AN EFFECTIVE IMAGE DEBLURRING SCHEME USING CLUSTER BASED SPARSE REPRESENTATION

Yediga Ravi Sankaraiah, Sourirajan Varadarajan

Department of Electronics and Communication Engineering, Jawaharlal Nehru Technological University, Kakinada, India, e-mail: ravinov12@gmail.com

Abstract

Sparse based representation is being used extensively for image restoration. The dictionary learning through patch extraction is central to the sparse based schemes. In the process of dictionary learning, a large number of patches will be extracted from high quality images and dictionary will be formed. Hence, over-complete dictionaries will be built. To overcome the complexity associated with over-complete dictionaries many schemes were proposed. Of them, the adaptive sparse domain is the popular one. Many variations of adaptive sparse domain schemes were proposed. Selection of obvious patches is common to all. In all these schemes, individual patches will be considered as the basic entity and will be used. This is the reason for the complexity involved in sparse representation. In this paper, to avoid the complexity, the patches are grouped according to the similarity among the patches. In addition to reduce the complexity the proposed cluster based scheme considers the self-similarity of the patches involved. Hence better performance with less complexity is possible with the proposed schemes. In the process of testing, in addition to uniform blur and Gaussian blur, a combination of the two blurs is also considered.

Keywords: Cluster of patches, Image restoration, Mixed blur, Self-similarity, Sparse coding

Introduction

The purpose of image restoration is to recover the original undistorted image from a degraded image. This mechanism is formulated as:

$$Y = Hx + n$$

Here, '$\mathbf{y}$' is the observed image, '$\mathbf{x}$' is the original image and '$\mathbf{n}$' is the additive noise. Depending on the nature of '$\mathbf{H}$' the degrading phenomenon will have different effect on the original image. In the context of image blurring, '$\mathbf{H}$' is an image blur operator [1][2]. Ill-posed characteristic of image deblurring necessitates the prior knowledge and the deblurring has been modelled as follows.

$$\arg\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \lambda\Psi(x) \tag{1}$$

Here $\frac{1}{2}\|\mathbf{H}x - y\|_2^2$ is the l_2 fidelity term, Ψ is the regularization term and λ is regularization parameter. Based on the above model, many optimization schemes were proposed in the

literature [3]-[10]. Image priors are crucial which affects the overall performance of image restoration algorithms. Hence, the task of defining the regularization terms corresponding to image priors need to be done carefully. Traditional schemes like total variation [11][12], Mumford Shah model [13] are built with an assumption that images are smooth in a local area except at the edges. The regularization terms in these models are defined by considering these structures and correspondingly these models can represent the smooth regions and sharp edges. But these schemes, which are basically designed based on smooth regions and sharp edges, cannot hold and retain the fine details and lose image details.

Later on, the sparse coding based schemes came into existence and shown superior performance in many image processing tasks. In sparse coding, a set high quality images will be chosen from which a set of patches will be collected. These patches will form a structure called a dictionary. The degraded image will be broken down into small exclusive patches. The patches in the dictionary will be used in different ways to restore patches of the degraded image. The reconstructed patches will be grouped to form the restored image. The patch based spare coding schemes requires high range of computational capacity as the number of patches to be processed is too high.

The finite set of patches extracted from either the test signal or the patches collected from high quality images in the process of building the dictionary. Another factor to consider in patch based sparse coding is the similarity among the patches and its consequence [14]. The patches will have similarity with each other in some extent and if these patches mapped somehow with each other, the comparison and correlation operations can be localized based on the cluster of similar patches. Hence, the overall computational complexity will be reduced. Correspondingly, in this paper, a combination of patches called cluster is used as the fundamental unit of sparsity. A cluster dictionary will be built which is self-adaptive and concise. This lifts the barrier of complexity in sparse coding. The cluster based sparse coding can yield the non-local self-similarity automatically and produce high quality restored image.

Conventional Sparse Representation

Suppose that $\{\Phi_k\}$, $k=1, 2\ldots K$, is a set of K orthonormal sub-dictionaries. Let x be an image vector, and $x_i=R_ix$, $i = 1, 2\ldots N$, be the ith patch (size: root (n) $\times$ root (n)) vector of x, where R_i is a matrix extracting patch x_i from x. For patch x_i, suppose that a sub-dictionary, Φ_k is selected for it. Then, x_i can be approximated as $\hat{x}_i = \Phi_{k_i}\alpha_i$, $\|\alpha_i\|_1 \leq T$, via sparse coding. The whole image x can be reconstructed by averaging all the reconstructed patches $\hat{x}_i$, which can be mathematically written as [15].

$$\hat{x} = \left(\sum_{i=1}^{N} R_i^T R_i \right)^{-1} \sum_{i=1}^{N} R_i^T \Phi_{k_i}\alpha_i \qquad (2)$$

In Equation (2), the matrix to be inverted is a diagonal matrix, and hence the calculation of Equation (2) can be done in a pixel-by-pixel manner [15]. Obviously, the image patches can be overlapped to better suppress noise [15] [16] and block artifacts. For the convenience of expression, we define the following operator "o":

$$\hat{x} = \Phi \, o \, \alpha \equiv \left(\sum_{i=1}^{N} R_i^T R_i \right)^{-1} \sum_{i=1}^{N} R_i^T \Phi_{k_i} \alpha_i \tag{3}$$

where Φ is the concatenation of all sub-dictionaries {Φ$_k$} and α is the concatenation of all α$_i$. The notation 'o' is used for convenience of expression and relates Φ and α as per the Equation

(2). Let $y = DHx + v$ be the observed degraded image; the goal is to recover the original image x from y. Here, D and H are degrading operators and in the case of deblurring, D is identity and H is blurring operator. With adaptive sparse domain selection (ASDS) and the definition in Equation (3), the image restoration (IR) problem can be formulated as follows:

$$\hat{\alpha} = \begin{array}{c} \arg\min \\ \alpha \end{array} \left\{ \left\| y - DH\Phi \, o \, \alpha \right\|_2^2 + \lambda \left\| \alpha \right\|_1 \right\} \tag{4}$$

The minimization problem present in the above equations is non-convex and computationally highly complex. In the literatures many approximation schemes are appointed to tackle the minimization problem including SVD [17], K-SVD [18], and MOD [19]. These schemes produced high quality restored images. Though the approximation schemes are involved, the computational complexity is still high. This is inevitable because the dictionary is built by considering all the patches. Apart from the computational complexity, the sparse coding is not exploiting the similarity among the patches. Hence, a concept that somehow relates the patches and groups by considering a parameter is obvious to exploit the self-similarity between the patches. This also alleviates the complexity associated with the sparse coding and dictionary learning.

Cluster based Sparse Representation

Building a Cluster

As mentioned in the previous section, the patch-based sparse coding scheme is built by considering the patch as the basic unit. The drawbacks of the patch-based sparse coding scheme can be removed by considering a cluster of patches as the reference for sparsity. This exploits the self-similarity among the similar patches and then the dictionary learning takes place for each cluster separately. This reduces the complexity involved with regular dictionary learning. The original image **x** will be divided into patches of $\sqrt{n_p} \times \sqrt{n_p}$. These patches may be overlapping patches. These patches are denoted by x_k, k = 1, 2, …., n.
The patches are extracted from the image using the following relation.

$$x_k = P_k(x)$$

Here P_k is the process of extracting kth patch from **x**. If patch extraction is overlapping then large number of patches will be generated giving raise to complexity. Figure 1 shows non-overlapping patch extraction as the best case. Still, here there exists correlation among the

patches extracted. The patches bounded with same color has high correlation. The size of the input image is 512×512. Figure 1 shows the non-overlapping patches of size 16×16. Hence a total of 32×32 = 1,024 patches are extracted.
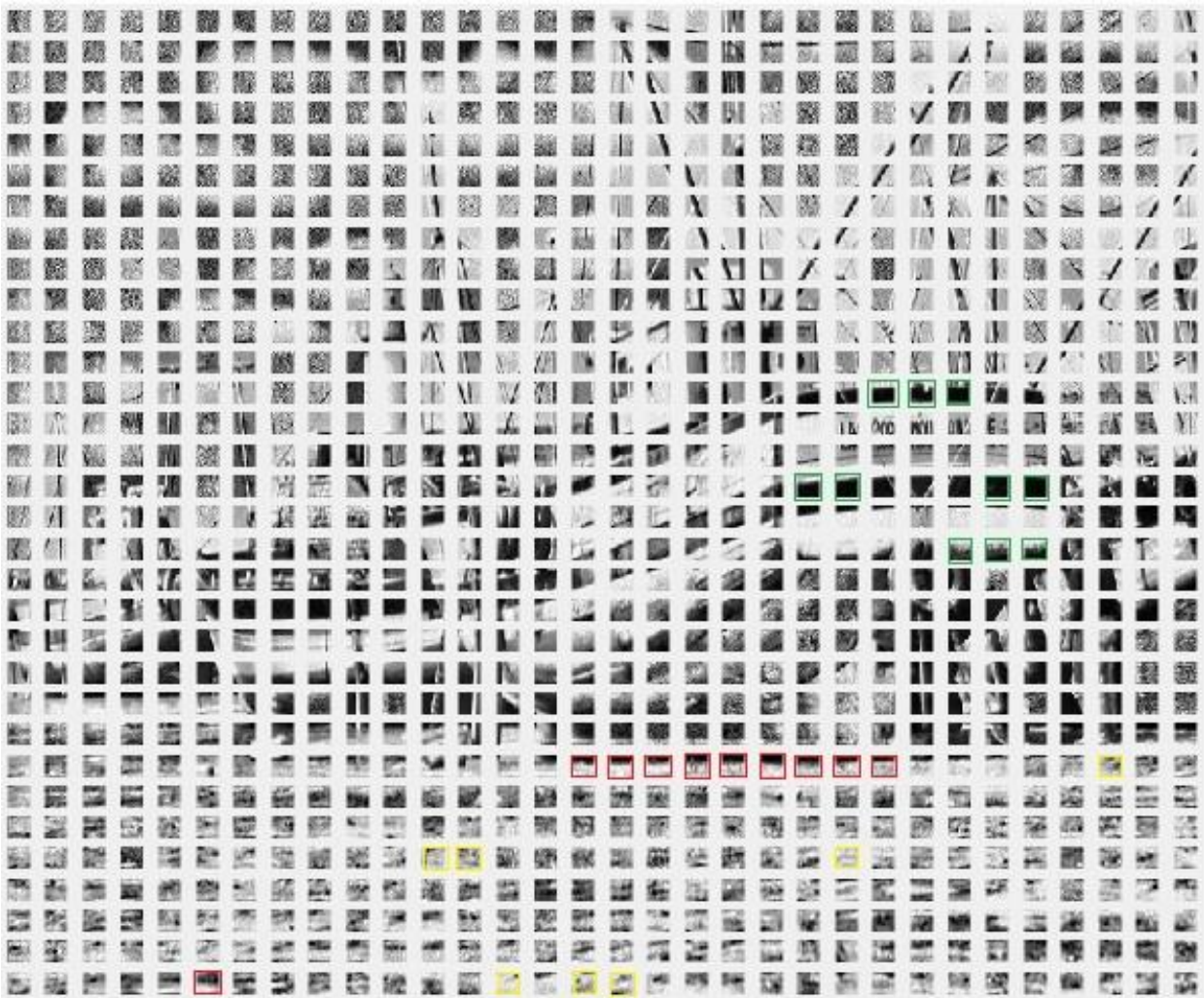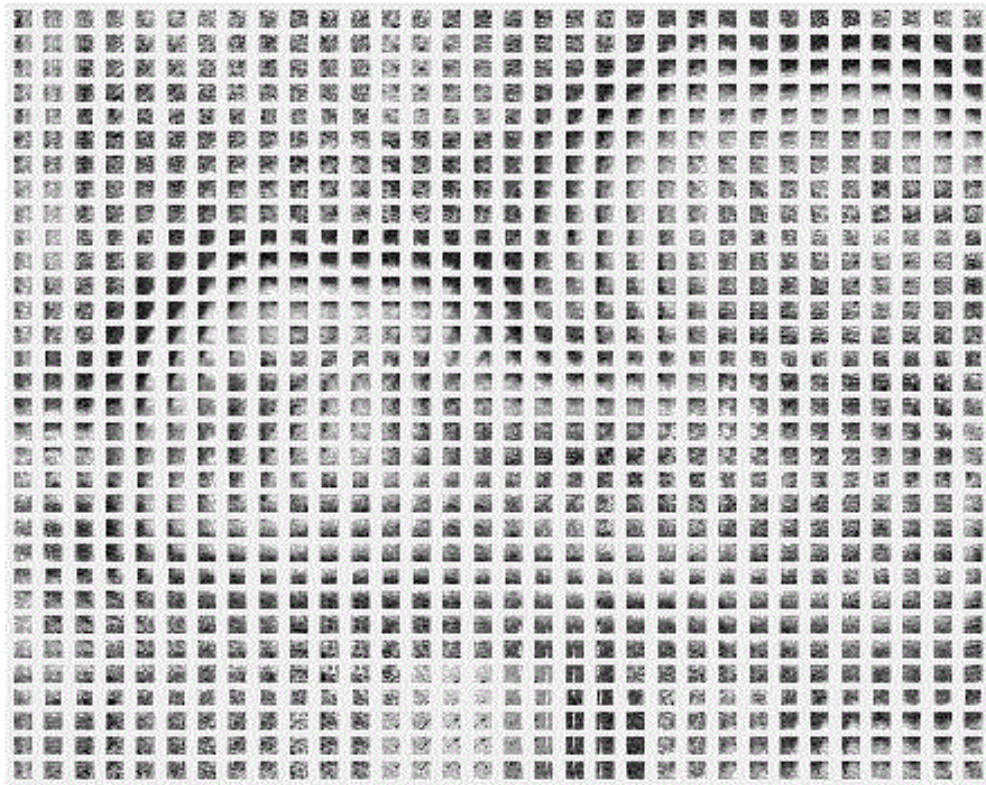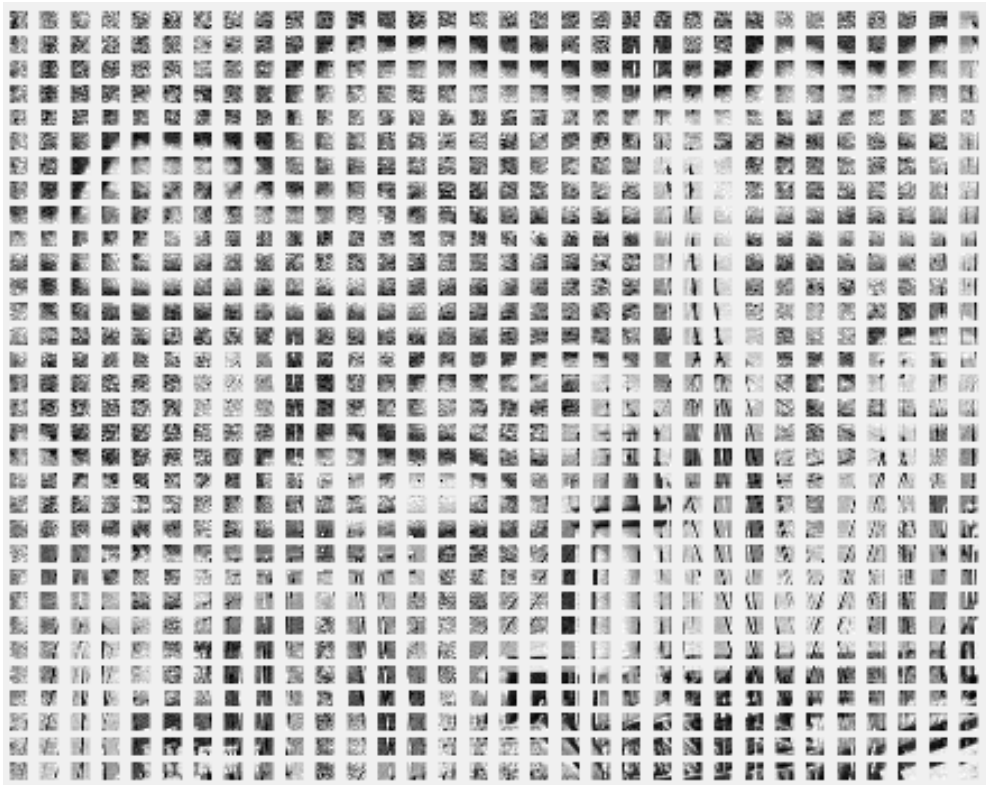

Figure 1. Non-overlapping Patch extraction and clustering

Hence, the patches with certain amount of correlation may be grouped to form the cluster. In practice there will be more number of patches in each cluster and based on the level of the similarity considered, there will be lesser number of clusters. This also depends on how these patches are overlapped. Figure 2 shows the overlapped patches with different levels of overlapping. The first part of Figure 2 shows the overlapping patches with same size that of Figure 1, but with only 4×4 pixels unique and the next part with 8×8 pixels unique. The similar patches are grouped into a cluster. The similarity is defined in terms of Euclidean distance. An additional rule was introduced to consider only $n_p/2$ number of patches in the cluster. Indirectly, an operator may be defined which extracts clusters from the image as shown below.

$$xc_k = C(x_k)$$

(a)

(b)

Figure 2. Overlapping patch extraction with only (a) 4×4 pixels unique (b) 8×8 pixels unique

This is analogous to the process of extracting patches from the image. By averaging all the clusters, recovery of input image **x** is always possible. The process of building cluster of patches is given in Figure 3.
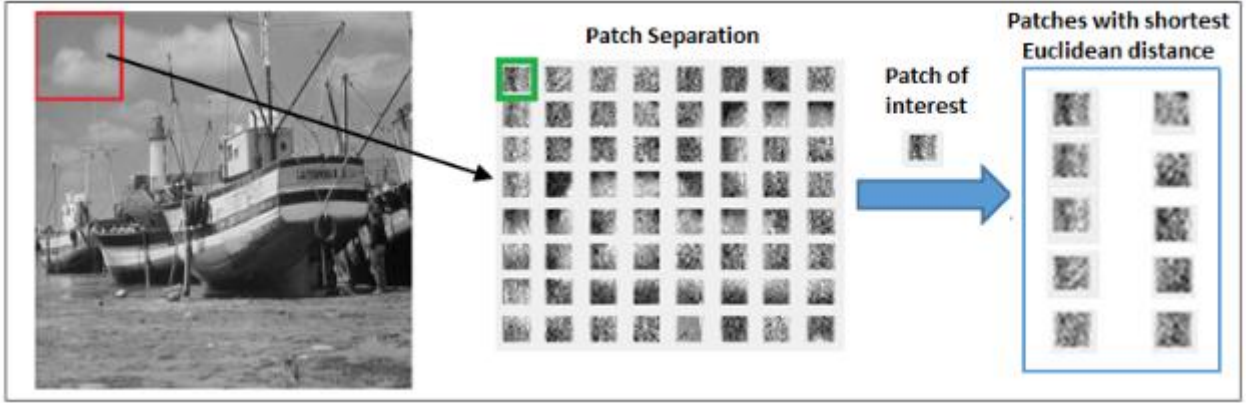

Figure 3. Process of building cluster of patches

Consider the 'ship' image of Figure 3. An 8×8 part of the image considered and patches are linearly extracted and shown in the same figure. First, for a patch of interest, patch surrounded by a green box, similar patches are selected by calculating the Euclidean distance. As a rule, out of these similar patches, only $n_p/2$ number of patches with least Euclidean distance are chosen. This kind of clustering will be continued till the last patch by excluding the patches which are already clustered in one of the clusters. **xc$_k$** contain the patches with similar structures and is referred as cluster of **x$_k$**. Now another matrix **X$_k$** is defined to hold all these similar patches of exclusive set of reference patches. The clusters are written as columns of the proposed matrix with index refereeing to reference patch. The order of the matrix **X$_k$** will be **Nc** X $n_p/2$. **Nc** is the number of clusters. The process of extracting the matrix **X$_k$** is attributed to, say, **O$_k$**. Therefore **X$_k$ = O$_k$(x)**. Now, the original image **x** may be represented in terms of **X$_k$** and **O$_k$** by the following expression.

$$x = \sum_{k=1}^{n} O_k^{T}(X_k) \bigg/ \sum_{k=1}^{n} O_k^{T}(1) \qquad (5)$$

Impact of Clustering on Sparse Coding

Local sparsity and non-local similarity over the entire image domain are incorporated simultaneously using a model defined in terms of clusters. This model presumes that every cluster of a test image can be represented using the objects of a dictionary **D$_c$**. Each object of the dictionary **D$_c$** are of same size as that of a cluster **x$_k$**. The sparse coding of each cluster **x$_k$** over the dictionary **D$_c$** is to find a vector **α$_c$** used in the process of restoration along with the dictionary. In the domain of clustering, the entire image **x** may be sparsely represented by the vector **α$_c$** as follows.

$$x = D_C \circ \alpha_C = \sum_{k=1}^{n} O_k^{T}(D_C \alpha_C) \bigg/ \sum_{k=1}^{n} O_k^{T}(1) \qquad (6)$$

When the degraded model is considered, the regularization model can be written as follows.

$$\tilde{\alpha}_C = \frac{\arg\min}{\alpha_C} \frac{1}{2}\left\|HD_C \circ \alpha_C - y\right\|_2^2 + \lambda\left\|\alpha_C\right\|_0 \tag{7}$$

The reconstructed image can be expressed as follows.

$$\hat{x} = D_C \circ \hat{\alpha}_C$$

The sparse based sparse coding now can be considered as a special case of cluster based sparse coding where the cluster contains only one patch. The cluster of patches grouped from blurry image will be recovered from the dictionary of clusters generated from high quality image dictionary. The cluster will have all the patches with nearest equilibrium distance. One or few of the patches among the cluster members will have highest degree of similarity, hence a better approximation and recovery of lost image information is possible. The whole difference between the patch-based schemes and the proposed scheme can be understood by comparing the eqns. (4) and (7). The main differences are based on the reference entity, i.e., patch vs cluster and the dictionary built based on the entities. There are mainly three improvements of the proposed scheme. They are

i. The proposed scheme requires less computational complexity, hence can be applied to large scale optimization problems.
ii. The clusters which were build based the similarity among the patches helps to exploit the self-similarity and hence this similarity among the test images will be exploited in both dictionary learning and sparse coding.

The proposed system presents a concept of self-adaptive dictionary in contrast to that of a dictionary for all the patches in earlier schemes. Hence the proposed system will work more effectively for a specific test patch or for the whole test image. When compared with the scheme proposed by Daniel Yair Weiss [20] and Vardan Papyan and Michael Elad [21], the proposed scheme is advantageous as these schemes propose building a general dictionary rather than a self-adaptive dictionary. The steps used in the proposed scheme are given below.

Algorithm

1: Input: The observed image and degraded operator
2: Build clusters of pixels
    a. Perform patch separation from high quality images.
    b. Perform clustering of these patches based on Euclidean distance.
3: Use the standard sparse representation for image restoration.
    a. Repeat
    b. For each cluster
    c. Construct dictionary
    d. Reconstruct alpha
    e. End
    f. End
    g. Update the dictionary D using dictionaries corresponding to each cluster
    h. Update alpha using alpha corresponding to each cluster
    i. The restored image $\hat{x} = D_C \ o \ \hat{\alpha}_C$

Mixed Blur

Image blurring causes significant degradation of image quality. In many cases, though the cause of blurring is alone, the effect of blurring is too high and creates huge overlapping of image information. The overlapping of image is not as a whole, but at molecular level. The size of pixels which are dispersed to other pixels has inverse relation to the overall degradation. In the literature a number of deblurring schemes are proposed by considering mostly either uniform or Gaussian blur. But, in practice the cause of blurring will be more than a single source. In true sense, the source is undefined and will be a combination of many sources. The amount of blurring is generally defined in terms of blurring parameter of specific type blur. In the case of uniform blur, the kernel size is used as the blurring parameter and in Gaussian blur, the standard deviation σ is used as the blurring parameter. The effect of uniform and Gaussian blurs for different values of blur parameter is shown in Figures 4 and 5 respectively.

The range of kernels in uniform blur is taken from 6×6 to 13×13. The degradation raises as the size of the kernel increases. The PSNR with 6×6 kernel is 27.52dB and SSIM is 0.80. When a 13×13 kernel is employed, the PSNR is 24.09dB and SSIM 0.68. The standard deviation in the case of Gaussian blur is varied from 2.5 to 6 in steps of 0.5. The PSNR and SSIM with 2.5 standard deviation are 26.80dB and 0.78 and with 6 are 22.95dB and 0.66. It can be understood that though the way of blur is different, the amount of blur on the image is almost similar with equivalent quantities of blur parameters. But, as mentioned earlier, the real concern is about the case of mixed blur. Let's consider the case of uniform blur followed by Gaussian blur. This case with different values of kernel size and standard deviation is presented in Figure 6. The values of PSNR and SSIM suggests that the effect of mixed blur is too significant. The blur created by the sequence of blurring operators modifies the images rigorously, but very locally.

(a)　　　　　　　　　(b)

(c)　　　　　　　　　(d)
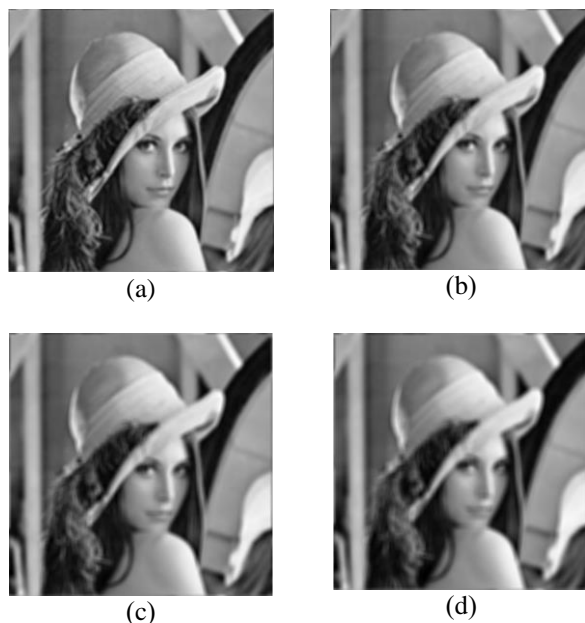
Figure 4. Blurred images with Uniform blurs of different dimensions with corresponding PSNR and SSIM values (a) 7×7, 27.17dB and 0.78 (b) 9×9, 25.84dB and 0.74 (c) 11×11, 24.87dB and 0.7 (d) 13×13, 24.09dB and 0.68

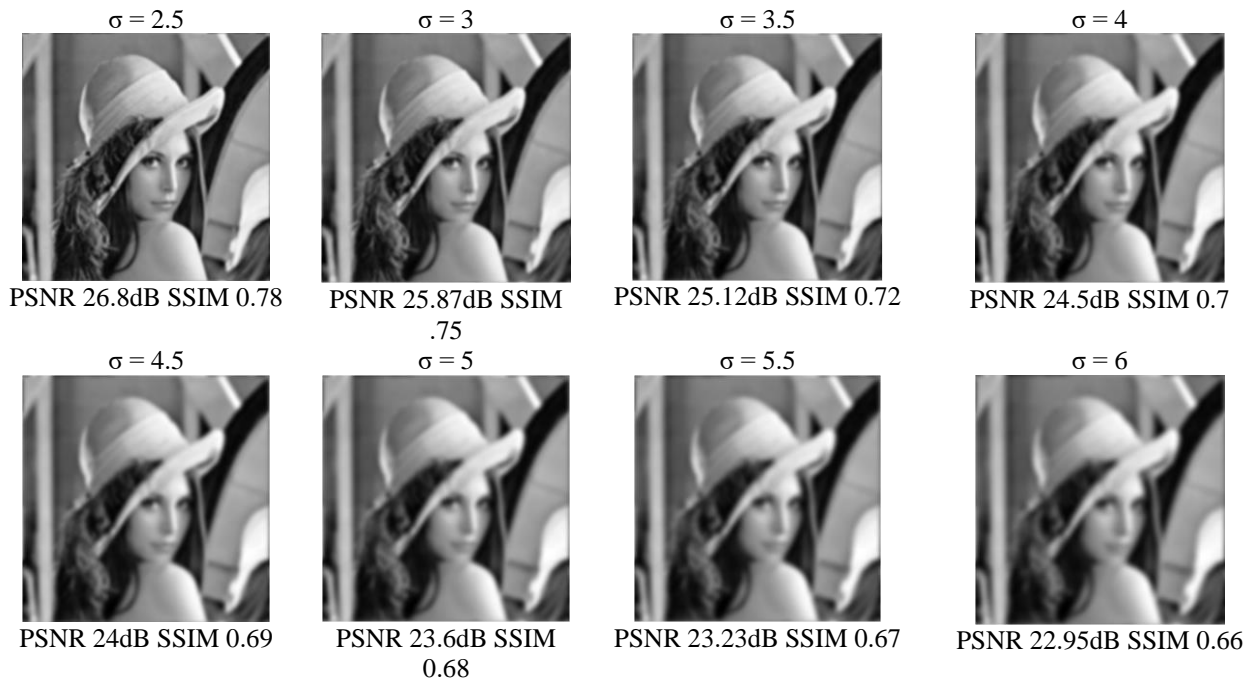Figure 5. Blurred images with Gaussian blurs of different standard deviations with corresponding PSNR and SSIM values
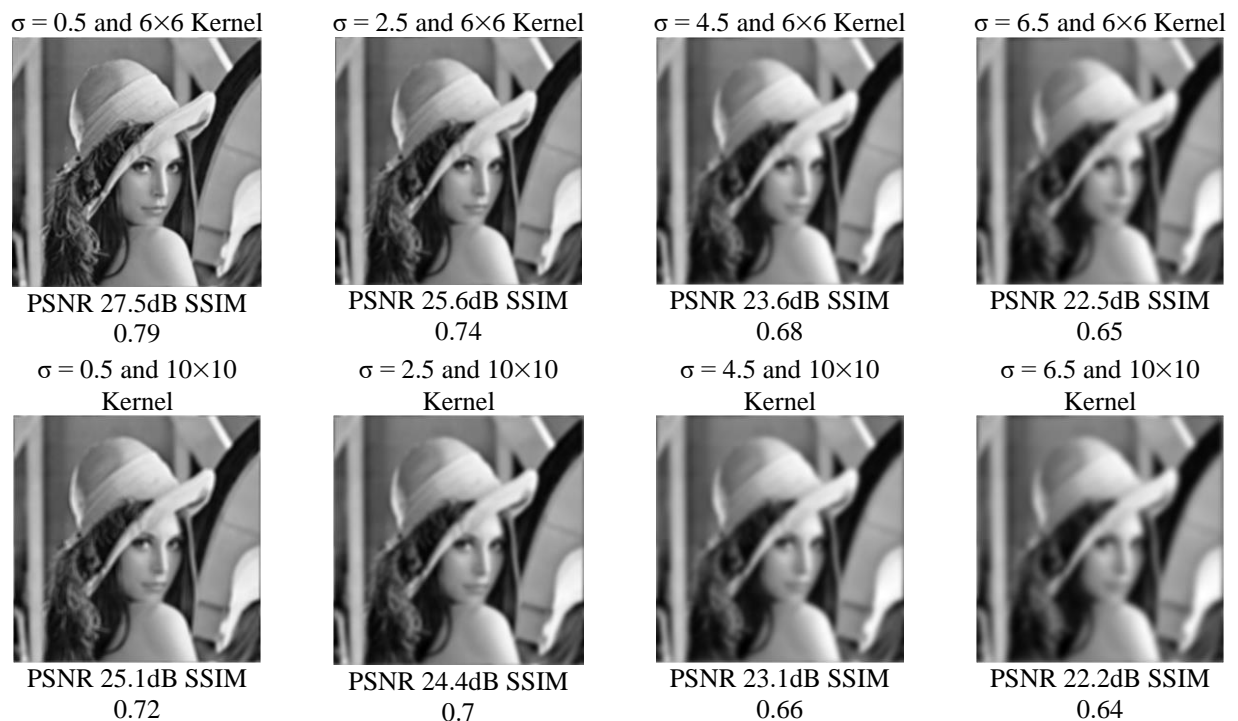

Figure 6. Blurred images with mixed blur of different blur parameters with corresponding PSNR and SSIM values

Simulation Results

In this section, the deblurring results using cluster based sparse coding is presented. The uniform, Gaussian and mixed blurs are considered. The uniform blurring was applied using kernels of size 2×2, 3×3, 4×4, 5×5, 6×6, 7×7, 8×8 and 9×9. The standard deviation of Gaussian blur is varied from 0.5 to 4 in steps of 0.5. Hence at total of 16 cases are derived. In the mixed blur case, 8 combinations are simulated. The simulation results with Uniform and Gaussian blurs are given in Tables 1 and 2.

Table 1. Deblurring Results with Uniform Blur Using Cluster-based Sparse Coding

Kernel Order		3×3	5×5	7×7	9×9
PSNR	**Blurred**	33.20	29.24	27.17	25.84
	Deblurred	40.85	36.70	34.22	32.59
SSIM	**Blurred**	0.91	0.84	0.78	0.74
	Deblurred	0.97	0.93	0.91	0.88

Table 2. Deblurring Results with Gaussian Blur Using Cluster-based Sparse Coding

Standard Deviation		0.5	1	1.5	2	2.5	3	3.5	4
PSNR	**Blurred**	41.87	32.78	29.81	28.03	29.86	25.87	25.12	24.50
	Deblurred	58.19	39.09	35.21	32.81	30.98	29.69	28.72	27.97
SSIM	**Blurred**	0.99	0.91	0.86	0.81	0.85	0.75	0.72	0.70
	Deblurred	1.00	0.96	0.92	0.89	0.86	0.83	0.81	0.79

The simulation results in above tables show the performance of the proposed schemes. The results when compared with the state-of-art schemes, the proposed deblurring method is found to outperform all the existing schemes. The performance analysis is given in Tables 3, 4, 5 and 6. In addition to the individual uniform and Gaussian blur, in this paper a mixture of these two blurs is also considered. This is the first of its kind in the literature. The essence of considering mixed blur is already discussed in previous section. Table 7 shows the performance of proposed method on mixed blur. From the Table 7, it can be understood that the local overlapping of image pixels is too complex. Deblurring of blurred image with mixed blur is feasible up to specific level of blur quantity.

Table 3. Performance Analysis Using PSNR Values with 9×9 Uniform Blur Kernel

	LENA	HOUSE	CAMERAMAN	PEPPER	BOAT	BARBARA
TVMM [1]	28.6	32.4	26.8	28.7	29.3	26.0
L0_ABS [22]	28.7	33.01	27.1	28.6	29.7	26.4
BM3D [23]	29.7	34.4	28.5	29.6	31.2	27.9
NCSR [24]	30.1	34.3	**28.6**	29.6	31.08	**28.1**
PROPOSED	**32.59**	**35.4**	28.15	**30.06**	**31.2**	27.5

Table 4. Performance Analysis Using SSIM Values with 9×9 Uniform Blur Kernel

	LENA	HOUSE	CAMERAMAN	PEPPER	BOAT	BARBARA
TVMM [1]	0.89	0.91	0.86	0.89	0.89	0.85
L0_ABS [22]	0.9	0.92	0.87	0.9	0.9	0.86
BM3D [23]	0.91	0.93	0.9	0.92	0.93	0.9
NCSR [24]	0.92	0.94	0.9	0.92	0.929	0.91
PROPOSED	0.88	**0.944**	**0.914**	**0.933**	0.917	**0.916**

Table 5. Performance Analysis Using PSNR Values with Gaussian Blur with σ =1.6

	LENA	HOUSE	CAMERAMAN	PEPPER	BOAT	BARBARA
TVMM [1]	30.7	33.0	27.0	29.2	30.4	24.8
L0_ABS [22]	30.6	33.0	27.3	28.8	30.5	24.7
BM3D [23]	31.4	**34.0**	28.1	29.9	31.6	27.1
NCSR [24]	31.2	33.6	**28.3**	30.1	31.4	27.9
PROPOSED	**34.96**	33.86	27.5	**30.72**	**31.87**	**28.4**

Table 6. Performance Analysis using PSNR Values with Gaussian Blur with σ =1.6

	LENA	HOUSE	CAMERAMAN	PEPPER	BOAT	BARBARA
TVMM [1]	0.92	0.91	0.89	0.92	0.89	0.85
L0_ABS [22]	0.93	0.92	0.89	0.93	0.9	0.86
BM3D [23]	0.94	0.93	0.91	0.93	**0.93**	0.9
NCSR [24]	0.93	0.93	0.9	0.93	0.92	0.91
PROPOSED	**0.947**	**0.942**	**0.926**	**0.934**	0.926	**0.93**

Table 7. Deblurring Results with Mixed Blur using Cluster Based Sparse Coding

Blur Parameters		*3×3, 1*	*5×5, 2*	*7×7, 3*	*9×9, 4*
PSNR	**Blurred**	30.69	26.84	24.92	23.67
	Deblurred	32.60	28.00	25.86	24.50
SSIM	**Blurred**	0.88	0.78	0.72	0.68
	Deblurred	0.91	0.81	0.74	0.70

Conclusions

Sparse coding scheme is well-known successor of dictionary based learning. In comparison with straight dictionary learning and frequency or wavelet domain, the sparse coding is advantageous and possess many similarities with these. Translated and dilated version of basis function are utilized to represent the signal quantity in wavelet domain and properties of high quality images are used in dictionary learning. The optimum choice and concise design of dictionary lead to sparse coding. In sparse coding, the basic unit is patch and architecture is built based on patch. The result is high computational complexity and also the self-similarity among the patches are not being exploited. Hence, a cluster is defined with a specific similarity measure between patches and the cluster is now used as the basic component by which the complexity is reduced

and at the same time the self-similarity is exploited. In addition to the Uniform blur and Gaussian blur, a mixed blur is simulated and used in testing the proposed scheme. The simulation results proven the superiority of the proposed scheme in comparison with the state-of-art methods. Also, the difficulty of deblurring the blurred image by mixed blur is presented.

References

[1] J.M. Bioucas-Dias, M.A.T. Figueiredo and J.P. Oliveira, "Total variation-based image deconvolution: A majorization-minimization approach," Paper presented at *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, IEEE, Toulouse, France, 2006. doi: 10.1109/ICASSP.2006.1660479

[2] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Transactions on Image Processing*, Vol. 19, pp. 2345–2356, 2010. doi: 10.1109/TIP.2010.2047910

[3] J.K. Sunkara, K. Purnima, S. Muchakala, Y. Ravisankariah, "Super-resolution based image reconstruction," *International Journal of Computer Science and Technology*, Vol. 2, pp. 272-281, 2011.

[4] J. Bioucas-Dias, and M. Figueiredo, "A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Transactions on Image Processing*, Vol. 16, pp. 2992–3004, 2007. doi: 10.1109/TIP.2007.909319

[5] J.K. Sunkara, "Selection of wavelet basis function for image compression – a review," *Electronic Letters on Computer Vision and Image Analysis,* Vol. 18, pp. 1-20, 2019. doi: https://doi.org/10.5565/rev/elcvia.1094

[6] Y. Zhang, H. Pu, and J. Lian, "Quick response barcode deblurring via l_0 regularisation based sparse optimisation," *IET Image Processing*, Vol. 13, pp. 1254-1258, 2019. doi: 10.1049/iet-ipr.2018.5738

[7] S. Liu, Y. Feng, S. Zhang, H. Song, and S. Chen, "L_0 sparse regularization-based image blind deblurring approach for solid waste image restoration," *IEEE Transactions on Industrial Electronics*, Vol. 66, pp. 9837-9845, 2019. doi: 10.1109/TIE.2019.2892681

[8] T. An, D. Choi, S. Cho, K. Hong, and S. Lee, "Blind deblurring using coupled convolutional sparse coding regularisation for noisy-blurry images," *Electronics Letters*, Vol. 54, pp. 874-876, 2018. doi: 10.1049/el.2018.0901

[9] M. Tofighi, Y. Li, and V. Monga, "Blind image deblurring using row–column sparse representations," *IEEE Signal Processing Letters*, Vol. 25, pp. 273-277, 2018. doi: 10.1109/LSP.2017.2782570

[10] J. Zhang, D. Zhao, and W. Gao, "Group–based sparse representation for image restoration," *IEEE Transactions on Image Processing*, Vol. 23, pp. 3336-3351, 2013. doi: 10.1109/TIP.2014.2323127

[11] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, Vol. 60, pp. 259–268, 1992. doi: 10.1016/0167-2789(92)90242-F

[12] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, Vol. 20, pp. 89–97, 2004. doi: 10.1023/B:JMIV.0000011325.36760.1e

[13] D. Mumford, and J. Shah, "Optimal approximation by piecewise smooth functions and associated variational problems," Communications on Pure and Applied Mathematics, Vol. 42, pp. 577–685, 1989. doi: 10.1002/cpa.3160420503

[14] J.K. Sunkara, S. Eswarawaka, K. Darisi, S. Dara, P.K. Dasari, and P. Dara, "Intensity non-uniformity correction for image segmentation", *IOSR Journal of VLSI and Signal Processing*, Vol. 1, pp. 49-57, 2013. doi: 10.9790/4200-0154957

[15] J. Zhang, D. Zhao, C. Zhao, R. Xiong, S. Ma, and W. Gao, "Compressed sensing recovery via collaborative sparsity," Paper presented at *2012 Data Compression Conference*, IEEE, Snowbird, Utah, United States, 2012. doi: 10.1109/DCC.2012.71.

[16] D. Geman, and G. Reynolds, "Constrained restoration and the recovery of discontinuities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, pp. 367–383, 1992. doi: 10.1109/34.120331

[17] G.A.E. Satish Kumar, and J.K. Sunkara, "Multiresolution SVD based image fusion," *IOSR Journal of VLSI and Signal Processing*, Vol. 7, pp. 20-27, 2017. doi: 10.9790/4200-0701012027

[18] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, Vol. 54, pp. 4311–4322, 2006. doi: 10.1109/TSP.2006.881199

[19] K. Engan, S.O. Aase, and J.H. Hakon-Husoy, "Method of optimal directions for frame design," Paper presented at *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings,* IEEE, Phoenix, Arizona, United States, 1999, doi: 10.1109/ICASSP.1999.760624.

[20] D. Zoran, and Y. Weiss, "From learning models of natural image patches to whole image restoration," Paper presented at *2011 International Conference on Computer Vision*, IEEE, Barcelona, Spain, 2011, doi: 10.1109/ICCV.2011.6126278.

[21] V. Papyan, and M. Elad, "Multi-scale patch-based image restoration," *IEEE Transactions on Image Processing*, Vol. 25, pp. 249-261, 2016. doi: 10.1109/TIP.2015.2499698

[22] J. Portilla, "Image restoration through l_0 analysis-based sparse optimization in tight frames," Paper presented at *2009 16th IEEE International Conference on Image Processing*, IEEE, Cairo, Egypt, 2009, doi: 10.1109/ICIP.2009.5413975

[23] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Transactions on Image Processing*, Vol. 21, pp. 1715–1728, 2012. doi: 10.1109/TIP.2011.2176954

[24] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally centralized sparse representation for image restoration," *IEEE Transactions on Image Processing*, Vol. 22, pp. 1620–1630, 2013. doi: 10.1109/TIP.2012.2235847