham, N. Linh-Trung, and T. Q. Nguyen. "An effective example-based learning method for denoising of medical images corrupted by heavy Gaussian noise and poisson noise". In: *2014 IEEE International Conference on Image Processing (ICIP)*. Oct. 2014, pp. 823–827.

[5] M. Elad and M. Aharon. "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries". In: *IEEE Transactions on Image Processing* 15.12 (Dec. 2006), pp. 3736–3745.

[6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. "Non-local sparse models for image restoration". In: *International Conference on Computer Vision (ICCV)*. Sept. 2009, pp. 2272–2279.

[7] V. Jain and S. Seung. "Natural Image Denoising with Convolutional Networks". In: *Advances in Neural Information Processing Systems 21*. 2008, pp. 769–776.

[8] J. Xie, L. Xu, and E. Chen. "Image Denoising and Inpainting with Deep Neural Networks". In: *Proceedings of the 25th International Conference on Neural Information Processing Systems*. NIPS'12. 2012, pp. 341–349.

[9] A. Buades, B. Coll, and J. M. Morel. "A non-local algorithm for image denoising". In: *CVPR*. Vol. 2. June 2005, 60–65 vol. 2.

[10] M. Aharon, M. Elad, and A. Bruckstein. "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation". In: *IEEE Trans. on Signal Processing* 54.11 (Nov. 2006), pp. 4311–4322.

[11] J. Mairal, M. Elad, and G. Sapiro. "Sparse Representation for Color Image Restoration". In: *IEEE Transactions on Image Processing* 17.1 (Jan. 2008), pp. 53–69.

[12] W. Dong, X. Li, L. Zhang, and G. Shi. "Sparsity-based image denoising via dictionary learning and structural clustering". In: *CVPR*. June 2011, pp. 457–464.

[13] H. Zou and T. Hastie. "Regularization and variable selection via the Elastic Net". In: *Journal of the Royal Statistical Society, Series B* 67 (2005), pp. 301–320.

[14] J. A. Tropp and A. C. Gilbert. "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit". In: *IEEE Transactions on Information Theory* 53.12 (Dec. 2007), pp. 4655–4666.

[15] R. Chartrand. "Exact Reconstruction of Sparse Signals via Nonconvex Minimization". In: *IEEE Signal Processing Letters* 14.10 (Oct. 2007), pp. 707–710.

[16] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. "Least angle regression". In: *Annals of Statistics* 32 (2004), pp. 407–499.

[17] I. Naseem, R. Togneri, and M. Bennamoun. "Linear Regression for Face Recognition". In: *IEEE Trans. on Patt. Ana. and Mac. Intel.* (Nov. 2010), pp. 2106–2112.

[18] H. Lu, A. C. Kot, and Y. Q. Shi. "Distance-reciprocal distortion measure for binary document images". In: *IEEE Signal Processing Letters* 11.2 (Feb. 2004), pp. 228–231.

[19] J. Besag. "On the statistical analysis of dirty pictures". In: *Journal of the Royal Statistical Society: Series B* 48.3 (1986), pp. 48–259.

[20] V. Kolmogorov and R. Zabin. "What energy functions can be minimized via graph cuts?" In: *IEEE Trans. on Patt. Ana. and Mac. Intel.* 26 (Feb. 2004), pp. 147–159.
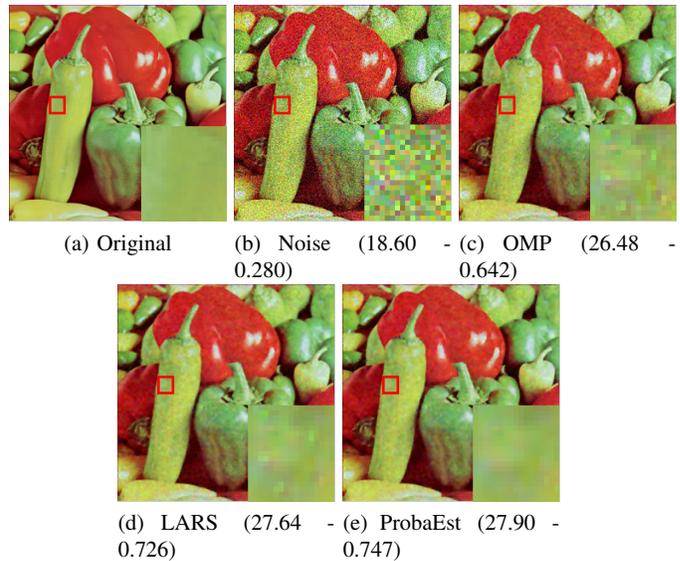
(a) Original　　(b) Noise (18.60 - 0.280)　　(c) OMP (26.48 - 0.642)

(d) LARS (27.64 - 0.726)　　(e) ProbaEst (27.90 - 0.747)

Fig. 9: Results of denoising on image of peppers with $\sigma = 30$.