Advances in Signal Processing 2(2): 55-74, 2014
DOI: 10.13189/asp.2014.020204 http://www.hrpub.org

Dictionary Learning for Scalable Sparse Image Representation with Applications

Bojana Begovic*, **Vladimir Stankovic**, **Lina Stankovic**

Department of Electrical and Electronic Engineering, University of Strathclyde, Glasgow, G1 1XW, UK

*Corresponding Author: bojana@begovic.co.uk

Abstract This paper introduces a novel design for the dictionary learning algorithm, intended for scalable sparse representation of high motion video sequences and natural images. The proposed algorithm is built upon the foundation of the K-SVD framework originally designed to learn non-scalable dictionaries for natural images. Proposed design is mainly motivated by the main perception characteristic of the Human Visual System (HVS) mechanism. Specifically, its core structure relies on the exploitation of the high-frequency image components and contrast variations in order to achieve visual scene objects identification at all scalable levels. Proposed design is implemented by introducing a semi-random Morphological Component Analysis (MCA) based initialization of the K-SVD dictionary and the regularization of its atom's update mechanism. In general, dictionary learning for sparse representations leads to state-of-the-art image restoration results for several different problems in the field of image processing. In experimental section we show that these are equally achievable by accommodating all dictionary elements to tailor the scalable data representation and reconstruction, hence modeling data that admit sparse representation in a novel manner. Performed simulations include scalable sparse recovery for representation of static and dynamic data changing over time (e.g., video) together with application to denoising and compressive sensing.

Keywords Scalable Video Representation, Sparse Coding, Regularization, Contrast Variation, Denoising, Compressive Sensing

1 Introduction

Over the past couple of decades, image processing applications have undergone significant improvements. A recent critical factor in this growth is the sparse coding paradigm introduced firstly by [1], based on the assumption that signals (e.g., natural images) admit a sparse decomposition over a learned representational basis i.e., dictionary. This so-called sparseland model [2, 3, 4] has led to numerous state-of-the-art algorithms for several image processing problems [3] specifically in the context of dictionary $\mathbf{D} \in R^{n \times K}$ learning for any image signal class. Commonly, the representation of image $\mathbf{Y} \in R^{b \times b}$, is broken down into a set of N extracted patches $\{\mathbf{y}_i\}_{i=1}^{N} \in R^n$ which are in turn sparsely represented. Typically (but not necessarily) it is assumed that dictionary $\mathbf{D}$ is overcomplete i.e., the number of its basis vectors (atoms) is greater than the original signal's dimension ($K > n$). Given one of the pursuit algorithms e.g., [5, 6, 7, 8, 9] and a dictionary $\mathbf{D}$, one can estimate matrix $\mathbf{X}$ containing sparse approximations $\{\mathbf{x}_i\}_{i=1}^{N} \in R^K$ for each $\mathbf{y}_i$. Hence, a set of weighted linear combinations of few atoms in $\mathbf{D}$ satisfactorily approximates each patch $\mathbf{y}_i \in \mathbf{Y}$ with image denoted as $\widehat{\mathbf{Y}} \approx \mathbf{DX}$. The applications of dictionary learning [10, 11] include areas such as classification [12, 13], efficient face recognition [14], inpainiting [15], denoising [16, 17], super-resolution [18, 19], Morphological Component Analysis (MCA) [20, 21] and those designed for sparse color image processing [22, 23].

In this paper, we provide a detailed presentation of the *scalable* and sparse modeling dictionary learning framework which basic outline was originally presented in [24, 25]. Our focus is placed on designing a procedure for learning a dictionary capable of adapting both to a specific dataset and providing its effective *scalable* reconstruction. Given that current work on *scalable* data recovery is only based on the the conventional predefined dictionaries such as DCT [26] we find that it is important to offer an alternative one in a form of an adaptive dictionary sparse representation. In addition, to the best of our knowledge existing literature just provides the dictionary learning algorithms such as K-SVD [3, 10] that only assume fine resolution as the representational output. This is not sufficient nor tailored to provide the progressive image recovery over its trained sparse representation. Thus, we take and extend the classical form of the K-SVD where the proposed learning scheme differs from the regular one [3, 10] in the sense that:

- Dictionary is initialized in a controlled, semi-random manner using image's MCA properties [20, 21];
- A novel learning design is introduced as a regular-

ization of a dictionary training procedure prior to Singular Value Decomposition (SVD) [27, 28];

- It allows more flexibility for adapting the *scalable* representation to specific data by removing constraints originally imposed on redundant atoms (i.e., mutually coherent or rarely used) in [5];

- A firm spatial frequency distribution is enforced over dictionary atoms as a built in feature;

- OMP pursuing algorithm is replaced via a simple matrix inversion for sparse coefficient estimation given that proposed dictionary is complete.

Specifically, proposed implementation is carried out by introducing regularization of the K-SVD atoms update stage aiming for *scalable* sparse image reconstruction which would improve gradually as we take more and more entries per each coefficient $\mathbf{x_i} \in \mathbf{X}$ to restore $\{\mathbf{y}_i\}_{i=1}^{N}$ patches. In particular, we emphasize the penalization of the low and high spatial frequency components of the image patches and dictionary, imposing the learning model that mimics the main HVS [29, 30] system properties. That is, incorporating HVS's high sensitivity to contrast light information and to the patterns orientation [31, 32] at high spatial frequencies (originally shown via contrast sensitivity function map [33, 34]). The HVS features are proven to be essential modeling elements for many image processing algorithms [35] and image quality assessment tools [36, 37].

Furthermore, in modern video broadcasting networks, an image or a video source is transmitted to numerous clients with various receiver characteristics. These consumers differ primarily in accessible: (*i*) channel capacity; (*ii*) display resolutions; (*ii*) computing resources. The interesting question is how to support and deliver a controlled quality of the displayed data of a wide range of applications that differ in the users equipment heterogeneity, communication channels and QoS demands? It would be appealing somehow for a video or image signal to be processed in a such manner that would enable its optimal usability by all diverse clients. For example, the limited frequency space shared by mobile video streaming users would be effectively exploited by a generic *scalable* i.e., progressive data reconstruction such as proposed here. In other words, progressive reconstruction framework would be applied on the source signal prior to its transmission producing its scaled representation form. Once delivered at the client side, depending on its technical specifications, signal would be restored at different quality levels. Thus, signal's generic scalability is desirable in many applications since it will be able to support heterogeneity in users equipment, QoS demands, and communication channels.

This paper makes following contributions:

1. It tackles for the first time the problem of creating a dictionary tailored to *scalable* image restoration, offering a novel model for data that admits sparse representation;

2. Enforces specific spatial frequency distribution as a built-in feature over trained dictionary;

3. As a solution to the *scalable* image restoration problem, this paper provides an extension and upgrade of the K-SVD dictionary learning concept from non-scalable to *scalable* adaptive image reconstruction by introducing semi-random dictionary initialization based on the MCA activity norm [3] and by regularizing the learning process of dictionary elements overall promoting the HVS perceptual mechanism features;

4. The potential of the proposed method is shown for the adaptive *scalable* denoising and CS;

Evaluation of the *scalable* recovery is done using high-motion test video sequences and several natural images, successfully attaining progressive frame-to-frame and image *scalable* restoration. Experimental results confirm that the proposed *scalable* scheme outperforms significantly conventional K-SVD at different *scalable* image recovery levels. Specifically, in terms of application, our focus is placed on applying our proposed *scalable* scheme to denoising [15] and compressive sensing (CS) [5, 38, 39, 40, 41]. Mainly, in relation to denoising, we tackle processing and computational demands of the [16] given that experimental results in [42] suggest that objective quality improvement of current state-of-the-art image denoising schemes cannot be improved by more than 0.1 [dB]. This conclusion is a result of comparison between the lowest error rates given a simple statistical measure derived from a huge image patch distribution [42] and the empirical errors of state-of-the-art denoising algorithms. Lastly, given the CS results in [43, 44] we test the performance of proposed *scalable* dictionary learning method in one of the CS sampling scenarios.

2 Problem statement and proposed approach

Adhering closely to the notation used in [10], this section provides the detailed description of the proposed dictionary learning scheme for *scalable* image reconstruction. We build on the regular K-SVD algorithm [10] by altering its initialization and atom's update step. In general, we are given a set of N signals i.e., overlapping image patches of size $\sqrt{n} \times \sqrt{n}$ vectorized as $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N] \in R^{n \times N}$. The classical configuration of the K-SVD algorithm aims to approximate representation of these signals in a sparse way as weighted linear combinations of a few dictionary elements i.e., the columns of matrix $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_K] \in R^{n \times K}$.

Note however that this conventional approach is not capable of providing *scalable* image reconstruction that would be based on progressive recovery of each image patch $\mathbf{y}_i$. For instance, one can form $\{a \,|1 \leq a \leq \lfloor K/m \rfloor = s\}$ number of recovery layers for each patch leading to reconstructed image denoted as L_a. In general, m can vary and take on different values i.e., $1 < m \leq K$ resulting in a number of *scalable* recovery layers having m as the scaling parameter. This leads to a progressive image restoration provided as a sequence of L_a image layers each generated as a combination of the truncated versions of sparse representation $\mathbf{X}$ and dictionary $\mathbf{D}$. At the beginning of the progressive

recovery, the base layer L_1 is rebuilt out of the first m sparse coefficients entries per patch. That is, for each patch i we take $[\mathbf{x}_i[1]\ \mathbf{x}_i[2] ... \mathbf{x}_i[m]]$ while remaining entries are set to zero $\mathbf{x}_i = 0$ for $m < i \leq K$. These are combined together with the first m corresponding atoms i.e. $[\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_m]$ leading to a compression rate of m/n. Afterwords, while reconstructing each subsequent layer L_a $(a > 1)$ additional m coefficients are added. That is, $[\mathbf{x}_i[1]\ \mathbf{x}_i[2] ... \mathbf{x}_i[am]]$ ($\mathbf{x}_i = 0$ for $am < i \leq K$) and $[\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{am}]$ producing compression ratio of $(ma)/n$.

The way in which we achieve effective sparse adaptive *scalable* image reconstruction is by introducing:

1. MCA based semi-random initialization of the dictionary at the very beginning of the training procedure;
2. A regularization scheme over the second K-SVD iterative stage i.e., the dictionary atom's update which enforces significance of the high frequency components during the regularized atom's update.

The following terms will be used in the remainder of this paper:

- $\mathbf{Y} \in R^{n \times N}$ - matrix with N overlapping image patches $\mathbf{y}_i \in R^n$;
- $\mathbf{D}_{sc} \in R^{n \times K}$ - proposed *scalable* dictionary;
- $\mathbf{D} \in R^{n \times K}$ - conventional non-scalable dictionary obtained using standard K-SVD [1][5];
- K - the number of dictionary atoms in $\mathbf{D}_{sc}$ or $\mathbf{D}$;
- $\mathbf{X} \in R^{K \times N}$ - sparse matrix with sparse coefficient vectors $\mathbf{x}_i \in R^K$.

2.1 Dictionary initialization

In classical K-SVD, prior to any of the two training stages, dictionary $\mathbf{D}$ is initialized with K randomly extracted image training patches $\mathbf{y}_i$ [10] from the set of total N. In contrast, prior to initialization we divide the N training patches in two classes C_1 and C_2 each containing smooth and texture image content, respectively. As a classification criteria we use the activity measure similar to TV norm originally used within the K-SVD MCA setup [3] and defined as:

$$Activity(\mathbf{y}_i) = \sum_{j=2}^{n}\sum_{k=1}^{n} |\mathbf{y}_i[j,k] - \mathbf{y}_i[j-1,k]| + \sum_{j=1}^{n}\sum_{k=2}^{n} |\mathbf{y}_i[j,k] - \mathbf{y}_i[j,k-1]|. \quad (1)$$

Subsequently, *Activity* is normalized in a way which sets its range from 0 to 1. These values are reflecting the degree of "smoothness" and "textureness" in each image patch [3]. The higher the *Activity* the higher the level of the texture will be within the patch. Thus, the classification is performed via simple tresholding using heuristically set value A. This value is taken from [3] where it is shown that it provides the best possible classification performance for smooth and texture element separation. Specifically, classifying parameter A indicates classification of patches into two classes C_1 or C_2. That is:

- $\mathbf{y}_i \in C_1$ for $Activity(\mathbf{y}_i) \leq A$;
- $\mathbf{y}_i \in C_2$ for $Activity(\mathbf{y}_i) > A$.

Thereafter, the first $K/2$ atoms of the proposed dictionary $\mathbf{D}_{sc}$ are initialized randomly choosing $K/2$ image patches from the C_1 class, that is, the smooth group. The rest of the $K/2$ atoms are randomly picked from the C_2 class i.e., the texture group. In this way, we enforce semi random initialization which directly controls and effects the starting dictionary structure by placing low frequencies (smooth image areas) within its first half of $\mathbf{d}_j$ atoms $(1 \leq j \leq K/2)$ and high ones (texture image areas) within the last half $(K/2 < j \leq K)$. In return, this sets a foundation for further design which is organized around applying proposed regularization scheme and subsequently tuning dictionary learning to the main HVS perception characteristic.

2.2 Sparse coding

The first of the two iterative dictionary learning stages (sparse coding) is posed as a constraint optimization problem defined in [10] as:

$$\min_{\mathbf{X}} \left\{ \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \right\} \quad s.t. \ \forall\, i \quad \|\mathbf{x}_i\|_0 \leq T_0 \quad (2)$$

given the current estimation of the dictionary $\mathbf{D}$ which is kept fixed during this process. The expression $\|\mathbf{x}_i\|_0$ accounts for the number of non-zero elements in each vector $\mathbf{x}_i$ by the means of the l_0 pseudo norm where T_0 impose the top limit on $\|\mathbf{x}_i\|_0$. Signal $\mathbf{y}_i \in \mathbf{Y}$ $(i = 1, \ldots, N)$, extracted from the original image, is mapped into its sparse representation $\mathbf{x}_i$ commonly via OMP [3, 6]. However, given that we train a complete dictionary as proposed, OMP is not needed for the sparse coding step. That is, the exact solution for the *scalable* dictionary is attained via simple matrix inversion as $\mathbf{x}_i = \mathbf{D}'_{sc}\mathbf{y}_i$ by maintaining up to T_0 largest non-zero coefficient entries. Each of K entries $\mathbf{x}_i[j]$ corresponds to one of the atoms $\mathbf{d}_j \in \mathbf{D}_{sc}$ $(j=1,\ldots,K)$ where nonzero entry $\mathbf{x}_i[j] \neq 0$ means that particular atom $\mathbf{d}_j$ participates in the sparse representation of the signal $\mathbf{y}_i$ [10]. We relax the sparsity constraint, permitting T_0 to take a higher value than in [10] where the relation $T_0 << n$ is still maintained. This allows the *scalable* signal recovery to be established while introducing a T_0 value on an empirical basis that still promotes the sparsity prior of the signal.

2.3 Regularized dictionary update stage

Once stage described in 2.2 is completed, we move to the atom $\mathbf{d}_j$ update stage. Usually, the new basis atom is estimated by processing a current representational residual $\mathbf{E}_j$ [10] constructed to account for the error of all N patches when the atom $\mathbf{d}_j$ is removed:

$$\left\| \left(\mathbf{Y} - \sum_{k \neq j} \mathbf{d}_k \mathbf{x}_T^k \right) - \mathbf{d}_j \mathbf{x}_T^j \right\|_F^2 = \left\| \mathbf{E}_j - \mathbf{d}_j \mathbf{x}_T^j \right\|_F^2 \quad (3)$$

$\mathbf{x}_T^k$ represents coefficients from the k_{th} row in $\mathbf{X}$ (3) [9] where $\mathbf{x}_T^k[i] \neq 0$ denotes that the sparse approximation

for the patch $\mathbf{y}_i$ includes atom $\mathbf{d}_k$. Prior to update each atom $\mathbf{d}_j$ is set to zero while the remaining basis elements are kept fixed. Subsequently, error matrix $\mathbf{E}_j$ is subject to shrinking [10] which results in reduction of her compositional structure to one which only contains error columns of the patches that use $\mathbf{d}_j$. Update of the pair $\left[\mathbf{d}_j, \mathbf{x}_T^j\right]$ is obtained via SVD decomposition of such interchanged matrix. Shrinking is necessary in order to preserve the sparsity constraint. That is, the new vector $\mathbf{x}_T^j$ will keep the sparsity property and it is not going to be fully filled with non-zero entries after subject to SVD. It is performed by identifying all patches that at the moment of the update use atom $\mathbf{d}_j$ as $\omega_j = \left\{ i | 1 \leq i \leq N, \mathbf{x}_T^j(i) \neq 0 \right\}$ followed with the formation of the matrix Ω_j size $N \times |\omega_j|$. This matrix contains ones only at the $(\omega(i), i)$ positions. Remaining entries are zeros. Multiplying (3) with Ω_j achieves necessary shrinking.

However, as already stated, this is insufficient to generate dictionary tailored for the *scalable* image restoration. That is why we decide to redefine the structure of the $\mathbf{E}_j$ (3) by introducing the regularization scheme. The proposed procedure is mainly motivated by the HVS functional mechanism properties. Specifically, human eyes tend to pay more attention to the edges of an object given the high firing rate of the visual cortex neurons at the moment of perception, primarily identifying objects by their bounding shapes [31, 32]. Thus, in order to facilitate effective *scalable* recovery, we find that it is necessary to ensure that the main object shapes are identified from the beginning of the image reconstruction. This effect would resemble to some extent to the object recognition procedures [35] which mostly rely on exploitation of image's high frequency information for this task. Hence, spatial higher frequencies should be more relevant to *scalable* dictionary learning. We address this by appropriately favoring the significant changes associated with the edges in the image patches (i.e., the texture) during the $\mathbf{D}_{sc}$ training. This is carried out by dividing the current sparse approximations of all patches in $\mathbf{E}_j$ (3) as:

$$\mathbf{E}_j^R = \left(\mathbf{Y} - v_0 \sum_{k=1}^{\frac{K}{2}} \mathbf{d}_k \mathbf{x}_T^k - v_1 \sum_{k=\frac{K}{2}+1}^{K} \mathbf{d}_k \mathbf{x}_T^k \right) \Omega_j, k \neq j. \tag{4}$$

Superscript R stands for regularized and pair $[v_0, v_1]$ denotes regularization terms. Each batch corresponds to the low and high frequency components of the training image patches:

- First batch (with v_0) contains only atoms initialized from the C_1 smooth class;
- Second batch (with v_1) contains only atoms initialized from the C_2 texture class.

This separation is plausible due to semi-random initialization described in Sec. 2.1. Proposed design is a result of testing various weight pairs $[v_0, v_1]$ under the constraint $v_0 + v_1 = 1$ in order to avoid degeneracy of the learned representation. Outcomes show that carefully introduced regularization over the smooth and texture image components is able to yield the appropriate dictionary for the *scalable* data representation (Sec. 3).

Further, by introducing:

- $\mathbf{Y}_j = \mathbf{Y}\Omega_j$
- $\mathbf{D}_{sc}^{\mathbf{low}}\mathbf{X}_j^{\mathbf{low}} = \left(\sum_{k=1}^{K/2} \mathbf{d}_k \mathbf{x}_T^k \right) \Omega_j$;
- $\mathbf{D}_{sc}^{\mathbf{high}}\mathbf{X}_j^{\mathbf{high}} = \left(\sum_{k=K/2+1}^{K} \mathbf{d}_k \mathbf{x}_T^k \right) \Omega_j$;

the proposed regularized error matrix (4) can be rearranged as:

$$\mathbf{E}_j^R = \left(\mathbf{Y}_j - v_0 \mathbf{D}_{sc}^{\mathbf{low}} \mathbf{X}_j^{\mathbf{low}} - v_1 \mathbf{D}_{sc}^{\mathbf{high}} \mathbf{X}_j^{\mathbf{high}} \right) \tag{5}$$

where $\mathbf{Y}_j$ represents a subset of the image patches $\mathbf{y}_i$ from $\mathbf{Y}$ with indices given in ω_j. Superscripts **low** and **high** denote smooth and texture frequency content associated with the weight pair $[v_0, v_1]$ which regularizes contribution of their residual components to the $\mathbf{E}_j^R$. Consequentially, this separation controls the type of the information used for the $\mathbf{d}_j$ atom's update.

For all atoms $j = 1, ..., K$, the proposed update stage is summarized as:

1. STEP 1 - Allocate corresponding image patches which current sparse approximation given as a linear superposition $\mathbf{D}_{sc}\mathbf{x}_i$ includes atom $\mathbf{d}_j$ as it is done in [10], map them accordingly with ω_j and denote as a subset of patches $\mathbf{Y}_j$, that is a subset of sparse coefficients $\mathbf{X}_j$;
2. STEP 2 - In contrast to [10], split each current sparse approximation element $\mathbf{x}_i \in \mathbf{X}_j$ $(i \in \omega_j)$, associated with atom $\mathbf{d}_j$, in two parts using binary vectors $\mathbf{T}^{\mathbf{low}}, \mathbf{T}^{\mathbf{high}} \in R^K$ as:
   - $\mathbf{x}_i^{\mathbf{low}} = \mathbf{x}_i \mathbf{T}^{\mathbf{low}}$ and $\mathbf{x}_i^{\mathbf{high}} = \mathbf{x}_i \mathbf{T}^{\mathbf{high}}$;

   where $\mathbf{T}^{\mathbf{low}}, \mathbf{T}^{\mathbf{high}} \in R^K$ are binary vectors that cancel any $\mathbf{x}_i[l]$ element for $l > \frac{K}{2}$ (associated with the dictionary elements initialized with class C_1) and $l < \frac{K}{2}$ (associated with the dictionary elements initialized with class C_2) as follows:
   - $\mathbf{T}^{\mathbf{low}}[l] = \begin{cases} 1 & \text{if} \quad l \leq \frac{K}{2}, \\ 0 & \text{if} \quad l > \frac{K}{2}. \end{cases}$
   - $\mathbf{T}^{\mathbf{high}}[l] = \begin{cases} 0 & \text{if} \quad l \leq \frac{K}{2}, \\ 1 & \text{if} \quad l > \frac{K}{2}. \end{cases}$

   In this way the smooth and texture patch content are extracted finally as $\mathbf{D}_{sc}^{\mathbf{low}}\mathbf{X}_j^{\mathbf{low}}$ and $\mathbf{D}_{sc}^{\mathbf{high}}\mathbf{X}_j^{\mathbf{high}}$, respectively.
3. STEP 3 - After decomposing sparse representation of $\mathbf{Y}_j$ accordingly form newly proposed representational residual error term $\mathbf{E}_j^R$ associated with atom $\mathbf{d}_j$ as (5);
4. STEP 4 - Perform rank-one approximation of $\mathbf{E}_j^R$ via SVD and set the eigenvector corresponding to the largest eigenvalue as new $\mathbf{d}_j$ and the $|\omega_j|$ largest eigenvalues as the new non-zero entries for the $\mathbf{x}_T^j$ (as in [10]);
5. STEP 5 - Keep redundant atoms (unlike [10]): mutually coherent and rarely used ones;

Proposed regularization plays an important role given that weights v_0 and v_1 control which spatial frequency content will be joined to the $\mathbf{E}_j^R$. Consequently, the SVD decomposition (STEP 4) generates atoms of the *scalable* dictionary $\mathbf{D}_{sc}$ based on the information contained within $\mathbf{E}_j^R$. By keeping more of the original high frequency info ($v_1 < 0.5$) and suppressing the lower one ($v_0 > 0.5$) the algorithm regularizes the learning process which effectively generates dictionary $\mathbf{D}_{sc}$ suitable for *scalable* representation. This enables recovery of the basic image objects shapes from the base layer L_1 resulting in a learning procedure which is tailored to the characteristics of HVS.

2.4 Denoising and scalable dictionary scheme

Prior to presenting the *scalable* denoising process, we inspect the way in which noise is removed during the classical K-SVD dictionary training. Commonly, noise is iteratively discarded throughout two stages:

1. While performing sparse coding, OMP stops when the current approximated sparse solution reaches the sphere of radius $\sqrt{n}C\sigma$ in the patches space. This radius constrains the acceptable level of the recovered noise strength i.e., $\|e\|_2^2 \leq Cn\sigma^2$. Going bellow this boundary would result in direct noise reconstruction where C is a heuristically set constant and σ stands for the noise standard deviation;

2. During the dictionary's atom's update, noise is removed via SVD decomposition that estimates new "average" direction for each atom least influenced by the distortion.

The conventional K-VD denoising energy minimization problem [16, 17] is given as:

$$\left\{\widehat{\mathbf{x}}_i, \widehat{\mathbf{D}}, \widehat{\mathbf{y}}_i\right\} = \arg\min_{\mathbf{x}_i, \mathbf{D}, \mathbf{y}_i} \lambda \left\|\mathbf{y}_i - \mathbf{y}_i^{noisy}\right\|_2^2 + \sum_i \mu_i \|\mathbf{x}_i\|_0 + \sum_i \|\mathbf{D}\mathbf{x}_i - y_i\|_2^2 \quad (6)$$

We simplify this complex minimization task by relaxing the regularization process with the introduction of the proposed *scalable* dictionary $\mathbf{D}_{sc}$ as follows:

$$\arg\min_{\mathbf{D}_{sc}, \mathbf{y}_i} \lambda \left\|\mathbf{y}_i - \mathbf{y}_i^{noisy}\right\|_2^2 + \sum_i \|\mathbf{D}_{sc}\mathbf{x}_i - y_i\|_2^2 \quad (7)$$

In (7) we decide to discard the sparse coding phase while merely performing noise removal during the *scalable* dictionary $\mathbf{D}_{sc}$ update. Our detailed study of denoising scheme in [16, 17] suggests that the initial sparseness level i.e., the average number of the non-zero coefficients nearly stays fixed during the dictionary training in the classical K-SVD setup. That is, the one established after the first OMP sparse coding over the initialized dictionary [16]. Furthermore, we impose assumption that the noise less distorts texture than smooth image components due to the high-frequency nature of the texture information. Justification for this is provided in Sec. 4.3 where we illustrate how various level of noise distort smooth and texture image blocks based on their standard deviation estimated before and after noise is introduced.

Thus, we promote the idea that, after the initial matrix inversion $\mathbf{X} = \mathbf{D}'_{sc}\mathbf{Y}$ (substitute for OMP), we could neglect subsequent ones during the dictionary learning while still obtaining a satisfactory denoising results given that texture information prevails for the modified dictionary update. For this setup, the coefficient entries $\mathbf{x}_T^j$ are only updated during the SVD decomposition employed for the atom's update. Hence, the introduced modification is expected to result in a considerably shorter computational processing time while achieving comparable quality as obtained with the non-scalable K-SVD denoising scheme.

2.5 Computational Complexity

The proposed design does not incur the cost of the original dictionary learning in [3, 10] in case of training strictly representative dictionary over the noise free image. Given that there are no additional transforms employed but just linear separation of the low and high frequencies components via semi-random initialization and introduced error matrix regularization (as shown in Sec. 2.3) the computational complexity remains of the same order as that of the conventional non-scalable K-SVD. That is, the number of operations per pixel is still $O(nT_0I)$ where I stands for the number of iterations. By setting the number of atoms $K = n$ and replacing OMP via simple matrix inversion, we manage to even decrease the processing demands while achieving good signal recovery (typically in [3, 10] K is equal to $2n$, $3n$ or $4n$). This is in particular transparent in relation to when applying proposed scheme to denoising given that sparse coding stage is removed. More details on processing time necessary for denoising are shown in Sec. 3.2.

3 Simulation results

The performance of the proposed *scalable* K-SVD method is evaluated in the set of experiments applied to:

- Standard CIF high motion video test sequences "Stephan" and "Tempete" at resolution 352×288 and a frame rate of $30Hz$;

- Several natural images e.g,, "Boat" and "Peppers", size 512×512.

Variables and parameters for all simulations are summarized in Tab. 1 together with their values and roles. Prior to processing, every frame is broken down into $N = 96,945$ or every natural image into $N = 255,025$ overlapping patches size 8×8 pixels. Thus, the vectorized dimension of the signals used for the *scalable* dictionary learning algorithm is $n = 64$ pixels, while both dictionaries $\mathbf{D}_{sc}$ and $\mathbf{D}$ contain $K = n$ atoms with redundancy factor $r = 1$. Sparsity level T_0 is set to 10 both for training and reconstruction phase. This provides the best processing effectiveness (in terms of PSNR values) for the proposed *scalable* learning design after testing the wide range of sparsity levels e.g., $T_0 = [4, 5, ..., 28]$.

The number of progressively recovered layers L_a is defined with scaling parameter $m = 4$ as $\lfloor K/m \rfloor = s = 16$ for every layer of *scalable* patch recovery and therefore image with a=1, ..., 16.

Starting from the first scalable version of the processed image i.e., base layer L_1, the reconstruction is carried out using only first $m = 4$ entries (i.e., 6.25%) per each sparse coefficient $\mathbf{x}_i$. This first level of truncated coefficients from $\mathbf{X}$ is denoted with $\mathbf{X}_1 \in R^{(4)\times N}$. Along this, we employ a truncated version of trained dictionary $\mathbf{D}_{sc}$: $\mathbf{D}_{sc}^1$ with only first m=4 atoms i.e., $[\mathbf{d}_1\mathbf{d}_2\mathbf{d}_3\mathbf{d}_4]$. The remaining recovery levels are progressively enhanced by adding four (i.e., m) additional entries in each representational vector and four (i.e., m) additional atoms. In general, (for any m value) the progressive recovery of the each image patch $\mathbf{y}_i$ for new scalable layer L_a starts by first taking all $m(a-1)$ entries from the associated sparse coefficient $\mathbf{x}_i$ previously employed for the estimation of the $\mathbf{y}_i$ scalable version at the level L_{a-1}. This continues by adding subsequent m values from the sparse coefficients $\mathbf{x}_i$ that are indexed as:

- $\mathbf{x}\,[m(a-1)+1]_i$;
- $\mathbf{x}\,[m(a-1)+2]_i$;
- ...
- $\mathbf{x}\,[m(a-1)+m]_i$

with reconstruction for scalable patch at L_a given as $\mathbf{D}_{sc}^{ma}\mathbf{x}_i^{ma}$ with $\mathbf{x}_i^{ma} \in \mathbf{X}^{ma}$. The end result is that each recovered patch at the new layer L_a will contain the first $m(a-1)$ reconstructed elements as patches in L_{a-1} and newly estimated m. For shown case, this is done until the final L_{16} restoration level is attained using full sparse representation $\mathbf{X}_{16} = \mathbf{X}$ and all atoms in dictionary $\mathbf{D}_{sc}^{16} = \mathbf{D}_{sc}$. That is, the recovery scheme for each image layer is given as:

- $L_1 = \mathbf{D}_{sc}^1\mathbf{X}_1$: 4 atoms and entries per sparse coefficient;
- $L_2 = \mathbf{D}_{sc}^2\mathbf{X}_2$: 8 atoms and entries per sparse coefficient;
- ...;
- $L_{15} = \mathbf{D}_{sc}^{15}\mathbf{X}_{15}$: 60 atoms and entries per sparse coefficient;
- $L_{16} = \mathbf{D}_{sc}^{16}\mathbf{X}_{16}$: 64 atoms and entries per sparse coefficient.

Unlike the classical, non-scalable sparse dictionary learning where practice is to train an overcomplete dictionary ($K >> n$, $r > 1$), we promote training of a complete one. One of the main reasons for this arises from our observation that whether we train a complete or overcomplete basis $\mathbf{D}_{sc}$, the achieved restoration quality for *scalable* image representation is highly comparable. Tab. 2 and Tab. 3 show the averaged comparison at ever *scalable* recovery level L_a given the complete and overcomplete *scalable* scheme for video sequence "Stephan" and image "Peppers", respectively. The number of atoms for the overcomplete $\mathbf{D}_{sc}$ dictionary is $K = 128$ ($r = 2$) thus having greater number of the recovery levels $\lfloor K/m \rfloor = s = 32$ than with the complete scheme given the same scaling factor $m = 4$. On average, when we take into account all testing results, the difference of the highest recovered layers i.e., L_{16} goes around 0.15[dB] (for frame size 352×288) and 0.66 [dB] (for image size 512×512) in favor of the overcomplete $\mathbf{D}_{sc}$ dictionary. We can see a comparison of every two recovery levels of the overcomplete $\mathbf{D}_{sc}$ dictionary with one of the recovery level of the complete $\mathbf{D}_{sc}$ in Tab. 2 and Tab. 3 (e.g., L_{25} and L_{26} with L_{13}). Conclusion follows that the *scalable* performance of the complete $\mathbf{D}_{sc}$ overruns the overcomplete $\mathbf{D}_{sc}$ at all recovery levels (bold values) except for the final L_{16} that is L_{32} for "Stephan" and almost all levels for "Peppers". Having this in mind together with the fact that lesser number of trained atoms:

- Minimizes the amount of information necessary for training and signal's recovery;
- Lowers computational complexity;

we chose $K = 64$. Prior to defining the proposed *scalable* scheme, we performed exhausting simulations in order to evaluate performance given the various P_i regularization parameters pairs $[v_0, v_1]$ listed as:

1. P_1: $[v_0, v_1] = [0, 1]$;
2. P_2: $[v_0, v_1] = [0.1, 0.9]$;
3. P_3: $[v_0, v_1] = [0.3, 0.7]$;
4. P_4: $[v_0, v_1] = [0.5, 0.5]$;
5. P_5: $[v_0, v_1] = [0.7, 0.3]$;
6. P_6: $[v_0, v_1] = [0.9, 0.1]$;
7. P_7: $[v_0, v_1] = [1, 0]$;

searching for the one which provides the most effective dictionary $\mathbf{D}_{sc}$ for *scalable* image restoration. Fig. 1 and Fig. 2 present the averaged PSNR and SSIM estimates at every recovery layer L_a of *scalable* restoration given the high motion video sequence "Stephan" and 10 averaged iterations of the natural image "Peppers" with $K = 64$ number of atoms. As we can see, out of seven P_i regularization ($1 \leq i \leq 7$) scenarios (Fig. 1 and Fig. 2), the P_7 results with the dictionary that is most effectively tailored to the *scalable* sparse image representation given that, overall, results with the highest PSNR and SSIM restoration values. Similar results are achieved for video sequence "Tempete" and several other natural images such as "Boat". For each of the testing, *Activity* measure is set to the $A = 0.27$ as in [3]. In addition, we provide results in Fig. 3 (PSNR) and Fig. 4 (SSIM) for all P_i regularization variations when training overcomplete $\mathbf{D}_{sc}$ dictionary again with $K = 128$ atoms for the $K/m = 32$ number of L_a recovery levels. Again, P_7 achieves most optimal *scalable* recovery performance.

To reiterate, the v_0 is associated with the $\mathbf{D}_{sc}^{\mathbf{low}}$ elements, which capture spatial low-frequencies. These atoms represent a compositional structure of patches extracted from large, smooth, low-variance areas, lacking in harsh edges i.e., the tennis field in the "Stephan" sequence, or the sky background in the "Tempete" sequence. On the other hand, v_1 weights the contribution

Table 1. Table of parameters

Parameter	Definition	Role
$N = 96,945$	Number of image patches	Limits the size of the training set for frames size 352×288
$N = 255,025$	Number of image patches	Limits the size of the training set for images size 512×512
$n = 64$	Constant integer	Dimension of image patch vector and each atom
K=64	Number of dictionary atoms	Limits the size of the representational basis
$K/n = r = 1$	Redundancy factor	Defines overcompleteness of the dictionary
v_0=1	1^{st} regularization parameter	Weights smooth pathc spare presentation for the atom's update
v_1=0	2^{st} regularization parameter	Weights texture patch spare presentation for the atom's update
$A = 0.27$	Activity measure	Classification threshold for smooth and texture image patches $\mathbf{y}_i$
T_0=10	Sparsity level	Limits the number of non zero entries per sparse coefficient $\mathbf{x}_i$
$l \in \{1, \ldots, 64\}$	Integer index	Defines the entry for a sparse coefficient $\mathbf{x}_i$
$m = 4$	Scaling parameter	Defines number of *scalable* layers
$\lfloor K/m \rfloor = s = 16$	General scalability level	Total number of the *scalable* layers
$L = 9$	CS scalability level	Limits number of the *scalable* recovery layers for CS setup
$s_1 < s_2, \ldots, < s_L$	Progressive CS samples	Limit number of samples per each CS *scalable* recovery layer
$S = s_L = 50$	Maximal CS sample	Limits total number of the CS samples

of the $\mathbf{D}_{sc}^{\mathbf{high}}$ atoms that contain higher spatial frequencies, that is, the areas of high detail with many contrasting edges such as the audience in "Stephan" or the flower object in "Tempete". By looking at (5) in Sec. 2.3, with P_7 parametrization ($[v_0, v_1] = [1, 0]$), we can conclude that the regularization process will in each iteration:

- Discard texture sparse approximation $\mathbf{D}_{sc}^{\mathbf{high}}\mathbf{X}_j^{\mathbf{high}}$ given the $v_1 = 0$;
- Keep the smooth part $\mathbf{D}_{sc}^{\mathbf{low}}\mathbf{X}_j^{\mathbf{low}}$ with $v_0 = 1$.

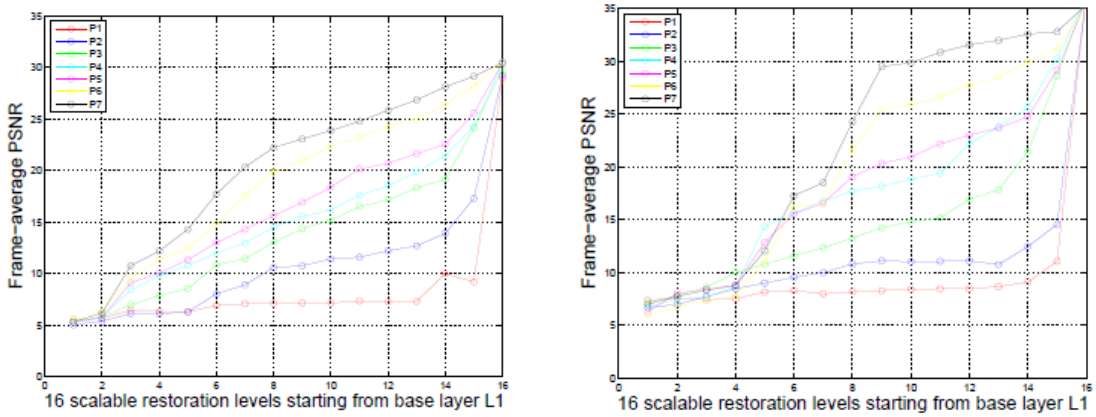
This will determine the final content of the regularized error matrix $\mathbf{E}_j^R$ where the texture patches ($Activity\,(\mathbf{y}_i) > 0.27$) are dominant information being directly included into the error synthesis rather than being just a part of the representational residual as the smooth one ($Activity\,(\mathbf{y}_i) \leq 0.27$). Furthermore, Sec. 4 provides a detailed discussion on effective structural differences between trained dictionaries and how $\mathbf{D}_{sc}$ is better tailored to the HVS perception system than the non-scalable, conventional dictionary $\mathbf{D}$. Some of these dictionaries are shown in Fig. 5a and Fig. 5b illustrating examples of *scalable* ("SC K-SVD") and non-scalable ("NSC K-SVD") dictionary trained over the first frame of the "Stephan" sequence. Given the first frame for either of sequences i.e., the *training frame*, the intro-

Table 2. Averaged PSNR quality assessment for scalable restoration given the "Stephan" video sequence for two sizes dictionary, $K = 64$ and $K = 128$.

$K = 128$	Stephan [dB]	$K = 64$	Stephan [dB]
L_{32}	**30.62**	L_{16}	30.48
L_{31}	28.93		
L_{30}	27.92	L_{15}	**29.14**
L_{29}	26.41		
L_{28}	26.16	L_{14}	**28.12**
L_{27}	25.86		
L_{26}	25.25	L_{13}	**26.87**
L_{25}	24.97		
L_{24}	24.67	L_{12}	**25.85**
L_{23}	24.13		
L_{22}	23.78	L_{11}	**24.77**
L_{21}	23.42		
L_{20}	22.91	L_{10}	**23.89**
L_{19}	22.39		
L_{18}	22.22	L_9	**23.06**
L_{17}	21.94		
L_{16}	21.50	L_8	**22.22**
L_{15}	20.36		
L_{14}	17.32	L_7	**20.34**
L_{13}	16.14		
L_{12}	15.03	L_6	**17.73**
L_{11}	13.79		
L_{10}	10.66	L_5	**14.29**
L_9	9.95		
L_8	9.51	L_4	**12.19**
L_7	8.92		
L_6	8.55	L_3	**10.77**
L_5	8.06		
L_4	**7.36**	L_2	6.18
L_3	5.90		
L_2	**5.52**	L_1	5.30
L_1	5.16		

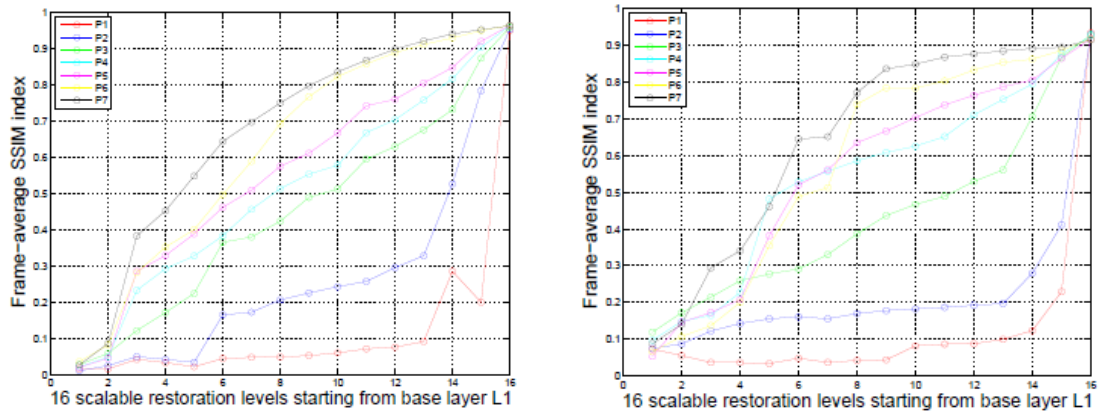
Table 3. Averaged PSNR quality assessment for scalable restoration given the "Peppers" natural image for two sizes dictionary, $K = 64$ and $K = 128$.

$K = 128$	Peppers [dB]	$K = 64$	Peppers [dB]
L_{32}	**35.78**	L_{16}	35.38
L_{31}	34.59		
L_{30}	**34.45**	L_{15}	32.82
L_{29}	33.52		
L_{28}	31.57	L_{14}	**32.60**
L_{27}	30.62		
L_{26}	30.46	L_{13}	**32.00**
L_{25}	30.10		
L_{24}	30.06	L_{12}	**31.58**
L_{23}	29.96		
L_{22}	29.66	L_{11}	**30.86**
L_{21}	29.24		
L_{20}	29.01	L_{10}	**29.88**
L_{19}	28.70		
L_{18}	28.51	L_9	**29.52**
L_{17}	28.41		
L_{16}	**28.25**	L_8	24.30
L_{15}	24.20		
L_{14}	**22.11**	L_7	15.61
L_{13}	18.69		
L_{12}	**16.08**	L_6	15.07
L_{11}	15.06		
L_{10}	**14.07**	L_5	12.37
L_9	11.33		
L_8	**10.99**	L_4	8.17
L_7	9.99		
L_6	**9.25**	L_3	7.08
L_5	8.97		
L_4	**7.97**	L_2	6.80
L_3	7.68		
L_2	**7.40**	L_1	6.35
L_1	6.12		



(a) "Stephan" video sequence **(b)** "Peppers" image

Figure 1. Averaged PSNR scalable results given the seven different setups for regularization parameters $[v_0, v_1]$ and $K = 64$ number of atoms.



(a) "Stephan" video sequence **(b)** "Peppers" image

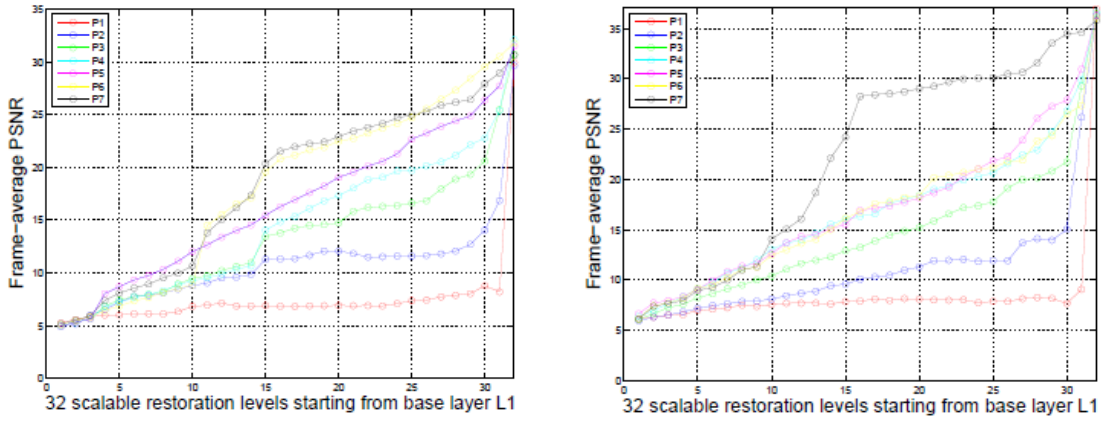
Figure 2. Averaged SSIM scalable results given the seven different setups for regularization parameters $[v_0, v_1]$ and $K = 64$ number of atoms.

duced training scheme is carried out only once generating the $\mathbf{D}_{sc}$ dictionary. Subsequently, while reconstructing each incoming frame we use this single $\mathbf{D}_{sc}$ thus, not training any new dictionary. This approach considerably reduces the computational complexity of the *scalable* sparse video representation, since training is done only once instead for each incoming frame. This is immensely important in the context of real-time *scalable* image/video applications development. It is necessary to mention that in general, when the video scene undergoes significant changes with respect to the *training frame*, a new training frame should be inserted. This is necessary in order to accommodate for the difference in the compositional structure of previous frames and newly changed one.

3.1 Scalability Performance

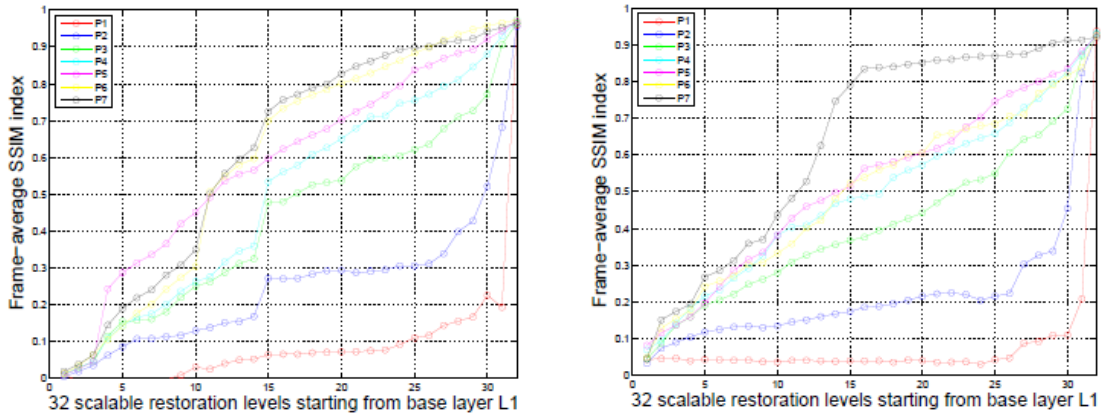
The comparison of the restoration quality is done for the proposed regularized *scalable* "SC K-SVD" and conventional non-scalable "NSC K-SVD" algorithm. Fig. 6a and Fig. 6b illustrate the PSNR estimates for video sequences "Stephan" and "Tempete" respectively. Shown results are averaged over all frames given each of 16 recovery $\{L_a\}_{a=1}^{16}$ layers. Minor exception is the "Stephan" sequence which frames are divided into two groups: [1, 270] and [271, 300]. This frame separation is carried out in order to demonstrate the variation in the quality of the restored image, when a new object is introduced in the frame 271. We would expect certain degradation in the restoration quality given that $\mathbf{D}_{sc}$ is trained over the *training frame* which does not contain a newly introduced visual object.

Depicted results clearly demonstrate that the proposed regularized scheme considerably outperforms the standard [10] over all recovery levels L_a, where average gain of 11.32 [dB] ("Stephan", first 270 frames) and 8 [dB] ("Tempete", all frames) is achieved. This proves superiority of the proposed work in terms of *scalable* recovery. Once a new image object appears (e.g., the tennis net in the "Stephan" sequence) a noticeable drop in the *scalable* recovery quality can be noticed in Fig. 6a for the second frame group [271, 300]. Specifically, on average "SC K-SVD" PSNR declines for 1.84 [dB] while still outperforming the "NSC K-SVD" for 9.43 [dB]. Overall, only in the case when all the information on the sparse coefficients is available ($\mathbf{X_{16}} \in R^{(64)\times N}$), the regular K-SVD algorithm has a slight advantage over the proposed scheme. Besides the standard objective quality assessment i.e., PSNR, we consider an alternative quality measure, so-called Structural Similarity Index (SSIM) [37]. It is designed to quantify the degree to



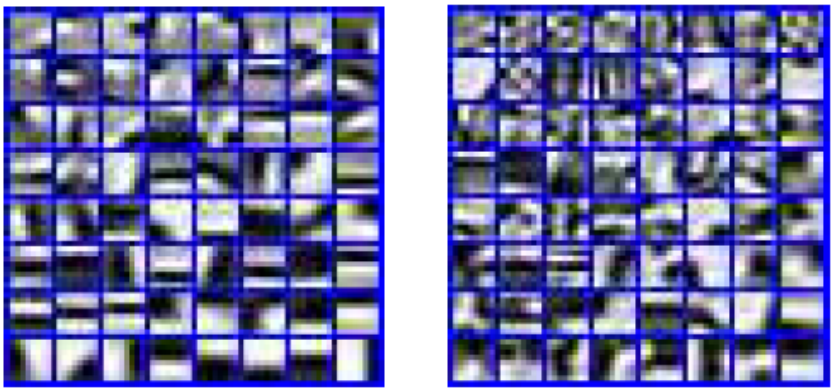
(a) "Stephan" video sequence **(b)** "Peppers" image

Figure 3. Averaged PSNR scalable results given the seven different setups for regularization parameters $[v_0, v_1]$ and $K = 128$ number of atoms.



(a) "Stephan" video sequence **(b)** "Peppers" image

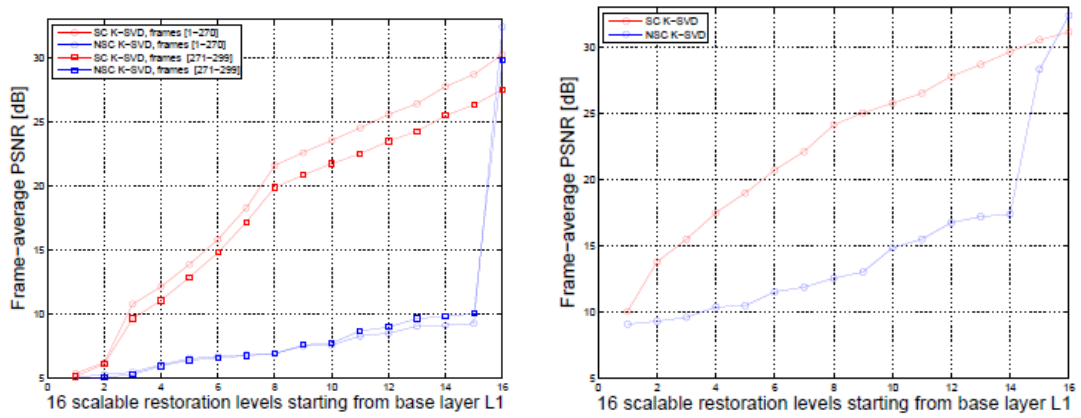
Figure 4. Averaged SSIM scalable results given the seven different setups for regularization parameters $[v_0, v_1]$ and $K = 128$ number of atoms.



(a) "SC K-SVD" **(b)** "NSC K-SVD"

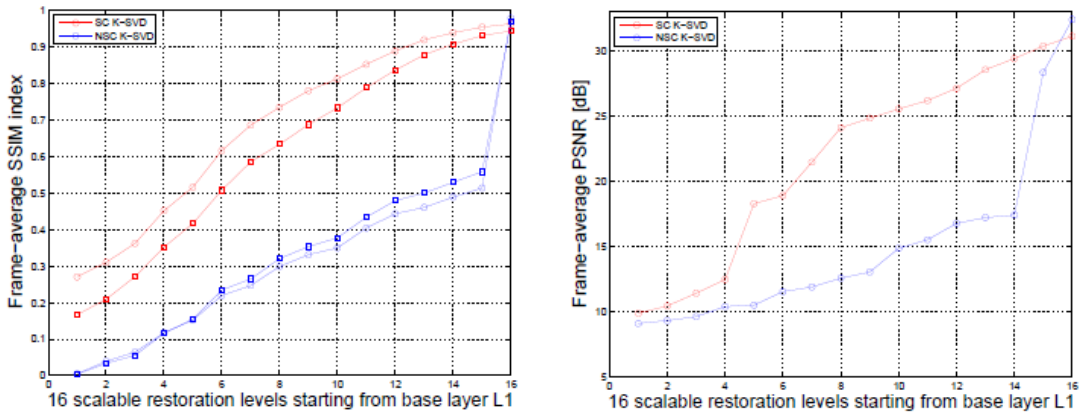
Figure 5. Scalable and non-scalable dictionaries for the 1^{st} "Stephan" frame.

which image structural information is degraded by calculating a quality index ranging from 0 (denoting highest distortion) up to 1 (denoting no distortion). This measure is specially appealing for the evaluation of the proposed *scalable* image restoration framework due to the fact that the SSIM is based on the discussed HVS characteristics. Specifically, it takes into account local pixels distortions of the luminance and contrast information. The higher the SSIM index value gets, the more successful retrieval of the HVS perception information at each *scalable* layer L_a will be. This results in a better visual information thus providing progressive image recovery of better quality. Therefore, SSIM index values shown in Fig. 7 quantify the degree of the degradation of structural information in a frame at each *scalable* reconstruction level L_a. Once again, these estimates are averaged over all frames for both testing video sequences. As in the case of PSNR, in Fig. 7a , for the "Stephan" sequence, we can see that the proposed *scalable* method surpasses in general the non-scalable for 0.37 (first frame



(a) "Stephan" video sequence **(b)** "Tempete" video sequence

Figure 6. Frame-average PSNR of the scalable reconstructed video test sequences ("Stephan" and "Tempete") given for each layer L_a of the scalable video reconstruction using the scalable and non-scalable K-SVD algorithm.



(a) "Stephan" video sequence **(b)** "Tempete" video sequence

Figure 7. Frame-average SSIM index of the scalable reconstructed video test sequences ("Stephan" and "Tempete") given for each layer L_a of the scalable video reconstruction using the scalable non-scalable K-SVD algorithm.

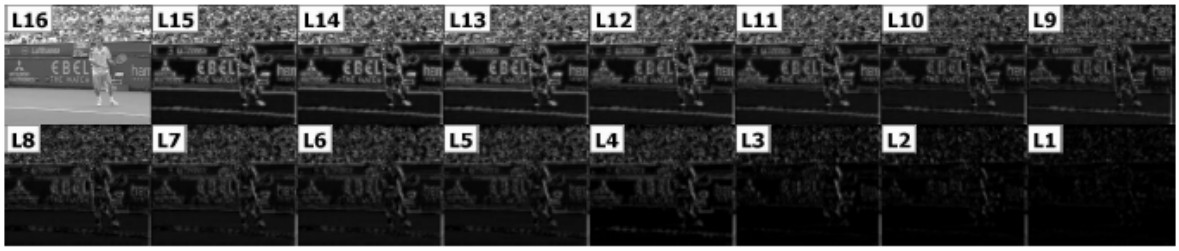
group) and 0.28 (second frame group). Similarly, we can see in Fig. 7b, the SSIM difference of 0.27 for the "Tempete" sequence over all recovery levels L_a between two dictionary learning algorithms. Interestingly, for SSIM evaluation we have a different trend than in the case of PSNR where once we switch to the second frame group the quality assessment shows a high drop for restoration levels L_{14}, L_{15}, L_{16}. In contrast, Fig. 7a denotes a high similarity in the SSIM values for L_{14}, L_{15}, L_{16} at around 0.94 given both frame groups, meaning that the structural information of the image is preserved despite the fact that we have new visual object in the scene.

Visualization of the results is provided in Fig. 8, Fig. 9 for "Stephan" and in Fig. 10, Fig. 11 for "Tempete" sequence in order to emphasize the subjective perceptual quality. These figures illustrate the *scalable* reconstruction outcome at every recovery level L_a for the so-called *training frame* (Fig. 8 and Fig. 10). The last frames for both video sequences are shown in Fig. 9 and Fig. 11. Here one can observe the visual variations in the restoration quality when the new object containing the high-frequency content structure is introduced (i.e., the tennis net in Fig. 9) or the more spatial low-frequencies are added i.e., the background in "Tempete" in Fig. 11. All *scalable* restorations are performed over the single trained dictionary given the first *training* frame. From these figures one can notice that the proposed *scalable* scheme is able to recover the frame at a recovery level L_3 ($\mathbf{D}_{sc}^3 \in R^{(64)\times 12}$ and $\mathbf{X}_3 \in R^{(12)\times N}$) whereas [1] fails to show any *scalable* characteristics overall up to L_{15} ($\mathbf{D}_{sc}^{15} \in R^{(64)\times 60}$ and $\mathbf{X}_{15} \in R^{(60)\times N}$) for "Stephan" and L_8 ($\mathbf{D}_{sc}^{8} \in R^{(64)\times 32}$ and $\mathbf{X}_8 \in R^{(32)\times N}$) for "Tempete". It should be said that the "NSC K-SVD" does show slight scalability with the "Tempete" sequence. However, this is still far from the performance of the proposed method that keeps its reconstruction efficiency consistent for quite different video sequences, hence showing its processing stability.

3.2 Application to image processing 1: denoising

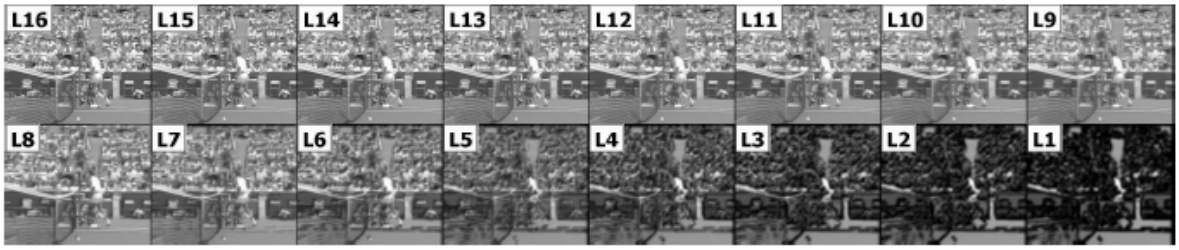
This experimental section justifies and validates advantages of the proposed scheme for the denoising application. We compare introduced *scalable* (denoted as "SC") over the non-scalable (denoted as "NSC") together with the overcomplete classical K-SVD dictionary "Org". For the aforementioned algorithm setups, we discuss objective quality assessment and time processing complexity. However, unlike in Sec. 3.1 where dictionary training is done only once over the first non distorted frame in the video sequence, for denoising, the dictio-

(a) Training frame, "Stephan" test sequence ("SC K-SVD")

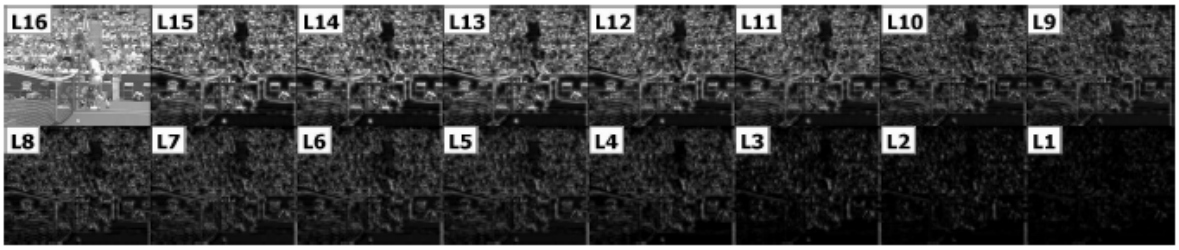


(b) Training frame, "Stephan" test sequence ("NSC K-SVD")

Figure 8. Visual assessment of the scalable reconstruction using the scalable and non-scalable K-SVD at every recovery level L_a.

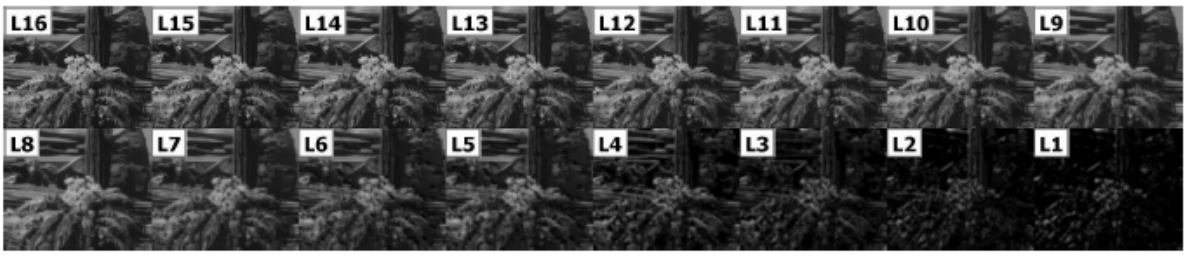


(a) Last frame i.e., 300th, "Stephan" test sequence ("SC K-SVD")

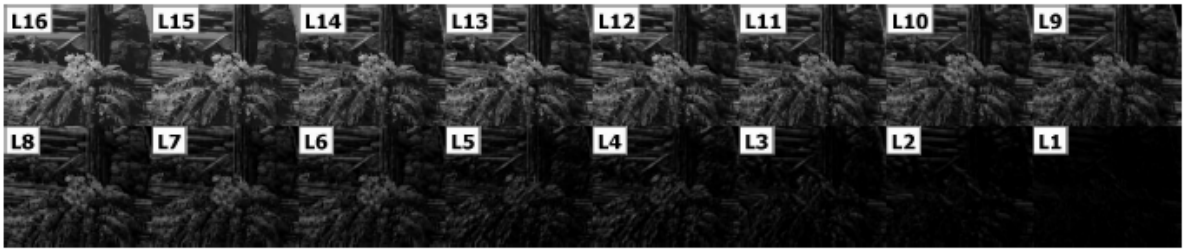


(b) Last frame i.e., 300th, "Stephan" test sequence ("NSC K-SVD")

Figure 9. Visual assessment of the scalable reconstruction using the scalable and non-scalable K-SVD at every recovery level L_a.

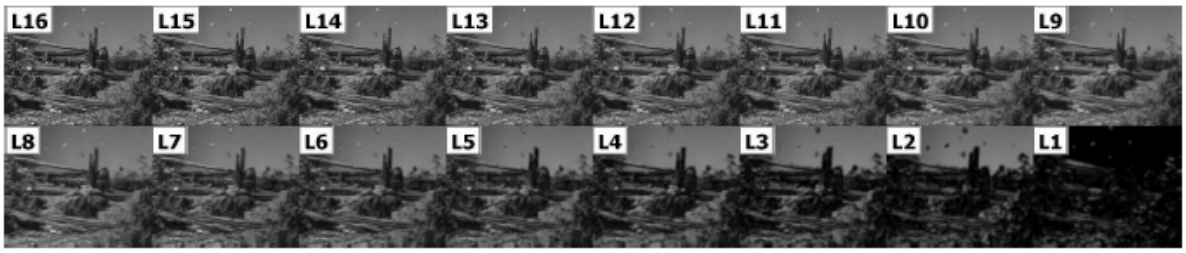


(a) Training frame, "Tempete" test sequence("SC K-SVD")

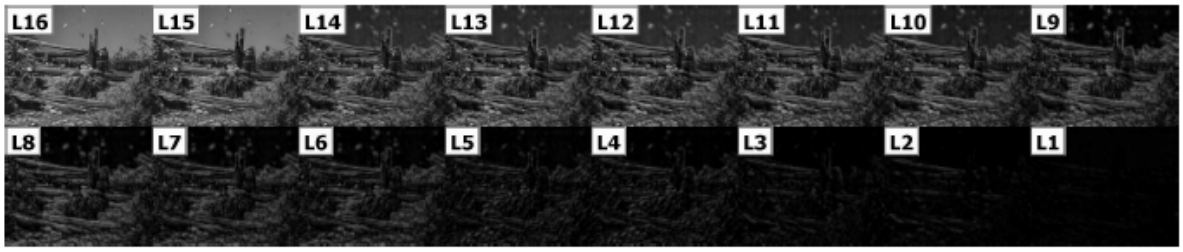


(b) Training frame, "Tempete" test sequence ("NSC K-SVD")

Figure 10. Visual assessment of the scalable reconstruction using the scalable and non-scalable K-SVD at every recovery level L_a.



(a) Last frame i.e., 260th, "Tempete" test sequence("SC K-SVD")



(b) Last frame i.e., 260th, "Tempete" test sequence ("NSC K-SVD")

Figure 11. Visual assessment of the scalable reconstruction using the scalable and non-scalable K-SVD at every recovery level L_a.

nary is trained for each incoming noisy frame, likewise in [15]. All provided results are averaged over two video sequences frames i.e., "Stephan" and "Tempete" together with the additional estimates for the several conventional images i.e., "Boat" and "Peppers". For experiments we consider the range of five different noise standard deviations: $\sigma = [20, 40, 60, 80, 100]$. The restoration of every *scalable* level L_a is carried out in the same way as in the previous section. Starting from Tab. 4 to Tab. 7 we can see comparison for denoising outcomes at every *scalable* recovery layer L_a for all mentioned data. Additionally, each level is compared against the denoising estimates of the overcomplete K-SVD scheme "Org" (red, bold values) in order to emphasize the effectiveness of the proposed scheme. From the provided results conclusion follows that PSNR values of the "SC" at the final restoration level L_{16} (Tab. 4 to Tab. 7) are, at most cases, comparable or surpass (black bold values) denoising performance of the classical K-SVD setup once the noise passes value $\sigma = 60$. This better performance indicates that the higher frequencies are less influenced by the noise since they are enforced as the most important content of the trained dictionary, contributing most to the restored frame or image unlike in the conventional K-SVD. Overall, the proposed method achieves better denoising performance with lowest and highest gain of 0.1 [dB] and 5.7 [dB], respectively.

In addition, we performed testing for the scenario where the sparse coding stage is also removed from the classical non-scalable K-SVD scheme in order to further validate the practicality of the proposed *scalable* design. After simulations final estimates show that there is a drop of 2 [dB] for "NSC" without sparse coding stage when compared to the best denoising results of the "SC". Hence, newly introduced regularization scheme is efficient when it comes down to noise removal given that we only keep atom's update out of two iterative stages for dictionary learning over the corrupted image. The greatest benefit of the *scalable* denoising is direct reduction of both, computational complexity and processing time where Tab. 8 shows the total denoising run times in seconds for two image sizes:

1. 352x288 - size of the video sequences frames;
2. 512x512 - size of the conventional images.

Illustrated times are outcomes of processing on the Dell operating system with 64 bit Intel core, 8 GB RAM memory and 2.40 GHz processor. The number of iterations for the provided results is fixed and set to sixteen. Based on the averaged run times we can see reduction in:

1. approximately 6.5 times for data of size 352x288 when comparing "SC" vs "Org";
2. approximately 7.3 times for data of size 512x512 when comparing "SC" vs "Org";
3. approximately 10.8 times for data of size 352x288 when comparing "SC" vs "NSC";
4. approximately 11 times for data of size 512x512 when comparing "SC" vs "NSC";

provided that we achieve still highly comparable (lower levels of noise) or better results (higher levels of noise). The forth column of the Tab. 8 illustrates the time for the error matrix formation per each iteration. These numbers are aiming to show that introduced modification of the atom's update in the form of a new error matrix scheme influences processing complexity on a minor scale by being increased on average for two seconds.

Finally, Fig. 12, Fig. 13, Fig. 14 and Fig. 15 illustrate visual preview for all discussed data at the recovery level L_{16} after the noise $\sigma = 40$ is removed given the *scalable*, non-scalable complete or overcomplete K-SVD scheme. After additional subjective quality assessment we can conclude that provided results are highly comparable where, as emphasized, proposed method "SC" put considerably less computational demands than both classical K-SVD setups.

Table 4. PSNR quality assessment for scalable denoising via the scalable and non-scalable K-SVD dictionary, "Stephan" sequence.

σ	$PSNR_i\,[dB]$	*Alg.*	Scalable recovery levels L_i															
			L_{16}	L_{15}	L_{14}	L_{13}	L_{12}	L_{11}	L_{10}	L_9	L_8	L_7	L_6	L_5	L_4	L_3	L_2	L_1
20	**22.14**	**SC**	28.71	**28.73**	**28.74**	**28.75**	**28.71**	**28.71**	**28.50**	**28.47**	**27.79**	**27.67**	**26.37**	**26.18**	**24.34**	**24.13**	**21.57**	**21.07**
		NSC	**28.88**	28.58	28.22	27.66	26.93	26.57	26.17	25.90	25.39	25.05	24.37	23.97	23.11	22.75	21.08	20.46
		Org	29															
40	**16.13**	**SC**	24.30	24.32	24.34	**24.36**	**24.36**	**24.38**	**24.34**	**24.35**	**24.18**	**24.17**	**23.76**	**23.71**	**22.88**	**22.79**	**21.02**	**20.68**
		NSC	**24.64**	**24.52**	**24.35**	24.09	23.60	23.14	22.84	22.68	22.45	22.28	21.97	21.66	21.17	20.78	20.07	19.51
		Org	24.73															
60	**12.56**	**SC**	21.93	21.96	21.98	22.00	**22.01**	**22.03**	**22.03**	**22.05**	**22.00**	**22.01**	**21.86**	**21.84**	**21.48**	**21.45**	**20.35**	**20.13**
		NSC	**22.31**	**22.28**	**22.13**	**22.03**	21.90	21.87	21.70	21.56	21.38	21.18	21.02	20.86	20.52	20.28	19.70	19.29
		Org	22.32															
80	**10.05**	**SC**	20.49	20.51	20.54	20.56	20.58	20.60	20.61	**20.63**	**20.61**	**20.63**	**20.57**	**20.58**	**20.41**	**20.40**	**19.72**	**19.58**
		NSC	**20.78**	**20.79**	**20.72**	**20.69**	**20.66**	**20.66**	**20.64**	20.61	20.45	20.41	20.30	20.21	19.96	19.79	19.39	19.07
		Org	20.75															
100	**8.13**	**SC**	19.45	19.48	19.52	19.55	19.57	19.60	**19.62**	**19.65**	**19.66**	**19.68**	**19.67**	**19.68**	**19.60**	**19.61**	**19.17**	**19.10**
		NSC	**19.63**	**19.64**	**19.63**	**19.64**	**19.64**	**19.64**	19.60	19.60	19.57	19.55	19.50	19.48	19.35	19.31	18.99	18.77
		Org	19.59															

Table 5. PSNR quality assessment for scalable denoising via the scalable and non-scalable K-SVD dictionary, "Tempete" sequence.

σ	$PSNR_i\,[dB]$	*Alg.*	Scalable recovery levels L_i															
			L_{16}	L_{15}	L_{14}	L_{13}	L_{12}	L_{11}	L_{10}	L_9	L_8	L_7	L_6	L_5	L_4	L_3	L_2	L_1
20	**22.12**	**SC**	28.29	**28.3**	**28.21**	**28.21**	**28.03**	**28.02**	**27.73**	**27.7**	**27.27**	**27.22**	**26.63**	**26.55**	**25.62**	**25.49**	**23.86**	**23.56**
		NSC	**28.27**	28.11	27.97	27.81	27.59	27.41	26.99	26.75	26.25	25.97	25.42	25.15	24.52	24.08	23.06	22.56
		Org	28.40															
40	**16.07**	**SC**	24.61	24.63	24.62	**24.64**	**24.61**	**24.62**	**24.56**	**24.56**	**24.45**	**24.46**	**24.28**	**24.27**	**23.92**	**23.88**	**23.01**	**22.87**
		NSC	**24.71**	**24.70**	**24.64**	**24.64**	24.58	24.49	24.29	24.16	24.08	23.88	23.77	23.51	23.38	22.68	22.20	21.72
		Org	24.76															
60	**12.55**	**SC**	**22.59**	**22.61**	**22.63**	**22.65**	**22.65**	**22.67**	**22.66**	**22.67**	**22.64**	**22.66**	**22.6**	**22.61**	**22.48**	**22.47**	**22.07**	**22**
		NSC	22.59	22.60	22.60	22.60	22.58	22.58	22.53	22.51	22.44	22.41	22.29	22.23	21.96	21.85	21.51	21.23
		Org	22.52															
80	**10.07**	**SC**	**21.18**	**21.2**	**21.23**	**21.25**	**21.27**	**21.29**	**21.3**	**21.32**	**21.32**	**21.34**	**21.33**	**21.34**	**21.3**	**21.3**	**21.13**	**21.11**
		NSC	21.12	20.32	20.19	19.35	17.94	15.79	15.32	15.29	15.06	14.10	14.00	13.75	13.72	12.22	12.18	11.62
		Org	21.01															
100	**8.14**	**SC**	**20.04**	**20.07**	**20.09**	**20.12**	**20.14**	**20.17**	**20.18**	**20.21**	**20.22**	**20.25**	**20.26**	**20.28**	**20.27**	**20.28**	**20.22**	**20.22**
		NSC	19.92	19.65	18.72	17.86	16.61	16.01	15.35	14.23	13.65	13.63	13.36	12.68	11.92	11.38	11.12	10.92
		Org	19.69															

Table 6. PSNR quality assessment for scalable denoising via the scalable and non-scalable K-SVD dictionary, "Boat" image.

σ	$PSNR_i\,[dB]$	*Alg.*	Scalable recovery levels L_i															
			L_{16}	L_{15}	L_{14}	L_{13}	L_{12}	L_{11}	L_{10}	L_9	L_8	L_7	L_6	L_5	L_4	L_3	L_2	L_1
20	**22.11**	**SC**	30.13	30.14	30.01	**30.02**	**29.82**	**29.82**	**29.49**	**29.49**	**28.94**	**28.94**	**28.06**	**28.06**	**26.60**	**26.59**	**24.57**	**24.55**
		NSC	**30.31**	**30.26**	**30.14**	30.02	29.79	29.59	29.31	29.04	28.67	28.40	27.75	27.29	26.39	26.01	24.50	24.15
		Org	30.52															
40	**16.09**	**SC**	26.80	26.83	26.81	26.84	26.80	26.82	**26.74**	**26.76**	**26.60**	**26.62**	**26.25**	**26.26**	**25.47**	**25.48**	**24.05**	**24.06**
		NSC	**27.11**	**27.13**	**27.09**	**27.09**	**26.91**	**26.87**	26.72	26.56	26.32	26.23	25.88	25.84	25.24	24.83	24.04	23.71
		Org	27.11															
60	**12.55**	**SC**	24.86	24.89	24.90	24.93	24.94	24.97	24.95	24.98	**24.93**	**24.96**	**24.80**	**24.82**	**24.39**	**24.41**	**23.49**	**23.51**
		NSC	**25.11**	**25.15**	**25.13**	**25.16**	**25.10**	**25.12**	**25.04**	**25.05**	24.87	24.88	24.70	24.68	24.23	24.12	23.54	23.34
		Org	25.02															
80	**10.05**	**SC**	23.46	23.50	23.53	23.56	23.58	23.62	23.63	23.67	**23.66**	**23.69**	**23.62**	**23.65**	**23.42**	**23.45**	**22.89**	**22.92**
		NSC	**23.59**	**23.63**	**23.65**	**23.68**	**23.69**	**23.73**	**23.70**	**23.73**	23.65	23.68	23.57	23.59	23.31	23.32	22.94	22.88
		Org	22.83															
100	**8.15**	**SC**	22.37	22.41	22.44	22.48	22.51	22.55	22.57	22.61	**22.62**	**22.66**	**22.63**	**22.67**	**22.53**	**22.56**	**22.22**	**22.26**
		NSC	**22.43**	**22.48**	**22.52**	**22.57**	**22.59**	**22.63**	**22.61**	**22.63**	22.62	22.65	22.58	22.60	22.46	22.46	22.22	22.21
		Org	21.46															

Table 7. PSNR quality assessment for scalable denoising via the scalable and non-scalable K-SVD dictionary, "Peppers" image.

σ	$PSNR_i\,[dB]$	*Alg.*	Scalable recovery levels L_i															
			L_{16}	L_{15}	L_{14}	L_{13}	L_{12}	L_{11}	L_{10}	L_9	L_8	L_7	L_6	L_5	L_4	L_3	L_2	L_1
20	**22.12**	**SC**	31.94	31.95	31.97	**31.98**	**31.96**	**31.97**	**31.90**	**31.91**	**31.29**	**31.29**	**30.81**	**30.79**	**29.52**	**29.49**	**26.88**	**26.66**
		NSC	**31.99**	**32.00**	**31.98**	31.85	31.56	30.61	29.70	29.25	29.08	28.98	28.74	28.43	27.89	27.52	26.32	25.82
		Org	32.24															
40	**16.08**	**SC**	28.83	28.86	28.88	28.91	**28.91**	**28.94**	**28.94**	**28.96**	**28.71**	**28.74**	**28.59**	**28.61**	**28.00**	**28.00**	**26.23**	**26.07**
		NSC	**29.04**	**29.09**	**29.04**	**29.08**	28.69	28.71	28.67	28.69	27.88	27.89	27.82	27.65	27.05	26.99	26.07	25.49
		Org	29.07															
60	**12.58**	**SC**	26.54	26.59	**26.62**	**26.67**	**26.66**	**26.71**	**26.72**	**26.76**	**26.70**	**26.74**	**26.72**	**26.75**	**26.52**	**26.54**	**25.41**	**25.33**
		NSC	**26.68**	**26.73**	25.89	25.93	25.93	25.97	25.91	25.94	25.85	25.86	25.14	25.14	24.85	24.83	24.65	24.21
		Org	26.05															
80	**10.05**	**SC**	**24.85**	**24.89**	**24.93**	**24.97**	**24.99**	**25.03**	**25.06**	**25.11**	**25.13**	**25.17**	**25.17**	**25.21**	**25.08**	**25.12**	**24.50**	**24.46**
		NSC	24.70	24.74	19.06	19.07	18.98	18.99	18.99	19.00	18.99	19.00	18.99	18.99	17.31	17.31	17.27	17.16
		Org	23.91															
100	**8.12**	**SC**	**23.41**	**23.46**	**23.49**	**23.53**	**23.56**	**23.60**	**23.63**	**23.67**	**23.70**	**23.74**	**23.77**	**23.81**	**23.80**	**23.83**	**23.55**	**23.54**
		NSC	22.61	22.66	22.14	22.17	22.19	22.22	17.23	17.24	17.24	17.25	17.23	17.23	13.54	13.53	13.27	11.75
		Org	22.13															

Table 8. Comparison of the processing time given tree denoising schemes

Image size	Alg.	Total denoising run time [s]	Error formation time [s]
352x288	Org	2897.6	1.26
	NCS	3515.7	2.61
	SC	**325.1**	4.42
512x512	Org	7565.5	1.59
	NCS	7337.3	1.58
	SC	**665.5**	3.29

3.3 Application to image processing 2: compressive sensing

Following closely the experimental layout suggested in [26], we investigate the effectiveness and the performance of the proposed *scalable* CS video acquisition scheme when combined with a learned basis rather than a predefined one. In particular, the proposed framework aims for the frame-by-frame progressive CS recov-

(a) "SC" **(b)** "NSC" **(c)** "Org"

Figure 12. Visual assessment for denoising via the scalable, non-scalable complete and over complete K-SVD at L_{16} recovery level of the first *training* frame of the "Stephan" video sequence, $\sigma = 40$

(a) "SC" **(b)** "NSC" **(c)** "Org"

Figure 13. Visual assessment for denoising via the scalable, non-scalable complete and over complete K-SVD at L_{16} recovery level of the first *training* frame of the "Tempete" video sequence, $\sigma = 40$

(a) "SC" **(b)** "NSC" **(c)** "Org"

Figure 14. Visual assessment for denoising via the scalable, non-scalable complete and over complete K-SVD at L_{16} recovery level given the image "Boat", $\sigma = 40$

ery while analysing the implications of the sub-Nyquist CS paradigm in both the *scalable* and *adaptive* representational domain. Likewise in previous experimental sections and as in [26], the image is processed block by block. Mainly, we take into consideration two cases of the CS *scalable* recovery:

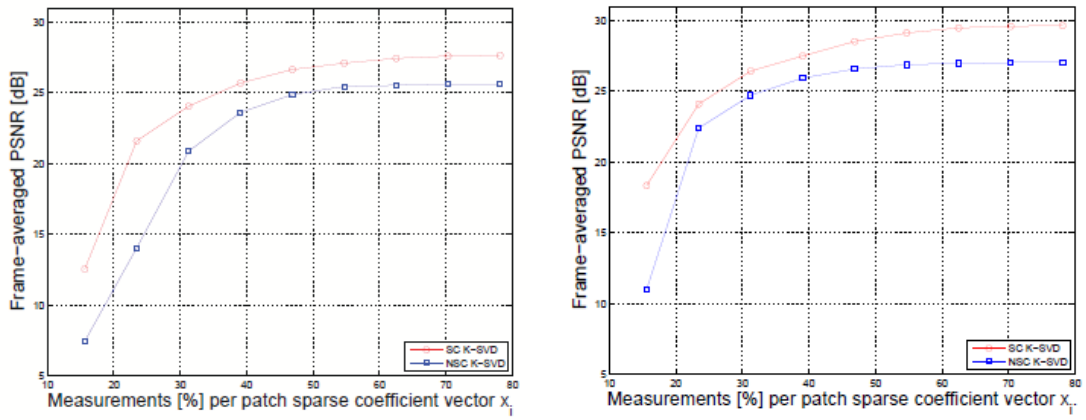
- With the proposed *scalable* K-SVD dictionary tailored to this task;
- With the conventional non-scalable K-SVD dictionary.

Rather than taking the full number of measurements [5, 45, 46] over every incoming frame, CS sampling is carried out in incremental steps. Note that this is applicable only for the CS *scalable* sensing scenario. Given the sufficient number of progressive measurements per patch as $s_1, s_2, \ldots, s_L$ $(s_i < n)$ we are able to recover the frame gradually after incrementally retrieving entries of sparse vector coefficients in $\mathbf{X}_i$ via OMP. Furthermore, each incremental number of samples s_i satisfies the fundamental result of the CS theory [2] that imposes the limit on the necessary number of measurements for satisfactory signal reconstruction.

Unlike the conventional CS for our testing we apply specially structured sampling matrix $\mathbf{\Phi}$. This aims to achieve efficient *scalable* acquisition of samples over each image layer commonly denoted as $\mathbf{y}_{CS} = \mathbf{\Phi}\mathbf{y} = \mathbf{\Phi}\mathbf{D}_{sc}\mathbf{x}$. Implementation is carried out via the systematic non-adaptive approach as in [26] that generates the structural sampling matrix $\mathbf{\Phi}$ optimally suited for the *scal-*

(a) "SC" **(b)** "NSC" **(c)** "Org"

Figure 15. Visual assessment for denoising via the scalable, non-scalable complete and over complete K-SVD for the L_{16} recovery level given the image "Peppers", $\sigma = 40$



(a) "Stephan" video sequence **(b)** "Tempete" video sequence

Figure 16. Frame-average PSNR of the scalable CS reconstruction for two test video sequences as a function of the measurement procent using the scalable ("SC K-SVD") and non-scalable ("NSC K-SVD") algorithm.

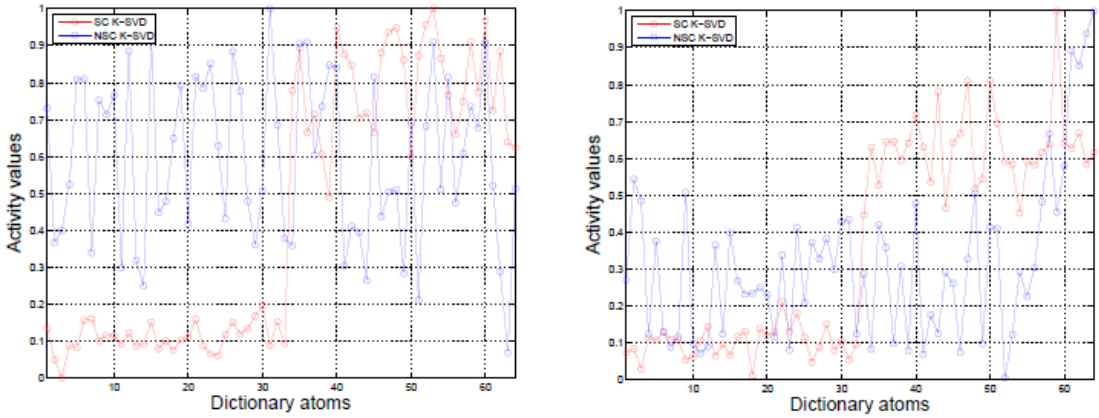
able task at hand. For each recovery step (as in [26]) we scale sampling matrix size-wise into its truncated versions as $\mathbf{\Phi}_i \in R^{s_i \times n}$. Once the sampling is done we attain a group of samples each denoted as $\mathbf{y}^i_{CS}$. The sampling is structured in a way that the basic level is collected via $\mathbf{\Phi}_1$ that contains binary entries generated from the Gaussian distribution. Remaining measurements are sampled via Bernoulli binary distributed entries of $\mathbf{\Phi}_i$ consecutively added up to the basic layer for the *scalable* restoration. Again, starting from a base level $i = 1$ and with $\mathbf{y}^1_{CS} = \mathbf{\Phi}_1\mathbf{y} = \mathbf{\Phi}_1\mathbf{D}_{sc}\mathbf{x}$ (approximately 15% of original patch image size $\mathbf{y}_i$) we advance through enhancement layers by uniformly collecting additional number of samples (e.g., s_2, $s_3, ..., s_L = S$) in each step until the total number of $S < n$ samples is reached. Hence, given the single trained dictionary $\mathbf{D_{sc}}$ (as in Sec. 3.1) learned over *training* frame for either of video test sequences, one can define an arbitrary number of sampled layers over extracted image patches.

Fig. 16 shows reconstruction results obtained via the proposed *adaptive scalable* CS approach averaged over the frames starting with $s_1 = 10$ and adding five more samples per each patch as frame recovery progresses (e.g., $s_2 = 15$, $s_3 = 20$, etc.). Therefore, we define in total nine sampling levels resulting in nine patch, that is, frame reconstruction layers. Thus, full number of measurements is $S = 50$ ($n > 50$) which accounts for roughly 80% of the information of the sampled signal y_i. The gap between the performance of the two methods is evident for the layers sampled both at low (e.g., 15%, 23% , 31% and 39%) and high subrates (47%, 55%, 62%, 70% and 80%) of sampling information, at around 3.03 [dB] in case of "Stephan" sequence and for the "Tempete" frames at 2.96 [dB]. We can see that the proposed design is successful for the subsampling factors at different rates whereas the conventional K-SVD has a comparable but not better performance as more measurements are added.

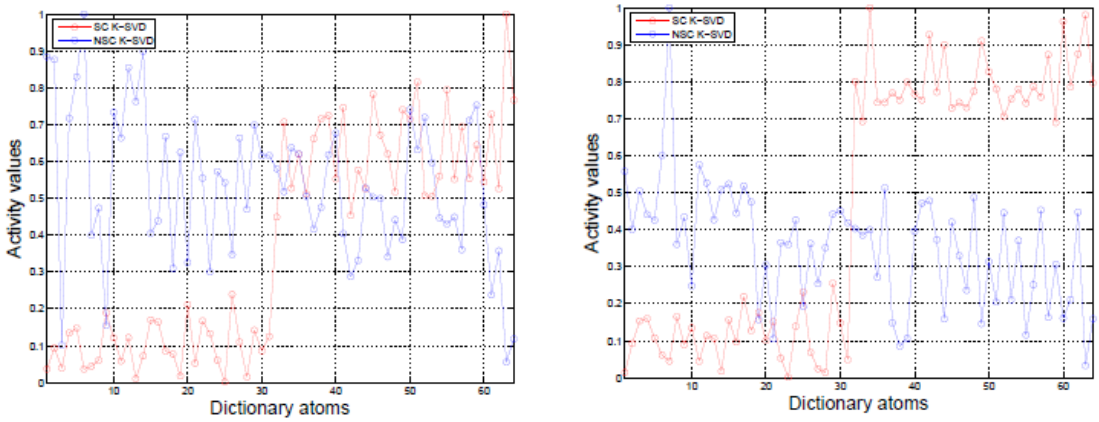
4 Discussion

Training the dictionary for the *scalable* sparse data representation and applying it to the denoising adopts a different approach than the one originally introduced by K-SVD [3, 9, 15]. Mainly, the atom update illustrated in Sec. 2.3 and denoising proposed Sec. 2.5 are grounded in the following assumptions:

- The progressive and quality wise scaled recovery of the image/frame can be attained via learned dictionary by modeling the main HVS perception mechanism properties and integrating them during dictionary's training;
- This implementation should be taken forward by MCA based semi-random initialization, allocation,



(a) "Stephan" **(b)** "Tempete"



(c) "Boat" **(d)** "Peppers"

Figure 17. *Activity* atom's pattern for the dictionaries of the two video sequences and two natural images.

separation and regularization of low and high spatial frequencies information captured by the atoms during the dictionary training procedure;

- Texture image components are less distorted by noise than the smooth ones thus with the newly introduced design SVD over proposed regularized error matrix $\mathbf{E}_j^R$ is sufficient for noise removal.

These hypotheses give rise to a series of questions:

1. How are spatial frequencies distributed over *scalable* and non-scalable dictionary's atoms?;
2. Could this distribution be denoted as a built-in property of the trained dictionaries?
3. Does the proposed design properly adopt the HVS perception mechanism properties?
4. To what degree noise effects smooth and texture image properties?

The following sections aim to look into some answers to these questions.

4.1 Spatial frequencies distribution

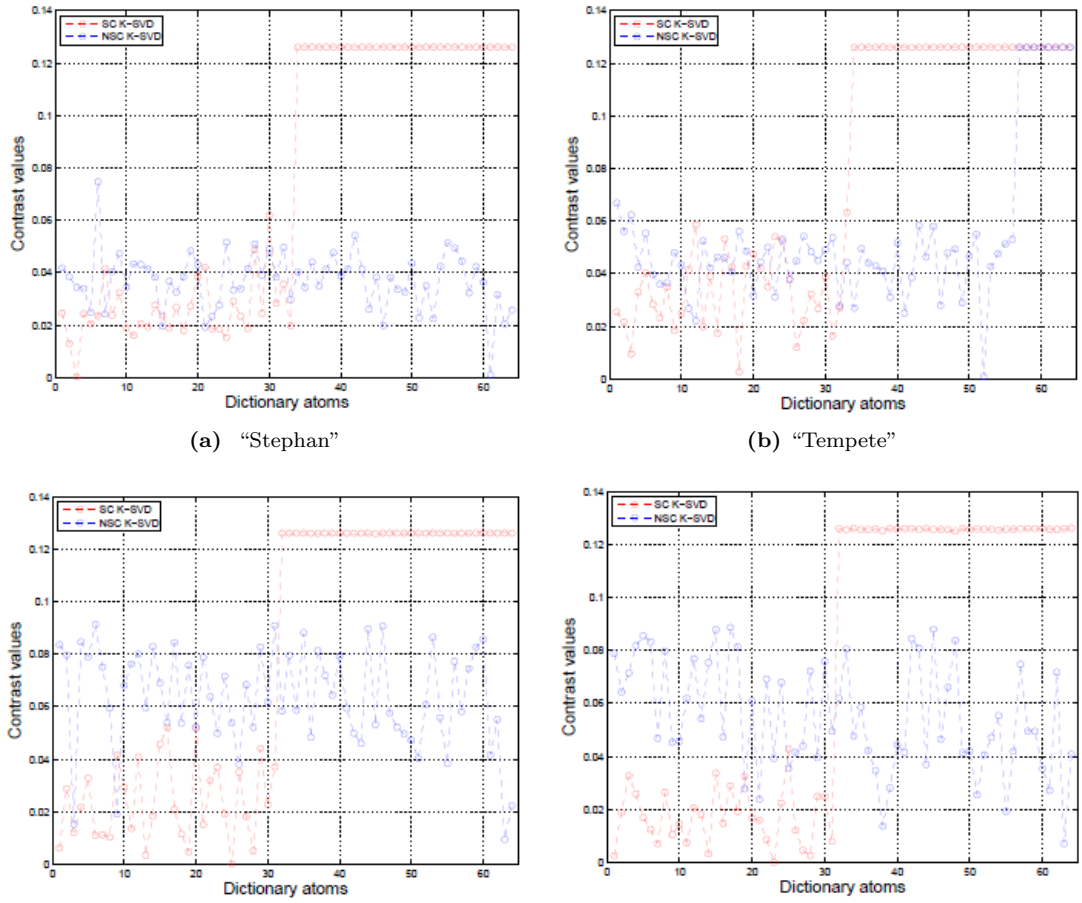
In Sec. 2.1 we gave a detailed explanation on semi-random dictionary initialization where we enforce allocation and separation of the dictionaries atoms into smooth and texture ones. As explained, the classification criteria we use is formulated via *Activity* norm in [3]. Thus, we further assess the spatial frequencies distributions for both dictionaries by looking at and analyzing *Activity* trend once the training is done. This is illustrated with Fig. 17. Whether we consider frames of the video sequence (Fig. 17a and Fig. 17b) or some conventional image (Fig. 17c and Fig. 17d) we can conclude that classical K-SVD scheme results in dictionaries which do not show any specific structural features in terms of how smooth and texture information are learned and allocated. In contrast, proposed design shows clear distinction between atoms that carry:

- Low spatial frequency: $Activity(\mathbf{d}_j)_{j=1}^{j=K/2} \leq A = 0.27$ (first $K/2$ atoms);
- High spatial frequency: $Activity(\mathbf{d}_j)_{j=K/2}^{j=K} > A = 0.27$ (last $K/2$ atoms);

thus successfully implementing this specific distribution as a built-in property of the *scalable* dictionary $\mathbf{D}_{sc}$ unlike the classical K-SVD scheme.

4.2 Contrast variation

Proper integration of the HVS sensitivity properties is done adequately if the proposed *scalable* design reinforces learning of the spatial high-frequency components (see Sec. 2.3) which represent regions of a high contrast variation [33, 34]. By examining in what ways atoms of $\mathbf{D}$ and $\mathbf{D}_{sc}$ differ in terms of their composition



(a) "Stephan" **(b)** "Tempete"

(c) "Boat" **(d)** "Peppers"

Figure 18. Atoms contrast variation levels via *std* function for the scalable and non-scalable K-SVD algorithm for the dictionaries of the two video sequences and two natural images.

structure (i.e., contrast variation) we verify the credibility of HVS properties modeling. This is taken forward by estimating contrast levels captured within the atoms during dictionary learning procedure. Assessment of the contrast levels is done by finding the standard deviation (*std*) over the atom's pixel intensity. Estimates are averaged over several dictionaries trained over the frames of the same video sequence or several times over the same image. The proposed computation is adopted from [37] where authors use *std* as a measure to estimate contrast image levels. Likewise, in Fig. 14 we depict standard deviation values for contrast levels of atoms both from $\mathbf{D}$ and $\mathbf{D}_{sc}$ for the same set of data as in the Sec. 3.1. We can notice distinct pattern for the contrast levels of the *scalable* "SC K-SVD" dictionary for all results likewise for *Activity* values shown in Sec. 4.1. Specifically, for the first $K/2$ atoms of each of the presented scalable dictionaries $\mathbf{D}_{sc}$ the contrast is considerably lower with some slight fluctuations (Fig. 18, notation "SC K-SVD") with highest contrast variation of 0.06. The remaining atoms reach quite high contrast levels with a steep jump up to around 0.13, creating a distinct threshold in distributed contrast variation over the all four $\mathbf{D}_{sc}$ dictionaries. The clear contrast variation borderline which clearly splits atoms in two groups, e.g., those with low and those with high contrast variation, is the final processing effect of the enforced semi-random initialization and regularization. In case of the conventionally K-SVD i.e., "NSC K-SVD" shown trend does not exist. Thus, this directly proves that proposed design complies with the characteristic of the HVS perception mechanism [29, 30] given that it is more efficient in extracting contrast information from the training images. In addition, this is significant since a proper visual understanding of the scene at hand [31, 32, 34] depends on how well contrast variations are captured with the image representational elements, that is atoms.

4.3 Noise distortion of the smooth and texture image patches

We posed an assumption in Sec. 2.5 that noise effects more smooth than texture image components. Specifically, oscillatory components of the scene i.e., texture exhibit regularity in terms of the frequency content that repeats to some extent over the image. Thus, noise which represents random signal (without any consistency in its change) should have a higher impact on image parts which do not exhibit periodic spatial variations i.e., smooth one. This is shown by estimating changes in *std* variation before and after noise is added to specific image blocks of smooth and texture areas. Several of these blocks are depicted in Fig. 19 where first row represents smooth and second texture image blocks of size 30×30. Given the five noise levels as in Sec. 3.3, in Tab. 9 we show how averaged *std* of the texture and smooth image patches varies before and after noise is added. Given the smooth group we can see rel-

Figure 19. Visual overview of the image patches size $30x30$ used for *std* noise impact analysis. First row represents smooth image content while seconds one depicts texture.

Table 9. *Std* variation assessment averaged over group of smooth and texture image blocks size 30×30.

Smooth	$\sigma = 0$	$\sigma = 20$	$\sigma = 40$	$\sigma = 60$	$\sigma = 80$	$\sigma = 100$
	6.67	21.42	35.82	57.59	70.44	80.22
Texture	$\sigma = 0$	$\sigma = 20$	$\sigma = 40$	$\sigma = 60$	$\sigma = 80$	$\sigma = 100$
	46.42	50.11	56.85	67.21	76.40	84.47

ative jumps of 14.75, 29.15, 50.92, 63.77, 73.55 for each noise level from the initial noise free level of *std* 6.67. In contrast, for texture areas this change is not that steep starting from noise free *std* of 46.42 with relative changes of 3.69, 10.43, 20, 79, 29.89 and 38.05. Conclusion follows that noise disturbs original smooth image content on a much larger scale than the texture areas.

5 Conclusion

This work introduces a design for learning dictionary for *scalable* image recovery by enforcing semi-random dictionary initialization and regularization of the K-SVD atom's update stage. To the best of our knowledge this problem has not been addressed before i.e., creating learned sparse representations for the purpose of the *scalable* image restoration. The proposed technique is evaluated over two different video test sequences, "Stephan" and "Tempete" and several conventional images. We demonstrate its practical utilization for dynamic data changing over time given single trained dictionary $\mathbf{D}_{sc}$. Mainly, three potential applications schemes are tested:*scalable* video recovery, *scalable* denoising and *scalable* CS. Interestingly, the proposed approach for learning dictionary for *scalable* image recovery, significantly outperforms or it is highly comparable with the classical K-SVD setting for the all aforementioned purposes: (i) best for 11.32 [dB]; (ii) achieves comparable results with best decreased computational demands for 7.3 times; (iii) best for all subsampling levels at around $3.03[dB]$). Future work will address joint design and optimization of the *scalable* CS sampling matrix and dictionary for *scalable* image recovery aiming for considerably decreasing current coherence level (i.e., 7.7). Overall, the proposed method can be successfully used for real-time *scalable* image/video display applications, where video streams, tailored to the needs of a divers user pool operating heterogeneous display equipment, are required.

REFERENCES

[1] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?," *Vision Research*, 37, 1997.

[2] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, vol. 51, no. 1, pp. 34-81, 2009.

[3] M. Elad, "Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing," *SPRINGER,* 2010.

[4] R. Rubinstein, A.M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of IEEE, Special Issue on Applications of Compressive Sensing and Sparse Representation*, vol. 98, no. 6, pp. 1045–1057, June 2010.

[5] D. L. Donoho, "Compressed sensing ," *IEEE Trans. on Information Theory*, vol. 52, pp. 1289-1306, Apr. 2006.

[6] G. Davis, S. Mallat, and Z. Zhang, "Adaptive time-frequency decompositions," *Opt. Eng*, vol. 33, no. 1, pp. 2183–2191, Jan. 1994.

[7] T, Cai and L. Wang, "Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Trans. Inform. Theory*, vol. 57, no. 7, pp. 4680–4688, 2011.

[8] D. L. Donoho, Y. Tsaig, and J.L. Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," *Technical Report*, March 2006.

[9] T. T. Do, L. Gan, N. Nguyen, and T. D. Tran, "Sparsity adaptive matching pursuit algorithm for practical compressed sensing," in Proc. of *The 42th Asilomar Conference on Signals, Systems, and Computers*, pp. 581-587, Oct. 2008.

[10] M. Aharon, M. Elad and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. Im. Proc.*, vol. 54, pp. 4311–4322, Dec. 2006.

[11] H. Lee, A. Battle, R. Raina, and A.Y. Ng, "Efficient Sparse Coding Algorithms," *Advances in Neural Information Processing Systems, MIT Press*, pp. 801-808, 2007.

[12] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," *Advances in Neural Information Processing Systems (NIPS08)*, pp. 1033-1040.

[13] K. Huang and S. Aviyente, "Sparse representation for signal classification," in *Adv. NIPS*, 2006.

[14] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 210-227, Feb. 2009.

[15] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *SIAM Multiscale Modelling and Simulation*, vol. 7, no. 1, pp. 214-241, Apr. 2008.

[16] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over Learned Dictionaries," *IEEE Trans. Image Proc.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[17] W. Dong, X. Li, L. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 52, pp. 457–464, June 2011.

[18] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image Super-Resolution via Sparse Representation," *IEEE Trans. on Image Processing*, vol. 19, no. 11, pp. 2861–2873, Nov. 2013.

[19] R. Zeyde, M. Elad and M. Protter, "On Single Image Scale-Up using Sparse-Representations," *7th International Conference on Curves and Surfaces, SMAI-AFA*, Avignon, France, June 2010 .

[20] J. Bobin, J.-L. Starck, M.J. Fadili, and Y. Moudden, "Sparsity, Morphological Diversity and Blind Source Separation," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2662–2674, 2007.

[21] M. Elad, J.-L Starck, D. Donoho and P. Querre, "Simultaneous Cartoon and Texture Image Inpainting using Morphological Component Analysis (MCA)," *Journal on Applied and Computational Harmonic Analysis ACHA*, vol. 19, pp. 340-358, Nov. 2005.

[22] J. Mairal, M. Elad and G. Sapiro, "Sparse representation for colour image restoration," *IEEE Transactions o Image Prcessing*, vol. 17, no. 1, pp. 53-69, Jan. 2008.

[23] J. Wright, Y. Ma, J. Mairal, G. Spairo, T. Huang, and S. Yan, "Sparse representations for computer vision and pattern recognition," *Proc. of IEEE, Special Issue on Applications of Compressive Sensing and Sparse Representation*, vol. 98, no. 6, pp. 1031–1044, June 2010.

[24] B. Begovic, V. Stankovic, L. Stankovic, "Learning Scalable Dictionaries With Application To Scalable Compressive Sensing," at *20th EUSIPCO, EURASIP conference (European Association for Signal, Speech, and Image Processing)*, Bucharest, Romania, Aug. 2012.

[25] B.Begovic, V. Stankovic, L. Stankovic, S. Cheng , "HVS Based Dictionary Learning for Scalable Sparse Image Representation ," at *46th Asilomar Conference on Signals, Systems and Computers* 2012.

[26] V. Stankovic, L. Stankovic, and S. Cheng, "Scalable compressive video," *ICIP-2011 IEEE Int. Conf, on Im. Proc.*, Brussels, Belgium , Sep. 2011.

[27] K. Baker, "Singular Value Decomposition Tutorial," *Available at www.cs.wits.ac.za/ michael/SVDTut.pdf*, 2005.

[28] P.C. Hansen, J.G. Nagy and D.P. O'Leary, "Fundamentals of Algorithms: Deblurring Images Matrices, Spectra and Filtering," *Siam* 2006.

[29] J. A. Ferwerda, "Elements of Early Vision for Computer Graphics," *IEEE Computer Graphics and Applications*, vol. 21, n. 5, pp. 22-33, Sep. 2001.

[30] R.C. Atkinson, "Stevens Handbook of Experimental Psychology," *John Wiley and Sons*, 2nd ed., New York, 1988.

[31] B. Wandell, "Foundations of Vision," *Sinauer Associates* 1995.

[32] J. Ferwerda, "Fundamentals of spatial vision," *In Applications of visual perception in computer graphics. Siggraph, Course Notes* 1998.

[33] F. Campbell and J. Robson, "Application of fourier analysis to the visibility of gratings," *Journal of Physiology*, vol. 197, pp. 551–566, 1968.

[34] E. Peli, E. L. Arend, M. G. Young and B.R. Goldstein, "Contrast sensitivity to patch stimuli: Effects of spatial bandwidth and temporal presentation," *Spatial vision*, vol. 7, pp. 1–14, 1993.

[35] O. A. Tun, M. Cadk, M. Karol and S. Hans-Peter, "Visually Significant Edges," *ACM Transactions on Applied Perception (TAP)* 2010.

[36] X. Gao, W. Lu, D. Tao, and X. Li, "Image quality assessment and human visual system," in *Proceedings of SPIE, Video Communications and Image Processing Conference*, vol. 7744, Huangshan, China, 2010.

[37] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Proc.*, vol. 13, pp. 600–612, April 2004.

[38] E. J. Candès and M. B. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, pp. 21-30, Mar. 2008.

[39] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Inf. Theory*, vol. 52, Feb. 2006.

[40] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 14-20, Mar. 2008

[41] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Mag.*, vol. 25, no. 2, pp. 83-91, March 2008.

[42] A. Levin and B. Nadler, "Natural image denoising: Optimality and inherent bounds,"in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011.

[43] M. Elad, "Optimized Projections for Compressed Sensing," *IEEE Tran. on Sig. Proc.*, vol. 55, pp. 5695-5702, Dec. 2007

[44] J. M. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Proc.*, vol. 18, no. 7, pp. 1395-1408, July 2009.

[45] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R.G. Baraniuk, "Compressive imaging for video representation and coding," *PCS-2006 Picture Coding Symposium*, vol. 52, pp. 1289-1306, Beijing, China , Apr. 2006.

[46] V. Stankovic, L. Stankovic, and S. Cheng, "Compressive video sampling," *Proc. EUSIPCO, Lausanne, Switzerland*, Aug. 2008.