# Structured Sparse Representation With Union of Data-Driven Linear and Multilinear Subspaces Model for Compressive Video Sampling

Yong Li, *Student Member, IEEE*, Wenrui Dai, *Member, IEEE*, Junni Zou, *Member, IEEE*, Hongkai Xiong, *Senior Member, IEEE*, and Yuan F. Zheng, *Fellow, IEEE*

***Abstract*—The standard compressive sensing (CS) theory can be improved for robust recovery with fewer measurements under the assumption that signals lie on a union of subspaces (UoS). However, the UoS model is restricted to specific types of signal regularities with predetermined topology for subspaces. This paper proposes a generalized model which adaptively decomposes signals into a union of data-driven subspaces (UoDS) for structured sparse representation. The proposed UoDS model leverages subspace clustering to derive the optimal structures and bases for the subspaces conditioned on the sample signals. For multidimensional signals with various statistics, it supports linear and multilinear subspace learning for compressive sampling. As an improvement for generic CS model, the basis which represents the sparsity of sample signals is adaptively generated via linear subspace learning method. Furthermore, a generalized model with multilinear subspace learning is considered for CS to avoid vectorization of high-dimensional signals. In comparison to UoS, the UoDS model requires fewer degrees of freedom for a desirable recovery quality. Experimental results demonstrate that the proposed model for video sampling is promising and applicable.**

***Index Terms*—Structured sparsity, compressive video sampling, union of data-driven subspaces, tensor subspace.**

## I. Introduction

SPARSITY is widely concerned in many areas such as statistics, machine learning and signal processing. A vector admits a sparse representation over a basis (or dictionary) if it can be represented by a linear combination of a few column vectors from the dictionary. In a certain sense, accompanied by a corresponding dictionary, sparsity has a close connection with the concept *subspace* spanned by its few column vectors. Recently, it has been proved that a trained dictionary is more effective than a predefined one (e.g., wavelets [1]) for many tasks such as image denoising [2]–[4], compressive sensing [5], [18] and classification tasks [6], [7]. The dictionary directly learned from input signals is shown to provide more adaptivity and sparsity than the off-the-shelf bases.

As an application of sparsity, compressive sensing (CS) is a desirable framework for signal acquisition [8]–[10]. CS attempts to acquire the dictionary-based sparse representation for unknown signals by randomly projecting them onto a space (observation) with much lower dimension. Recently, CS has been widely adopted in video acquisition and reconstruction [11]–[18]. In comparison to conventional video acquisition and compression approaches, CS can relieve the burden of video encoder by reducing the number of measurements to be sampled. At the decoder side, its recovery quality can be guaranteed with effective reconstruction methods based on a sparse representation with certain basis. Wakin *et al.* [11] first applied CS into video acquisition, which jointly made compressive sampling and reconstruction for video sequences with 3-D wavelet transform. Recognizing that compressive sampling was inadequate to the entire frame, a block-based CS (BCS) method [12]) utilized CS for non-overlapping blocks to exploit the local sparsity within the DCT domain. Later, temporal correlations were exploited to improve BCS. The distributed BCS framework [13] approximated each block by a linear combination of blocks in previous frames. Liu *et al.* [14] developed an adaptive CS strategy for blocks in various regions of independent motion and texture complexity. BCS-SPL framework [15] made smooth projected Landweber (SPL) reconstruction by incorporating motion estimation and compensation. Chen *et al.* [16] combined multi-hypothesis prediction with BCS-SPL to enhance the reconstruction performance. The approximate message passing (AMP) reconstruction based on the dual-tree complex wavelet transform was adopted in [17]. Liu *et al.* [18] improved the recovery performance of BCS framework by introducing Karhunen-Loève transform (KLT) basis in the decoder. To overcome the deficiency of sampling high-order signals, multidimensional CS techniques [28]–[32] have been developed for practical implementations. However, these methods assume that signals live in a single vector or tensor space, which ignore the structures within the sparse coefficients.

Manuscript received November 16, 2016; revised May 4, 2017; accepted June 10, 2017. Date of publication June 29, 2017; date of current version July 24, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xin Wang. The work was supported in part by the National Natural Science Foundation of China under Grant 61501294, under Grant 61622112, under Grant 61529101, under Grant 61472234, under Grant 61425011, in part by China Postdoctoral Science Foundation 2015M581617, and in part by Program of Shanghai Academic Research Leader 17XD1401900. *(Corresponding author: Hongkai Xiong.)*

Y. Li and H. Xiong are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: marsleely@sjtu.edu.cn; xionghongkai@sjtu.edu.cn).

W. Dai is with the Department of Biomedical Informatics, University of California, San Diego, CA 92093 USA (e-mail: wed004@ucsd.edu).

J. Zou is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: zou-jn@cs.sjtu.edu.cn).

Y. F. Zheng is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: zheng@ece.osu.edu).

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

Digital Object Identifier 10.1109/TSP.2017.2721905

To capture the underlying structures for recovery, the union of subspaces model [19]–[21] has been established to significantly reduce the number of measurements. In [19], a general sampling framework was studied, where sampled signals lived in a known union of subspaces with linear operators. Under the notion of block restricted isometry property (RIP), robust block sparse recovery with a mixed $\ell_2/\ell_1$ programming was developed for the UoS model in [21]. Blumensath [20] demonstrated that projected Landweber algorithm could recover signals from UoS for all Hilbert spaces and unions of such subspaces, as long as the sampling operator satisfied bi-Lipschitz embedding condition. The UoS model exploits the structural sparsity like tree-sparsity and block-sparsity [22], [23] to span subspaces with DCT and wavelet bases. However, it is restricted to signals with varying signal regularities, especially video sequences.

Furthermore, multi-linear subspaces (tensor subspaces) learning (MSL [33]–[35], [37]) was adopted under the assumption that high-dimensional signals live in a tensor product of subspaces. It is due to the fact that a predetermined vectorization of tensor data can obscure the statistical correlations among samples and discard important structural information. Tensor subspace analysis (TSA [33]) detected the intrinsic local geometric structures of the tensor space to generate a representation matrix for images. In the meantime, generalized low rank approximations of matrices (GLRAM [34]) iteratively made bi-directional linear projecting transform for feature extraction of images, which outperformed the traditional SVD-based methods. For efficient tensor subspace representation, multilinear principal component analysis (MPCA [35]) framework suggested feature extraction with dimensionality reduction to model a major part of input signals. Recently, L1-norm-based tensor analysis (TPCA-L1 [37]) replaced Frobenius norm and L2 norm with L1-norm to formulate the problem of tensor analysis with the robustness to outliers. Unfortunately, their performance for video sequences is degraded, as they failed to consider the temporal correlations with adaptive basis for each tensor subspace.

This paper proposes a generalized union of data-driven subspaces (UoDS) model based on linear or multi-linear subspaces for compressive sampling of multi-dimensional signals, especially video sequences. The contribution of this paper is twofold. Firstly, learning-based linear operators are developed to adaptively project patches onto a union of low-dimensional linear subspaces for structured sparse representation. Patches extracted from reference frames are fully sampled for generation of UoDS. They are adaptively categorized into finite clusters as estimation of the underlying subspaces with sparse subspace clustering. Incorporating linear subspace learning (LSL), the number and structures of these subspaces (clusters) are determined directly on the set of patches for learning. The non-reference frames are sparsely represented and recovered with the derived UoDS model. Its invertibility and stability of compressive sampling have been demonstrated under the block restricted isometric property (RIP) condition. Moreover, some ingredients, which affect the performance of the proposed model, are discussed including subspace clustering comparison, number and dimension of linear subspaces.

Furthermore, the UoDS model is generalized to the multilinear (tensor) subspaces to maintain the structural information of signals by avoiding vectorization. We first extend the CS theory to higher-order cases with data-driven basis to make signals compressible. The proposed compressive tensor sampling (CTS) method represents signals with the product of tensors and adaptively derives the basis for each tensor subspace with multilinear subspace learning (MSL). Moreover, it relieves the large-scale sensing matrices led by vectorization of high-dimensional signals and preserves the intrinsic structures in the original signals. Therefore, CTS is efficient and practical for sampling high-dimensional signals in comparison to previous schemes in literature.

The rest of the paper is organized as follows. Section II provides the background and motivation of UoDS. Section III proposes the UoDS model for compressive video sampling, including subspace clustering, linear subspace learning and stable recovery. Furthermore, the UoDS model is generalized to the multilinear subspaces in Section IV. Section V provides the experimental results for validation in terms of SR-PSNR performance and visual quality. Finally, Section VI concludes this paper.

## II. Definition and Motivation

To motivate the proposed UoDS model, we begin with a review of the compressive sensing models, including both single subspace model and union of subspace model.

### A. Single Subspace Model

Given an orthonormal basis $\Psi = \{\psi_i\}$, an $n$-dimensional signal $\mathbf{x} \in \mathbb{R}^n$ has a sparse representation in the form of $\mathbf{x} = \Psi\mathbf{c}$, where $\mathbf{c}$ is the representation vector with $k$ nonzero components. This fact implies that $\mathbf{x}$ lives in a $k$-dimension single subspace spanned by the $k$ basis vectors corresponding to the nonzero components in $\Psi$. Under the assumption of simple sparsity, this single subspace model is widely adopted in traditional compressive sampling. Thus, the measurement $\mathbf{y} \in \mathbb{R}^m$ is obtained by linear sampling $\mathbf{y} = \Phi\mathbf{x}$ based on a sensing matrix $\Phi \in \mathbb{R}^{m\times n}$ with $k < m \ll n$. Here, we define $m/n$ the sampling rate (SR). Denoting $\mathbf{A} = \Phi\Psi$, the linear sampling can be explicitly represented by $\mathbf{y} = \mathbf{A}\mathbf{c}$. When $\mathbf{A}$ is properly selected, $\mathbf{c}$ can be recovered exactly from $m = O(k \log(n/k))$ random measurements by solving $\ell_1$-norm minimization problem [8], [10]:

$$\min_{c} \|\mathbf{c}\|_1, \quad \text{s.t. } \mathbf{y} = \mathbf{A}\mathbf{c}. \tag{1}$$

### B. Union of Subspaces (UoS) Model

Instead of simple sparsity, union of subspaces model supposes a signal $\mathbf{x}$ to be lying on a union of subspaces [19]–[21]

$$\mathbf{x} \in \mathcal{U} \triangleq \bigcup_{\lambda\in\Lambda} \mathcal{S}_\lambda, \tag{2}$$

where $\Lambda$ is a list of indices, and $\mathcal{S}_\lambda$ is a subspace of Hilbert space $\mathcal{H}$ spanned by a predefined basis, e.g., DCT and wavelet basis. The UoS model considers block-sparse structure [22], [23] in the signals as shown in Fig. 1, where the representation vector $\mathbf{c}^T$ is segmented into $t$ blocks $[\mathbf{c}[1]^T, \cdots, \mathbf{c}[t]^T]$. $\mathbf{c}$ is called $k$-block-sparse, when at most $k$ blocks $\mathbf{c}[i] \in \mathbb{R}^{d_i}$ are nonzero. For a stable recovery, block restricted isometric property (Block-RIP) is imposed on $\mathbf{A}$, which guarantees to recover $\mathbf{c}$ with the convex formulation minimizing a mixed $\ell_2/\ell_1$ norm

$$\min_{\mathbf{c}} \|\mathbf{c}\|_{2,1}, \quad \text{s.t. } \mathbf{y} = \mathbf{A}\mathbf{c}, \tag{3}$$

where $\|\mathbf{c}\|_{2,1} \triangleq \sum_{i=1}^{t} \|\mathbf{c}_i\|_2$ with $\|\mathbf{c}_i\|_2 \geq 0$ for $1 \leq i \leq t$.

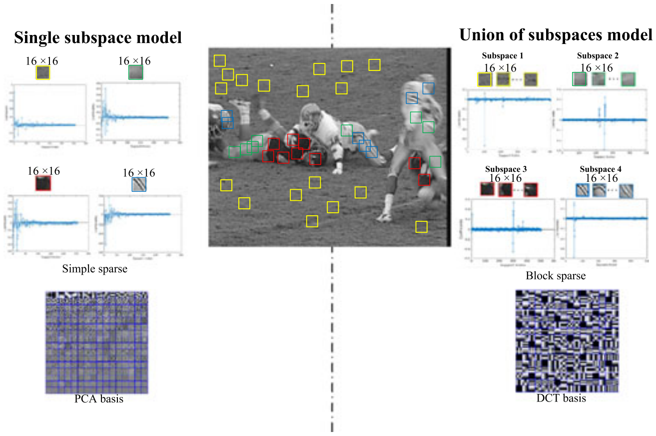



Fig. 1. The comparison of the single subspace model and union of subspaces model. As shown in this figure, the basic (simple) sparsity can be induced from data-driven basis based on single subspaces model. While based on union of subspaces model, the block (group) sparsity can be induced from predefined basis.

## C. Motivation

High-dimensional signals are characterized by various types of signal regularities. Taking video sequences for example, their spatial correlations within a frame are represented by regular textures and structures, while the temporal correlations are characterized by the motion trajectory of contents among neighboring frames. Consequently, most standard compressive video sampling methods treat the processing of video as the processing of a set of patches with single subspace model.

Instead of assuming high-dimensional signals lying on a single subspace, their sparse representation can be improved by considering intrinsic structures among representation vectors. Under the independent assumption for $\mathbf{c}$, the single subspace model obscures the dependencies and structures within the representation vector $\mathbf{c}$. Thus, it cannot utilize various types of prior knowledge for high-dimensional signals to make a sparse representation and stable recovery.

As an alternative, the UoS model projects such structures onto a union of low-dimensional linear subspaces while still preserving its information. It improves the applicability and efficiency of the single subspace model, especially for signals with noise and in the presence of under-sampling. However, it is still rigid to generate the geometry of a union of subspaces and describe the structure of each subspace with a predefined basis spanning. Thus, we propose a generalized model which adaptively endows structures and dependencies to sample high-dimensional signals with non-stationary statistics, especially for patch-based video sampling.

# III. Union of Data-driven Subspaces (UoDS) Model

## A. The Proposed Framework

Fig. 2 depicts the proposed framework of compressive video sampling with the UoDS model. For efficient sampling and stable recovery, a video sequence with group of pictures (GOP) is decomposed into the selected reference frames (RFs) and the remaining non-reference frames (NRFs). RFs are fully sampled and recovered with a trivial loss of quality. Therefore, they are suitable for generating a union of data-driven subspaces $\mathcal{U}^*$ to recover NRFs with learned structural information. In Section III-B, the UoDS model incorporates subspace clustering to adaptively group their patches based on their spatio-temporal structures. The overcomplete basis $\Psi^* = \{\Psi^*[i]\}$ for recovery is adaptively derived for each subspace with PCA-based learning methods, as shown in Section III-C.

On the other hand, NRFs are sparsely represented with a low sampling rate. For stable recovery, NRFs are reconstructed based on RFs to maintain the spatio-temporal consistency in video sequences. Hence, the trained overcomplete basis $\Psi^*$ is adopted to utilize the dependencies and structures within representation vectors $\mathbf{c}$. For each reconstructed patch $\mathbf{x}^*$, the block-sparse vector $\mathbf{c}^*$ is recovered by optimizing Eq. (15) in Section III-D. Consequently, $\mathbf{x}^*$ can be obtained based on the subspace consisting of patches with similar structures in $\mathcal{U}^*$. A detailed description of the proposed UoDS model can be found in Algorithm 1 in Section III-D, which considers both sampling and reconstruction for the UoDS model.

## B. Subspace Clustering for UoDS

Denote $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_K]$ the set of $K$ vectorized patches extracted from RFs. In this section, we incorporate sparse subspace clustering (SSC [26]) to segment $\mathbf{X}$ into $t$ clusters that naturally correspond to the underlying subspaces. SSC is suitable for training-based recovery, as it is rooted on the fact that each point in the union of subspaces has a sparse representation based on a trained dictionary constrained by the self-expressiveness property over the training set. Consequently, the representation vector $\mathbf{c}_i$ of the vectorized patch $\mathbf{x}_i \in \mathbb{R}^n$ can be obtained over $\mathbf{X}$

$$\min_{\mathbf{c}_i} \|\mathbf{c}_i\|_1 \quad \text{s.t. } \mathbf{x}_i = \mathbf{X}\mathbf{c}_i,\ \mathbf{c}_i[i] = 0. \tag{4}$$

Here, the extracted patches are overlapped to guarantee the smoothness of reconstructed frames. The non-zero coefficients of $\mathbf{c}_i$ correspond to patches from the same subspace. Fig. 3 provides an example for the generation of representation vector, where $\mathbf{c}_{122}$ is obtained over $\mathbf{X}$ with $K = 1505$. It shows that the structure of $\mathbf{x}_{122}$ is similar to those of $\mathbf{x}_{34}$, $\mathbf{x}_{78}$, $\mathbf{x}_{166}$, and $\mathbf{x}_{210}$, which implies that they are lying on the same subspace. Eq. (4) can be rewritten in matrix form by collecting all the patches in $\mathbf{X}$

$$\min_{\mathbf{C}} \|\mathbf{C}\|_1 \quad \text{s.t. } \mathbf{X} = \mathbf{X}\mathbf{C},\ diag(\mathbf{C}) = 0, \tag{5}$$

where $\mathbf{C} \triangleq [\mathbf{c}_1, \cdots, \mathbf{c}_K] \in \mathbb{R}^{K\times K}$ is a matrix of coefficients to generate the balanced similarity graph.

For a valid representation of the similarity, a weighted balanced similarity graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ is established based on the normalized columns $\mathbf{c}_i/\|\mathbf{c}_i\|_\infty$ of $\mathbf{C}$, where $\mathcal{V}$ is the set of vertices corresponding to the $K$ vectorized patches $\mathbf{x}_1, \cdots, \mathbf{x}_K$, and $\mathcal{E}$ is the set of directed edges $(v_i, v_j)$ indicating that $\mathbf{x}_j$ is one of the vectorized patch in the sparse representation of $\mathbf{x}_i$. Taking Fig. 3 for example, since $\mathbf{x}_{122} = 0.39\,\mathbf{x}_{78} + 0.22\,\mathbf{x}_{166} + \cdots$, there exist two edges $(v_{122}, v_{78})$ and $(v_{122}, v_{166})$ with weights $\mathbf{c}_{122}[78] = 0.39$ and $\mathbf{c}_{122}[166] = 0.22$, respectively. To ensure that $\mathcal{G}$ is balanced, we adopt the adjacency matrix $\mathbf{W} = |\mathbf{C}| + |\mathbf{C}|^T \in \mathbb{R}^{K\times K}$ with its elements $w_{ij} = |\mathbf{c}_i[j]| + |\mathbf{c}_j[i]|^T$. As shown in [24], each connected component in $\mathcal{G}$ suggests the vertices representing the vectorized patches in the same subspace. Fig. 4 provides an example of the similarity graph with four connected components

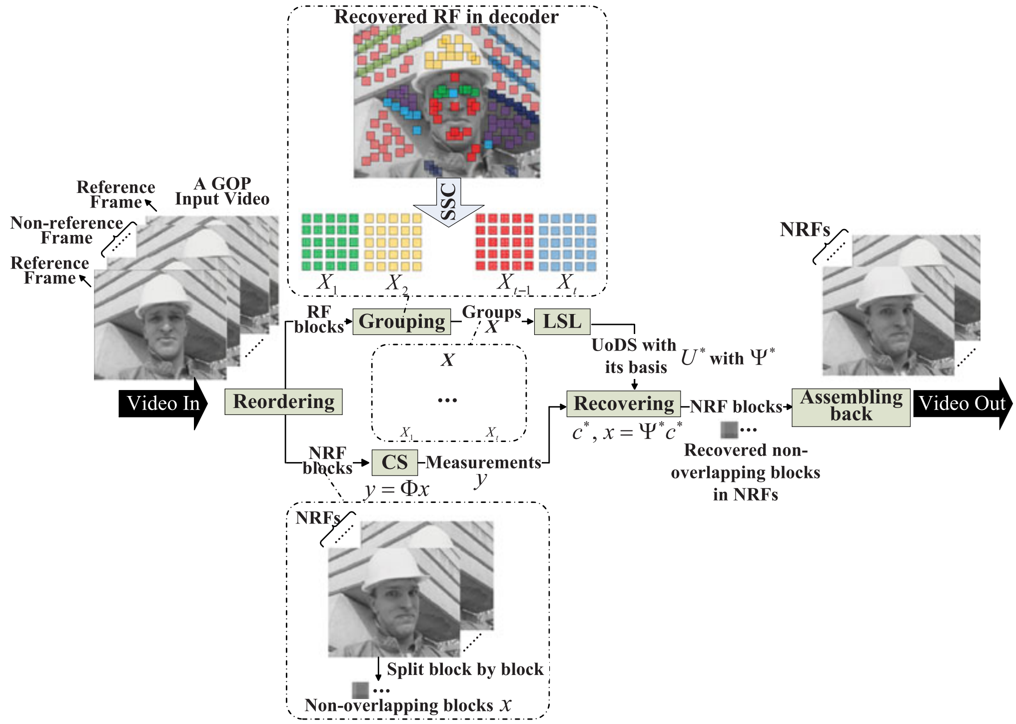



Fig. 2. The proposed union of data-driven subspaces (UoDS) model for compressive video sampling.

**Algorithm 1:** UoDS model for compressive video sampling and reconstruction.

**Task:** Sample and reconstruct a video with each group of $N$ frames $\{X^i\}, i = 1, \cdots, N$.

**Initialization:** Generate a random Gaussian matrix $\Phi \in \mathbb{R}^{m\times n}$, set $X^1$ as RF and $X^2, \cdots, X^N$ as NRFs.

**Sampling:** Fully sample non-overlapped blocks $\mathbf{x}^1$ in $\mathbf{X}^1$, and compressively sample those blocks in $\mathbf{X}^2, \cdots, \mathbf{X}^N$ with Eq. (13).

**Reconstruction:**

*Step 1:* Reconstruct each block $\hat{\mathbf{x}}^1$ to obtained recovered RF $\hat{\mathbf{X}}^1$

*Step 2:* Do SSC to the patches set $\mathbf{X}$ of $\hat{X}^1$ and clustering $X$ into $t$ clusters $X_1, \cdots, X_t$

*Step 3:* **for** $j = 1$ to $t$ **do**
Obtain $\Psi_j^*$ for $\mathcal{S}_j^*$ by LSL over $\mathbf{X}[j]$ with Eq. (8) or (9);
**end for**

*Step 4:* Concatenate $\Psi_j^*$ to form the dictionary $\Psi^* = [\Psi_1^*, \cdots, \Psi_t^*]$

*Step 5:* **for** $i = 2$ to $N$ **do**
Recover the block-sparse vector $c^*$ by Eq. (15) and reconstruct each non-overlapped block of $X^i$ by $x = \Psi^* c^*$,
then assemble them back to form the recovered NKF $\hat{X}^i$
**end for**

**Return** $\hat{X}^1, \cdots, \hat{X}^N$

for the UoDS model. It shows that vectorized patches in one connected component have similar structures.

Denote $\mathbf{L} = \mathbf{D} - \mathbf{W}$ the Laplacian matrix of $\mathcal{W}$, where $\mathbf{D}$ is a diagonal matrix with $D_{ii} = \sum_j w_{ij}$. We leverage spectral graph theory to segment $\mathbf{X}$ by applying $K$-means to the eigenvectors of the Laplacian matrix $\mathbf{L}$. The connected components of $\mathcal{G}$ can be determined from the eigenspace of the zero eigenvalue. Based on the $t$ eigenvectors corresponding to $t$ smallest eigenvalues, $\mathbf{X}$ is segmented into $t$ clusters $\{\mathbf{X}[i]\}_{i=1}^t$ with $K$-means. For $\mathbf{X}$ drawn from the underlying subspaces, each cluster corresponds to a low-dimensional linear subspace $\mathcal{S}_i^*$, as the similarity graph $\mathcal{G}$ has the $t$ connected components corresponding to these subspaces. Therefore, $\mathcal{U}^*$ is a union of low-dimensional linear subspaces by collecting all $t$ clusters $\mathcal{S}_i^*$.

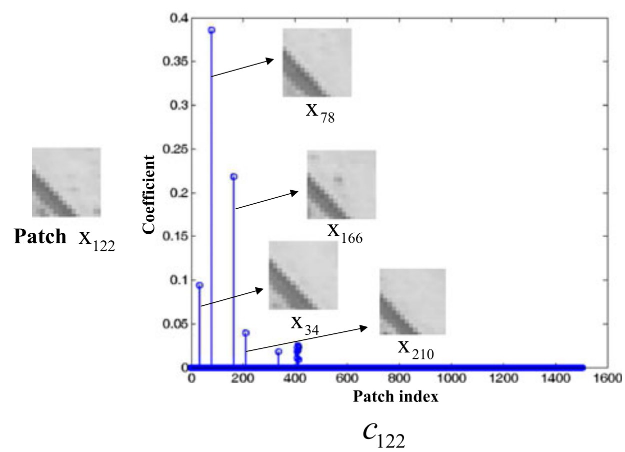



Fig. 3. The representation vector $\mathbf{c}_{122}$ of $16 \times 16$ patch $\mathbf{x}_{122}$ over a training set $\mathbf{X}$ of $K = 1505$ patches extracted from the first frame of *Foreman* sequence.

$$\mathbf{x} \in \mathcal{U}^* \triangleq \bigcup_{i=1}^{t} \mathcal{S}_i^*. \tag{6}$$

Remarkably, the number of the underlying subspaces can be estimated as the number of zero eigenvalues of $\mathbf{L}$. Thus, an

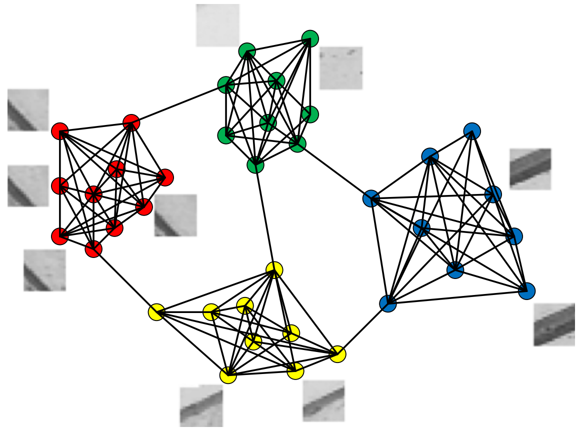


Fig. 4. The balanced similarity graph $\mathcal{G}$ with four connected components, where each subspace in $\mathcal{U}^*$ forms one component in $\mathcal{G}$.

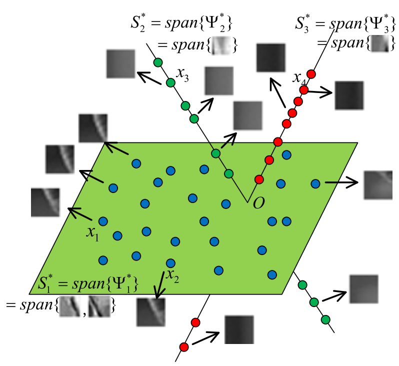



Fig. 5. Subspace clustering for the UoDS model. Vectorized patches are segmented into three low-dimensional linear subspaces, where $\mathcal{S}_1^*$ is a bi-dimensional plane and the dimension of $\mathcal{S}_2^*$ and $\mathcal{S}_3^*$ are one. For example, blue points like $\mathbf{x}_1$ and $\mathbf{x}_2$ live in $\mathcal{S}_1^*$ spanned by basis $\Psi_1^*$, and green and red points like $\mathbf{x}_3$ and $\mathbf{x}_4$ live in $\mathcal{S}_2^*$ spanned by basis $\Psi_2^*$ and $\mathcal{S}_3^*$ spanned by basis $\Psi_3^*$, respectively.

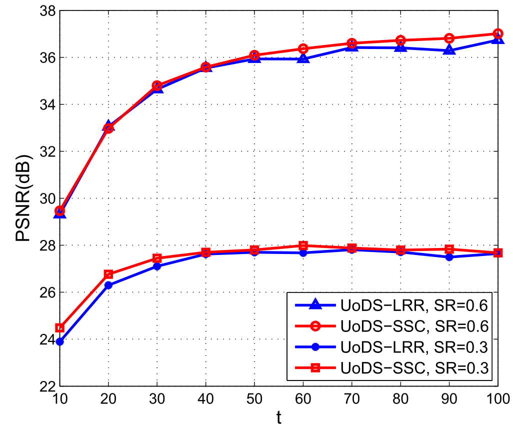
adaptive clustering of vectorized patches can be achieved, even though the number of their underlying subspaces are unknown. Fig. 5 shows a conceptual diagram for clustering the vectorized patches into three low-dimensional linear subspaces. We first solve Eq. (4) to each vectorized patch $x_i$ and thus solve Eq. (5) to $X$. Later, we can derive a balanced similarity graph $G$ similarly as shown in Fig. 4 but with three underlying connected components. Finally, we employ K-means algorithm with $K = 3$ to the eigenvectors of the Laplacian matrix $L$ of the balanced similarity graph $G$, and we get the final clustering result as shown in Fig. 5 with blue point $x_1, x_2$ are clustered together into common underlying data-driven subspace $\mathcal{S}_1^*$ spanned by basis $\Psi_1^*$ with two basis vectors, green and red points like $x_3$ and $x_4$ into $\mathcal{S}_2^*$ spanned by $\Psi_2^*$ with one basis vector and $\mathcal{S}_3^*$ spanned by $\Psi_3^*$ with one basis vector, respectively. Because the texture complexity for patches $x_1, x_2$ is larger than $x_3$ and $x_4$, the dimension of $\mathcal{S}_1^*$ is larger than $\mathcal{S}_2^*$ and $\mathcal{S}_3^*$.

*1) Subspace Clustering Comparison:* We compared SSC with another popular subspace clustering method, low-rank representation (LRR) [25]. Different from SSC, the nuclear norm $\|\mathbf{C}\|_*$ (sum of singular values) is used to minimize the rank of $\mathbf{C}$ in LRR

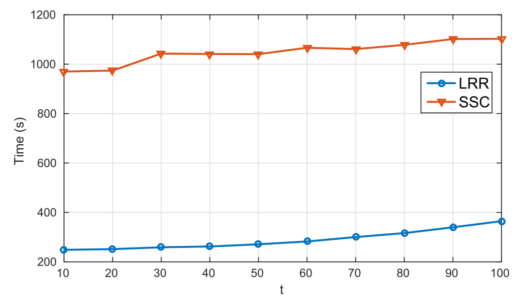
$$\min_{\mathbf{C}} \|\mathbf{C}\|_* \quad \text{s.t. } \mathbf{X} = \mathbf{XC},\ diag(\mathbf{C}) = 0. \tag{7}$$

For simplicity, we use UoDS-SSC and UoDS-LRR for the proposed UoDS incorporated with SSC and LRR, respectively. Fig. 6 compares UoDS-SSC and UoDS-LRR in terms of recovery performance. It is obvious that UoDS-SSC outperforms UoDS-LRR under different numbers of subspaces. SSC derives "cleaner" latent matrix $C$ in comparison to LRR. Thus, SSC can yield more accurate clustering results when $t$ is known, which performs better to fit the signals with the derived union of data-driven subspaces $\mathcal{U}^*$.

*2) Number of Subspaces Discussion:* Furthermore, we studied the number $t$ of subspaces in the union $\mathcal{U}^*$ for its effect on the accuracy of the UoDS model. As shown in Fig. 6, when the number of subspaces increases, the recovery accuracy increases rapidly at first and then slightly fluctuates around a certain maximum value. It is because that $t$ affects the distribution of these data-driven subspaces in $\mathcal{U}^*$. When $t$ is small, different underlying subspaces tend to merge into one cluster, which leads to a blend of structural information in these subspaces. Thus, this inaccurate clustering would produce a poor recovery accuracy. On the other hand, a large $t$ makes these subspaces independent as we assume. However, the clustering accuracy will be suppressed and the computational complexity will increase with the growth of $t$.




(a) PSNR (dB)




(b) Training speed (sec/GOP)

Fig. 6. The performance comparison between UoDS-SSC and UoDS-LRR for *Football* sequence with various numbers of subspaces $t$. UoDS-SSC and UoDS-LRR denote the proposed UoDS incorporated with SSC and LRR, respectively. SR denotes sampling rate.

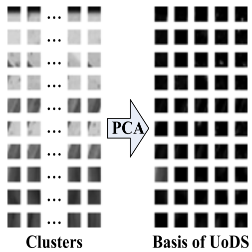



Fig. 7. PCA-based learning for bases of 10 clusters. Each cluster corresponds to a subspace and its dimension is fixed to 5. PCA is performed for each cluster separately.

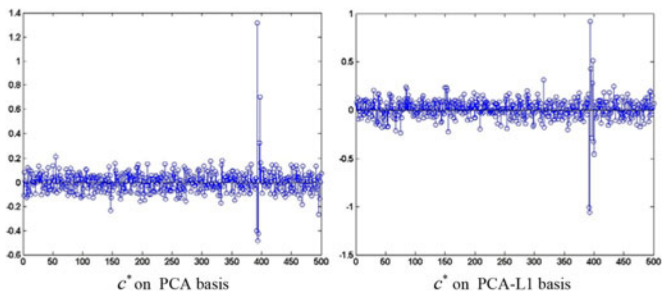



Fig. 8. Representation vector $\mathbf{c}^*$ obtained based on bases trained by PCA and PCA-L1. Dimension of each subspace is 10.

### C. Linear Subspace Learning for UoDS

When $\mathcal{U}^*$ is obtained, the basis for its subspaces can be derived based on the clustered training set $\mathbf{X} = \{\mathbf{X}[i]\}_{i=1}^t$. Principal component analysis (PCA) is leveraged for each cluster $\mathbf{X}[i]$ to learn the basis $\Psi_i^*$ for corresponding linear subspace $\mathcal{S}_i^*$ independently, as shown in Fig. 7. Therefore, $\Psi^* = [\Psi_1^*, \Psi_2^*, \ldots, \Psi_t^*]$ is the derived basis for $\mathcal{U}^*$. This fact implies that $\Psi_i^* \in \mathbb{R}^{n\times d_i}$ spans the $d_i$-dimensional subspace $\mathcal{S}_i^*$. In comparison to the UoS model, dimensions and bases of these subspaces are conditioned on structures embodied by their patches. Fig. 5 shows that patches from plain backgrounds lie in a lower-dimensional subspace than those patches from regions with dramatic change of details. Furthermore, $\Psi^*$ is nonlocal, as it is learned from the complete frame rather than a fixed local window of pixels.

To sufficiently capture the patch-based structures, $\Psi^* \in \mathbb{R}^{n\times r}$ is an overcomplete basis with $r = \sum_{i=1}^t d_i > n$. It is solved based on singular value decomposition (SVD) $\mathbf{X}[i] = \Psi_i^* \Sigma \mathcal{V}^T$

$$\Psi_i^* = \arg\max_{\Psi_i} \|\Psi_i^T \mathbf{X}[i]\|_2, \quad \text{s.t. } \Psi_i^T\Psi_i = \mathbf{I}_{d_i}. \tag{8}$$

Instead of classical PCA algorithm, we utilize the PCA-L1 algorithm [27] to obtain a subspace that is robust to outliers as well as invariant to rotations

$$\Psi_i^* = \arg\max_{\Psi_i} \|\Psi_i^T \mathbf{X}[i]\|_1 = \arg\max_{\psi_{iq}} \sum_{j=1}^{l_i} |\psi_{iq} x_{ij}|$$
$$\text{s.t. } \Psi_i^T\Psi_i = I_{l_i},\ q = 1, \ldots, d_i, \tag{9}$$

where $q$ is the index of vectors in $\Psi_i^* = [w_{i1}^*, w_{i2}^*, \ldots, w_{id_i}^*]$, and $l_i$ is the number of patches in $\mathbf{X}[i]$. The (sub-)optimal solution to $\Psi_i^*$ is obtained by the greedy algorithm. Fig. 8 shows the representation vectors of the same signal lying in the UoDS spanned by PCA and PCA-L1 basis, respectively.

*1) Subspace Dimension Selection:* Since dimensions of linear subspaces depend on the structures of their patches, they should be determined adaptively and independently. In the UoDS model, we employ the average gradient to measure the texture complexity for patches in each subspace, which is associated with the cost of representing the structures like textures and edges. Linear subspaces are classified into three categories with refined candidate dimensions for various texture complexity. The classification is based on adaptive calculation of average gradient $\overline{g}_i$ for $\mathbf{X}[i]$

$$\overline{g}_i = \frac{1}{K_i}\sum_{j=1}^{K_i} g(\mathbf{x}_j),\ \mathbf{x}_j \in \mathbf{X}[i], \tag{10}$$

where $K_i$ is the number of patches in cluster $\mathbf{X}[i]$, and $g(\mathbf{x}_j)$ is the average gradient for patch $\mathbf{x}_j$. Assuming that $\mathbf{x}$ is the vectorized version of patch $\mathbf{F} \in \mathbb{R}^{a\times a}$, we can obtain $g(\mathbf{x})$ by

$$g(\mathbf{x}) = \frac{1}{(a-1)^2}\sum_{i=1}^{a-1}\sum_{j=1}^{a-1}\sqrt{\frac{(F_{i,j}-F_{i+1,j})^2+(F_{i,j}-F_{i,j+1})^2}{2}}. \tag{11}$$

Therefore, the dimension of each subspace is selected according to the three categories.

$$d_i = \begin{cases} d_h, & T_2 \le \overline{g}_i \\ d_m, & T_1 \le \overline{g}_i < T_2\,, \\ d_l, & T_1 > \overline{g}_i \end{cases} \tag{12}$$

where $T_1$ and $T_2$ are related with the minimum and maximum of the average gradients.

### D. Stable Recovery

Stable recovery is significant for patches with a sparse representation in NRFs. Linear sampling in the UoDS model is based on the block diagonal measurement matrix $\mathbf{A}^*$

$$\mathbf{y} = \Phi\mathbf{x} = \Phi\Psi^*\mathbf{c}^* = \mathbf{A}^*\mathbf{c}^*, \tag{13}$$

where $\Phi$ is an i.i.d. random sensing matrix, $r = \sum_{i=1}^t d_i$, and

$$\mathbf{A}^* = \begin{bmatrix} \mathbf{A}_1^*, & & & \\ 0 & \mathbf{A}_2^*, & & \\ & \ddots & \ddots & \\ & & 0 & \mathbf{A}_t^* \end{bmatrix}.$$

The UoDS model inherits the merit of UoS model, which implies that input signals have a block-sparse representation over $\Psi^*$. When these subspaces are disjoint or independent, $\mathbf{c}^*$ is a 1-block-sparse vector that is more sparser than $\mathbf{c}$ in Eq. (3). Fig. 8 shows the sparsity of the proposed model for one selected block in $\mathbf{c}^*$. For a complete evaluation, the uniqueness and stability conditions are derived in this section.

Denote $\mathcal{S}_{ij}^* = \mathcal{S}_i^* \bigcup \mathcal{S}_j^*$ the convex hull of a set of two data-driven subspaces $\mathcal{S}_i^*$ and $\mathcal{S}_j^*$ and $k_{max} = \max_{i\neq j}\dim(\mathcal{S}_{ij}^*)$ its maximum dimension. In Proposition 1, we demonstrate the invertibility of the linear sampling operator $\Phi : \mathcal{U}^* \to \mathbb{R}^m$.

*Proposition 1:* Linear sampling operator $\Phi : \mathcal{U}^* \to \mathbb{R}^m$ is invertible for $\mathcal{U}^*$ if $m \ge k_{max}$.

*Proof:* Please refer to Appendix A. ∎

Proposition 1 provides the minimal number of samples required to guarantee a stable reconstruction for the UoDS model. It suggests that $k_{max}$ samples are required for invertible representation based on the UoDS model. Substituting $\Phi$ and $\mathbf{x}$ with $\mathbf{A}^*$ and the $k$-block sparse vector $\mathbf{c}^*$, we can draw Proposition 2.

*Proposition 2:* The measurement matrix $\mathbf{A}^*$ is stable for every $2k$-block sparse vector $\mathbf{u}$ if and only if there exists $C_1 > 0$ and $C_2 < \infty$ such that

$$C_1 \|\mathbf{u}\|_2^2 \leq \|\mathbf{A}^*\mathbf{u}\|_2^2 \leq C_2 \|\mathbf{u}\|_2^2, \tag{14}$$

where $\mathbf{u} = \mathbf{c}_1^* - \mathbf{c}_2^*$.

*Proof:* Please refer to Appendix A. ∎

Proposition 2 ensures the stable sampling and recovery of $\Phi$ in the UoDS model. In fact, it imposes the classical conditioning requirements on Gram matrices $\mathbf{A} = (< \psi_i, \phi_j >)_{1\leq i\leq m, 1\leq j\leq n}$ between the sets of vectors $\Phi = \{\phi_i\}_{i=1}^m$ and $\Psi = \{\psi_j\}_{j=1}^n$. Thus, $C_1$ and $C_2$ are the tightest stability bounds related with the minimum and maximum singular values of $\mathbf{A}$, respectively.

Under the invertibility of linear sampling, convex optimization is considered for efficient exact recovery. Theorem 1 proves that there exists a unique block-sparse vector $\mathbf{c}$ when $\mathbf{A}$ satisfies the block-RIP condition with constant $\delta + 2k < 1$.

*Theorem 1:* If $\mathbf{A}^*$ satisfies the block-RIP condition with $\delta_{2k} \leq \sqrt{2} - 1$, the vector $\mathbf{c}^*$ of Eq. (15) can be determined according to the convex second-order cone programming (SOCP) [21].

$$\min_{\mathbf{c}^*} \|\mathbf{c}^*\|_{2,\mathcal{I}} \quad \text{s.t. } \mathbf{y} = \mathbf{A}^*\mathbf{c}^*, \tag{15}$$

where $\mathcal{I}$ represents the group index set.

*Proof:* Please refer to Appendix B. ∎

Theorem 1 implies that the unique solution to Eq. (15) exists as long as $\delta_{2k}$ is small enough. SOCP determination is desirable in comparison to standard CS results, as it considers the block structure of $\mathbf{c}$ explicitly and adaptively. In practice, we reconstruct the block-sparse vector $\mathbf{c}^*$ by solving Eq. (15) with group-BP algorithm [39].

## IV. Tensor Generalization for UoDS

In Section III-C, PCA utilizes linear subspace learning (LSL) to deal with the vectorized tensor data (e.g., two-order tensors like image). For standard compressive sampling, its efficiency is degraded by the large-scale sampling matrix generated by the vectorization of high-dimensional signals, e.g., images and videos. For example, it is less practical to store and transmit the sampling matrix with size $1000 \times 10000$ if we sample and reconstruct an image with size $100 \times 100$ at $SR = 0.1$. Besides, reshaping high-order tensors into vectors destroys the intrinsic structure and correlation in the original tensors, making the representation less compact or useful. Therefore, we generalize it to the compressive tensor sensing (CTS) with the extension from LSL to MSL. The proposed CTS algorithm directly samples the tensor (signal) in each of its mode under the assumption that the tensor lives in a lower-dimensional tensor subspace. The data-driven basis is obtained by using multilinear subspace learning (MSL).

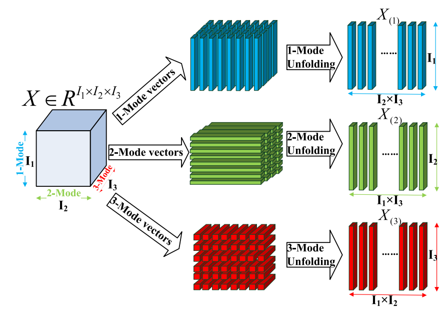



Fig. 9. Visual illustration of the $n$-mode unfolding of Tensor $\mathbf{X} \in \mathbb{R}^{I_1\times I_2\times I_3}$, $X_{(n)}$ is the $n$-mode unfolding matrix, $n = 1, 2, 3$.

### A. Generalized Formulation

Given arbitrary tensor $\mathbf{X}$, it can be decomposed according to standard multilinear algebra.

$$\mathbf{X} = \Theta \times_1 \Psi^{(1)} \times_2 \Psi^{(2)} \times \cdots \times_N \Psi^{(N)}, \tag{16}$$

where $\Theta$ is the tensor representation for $\mathbf{X}$, $\Psi^{(n)} = (\psi_1^{(n)} \psi_2^{(n)} \cdots \psi_{I_n}^{(n)})$ is an orthogonal $I_n \times I_n$ basis matrix, and $\times_n$ denotes the $n$-mode product of a tensor by a matrix. Given the matrix $\Psi$, the $n$-mode product of $\mathbf{X}$ is defined as

$$\Theta = \mathbf{X} \times_n \Psi \Leftrightarrow \Theta_{(n)} = \Psi \mathbf{X}_{(n)}, \tag{17}$$

where $\mathbf{X}_{(n)}$ is the $n$-mode unfolding matrix of tensor $\mathbf{X}$. Fig. 9 illustrates the $n$-mode unfolding of tensor $\mathbf{X}$ with $n \leq 3$.

MSL methods seek a tensor subspace that captures most of the variation in the original tensor objects and project $\mathbf{X}$ from the original tensor space $\mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \cdots \otimes \mathbb{R}^{I_N}$ onto the tensor subspace $\tilde{\mathcal{S}} = \mathbb{R}^{P_1} \otimes \mathbb{R}^{P_2} \otimes \cdots \otimes \mathbb{R}^{P_N}$, $P_n \ll I_n$. The projection $\tilde{\Theta} = \mathbf{X} \times_1 \tilde{\Psi}^{(1)^T} \times_2 \tilde{\Psi}^{(2)^T} \times \cdots \times_N \tilde{\Psi}^{(N)^T}$, $\tilde{\Psi}^{(n)}$ is an $I_n \times P_n$ basis matrix for the $n$-mode linear space $\mathbb{R}^{I_n}$. If a tensor $\mathbf{X}$ lies in a tensor subspace $\tilde{\mathcal{S}}$, the vectors corresponding to its modes admit sparse representation over the basis of corresponding modes. The data-driven orthonormal basis matrices $\{\tilde{\Psi}^{(n)} \in \mathbb{R}^{I_n\times P_n}, n = 1, \ldots, N\}$ can be learned based on the set of $K$ available tensor objects $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_K\}$ via MPCA [35] by solving the following problem.

$$\{\tilde{\Psi}^{(n)}, n = 1, \ldots, N\} = \arg \max_{\tilde{\Psi}^{(1)},\ldots,\tilde{\Psi}^{(N)}} \sum_{i=1}^K \|\tilde{\Theta}_i - \bar{\Theta}\|_F^2, \tag{18}$$

where $\tilde{\Theta}_i = \mathbf{X}_i \times_1 \tilde{\Psi}^{(1)^T} \times_2 \tilde{\Psi}^{(2)^T} \times \cdots \times_N \tilde{\Psi}^{(N)^T}$ and $\bar{\Theta} = (1/K)\sum_{i=1}^K \tilde{\Theta}_i$. The n-mode basis matrix $\tilde{\Psi}^{(n)}$ consists of the $P_n$ eigenvectors corresponding to the largest $P_n$ eigenvalues of the matrix

$$\Lambda^{(n)} = \sum_{i=1}^K (X_{i(n)} - \bar{X}_{(n)}) \cdot \tilde{\Psi}_{\Lambda^{(n)}} \cdot \tilde{\Psi}_{\Lambda^{(n)}}^T \cdot (X_{i(n)} - \bar{X}_{(n)})^T \tag{19}$$

where $\tilde{\Psi}_{\Lambda^{(n)}} = \tilde{\Psi}^{(1)} \otimes \cdots \otimes \tilde{\Psi}^{(n-1)} \otimes \tilde{\Psi}^{(n+1)} \otimes \cdots \otimes \tilde{\Psi}^{(N)}$. The higher-order singular value decomposition (HOSVD)[36] is used in multilinear principal component analysis.

Later, we define the compressive tensor sampling and recovery for arbitrary signals.

*Definition 1 (Compressive Tensor Sampling):* Given an $N$ th-order tensor $\mathbf{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ who is $K_1$-$K_2$-…-$K_N$-

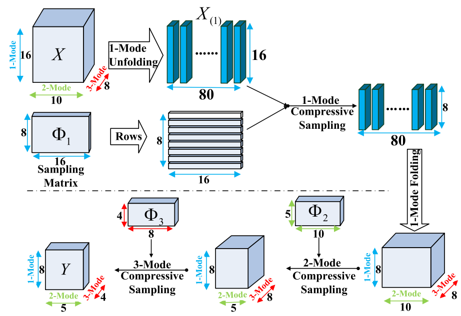



Fig. 10. 1, 2, 3-mode compressive sampling by projecting tensor $\mathbf{X} \in \mathbb{R}^{16\times10\times8}$ onto tensor $\mathcal{Y} \in \mathbb{R}^{8\times5\times4}$, where $\mathcal{Y} = \mathbf{X} \times_1 \Phi_1 \times_2 \Phi_2 \times_3 \Phi_3$ with sampling matrices $\Phi_1$, $\Phi_2$, and $\Phi_3$.

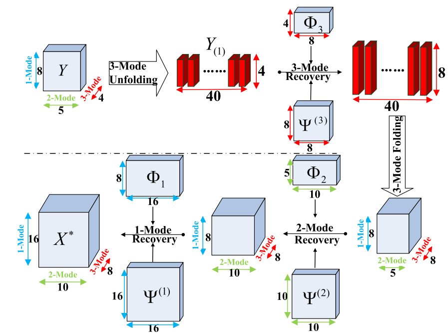



Fig. 11. 1, 2, 3 mode recovery of tensor $\mathbf{X}^* \in \mathbb{R}^{16\times10\times8}$ from tensor $\mathcal{Y} \in \mathbb{R}^{8\times5\times4}$, where $\mathcal{Y} = \mathbf{X} \times_1 \Phi_1 \times_2 \Phi_2 \times_3 \Phi_3$ with sampling matrices $\Phi_1$, $\Phi_2$, and $\Phi_3$, and basis $\Psi^{(1)}$, $\Psi^{(2)}$, and $\Psi^{(3)}$ for three modes.

sparse, the compressive tensor sampling (CTS) is defined by

$$\begin{aligned}\mathcal{Y} &= \hat{\Phi}\mathbf{X} = (\Phi_1 \otimes \Phi_2 \otimes \cdots \otimes \Phi_N)\mathbf{X}\\ &= \mathbf{X} \times_1 \Phi_1 \times_2 \Phi_2 \times \cdots \times_N \Phi_N,\end{aligned} \tag{20}$$

where $\otimes$ is the Kronecker product, $\mathcal{Y} \in \mathbb{R}^{M_1\times M_2\times\cdots\times M_N}$ is the measurement, and $\Phi_i \in \mathbb{R}^{M_i\times I_i}$ is the sensing matrix for the $i$-mode linear space. Here, $K_i < M_i \ll I_i$ for $1 \le i \le N$.

Rewriting Eq. (16) in the Kronecker product form, it obtains

$$\mathbf{X} = \hat{\Psi}\Theta = (\Psi^{(1)} \otimes \Psi^{(2)} \otimes \cdots \otimes \Psi^{(N)})\Theta. \tag{21}$$

Thus, $\mathbf{X}$ can be sampled with $\Psi$ and $\Phi$ by combining Eq. (20) and (21)

$$\begin{aligned}\mathcal{Y} &= \hat{\Phi}\mathbf{X} = \hat{\Phi}\hat{\Psi}\Theta\\ &= (\Phi_1 \otimes \Phi_2 \otimes \cdots \otimes \Phi_N)(\Psi^{(1)} \otimes \Psi^{(2)} \otimes \cdots \otimes \Psi^{(N)})\Theta\\ &= (\Phi_1\Psi^{(1)} \otimes \Phi_2\Psi^{(2)} \otimes \cdots \otimes \Phi_N\Psi^{(N)})\Theta.\end{aligned} \tag{22}$$

Fig. 10 provides an example for compressive tensor sampling.

### *B. Stable Recovery*

Given the sensing matrix $A^{(n)} = \Phi_n\Psi^{(n)}$ and the difference $u = \theta_1 - \theta_2$ of two $k$-sparse column vectors of $\Theta_{(n)}$, Proposition 3 provides the conditions for stable recovery.

*Proposition 3:* The $n$-mode sensing matrix $A^{(n)}$ is stable for every $2k$-sparse vector $u$ if and only if there exists $C_1 > 0$ and $C_2 < \infty$ such that

$$C_1\|u\|_2^2 \le \|A^{(n)}u\|_2^2 \le C_2\|u\|_2^2. \tag{23}$$

*Proof:* Please refer to Appendix A. ■

*Definition 2 ($n$-Mode Recovery):* Provided that each column vector $\theta$ of $\Theta_{(n)}$ is sparse, $\theta$ can be recovered by

$$\min \|\theta\|_1, \quad \text{s.t. } y_{(n)} = \Phi_n\Psi^{(n)}\theta = A^{(n)}\theta. \tag{24}$$

Here $A^{(n)}$ satisfies the RIP condition with $\delta_{2k} \le \sqrt{2} - 1$, and $y_{(n)}$ denotes each corresponding column vector of the $n$-mode unfolding matrix $Y_{(n)}$. Fig. 11 illustrates an example for $n$-Mode Recovery.

### *C. Generalized Model for Video Sampling and Recovery*

A detailed description of the proposed tensor sampling and recovery framework is given in Algorithm 2. Taking video sequences for example, we decompose each GOP into a set of RFs

---

**Algorithm 2:** Framework of CTS based on a tensor subspace.

---

**Task:** Sample and reconstruct the tensor $\mathbf{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$.
**Input:** Tensor $\mathbf{X}$, training set $\{\mathbf{X}_1, \ldots, \mathbf{X}_K\}$, random Gaussian sampling matrix $\Phi_1, \ldots, \Phi_N$.
**1 Sampling:**
  **for** $n = 1$ to $N$ **do**
    **n-Mode Compressive Sampling**: Calculate the n-Mode measurements $\mathcal{Y}_{(n)} = \Phi_n\mathbf{X}_{(n)}$ according to Eq. (20).
  **end for**
**2 Training:** Apply HOSVD to $\{\mathbf{X}_1, \ldots, \mathbf{X}_K\}$ and derive each mode basis $\Psi^{(n)} \in \mathbb{R}^{I_n\times I_n}$, $n = 1, \cdots, N$, the whole basis $\hat{\Psi} = [\Psi^{(1)} \otimes \Psi^{(2)} \otimes \cdots \otimes \Psi^{(N)}]$.
**3 Recovering**:
  **for** $n = 1$ to $N$ **do**
    **n-Mode Recovery**: Calculate each column vector $\theta$ of $\Theta_n$ with Eq. (24) and calculate $X^*_{(n)} = (\Psi^{(n)})^T\Theta_{(n)}$.
  **end for**
**Output:** Recovered tensor $\mathbf{X}^*$.

---

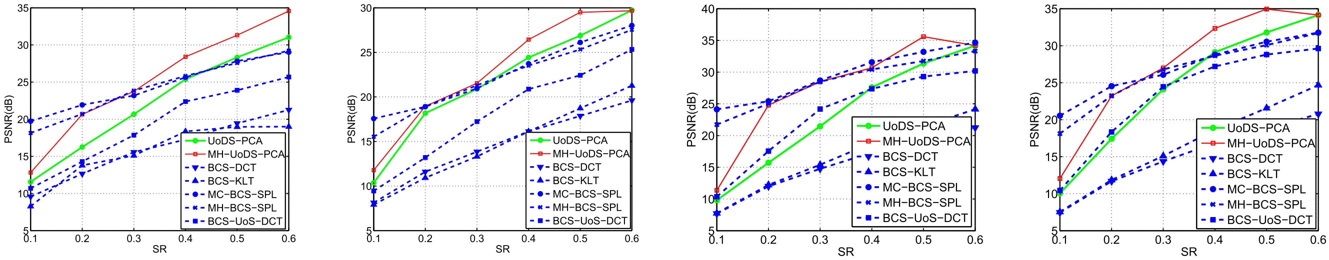
and the remaining NRFs. We first fully sample RFs while each of the NRFs cubes $X$ are sampled for each mode according to Eq. (22). Then, we train $K$ recovered cubes $X_1, \ldots, X_K$ of RFs by MPCA to derive each mode base $\Psi^{(1)}, \Psi^{(2)}, \Psi^{(3)}$ whose first k1, k2, k3 corresponding column vectors span the tensor subspace containing most of the variation of $X_1, \ldots, X_K$. Thus, $X$ can be considered as a "k1-k2-k3-sparse" tensor with respect to the whole basis $\hat{\Psi} = \Psi^{(1)} \otimes \Psi^{(2)} \otimes \Psi^{(3)}$. Subsequently, each of the NRFs cubes $X$ can be recovered for each mode according to (24).

The proposed algorithm can alleviate the computational and storage burden in sampling and recovery because of small-scale sampling matrix utilized for each mode. Moreover, in comparison to existing works [28]–[32], it makes compressible representation with sparse tensor based on data-driven basis, so that it can capture the non-stationarity in high-dimensional signals.

## V. Experimental Results

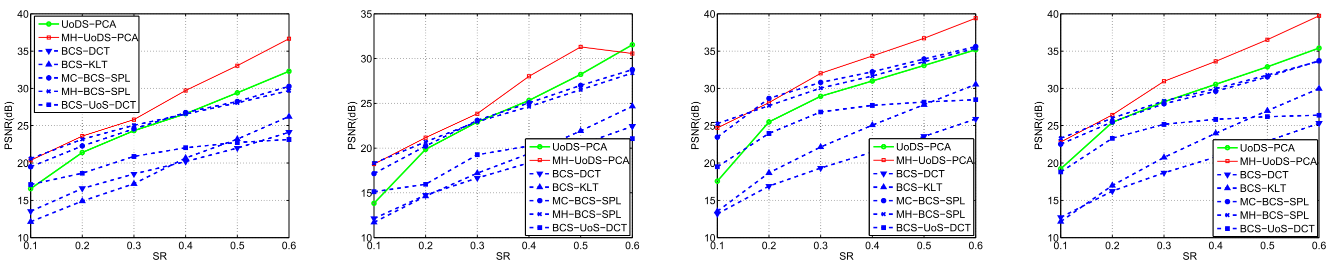
### *A. Implementation*

In this subsection, the UoDS model is employed on a variety of video sequences with multiple resolutions, including CIF ($352\times288$), DVD ($720\times480$), and 1080 p ($1920\times1080$).




(a) *Bike* (b) *NBA* (c) *Driving* (d) *Whale Show*

Fig. 12. Sampling-rate-distortion curves for the proposed UoDS-PCA and UoDS-MH-PCA, BCS-DCT, BCS-KLT, MC-BCS-SPL, MH-BCS-SPL, and UoS-BCS-DCT under $8 \times 8$ patches ($n = 64$) and dimension $d = 6$, respectively.




(a) *Bike* (b) *NBA* (c) *Driving* (d) *Whale Show*

Fig. 13. Sampling-rate-distortion curves for the proposed UoDS-PCA and UoDS-MH-PCA, BCS-DCT, BCS-KLT, MC-BCS-SPL, MH-BCS-SPL, and UoS-BCS-DCT under $16 \times 16$ patches ($n = 256$) and dimension $d = 10$, respectively.

Commonly, the size of each non-overlapping block is $8 \times 8$ and $16 \times 16$, thereby $n = 64$ and 256, respectively. As an exception, the block size for *Park Scene* is fixed to $32 \times 32$. The sampling matrix $\Phi \in \mathbb{R}^{m \times n}$ is an i.i.d. Gaussian random matrix with zero-mean and unit-variance. The sampling rate $SR \in \{0.1, 0.2, \ldots, 0.6\}$. The maximal value of $SR$ is set to 0.6, as it can already guarantee perfect recovery. Each GOP contains 10 grayscale frames. Without loss of generality, the first frame of each GOP is set as the RF, while NRFs for the remaining nine frames. The training set of patches for SSC[1] consists of overlapping blocks with a step of $\frac{1}{2}\sqrt{n}$ pixels along both rows and columns. For example, the training set for CIF sequences contains 1505 blocks, when block size is $16 \times 16$ and step is 8. Thus, representation matrix $C \in \mathbb{R}^{1505 \times 1505}$. In SSC, 50 clusters are adopted for spectral clustering of vectorized patches from RFs.

The basis of UoDS is learned for each cluster by PCA and PCA-L1, respectively. The dimension of each subspace $d_i$ selects from $\{2, 4, 6, 8, 10, 20\}$. The proposed models (UoDS-PCA and UoDS-PCAL1) are compared with five state-of-the-art sampling methods, including one UoS model UoS-BCS-DCT and four BCS based methods, e.g. BCS-DCT [12], BCS-KLT [18], BCS-MC-SPL [15], and BCS-MH-SPL [16]. Here, BCS based methods are single subspace models with adaptive KLT basis for BCS-KLT and predefined basis for BCS-DCT, BCS-MC-SPL and BCS-MH-SPL. To be concrete, BCS-MC-SPL incorporates motion compensation and BCS-MH-SPL utilizes multihypothesis prediction. For a fair comparison, we also incorporate multihypothesis prediction for an improved version MH-UoDS-PCA and MH-UoDS-PCAL1 of the UoDS model.

[1] Available at http://www.cis.jhu.edu/~ehsan/

The proposed models are implemented with the The PGL1 Matlab solver [39] under the configurations of 3.2-GHz CPU and 12-GB RAM.

## B. Results for Linear Subspaces

In this section, the UoDS model is evaluated for the union of data-driven linear subspaces. Figs. 12 and 13 show the distortion curves obtained under various sampling rates and block sizes for video sequences of various resolutions. We can see that the proposed UoDS model with multihypothesis prediction is competitive with BCS-based methods and the UoS model in the low SR region, and outperforms them when the sampling rate is high. The UoDS model can be sampled for a desirable recovery quality under a high sampling rate, as dependencies between RFs and NRFs can be exactly learned. Table I summarizes the results for six video sequences under various configurations of block sizes, dimensions and sampling rates, where the proposed model is shown to perform better than benchmark methods in most cases. The fact implies that the UoDS model can adaptively deal with structures with varying signal singularities to significantly decrease the necessary measurements for recovery in comparison to the single subspace and UoS models. Furthermore, the proposed model is suitable for a wide range of resolutions. Fig. 14 provides the comparative results for *Park Scene* sequence with a resolution of $1920 \times 1080$, where UoDS-MH-PCA still maintains its sampling and recovery performance.

Figs. 15–17 present the reconstructed fourth NRF in *Football*, *Whale Show*, and *Park Scene* sequences obtained by the proposed model, BCS-based models and the UoS model, respectively. These reconstructed frames are obtained under sufficient sampling and proper block sizes to contain local structures, i.e., $n = 256, SR = 0.6, d = 10$ for CIF and DVD

TABLE I
AVERAGE PSNR (dB) FOR RECONSTRUCTED VIDEO SEQUENCES WITH VARIOUS BLOCK SIZES $n$ AND DIMENSIONS $d$

| $(n, d)$ | Sequence | Sampling rate | UoDS-PCA | UoDS-PCAL1 | UoDS-MH-PCA | UoDS-MH-PCAL1 | BCS-DCT | BCS-KLT | BCS-MC-SPL | BCS-MH-SPL | UoS-BCS-DCT |
|---|---|---|---|---|---|---|---|---|---|---|---|
| (64, 6) | Bike | 0.1 | 11.57 | 11.81 | 12.82 | 12.89 | 9.57 | 8.27 | **19.25** | 18.14 | 10.69 |
| | | 0.2 | 16.27 | 16.23 | 20.65 | 20.66 | 12.65 | 13.79 | **21.93** | 20.71 | 14.30 |
| | | 0.3 | 20.67 | 20.65 | 23.75 | 23.81 | 15.56 | 15.12 | 23.18 | **23.85** | 17.86 |
| | | 0.4 | 25.39 | 25.35 | 28.41 | **28.47** | 17.27 | 18.35 | 25.64 | 25.82 | 22.38 |
| | | 0.5 | 28.33 | 28.37 | 31.31 | **31.41** | 19.44 | 18.97 | 27.83 | 27.57 | 23.89 |
| | | 0.6 | 31.05 | 31.04 | 31.57 | 31.08 | 21.25 | 19.00 | 29.03 | 29.23 | 25.68 |
| | Bus | 0.1 | 13.21 | 15.25 | 13.61 | 13.65 | 10.60 | 10.29 | **20.72** | 17.84 | 11.78 |
| | | 0.2 | 17.69 | 16.29 | 21.49 | 21.47 | 13.92 | 13.11 | **22.95** | 21.51 | 15.38 |
| | | 0.3 | 22.12 | 22.52 | 24.73 | 24.78 | 16.16 | 16.23 | **25.79** | 24.70 | 19.34 |
| | | 0.4 | 26.45 | 25.85 | **29.59** | 29.55 | 18.47 | 19.22 | 26.71 | 26.94 | 23.80 |
| | | 0.5 | 29.49 | 29.01 | **32.03** | 31.88 | 20.22 | 21.55 | 29.15 | 28.33 | 24.18 |
| | | 0.6 | 31.92 | **32.30** | 31.98 | 31.82 | 21.92 | 24.17 | 30.90 | 30.11 | 26.43 |
| | Football | 0.1 | 9.96 | 10.01 | 11.58 | 11.49 | 7.68 | 6.41 | **21.77** | 19.42 | 8.99 |
| | | 0.2 | 15.87 | 15.93 | 24.74 | 24.74 | 11.51 | 8.85 | **25.15** | 24.72 | 13.14 |
| | | 0.3 | 21.93 | 21.98 | **27.31** | 27.29 | 14.82 | 10.50 | 27.03 | 27.18 | 17.77 |
| | | 0.4 | 28.02 | 28.09 | 31.99 | **32.05** | 16.96 | 13.16 | 29.85 | 28.99 | 24.27 |
| | | 0.5 | 31.49 | 31.75 | 34.76 | **34.84** | 19.03 | 17.08 | 30.73 | 30.12 | 26.69 |
| | | 0.6 | 34.25 | 34.53 | 34.41 | **34.57** | 21.05 | 20.15 | 33.41 | 31.58 | 29.55 |
| | NBA | 0.1 | 10.38 | 8.32 | 11.79 | 11.52 | 8.13 | 7.92 | **17.57** | 15.53 | 9.45 |
| | | 0.2 | 18.19 | 15.26 | **18.95** | **18.95** | 11.60 | 10.93 | 18.90 | 18.93 | 13.20 |
| | | 0.3 | 20.89 | 20.25 | 21.54 | **21.57** | 13.85 | 13.31 | 20.96 | 21.31 | 17.23 |
| | | 0.4 | 24.43 | 24.49 | 26.44 | **26.55** | 16.13 | 16.12 | 23.73 | 23.51 | 20.88 |
| | | 0.5 | 26.89 | 27.65 | 29.50 | **29.62** | 17.87 | 18.73 | 26.12 | 25.33 | 22.43 |
| | | 0.6 | 29.73 | **29.99** | 29.67 | 29.64 | 19.61 | 21.26 | 28.02 | 27.54 | 25.33 |
| | Driving | 0.1 | 9.81 | 9.91 | 11.37 | 11.56 | 7.69 | 7.76 | **24.14** | 21.73 | 10.36 |
| | | 0.2 | 15.72 | 16.23 | 24.79 | 24.80 | 12.00 | 12.22 | **25.41** | 25.03 | 17.55 |
| | | 0.3 | 21.47 | 22.39 | 28.48 | 28.49 | 14.76 | 15.40 | **28.68** | 28.49 | 24.18 |
| | | 0.4 | 27.60 | 28.61 | 30.68 | **33.24** | 17.31 | 18.83 | 31.57 | 30.42 | 27.37 |
| | | 0.5 | 31.38 | 32.01 | 35.59 | **35.67** | 19.28 | 21.45 | 33.20 | 31.75 | 29.30 |
| | | 0.6 | 34.19 | 34.55 | 34.19 | 34.55 | 21.24 | 24.16 | **34.65** | 33.32 | 30.19 |
| | Whale Show | 0.1 | 10.06 | 10.05 | 12.06 | 12.03 | 7.46 | 7.55 | **20.54** | 18.12 | 10.44 |
| | | 0.2 | 17.43 | 17.46 | 23.22 | 23.21 | 11.68 | 11.85 | **24.53** | 23.24 | 18.35 |
| | | 0.3 | 24.11 | 24.25 | 27.00 | **27.01** | 14.36 | 15.10 | 26.08 | 26.79 | 24.46 |
| | | 0.4 | 29.13 | 29.29 | 32.35 | **32.37** | 16.85 | 18.61 | 28.79 | 28.65 | 27.20 |
| | | 0.5 | 31.80 | 31.92 | 34.96 | **34.98** | 18.76 | 21.57 | 30.54 | 30.07 | 28.80 |
| | | 0.6 | 34.14 | 34.29 | 34.14 | **34.29** | 20.77 | 24.67 | 31.77 | 31.70 | 29.62 |
| (256, 10) | Bike | 0.1 | 16.56 | 16.81 | 20.36 | 20.38 | 13.53 | 12.13 | 19.77 | **20.59** | 17.12 |
| | | 0.2 | 21.41 | 21.45 | **23.61** | 23.58 | 16.59 | 14.90 | 22.29 | 23.45 | 18.63 |
| | | 0.3 | 24.33 | 24.41 | **25.82** | 25.80 | 18.53 | 17.22 | 24.58 | 25.26 | 20.89 |
| | | 0.4 | 26.60 | 26.66 | 29.72 | **29.80** | 20.19 | 20.67 | 26.77 | 26.75 | 22.04 |
| | | 0.5 | 29.41 | 29.33 | **33.05** | 32.79 | 22.01 | 23.20 | 28.25 | 28.16 | 22.76 |
| | | 0.6 | 32.30 | 32.30 | **36.66** | 36.65 | 24.12 | 26.23 | 30.27 | 29.75 | 23.15 |
| | Bus | 0.1 | 18.96 | 19.37 | **21.35** | 21.05 | 13.96 | 13.82 | 20.61 | 20.77 | 18.49 |
| | | 0.2 | 23.24 | 23.06 | **24.59** | 23.62 | 16.87 | 17.65 | 23.94 | 24.15 | 20.47 |
| | | 0.3 | 25.68 | 25.55 | **28.31** | 28.21 | 19.10 | 20.61 | 26.72 | 26.05 | 23.00 |
| | | 0.4 | 27.95 | 27.84 | 30.52 | **30.57** | 20.73 | 22.72 | 28.03 | 27.44 | 24.02 |
| | | 0.5 | 29.99 | 30.25 | 33.43 | **33.61** | 22.73 | 25.19 | 29.86 | 28.97 | 24.50 |
| | | 0.6 | 32.86 | 32.81 | **37.28** | 37.12 | 25.02 | 28.05 | 32.10 | 30.64 | 24.79 |
| | Football | 0.1 | 17.00 | 16.97 | 24.10 | **24.11** | 13.20 | 9.49 | 22.95 | 23.92 | 18.47 |
| | | 0.2 | 24.35 | 24.37 | 27.02 | **27.05** | 16.83 | 15.20 | 25.92 | 26.32 | 21.58 |
| | | 0.3 | 27.84 | 28.06 | **28.83** | 28.82 | 19.11 | 16.21 | 27.99 | 27.91 | 25.04 |
| | | 0.4 | 30.44 | 30.53 | 33.47 | **33.49** | 21.18 | 18.94 | 30.15 | 29.06 | 26.23 |
| | | 0.5 | 33.15 | 33.08 | **36.71** | 36.65 | 23.29 | 22.84 | 31.88 | 30.47 | 26.79 |
| | | 0.6 | 36.09 | 35.99 | **40.22** | 40.19 | 25.40 | 25.39 | 34.10 | 31.96 | 27.09 |
| | NBA | 0.1 | 13.83 | 16.85 | 18.21 | **18.59** | 12.13 | 11.71 | 17.15 | 18.18 | 15.12 |
| | | 0.2 | 19.87 | 20.03 | 21.17 | **22.02** | 14.75 | 14.63 | 20.23 | 20.77 | 15.96 |
| | | 0.3 | 22.92 | 22.68 | **23.85** | 23.81 | 16.66 | 17.20 | 23.10 | 22.94 | 19.26 |
| | | 0.4 | 25.32 | 25.43 | 28.02 | **28.12** | 18.36 | 19.38 | 25.05 | 24.69 | 20.23 |
| | | 0.5 | 28.23 | 28.41 | **31.30** | 28.25 | 20.39 | 21.91 | 27.00 | 26.57 | 20.71 |
| | | 0.6 | 31.54 | 31.63 | 30.57 | **35.79** | 22.43 | 24.67 | 28.76 | 28.38 | 21.04 |
| | Driving | 0.1 | 17.57 | 17.83 | 24.70 | **25.52** | 13.18 | 13.52 | 23.47 | 24.86 | 19.49 |
| | | 0.2 | 25.51 | 25.84 | 28.10 | 28.11 | 16.91 | 18.69 | **28.66** | 27.78 | 23.97 |
| | | 0.3 | 28.95 | 29.10 | 32.02 | **32.08** | 19.32 | 22.15 | 30.80 | 29.42 | 26.85 |
| | | 0.4 | 31.00 | 31.08 | 34.35 | **34.49** | 21.46 | 25.07 | 32.22 | 30.66 | 27.73 |
| | | 0.5 | 33.07 | 33.18 | 36.72 | **36.92** | 23.56 | 27.82 | 33.94 | 32.13 | 28.18 |
| | | 0.6 | 35.18 | 35.21 | 39.41 | **39.89** | 25.90 | 30.54 | 35.61 | 33.78 | 28.48 |
| | Whale Show | 0.1 | 19.22 | 19.01 | 22.81 | **23.57** | 12.71 | 12.15 | 22.53 | 22.53 | 18.79 |
| | | 0.2 | 25.46 | 25.39 | 26.46 | **27.88** | 16.24 | 16.99 | 25.51 | 25.68 | 23.32 |
| | | 0.3 | 28.20 | 28.29 | **30.95** | **30.95** | 18.69 | 20.75 | 27.94 | 27.54 | 25.19 |
| | | 0.4 | 30.55 | 30.56 | **33.62** | 33.59 | 20.80 | 23.99 | 29.62 | 28.80 | 25.86 |
| | | 0.5 | 32.89 | 32.86 | 36.53 | **36.54** | 22.93 | 27.01 | 31.55 | 30.40 | 26.20 |
| | | 0.6 | 35.41 | 35.42 | **39.73** | **39.73** | 25.32 | 29.98 | 33.71 | 32.08 | 26.41 |

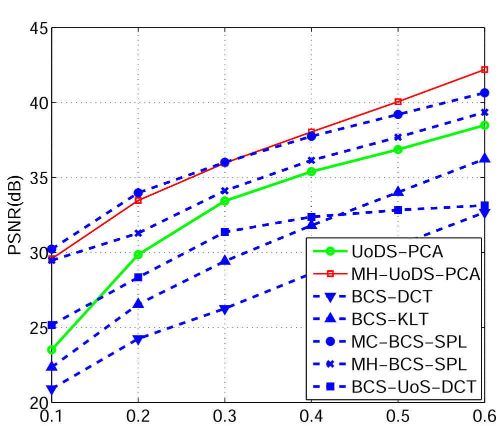



Fig. 14. Sampling-rate-distortion curves for *Park Scene* sequence obtained by the proposed UoDS-PCA and UoDS-MH-PCA, BCS-DCT, BCS-KLT, BCS-MC-SPL, BCS-MH-SPL, and UoS-BCS-DCT under $32 \times 32$ blocks ($n = 1024$) and dimension $d = 40$, respectively.

sequences, and $n = 1024, SR = 0.6, d = 40$ for 1080 p sequence. In overall, the proposed model achieves better visual quality in comparison to BCS-based models and the UoS model, especially in the texture regions like "grass", "water" and "tree" in the three video sequences. PSNR performance of the reconstructed frames is also provided for validation. The proposed model is shown to achieve best visual quality under such benchmark. These results are consistent with the sampling and recovery performance, which validates the UoDS model.

Moreover, Table I shows that $16 \times 16$ patches can achieve better recovery performance in comparison to $8 \times 8$ ones. It means that $c^*$ can be exactly reconstructed with a high probability when $n$ grows. The dimension of each subspace $d$ also affects the performance of the UoDS model, as shown in Fig. 18. In the region of low SR, recovery performance for small $d$, e.g., $d = 2$, and $SR = 0.1$, is superior to the large $d$ at a high SR, e.g., $d = 20$ and $SR = 0.1$. Since reconstructed RFs are decomposed into 50 clusters, we can obtain from Eq. (13) that $m = 6, r = 100$ for $t = 50, d = 2, SR = 0.1, n = 64$, while

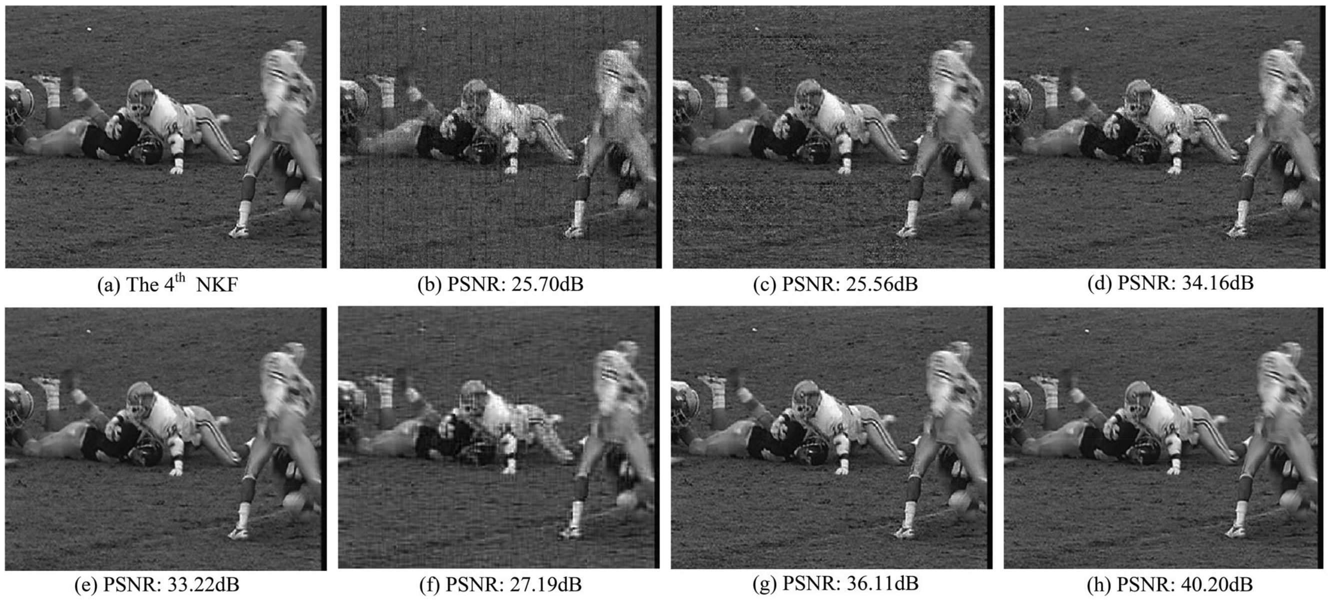



Fig. 15. Reconstructed frames for *Football* sequence under $n = 256$, SR $= 0.6$ and $d = 10$. (a) the 4th NRF; (b) BCS-DCT; (c) BCS-KLT; (d) BCS-MC-SPL; (e) BCS-MH-SPL; (f) UoS-BCS-DCT; (g) UoDS-PCA; (h) UoDS-MH-PCA.

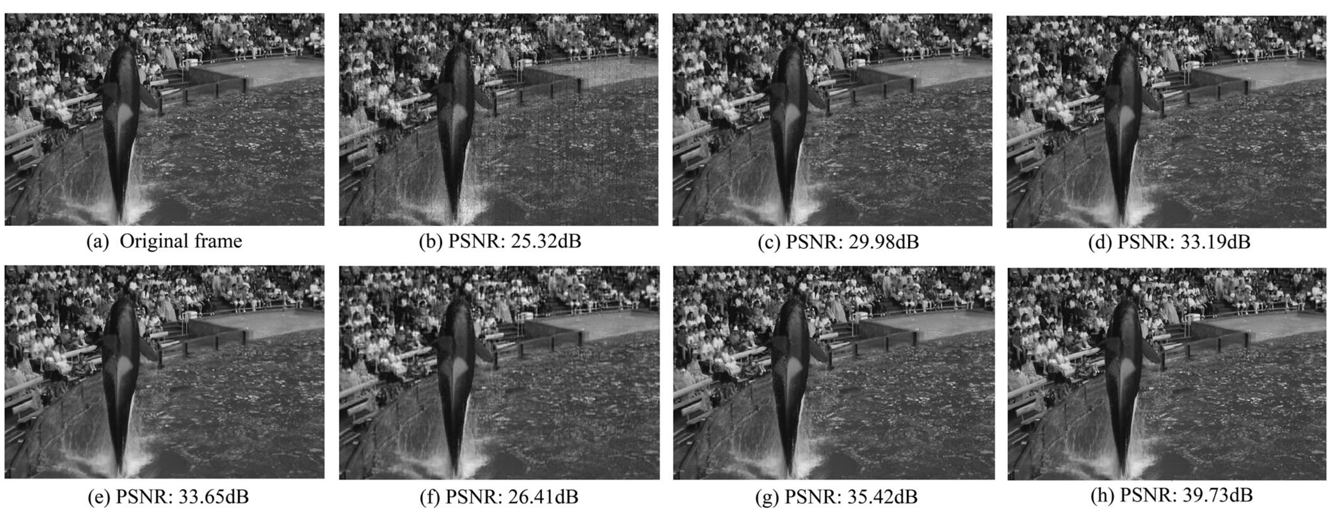
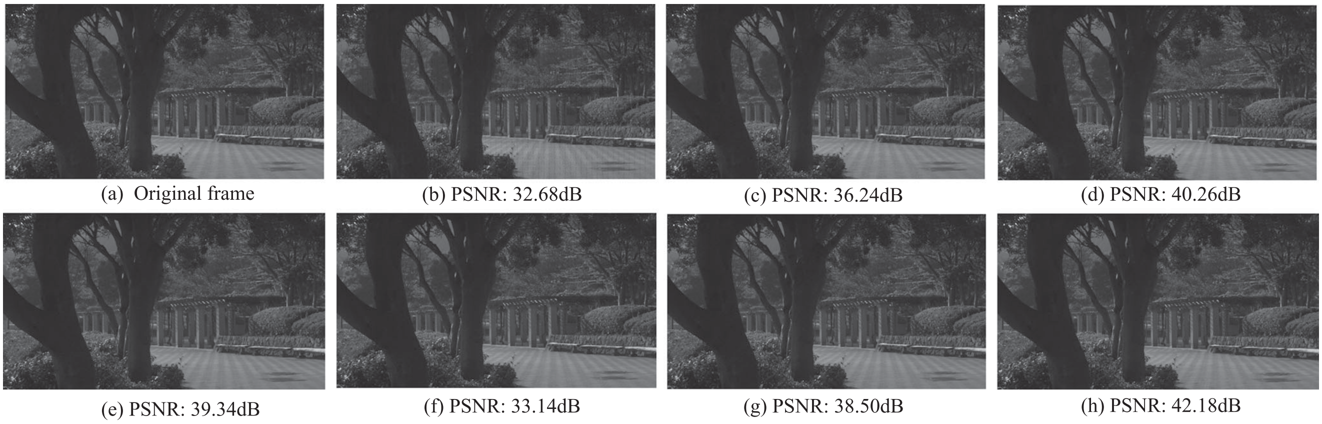



Fig. 16. Reconstructed frames for *Whale Show* sequence under $n = 256$, SR $= 0.6$ and $d = 10$. (a) the 4th NRF; (b) BCS-DCT; (c) BCS-KLT; (d) BCS-MC-SPL; (e) BCS-MH-SPL; (f) UoS-BCS-DCT; (g) UoDS-PCA; (h) UoDS-MH-PCA.

$m = 6, r = 200$ for $d = 4, SR = 0.1, n = 64$. Therefore, $c^*$ is more accurate for lower dimensions. However, Fig. 18(b) and (d) suggest that larger $d$ can improve the recovery performance at a high SR, as enough measurements are provided for subspaces with larger dimension. For example, $r = 1000, m = 614$, when $t = 50, d = 20, SR = 0.6, n = 1024$. Consequently, it is desirable to properly select $n$ and $d$ for an exact recovery. For example, $r = 1000, m = 38$ for $d = 20, SR = 0.6, n = 64$ would fail to recover $\mathbf{c}^*$ with necessary measurements.

### *C. Results for Multilinear Subspaces*

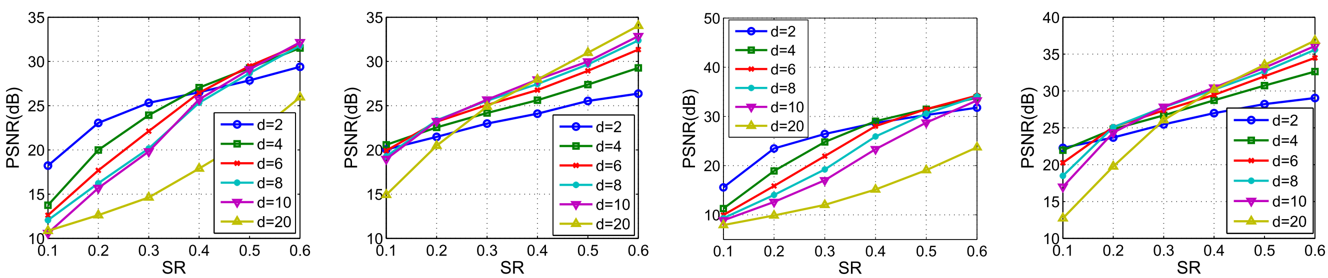
For further validation, we employ CTS in video sampling and reconstruction. In this subsection, experiments are conducted on a variety of video sequences with CIF ($352 \times 288$) (i.e., *Bike, Bus, Football, NBA*) and DVD ($720 \times 480$) resolution (i.e., *Driving, Whale Show*). We compare the proposed CTS method with the state-of-the-art GTCS [31], [32], which utilizes DCT basis to make video compressible in DCT domain. Video sequences are represented by a $352 \times 288 \times 16$ tensor. For CTS, we choose the first 3 frames and split them into overlapping $32 \times 32 \times 3$ tensors to generate the training set. MPCA is applied to this training set to derive 1-mode and 2-mode basis, which is adaptive and makes the tensors compressible. We use DCT basis for the 3-mode basis. For both methods, we sample each $32 \times 32 \times 16$ non-overlapping sub-video cube of the following 16 frames by i.i.d. random Gaussian matrix $\Phi_1, \Phi_2 \in \mathbb{R}^{32SR\times 32}$ with zero-mean and unit-variance for 1-mode and 2-mode sampling, respectively, and set $\Phi_3 \in \mathbb{R}^{16\times 16}$ to an identity matrix for 3-mode full sampling. Thus, the total number of measurements



(a) Original frame (b) PSNR: 32.68dB (c) PSNR: 36.24dB (d) PSNR: 40.26dB

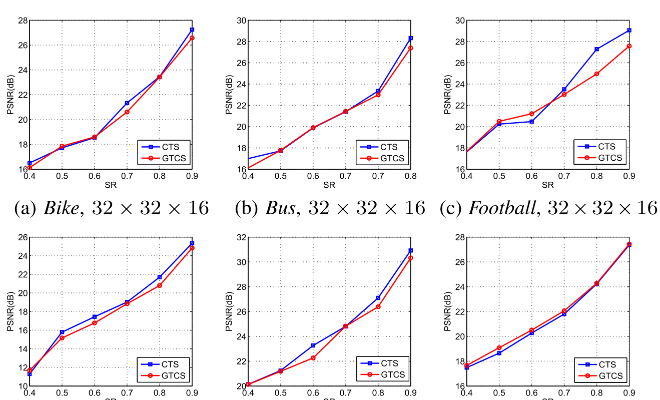
(e) PSNR: 39.34dB (f) PSNR: 33.14dB (g) PSNR: 38.50dB (h) PSNR: 42.18dB

Fig. 17. Reconstructed frames for *Park Scene* sequence under $n = 1024$, SR = 0.6 and $d = 40$. (a) the 4th NRF; (b) BCS-DCT; (c) BCS-KLT; (d) BCS-MC-SPL; (e) BCS-MH-SPL; (f) UoS-BCS-DCT; (g) UoDS-PCA; (h) MH-UoDS-PCA.




(a) *Bus*:8 × 8 (b) *Bus*:16 × 16 (c) *Football*:8 × 8 (d) *Football*:16 × 16

Fig. 18. The performance of the proposed scheme (UoDS-PCA) with various dimensions $d$ and patch sizes.




(a) *Bike*, 32 × 32 × 16 (b) *Bus*, 32 × 32 × 16 (c) *Football*, 32 × 32 × 16

(d) *NBA*, 32 × 32 × 16 (e) *Driving*, 64 × 64 × 16 (f) *WhaleShow*, 64 × 64 × 16

Fig. 19. Average PSNR performance (dB) of the 16 recovered frames with SR from 0.4 to 0.9. Tensor sizes 32 × 32 × 16 and 64 × 64 × 16 are adopted for CIF and DVD test sequences, respectively.

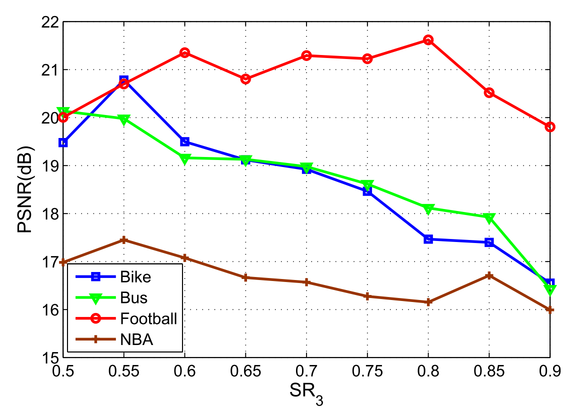



Fig. 20. The experimental result of sampling rate allocation for the CTS model over four sequences.

of each sub-video cube is $SR^2 \cdot 16384$. We also test tensor size of $64 \times 64 \times 16$ for *Driving* and *Whale Show* sequences with similar settings. CTS and GTCS are compared under SR ranging from 0.4 to 0.9. For the recovery of each mode, we apply the SPGL1 Matlab solver[38] for both schemes. Besides, a MATLAB Tensor Toolbox [40] is used in this experiment.

Fig. 19 shows the recovery performance for CTS and GTCS, where the proposed CTS outperforms GTCS in five out of six test sequences. In comparison to DCT basis adopted by GTCS, CTS is more flexible to fit the non-stationary statistics of video sequences with the adaptive basis of each mode of tensors to be sampled.

### D. Sampling Rate Allocation

Moreover, we discuss the rate allocation for the spatial and temporal sampling to obtain optimal overall performance. Instead of full sampling along the temporal dimension, the proposed CTS varies the temporal sampling rates $SR_3$ from 0.5

TABLE II
AVERAGE RECOVERY SPEED (SEC/FRAME) FOR VIDEO SEQUENCES *Bike*, *Bus*, *Football*, *NBA*, *Driving* AND *WhaleShow* OBTAINED UNDER BLOCK SIZE $n = 256$, DIMENSION $d = 10$ AND SAMPLING RATE $SR = 0.6$

| Sequence | UoDS-PCA | UoDS-PCAL1 | UoDS-MH-PCA | UoDS-MH-PCAL1 | BCS-DCT | BCS-KLT | BCS-MC-SPL | BCS-MH-SPL | UoS-BCS-DCT |
|---|---|---|---|---|---|---|---|---|---|
| *Bike* | 21.64 | 22.37 | 38.22 | 38.97 | **0.59** | 14.59 | 6.65 | 17.21 | 14.02 |
| *Bus* | 20.33 | 21.18 | 36.83 | 38.10 | **0.61** | 14.12 | 6.82 | 17.12 | 12.33 |
| *Football* | 19.72 | 19.98 | 36.15 | 36.30 | **0.67** | 14.45 | 6.86 | 17.11 | 13.79 |
| *NBA* | 20.44 | 20.54 | 36.98 | 37.01 | **0.64** | 14.31 | 7.04 | 17.16 | 11.99 |
| *Driving* | 63.82 | 64.01 | 122.20 | 122.97 | **2.02** | 48.65 | 22.04 | 60.45 | 40.89 |
| *WhaleShow* | 64.21 | 64.35 | 121.30 | 121.55 | **1.91** | 45.48 | 22.41 | 59.52 | 39.08 |

to 0.9 to exploit the compressible temporal components. Under the fixed overall sampling rate $SR = 0.5$, $\Phi_3 \in \mathbb{R}^{16 \cdot SR_3 \times 16}$ and $\Phi_1, \Phi_2 \in \mathbb{R}^{32 \cdot SR_1 \times 32}$ are adopted for temporal and spatial sampling with the rates $SR_1 = SR_2 = \sqrt{SR/SR_3}$.

Fig. 20 provides the sampling-rate-distortion curves obtained by CTS, where recovery performance is obtained under temporal sampling rates varying from 0.5 to 0.9 with a step of 0.05. It shows that the optimal sampling rate allocation is distinguishing for different test sequences. Specifically as shown in Fig. 20, lower temporal sampling rate is required for test sequences with smooth motion trajectories, e.g., *Bike* and *Bus*. On the contrary, recovery performance would be improved with higher temporal sampling rate for test sequences with sharp variation of corresponding pixels along motion trajectories, e.g., *Football*. As mentioned in Section III-C, the dimensions of subspaces are determined by the texture complexities, which serve as the measure of patch based structures. Thus, the distribution of sparsities would affect the sampling rate allocation, as block sparsities are related with the number of basis (dimensions). These facts imply that the allocation of sampling rates for different dimensions tends to follow the distribution of sparsity to guarantee the quality of reconstruction.

### E. Computational Complexity

The computational complexity of the proposed UoDS method comes from training and recovery process. For each group of pictures (GOP), the training process (Step 1-4 in Algorithm 1) utilizes the patches from recovered fully-sampled RF to derive the union of data-driven subspaces. Thus, the training speed is related to the subspace clustering and learning conditioned on the RF. In recovery process (Step 5 in Algorithm 1), the efficiency of UoDS depends on the recovery of NRFs, where second-order cone programming (SOCP) determination is leveraged to reconstruct the block-sparse vector $\mathbf{c}^*$ and multihypothesis prediction can be introduced to improve motion compensation.

The experiments for compressive video sampling and reconstruction are implemented with Matlab on a PC with 3.2 GHz CPU and 12 GB RAM. The complexity for the proposed UoDS model is evaluated in terms of recovery speed (sec/frame) and training speed (sec/GOP). Table II provides the recovery speeds for the proposed UoDS model and benchmarks for block CS, i.e., BCS-DCT, BCS-KLT, BCS-MC-SPL, BCS-MH-SPL and UoS-BCS-DCT, under block size $n = 256$, dimension $d = 10$ and sampling rate $SR = 0.6$. In comparison to UoS-BCS-DCT, the UoDS models UoDS-PCA and UoDS-PCAL1 require approximately 1.4 to 1.7 times the complexity of UoS-BCS-DCT, due to the SOCP determination that optimizes coefficients for data-driven subspaces. When compared with BCS-KLT, motion compensation leads to a 30% to 50% excess time cost in UoDS-PCA and UoDS-PCAL1. Moreover, it would require an additional 70% to 90% time cost to introduce multihypothesis compensated prediction in the UoDS model. These facts demonstrate that the recovery efficiency would be affected by the SOCP determination and motion compensation.

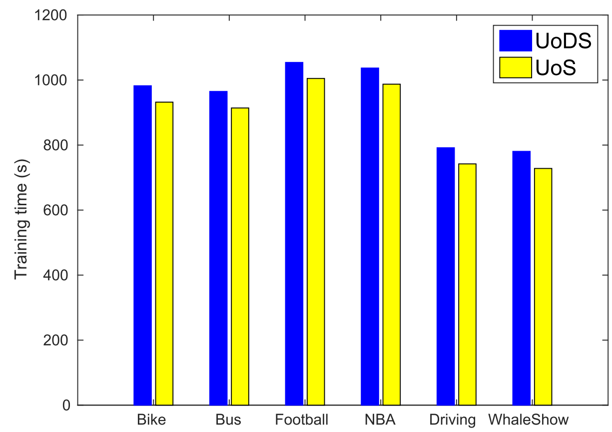



Fig. 21. Average training speed (sec/GOP) for video sequences *Bike*, *Bus*, *Football*, *NBA*, *Driving* and *WhaleShow* obtained by UoS and UoDS under SR = 0.6.

Furthermore, we evaluate the computational complexity for the training process. Fig. 21 compares the training speed (sec/GOP) for UoDS and UoS over the six video sequences. Here, subspace clustering and basis derivation are not required for the other benchmarks, as they are based on single subspace (simple sparsity). Fig. 21 shows that the proposed UoDS model is competitive with UoS in terms of training speed, but achieves a better reconstruction performance with the data-driven subspaces and corresponding bases. It should be noted that the training of union of subspaces and their bases is only performed once on the RFs for each GOP. It is also promising to leverage low-rank representation (LRR) to balance the complexity and performance of training process. Fig. 6 shows that the UoDS-LRR can significantly reduce the complexity (about 70%–90%) for the training process with a slight loss (up to 0.5 dB) of reconstruction performance, when the number $t$ of subspaces ranges from 10 to 100.

## VI. CONCLUSION

This paper proposes an explicit sampling scheme to recover an unknown signal from a union of data-driven subspaces (UoDS). It investigates neighboring data structures by clustering to form classified signal series. Subsequently, the union of subspaces is learned uniquely from the classified signal series by a linear subspace learning method thereby deriving an adaptive basis and enhancing the structured sparsity representation. Besides, the general sampling model which is based on tensor is also proposed. With the proof of stable reconstruction, the proposed scheme is fulfilled in video acquisition where the UoDS is learned from decoded reference frames. Experimental results show that the proposed method gets better performance in comparison to the other compressive video sampling methods. Additionally, we extend tradition CS to its tensor form for signals lying in a tensor subspace.

## APPENDIX A
## PROOF OF PROPOSITION 1—PROPOSITION 3

*Proof of Proposition 1:* The proof is similar with that of Proposition 3 in [19]. Since the UoDS model inherits the merit of the UoS model, $\mathcal{U}^*$ still follows the property of predefined UoS. Thus, $\Phi$ is one-on-one on each $\mathcal{S}_{ij}^*$ with $\dim(\mathcal{S}_{ij}^*) = \dim(\Phi) \leq m$. As a result, $m \geq k_{max} = \max_{i \neq j} \dim(\mathcal{S}_{ij}^*)$. ■

*Proof of Proposition 2:* Firstly, we obtain from Eq. (13) that $\mathbf{A}^* = \Phi\Psi^*$. Since PCA-based learning methods generate an orthornormal basis $\Psi_i^*$ for each linear subspace $\mathcal{S}_i^*$ and $\Phi$ is an i.i.d. random matrix, Proposition 2 is obtained according to Proposition 4 and 5 in [19]. ■

*Proof of Proposition 3:* Eq. (22) shows that $A^{(n)} = \Phi_n \Psi^{(n)}$ for the orthonormal basis $\Psi^{(n)}$ of $\tilde{\mathcal{S}}^{(n)}$ obtained by HOSVD and the i.i.d. random matrix $\Phi_n$. According to Proposition 4 and Proposition 5 in [19], we can easily obtain Proposition 3. ■

## APPENDIX B
## PROOF OF THEOREM 1

According to Proposition 1, $\delta_{2k} < 1$ makes $c^*$ unique. First, we assume that $c = c^* + b$ is a solution of Eq. (15). To prove $\mathbf{c}^*$ is the true solution of Eq. (15), we just need to prove $b = 0$. We know that $\mathbf{c}^*$ is a $k$-block sparse vector, so let $\mathcal{I}_0$ denote the indices for $\mathbf{c}^*$ where the coefficients are nonzero, $b_{\mathcal{I}_0}$ denotes the restriction of $b$ to these blocks. Then we can decompose $b$ as

$$b = \sum_{i=0}^{t-1} b_{\mathcal{I}_i} \tag{25}$$

where $b_{\mathcal{I}_i}$ is the restriction of $b$ to the set $\mathcal{I}_i$ which comprises of $k$ blocks, selected such that the norm of $b_{\mathcal{I}_0^c}$ over $\mathcal{I}_1$ is largest, the norm over $\mathcal{I}_2$ is the second largest, and so on. Therefore, we can prove that

$$\|b\|_2 = \|b_{\mathcal{I}_0 \cup \mathcal{I}_1} + b_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2 \leq \|b_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2 + \|b_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2$$

Because our goal is to prove $b = 0$, the work below is to prove both $\|b_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2 = 0$ and $\|b_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2 = 0$.

*Part 1:* $\|b_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2 \leq \|b_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2$.

First we have

$$\|b_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2 = \|\sum_{i=2}^{t-1} b_{\mathcal{I}_i}\|_2 \leq \sum_{i=2}^{t-1} \|b_{\mathcal{I}_i}\|_2 \tag{26}$$

which bounds $\|b_{\mathcal{I}_i}\|_2$ for $i \geq 2$. Then

$$\|b_{\mathcal{I}_i}\|_2 \leq k^{1/2}\|b_{\mathcal{I}_i}\|_{\infty,\mathcal{I}} \leq k^{-1/2}\|b_{\mathcal{I}_{i-1}}\|_{2,\mathcal{I}} \tag{27}$$

where we defined $\|a\|_{\infty,\mathcal{I}} = \max_i \|a[i]\|_2$. For there are at most $k$ nonzero blocks, $k\|b_{\mathcal{I}_i}\|_{\infty,\mathcal{I}} \leq \|b_{\mathcal{I}_{i-1}}\|_{2,\mathcal{I}}$. Therefore, combine Eq. (26) and (27), we have

$$\begin{aligned} \|b_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2 &\leq k^{-1/2} \sum_{i=1}^{t-2} \|b_{\mathcal{I}_i}\|_{2,\mathcal{I}} \leq k^{-1/2} \sum_{i=1}^{t-1} \|b_{\mathcal{I}_i}\|_{2,\mathcal{I}} \\ &= k^{-1/2}\|b_{\mathcal{I}_0^c}\|_{2,\mathcal{I}} \end{aligned} \tag{28}$$

To bound $\|b_(\mathcal{I}_0)^c\|_{2,\mathcal{I}}$, we use the fact that $\mathbf{c} = \mathbf{c}^* + \mathbf{b}_{\mathcal{I}_0} + \mathbf{b}_{\mathcal{I}_0^c}$, $\mathbf{c}^*$ is support on $\mathcal{I}_0$, $\mathbf{c}$ is the solution of Eq. (15), $\|\mathbf{c}^*\|_{2,\mathcal{I}} \geq \|\mathbf{c}\|_{2,\mathcal{I}}$, we have

$$\begin{aligned} \|\mathbf{c}^*\|_{2,\mathcal{I}} &\geq \|\mathbf{c}^* + \mathbf{b}_{\mathcal{I}_0}\|_{2,\mathcal{I}} + \|\mathbf{b}_{\mathcal{I}_0^c}\|_{2,\mathcal{I}} \\ &\geq \|\mathbf{c}^*\|_{2,\mathcal{I}} - \|\mathbf{b}_{\mathcal{I}_0}\|_{2,\mathcal{I}} + \|\mathbf{b}_{\mathcal{I}_0^c}\|_{2,\mathcal{I}} \end{aligned} \tag{29}$$

thereby we have

$$\|\mathbf{b}_{\mathcal{I}_0^c}\|_{2,\mathcal{I}} \leq \|\mathbf{b}_{\mathcal{I}_0}\|_{2,\mathcal{I}} \leq k^{1/2}\|\mathbf{b}_{\mathcal{I}_0}\|_2. \tag{30}$$

Combine Eq. (30) and (28), we have

$$\|\mathbf{b}_{(\mathcal{I}_0 \cup \mathcal{I}_1)^c}\|_2 \leq \|\mathbf{b}_{\mathcal{I}_0}\|_2 \leq \|\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2. \tag{31}$$

*Part 2:* $\|\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2 = 0$.

Since $\mathbf{y} = \mathbf{A}^*\mathbf{c}^* = \mathbf{A}^*\mathbf{c}$, $\mathbf{A}^*\mathbf{b} = 0$. Besides, we have the fact that $\mathbf{b} = \mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1} + \sum_{i \geq 2} \mathbf{b}_{\mathcal{I}_i}$.

$$\|\mathbf{A}^*\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2^2 = -\sum_{i=2}^{t-1} \langle \mathbf{A}^*(\mathbf{b}_{\mathcal{I}_0} + \mathbf{b}_{\mathcal{I}_1}), \mathbf{A}^*\mathbf{b}_{\mathcal{I}_i} \rangle. \tag{32}$$

From the block-RIP and the parallelogram identity, we have

$$|\langle \mathbf{A}^*\mathbf{c}_1, \mathbf{A}^*\mathbf{c}_2 \rangle| \leq \delta_{2k}\|\mathbf{c}_1\|_2\|\mathbf{c}_2\|_2 \tag{33}$$

for any two-block $k$-sparse vectors with disjoint support. Therefore we have

$$|\langle \mathbf{A}^*\mathbf{b}_{\mathcal{I}_0}, \mathbf{A}^*\mathbf{b}_{\mathcal{I}_i} \rangle| \leq \delta_{2k}\|\mathbf{b}_{\mathcal{I}_0}\|_2\|\mathbf{b}_{\mathcal{I}_i}\|_2 \tag{34}$$

also similarly for $\langle \mathbf{A}^*\mathbf{b}_{\mathcal{I}_1}, \mathbf{A}^*\mathbf{b}_{\mathcal{I}_i} \rangle$. Thereby, Eq. (32) becomes

$$\begin{aligned} \|\mathbf{A}^*\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2^2 &\leq \sum_{i=2}^{t-1} (|\langle \mathbf{A}^*\mathbf{b}_{\mathcal{I}_0}, \mathbf{A}^*\mathbf{b}_{\mathcal{I}_i} \rangle| + |\langle \mathbf{A}^*\mathbf{b}_{\mathcal{I}_1}, \mathbf{A}^*\mathbf{b}_{\mathcal{I}_i} \rangle|) \\ &\leq \delta_{2k}(\|\mathbf{b}_{\mathcal{I}_0}\|_2 + \|\mathbf{b}_{\mathcal{I}_1}\|_2) \sum_{i=2}^{t-1} \|\mathbf{b}_{\mathcal{I}_i}\|_2. \end{aligned} \tag{35}$$

From the Cauchy–Schwartz inequality, we have

$$\|\mathbf{b}_{\mathcal{I}_0}\|_2 + \|\mathbf{b}_{\mathcal{I}_1}\|_2 \leq \sqrt{2\,(\|\mathbf{b}_{\mathcal{I}_0}\|_2^2 + \|\mathbf{b}_{\mathcal{I}_1}\|_2^2)} = \sqrt{2}\|\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2$$

where the last equality is a result of the fact that $\mathbf{b}_{\mathcal{I}_0}$ and $\mathbf{b}_{\mathcal{I}_1}$ have disjoint support. Combine Eq. (27), (28), (30), and (35), we have

$$\begin{aligned} \|\mathbf{A}^*\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2^2 &\leq \sqrt{2}k^{-1/2}\delta_{2k}\|\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2\|\mathbf{b}_{\mathcal{I}_0^c}\|_{2,\mathcal{I}} \\ &\leq \sqrt{2}\delta_{2k}\|\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2\|\mathbf{b}_{\mathcal{I}_0}\|_2 \\ &\leq \sqrt{2}\delta_{2k}\|\mathbf{b}_{\mathcal{I}_0 \cup \mathcal{I}_1}\|_2^2 \end{aligned} \tag{36}$$

where the first inequality is obtained by Eq. (27) and (28), the second follows from Eq. (30), and the last from $\|\mathbf{b}_{\mathcal{I}_0}\|_2 \leq$

$\|\mathbf{b}_{\mathcal{I}_0\cup\mathcal{I}_1}\|_2$. Then with the Block-RIP, we have

$$(1-\delta_{2k})\|\mathbf{b}_{\mathcal{I}_0\cup\mathcal{I}_1}\|_2^2 \leq \|\mathbf{A}^*\mathbf{b}_{\mathcal{I}_0\cup\mathcal{I}_1}\|_2^2 \leq \sqrt{2}\delta_{2k}\|b_{\mathcal{I}_0\cup\mathcal{I}_1}\|_2^2. \tag{37}$$

Since $\delta_{2k} < \sqrt{2}-1$, Eq. (37) can only hold if $\|\mathbf{b}_{\mathcal{I}_0\cup\mathcal{I}_1}\|_2 = 0$, which completes the second part of the proof.

Since $\|\mathbf{b}_{(\mathcal{I}_0\cup\mathcal{I}_1)^c}\|_2 \leq \|\mathbf{b}_{\mathcal{I}_0\cup\mathcal{I}_1}\|_2$ which part 1 has proved, therefore $\|\mathbf{b}_{(\mathcal{I}_0\cup\mathcal{I}_1)^c}\|_2 = 0$, $\|\mathbf{b}\|_2 = 0$ which means $\mathbf{c} = \mathbf{c}^*$ and we complete the whole proof.

## References


[1] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. New York, NY, USA: Academic, Sep. 1999.

[2] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 54, no. 12, pp. 3736–3745, Dec. 2006.

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

[4] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 791–804, Apr. 2012.

[5] J. Zhang, D. Zhao, F. Jiang, and W. Gao, "Structural group sparse representation for image compressive sensing recovery," in *Proc. Data Compression Conf.*, Snowbird, UT, USA, Mar. 2013, pp. 331–340.

[6] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[7] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 1794–1801.

[8] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[9] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[10] E. J. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[11] M. Wakin *et al.*, "Compressive imaging for video representation and coding," in *Proc. Picture Coding Symp.*, Beijing, China, Apr. 2006, pp. 1–6.

[12] V. Stankovic, L. Stankovic, and S. Cheng, "Compressive video sampling," in *Proc. 16th Eur. Signal Process. Conf.*, Lausanne, Switzerland, Aug. 2008, pp. 1–5.

[13] J. Prades-Nebot, Y. Ma, and T. Huang, "Distributed video coding using compressive sampling," in *Proc. Picture Coding Symp.*, Chicago, IL, USA, May 2009, pp. 1–4.

[14] Z. Liu, A. Elezzabi, and H. Zhao, "Maximum frame rate video acquisition using adaptive compressed sensing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 11, pp. 1704–1718, Nov. 2011.

[15] S. Mun and J. E. Fowler, "Residual reconstruction for block-based compressed sensing of video," in *Proc. Data Compression Conf.*, Snowbird, UT, USA, Mar 2011, pp. 183–192.

[16] C. Chen, E. W. Tramel, and J. E. Fowler, "Compressed-sensing recovery of images and video using multihypothesis predictions," in *Proc. 45th Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2011, pp. 1193–1198.

[17] J. Ma, G. Plonka, and M. Yousuff Hussaini, "Compressive video sampling with approximate message passing decoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 9, pp. 1354–1364, Sep. 2012.

[18] Y. Liu, M. Li, and D. A. Pados, "Motion-aware decoding of compressed-sensed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 3, pp. 438–444, Mar. 2013.

[19] Y. M. Lu and M. N. Do, "A theory for sampling signals from a union of subspaces," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2334–2345, Jun. 2008.

[20] T. Blumensath, "Sampling and reconstructing signals from a union of linear subspaces," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4660–4671, Jul. 2011.

[21] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5302–5316, Nov. 2009.

[22] Y. C. Eldar, P. Kuppinger, and H. Böcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3042–3054, Jun. 2010.

[23] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.

[24] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 2790–2797.

[25] G. Liu *et al.*, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[26] E. Elhamifar and R. Vidal.,"Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[27] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1672–1680, Sep. 2008.

[28] L. Lim and P. Comon, "Multiarray signal processing: Tensor decomposition meets compressed sensing," *Comptes Rendus Mecanique*, vol. 338, no. 6, pp. 311–320, Jun. 2010.

[29] M. Duarte and R. Baraniuk, "Kronecker product matrices for compressive sensing," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Dallas, TX, USA, Mar. 2010, pp. 3650–3653.

[30] N. Sidiropoulos and A. Kyrillidis, "Multi-way compressed sensing for sparse low-rank tensors," *IEEE Signal Process. Lett.*, vol. 19, no. 11, pp. 757–760, Nov. 2012.

[31] Q. Li, D. Schonfeld, and S. Friedland, "Generalized tensor compressive sensing," in *Proc. IEEE Int. Conf. Multimedia Expo.*, San Jose, CA, USA, Jul. 2013, pp. 1–6.

[32] S. Friedland, Q. Li, and D. Schonfeld, "Compressive sensing of sparse tensors," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4438–4447, Oct. 2014.

[33] X. He, D. Cai, and P. Niyogi, "Tensor subspace analysis," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2005, pp. 499–506.

[34] J. Ye, "Generalized low rank approximations of matrices," *Mach. Learn.*, vol. 61, no. 1-3, pp. 167–191, Nov. 2005.

[35] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "MPCA: Multilinear principal component analysis of tensor objects," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 18–39, Jan. 2008.

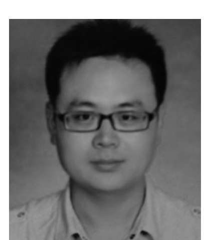
[36] D. L. Lieven, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[37] Y. Pang, X. Li, and Y. Yuan, "Robust tensor analysis with L1-norm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 2, pp. 172–178, Feb. 2010.

[38] E. van den Berg and M. P. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *SIAM J. Sci. Comput.*, vol. 31, no. 2, pp. 890–912, Nov. 2008.

[39] E. van den Berg and M. P. Friedlander, "Sparse optimization with least-squares constraints," *SIAM J. Optim.*, vol. 21, no. 4, pp. 1201–1229, 2011.

[40] T. G. Kolda *et al.*, MATLAB Tensor Toolbox Version 2.5, Jan. 2012. [Online]. Available: http://www.sandia.gov/ tgkolda/TensorToolbox/

**Yong Li** (S'15) received the B.S. degree from Xuzhou Normal University, Xuzhou, China, and the M.S. degree from China University of Mining and Technology, Xuzhou, China, in 2009 and 2012, respectively. He is currently working toward the Ph.D. degree with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. From November 2014 to May 2015, November 2015 to May 2016, he was with the Department of Biomedical Informatics, University of California, San Diego (UCSD), as a visiting scholar. His current research interests include compressive sensing, sparse representation, and image/signal processing.

**Wenrui Dai** (M'15) received the B.S., M.S., and Ph.D. degrees in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2005, 2008, and 2014. He is currently a Postdoctoral Scholar with the Department of Biomedical Informatics, University of California, San Diego, CA, USA. His research interests include learning-based image/video coding, image/signal processing, and predictive modeling.

**Junni Zou** (M'07) received the M.S. and the Ph.D. degrees in communication and information system from Shanghai University, Shanghai, China, in 2004 and 2006, respectively.

She is currently a full Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University (SJTU), Shanghai, China. From 2006 to 2016, she was with the School of Communication and Information Engineering, Shanghai University, Shanghai, and became a full Professor in 2014. From June 2011 to June 2012, she was with the Department of Electrical and Computer Engineering, University of California, San Diego, CA, USA, as a visiting Professor. Her research interests include multimedia communication, network resource optimization, wireless communication, and network information theory. She has published more than 80 IEEE journal/conference papers, and 2 book chapters, including 15 IEEE Transactions journal papers. She holds 12 patents and has more than 10 under reviewing patents.

Dr. Zou was granted National Science Fund for Outstanding Young Scholar in 2016. She received Shanghai Yong Rising Star Scientist Award in 2011. She received the First Prize of the Shanghai Technological Innovation Award in 2011, and the First Prize of the Shanghai Science and Technology Advancement Award in 2008. Also, she has served on some technical program committees for the IEEE and other international conferences.

**Hongkai Xiong** (M'01–SM'10) received the Ph.D. degree in communication and information system from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2003.

Since 2003, he has been with the Department of Electronic Engineering, SJTU, where he is currently a full Professor. From December 2007 to December 2008, he was with the Department of Electrical and Computer Engineering, Carnegie Mellon University (CMU), Pittsburgh, PA, USA, as a Research Scholar. From 2011 to 2012, he was a Scientist with the Division of Biomedical Informatics, University of California (UCSD), San Diego, CA, USA. His research interests include source coding/network information theory, signal processing, computer vision, and machine learning. He has published more than 180 refereed journal/conference papers.

Dr. Xiong was granted the Yangtze River Scholar Distinguished Professor from Ministry of Education of China in 2016, and the Youth Science and Technology Innovation Leader from Ministry of Science and Technology of China. He became Shanghai Academic Research Leader. In 2014, he was granted National Science Fund for Distinguished Young Scholar and Shanghai Youth Science and Technology Talent as well. In 2013, he received Shanghai Shu Guang Scholar. From 2012, he is a member of Innovative Research Groups of the National Natural Science. In 2011, he obtained the First Prize of the Shanghai Technological Innovation Award for Network-oriented Video Processing and Dissemination: Theory and Technology. In 2010 and 2013, he obtained the SMC-A Excellent Young Faculty Award of Shanghai Jiao Tong University. In 2009, he received New Century Excellent Talents in University, Ministry of Education of China. He served as TPC members for prestigious conferences such as ACM Multimedia, ICIP, ICME, and ISCAS. He received the Top 10% Paper Award at the 2016 IEEE Visual Communication and Image Processing, the Best Student Paper Award at the 2014 IEEE Visual Communication and Image Processing, the Best Paper Award at the 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, and the Top 10% Paper Award at the 2011 IEEE International Workshop on Multimedia Signal Processing (IEEE MMSP'11).

**Yuan F. Zheng** (F'97) received the M.S. and Ph.D. degrees in electrical engineering from The Ohio State University, Columbus, OH, USA, in 1980 and 1984, respectively. He received undergraduate education at Tsinghua University, Beijing, China, in 1970. From 1984 to 1989, he was with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA.

Since August 1989, he has been with The Ohio State University, where he is currently Winbigler Designated Chair Professor and was the Chairman of the Department of Electrical and Computer Engineering from 1993 to 2004. From 2004 to 2005, he spent sabbatical year at the Shanghai Jiao Tong University, Shanghai, China, and continued to be involved as the Dean of School of Electronic, Information and Electrical Engineering until 2008. His research interests include two aspects. One is in wavelet transform for image and video, radar waveform and signal, and object tracking. The other is in robotics which includes robotics for life science applications, multiple robots coordination, legged walking robots, and service robots. He has been on the editorial board of five international journals. For his research contributions, he received the Presidential Young Investigator Award from President Ronald Reagan in 1986, and the Research Awards from the College of Engineering of The Ohio State University in 1993, 1997, and 2007, respectively. He along with his students received the best conference and best student paper award a few times in 2000, 2002, 2006, 2009, and 2010, respectively, and received the Fred Diamond for Best Technical Paper Award from the Air Force Research Laboratory in Rome, New York, NY, USA, in 2006. In 2004 and 2005, he was appointed to the International Robotics Assessment Panel by the NSF, NASA, and NIH to assess the robotics technologies worldwide. Most recently in 2017, he received the Innovator of the Year Award from The Ohio State University.