Union of Data-driven Subspaces via Subspace Clustering for Compressive Video Sampling

Yong Li, Hongkai Xiong

Department of Electronic Engineering, Shanghai Jiao Tong Univ.
Shanghai, 200240, China
Email:{marsleely,xionghongkai}@sjtu.edu.cn

Abstract

The standard Compressive Sensing (CS) theory indicates that robust signals recovery can be obtained from just a few collection of incoherent projections. To further decrease the necessary measurements, an alternative to the generic CS framework assumes that signals lie on a union of subspaces (UoS). However, UoS model is limited to the specific type of signal regularity. This paper considers a more general and adaptive model which presumes that signals lie on a union of data-driven subspaces (UoDS). The UoDS model inherits the merit from UoS that signals have structural sparse representation. Meanwhile, it allows to recover signals using fewer degrees of freedom for a desirable recovery quality than UoS. To construct the UoDS model, a subspace clustering method is utilized to form an adaptive group set. The corresponding adaptive basis is learned by applying a linear subspace learning (LSL) method to each group. A corresponding recovery algorithm with provable performance is also given. Experiment results demonstrate that the proposed model for video sampling is valid and applicable.

1 Introduction

Compressive Sensing (CS)[1] is a new framework for signal acquisition and recovery. CS attempts to acquire the unknown signal which is sparse with respect to a given basis. It randomly projects the original signal onto a space (observation) whose dimension is much smaller. Recently, CS has been applied to video acquisition and recovery [4]–[9]. It can relax the hardware limitations and reduce the number of measurements to be sampled, thereby relieving the burden of the video encoder. Meanwhile the recovery can be guaranteed by a sparse representation with certain basis and an effect reconstruction method at the decoder side.

Wakin *et al.* [4] first applied CS to video acquisition where the video sequences are treated as a signal to be compressively sampled and afterwards 3-D wavelet transform is applied to jointly recover these sequences. In [5], a block-based CS (BCS) method was proposed to apply CS to sample non-overlapping blocks of frames and reconstruct them by discrete cosine transform (DCT) basis. Prades-Nebot *et al.* [6] proposed a distributed BCS framework where each block is approximated by a linear combination of blocks in previous frames. Liu *et al.* [7] proposed an adaptive framework where adaptive CS strategy is applied to blocks of different types. In [8], a motion-compensated BCS method was proposed with smooth projected Landweber (SPL) reconstruction. In [9], a BCS framework was proposed where Karhunen-Loève transform (KLT) basis is used to recover blocks in the decoder.

1068-0314/14 $31.00 © 2014 IEEE
DOI 10.1109/DCC.2014.21

The existing methods mostly consider simple sparsity which assumes that the signal x of interest lives in a given single subspace. Recently, there has been growing interest in assumption that x lives in a union of subspaces (UoS)[2],[3]. Because signals based on UoS model have structural sparsity (e.g. tree-sparse or block-sparse) which goes beyond simple sparsity, the necessary measurements are decreased. However, the bases of these subspaces are predefined (e.g. DCT, wavelet) to be irrespective of the nonstationarity of natural signals. Thus, UoS model is limited to the specific type of signal regularity. For multimedia signals, especially images and videos, there exists a variety of local structures with different types of signal regularity, which makes UoS model ineffective. Such problem motivates us to develop an adaptive and general model for video sampling and reconstruction.

In this paper, a union of data-driven subspaces (UoDS) model is proposed for compressive video sampling, where each patch is considered to be lying on a UoDS. We construct an adaptive group set for UoDS via sparse subspace clustering which separates signals according to their underlying subspaces. The corresponding adaptive basis is learned by applying a linear subspace learning (LSL) method to each group. In the encoder, key frames (KFs) are fully sampled while non-key frames (NKFs) are block-wise sampled by random sensing matrix. At the decoder side, the training set of the UoDS can be learned by sparse subspace clustering from previous decoded key-frame (KF). Subsequently, each non-overlapping patch of NKFs can be stably recovered with the adaptive basis of UoDS. The proposed model can leverage local and nonlocal structures which are embodied by video frames within nonlocal area. The spatio-temporal sparsity is enhanced and the nonlocal structural basis is adaptively derived, which makes the reconstruction more efficient than previous schemes.

2 Background

An unknown signal $x \in \mathbb{R}^{n\times 1}$ has a sparse representation over an orthonormal basis Ψ in form $x = \Psi c$, where c is the representation vector which has k nonzero components. In other words, x lives in a k-dimensional single subspace spanned by k basis vectors of Ψ. CS considers that x can be recovered from its m measurements $y_i, i = 1, \dots, m$. Commonly, the measurements are obtained by linear sampling $y = \Phi x$, where $y \in \mathbb{R}^{m\times 1}$, sensing matrix $\Phi \in \mathbb{R}^{m\times n}$, $m \ll n$, $m > k$. m/n denotes the sampling rate (SR). The linear sampling can be represented explicitly in form $y = Ac$, where $A = \Phi\Psi$. It is demonstrated that when A satisfies certain conditions, c can be recovered exactly from $m = O(k \log \frac{n}{k})$ random measurements by solving ℓ_1-norm minimization problem[1]:

$$\min_c \|c\|_1, \quad \textit{subject to} \quad y = Ac. \tag{1}$$

2.1 Union of Subspaces (UoS)

Instead of single subspace model, a general sampling framework considers x to be lying on a union of subspaces:

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

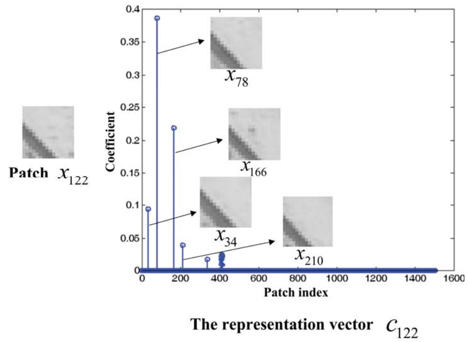

Figure 1: The representation vector c_{122} of 16×16 patch x_{122} under $X, K = 1505$ from the first frame of Foreman sequence.

where $\mathcal{S}_\lambda$ is a subspace of Hilbert space $\mathcal{H}$ and Λ is a list of indices, $\mathcal{S}_\lambda$ is spanned by a predefined basis (e.g. DCT or wavelet basis) [2].

Based on the UoS model, a block-sparse structure was considered in [3]. $c^T = [c[1]^T \ \cdots \ c[t]^T]$ is called k-block-sparse if at most k blocks $c[i]_{d_i \times 1}$ are nonzero. In this case, block restricted isometric property (Block-RIP) imposed on A was defined to guarantee a stable recovery. It was proved that if the sensing matrix A satisfies the block RIP, we can recover the block sparse vector c by a convex algorithm which is based on minimizing a mixed ℓ_2/ℓ_1 norm:

$$\min_{c} \|c\|_{2,1}, \quad \textit{subject to} \;\; y = Ac \tag{3}$$

where $\|c\|_{2,1} \triangleq \sum_{i=1}^{t} \|c[i]\|_2$, $\|c[i]\|_2 \geq 0, 1 \leq i \leq t$.

3 Union of Data-driven Subspaces (UoDS) model

Video contains different types of signal regularity. The spatial redundancy in a frame is often demonstrated by the repetitiveness of regular texture and structure. Meanwhile, the temporal redundancy in an video is displayed by tiny content-changing among neighboring frames. Thus, we desire a more general and adaptive model for patches in video.

3.1 UoDS

Given an image (e.g. key frame), the vectorized patches set $X = [x_1, x_2, \ldots, x_K]$ exacted from the image is segmented into t clusters by Sparse Subspace Clustering (SSC)[10] which aims at separating data in terms of their underlying subspaces. SSC is rooted on the fact that each point in a union of subspaces has a sparse representation with respect to a dictionary which is formed by all other data points with regard to the self-expressiveness property. It is noted that patches extracted from the image can be overlapped. For each vectorized patch $x_i \in \mathbb{R}^{n \times 1}$, we obtain its representation vector c_i under the dictionary X by

$$\min_{c_i} \|c_i\|_1 \quad \textit{subject to} \quad x_i = Xc_i, \quad c_i[i] = 0, \tag{4}$$

The non-zero coefficients of c_i corresponds to the patches from the same subspace. Fig. 1 shows the c_{122} under $X, K = 1505$. Note that patches $x_{34}, x_{78}, x_{122}, x_{166}, x_{210}$

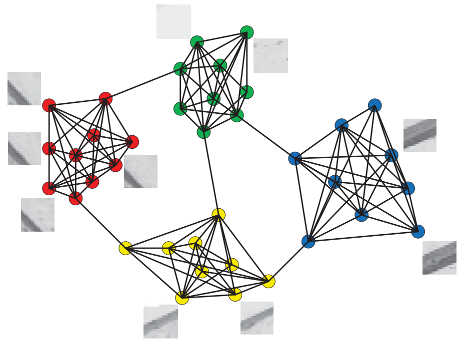

Figure 2: The similarity graph where there are four components. Each component denotes a cluster.

are similar and they are lying on the same subspace. For all patches, Eq. (4) can be rewritten in matrix form as

$$\min_{C} \|C\|_1 \quad subject\ to \quad X = XC, diag(C) = 0, \tag{5}$$

where $C \triangleq [c_1\ c_2\ \dots\ c_K] \in \mathbb{R}^{K\times K}$.

We normalize columns of C as $c_i \leftarrow \frac{c_i}{\|c_i\|_\infty}$ and build a weighted balanced similarity graph $\mathcal{G} = (V, E, W)$, V denotes the set of K vertices corresponding to K patches, E denotes the edge set with some edge (v_i, v_j) when patch x_i can be represented by a linear combination of some patches containing x_j. Take Fig. 1 for example, $x_{122} = 0.39\ x_{78} + 0.22\ x_{166} + \cdots$, therefore there exist edges (v_{122}, v_{78}) and (v_{122}, v_{166}) with weights $c_{122}[78] = 0.39$ and $c_{122}[166] = 0.22$. The similarity matrix $W = |C| + |C|^T \in \mathbb{R}^{K\times K}$ represents the weights of the edges and its element $w_{ij} = |c_i[j]|+|c_j[i]|^T$. Fig. 2 shows an example of the similarity graph model. Ideally, patches in the same subspace are connected while disconnected in different subspaces.

We adopt spectral clustering to the similarity matrix W, where the Laplacian matrix $L = D - W$ is initially attained and D is a diagonal matrix with $D_{ii} = \sum_j W_{ij}$. Applying K-means to the t eigenvectors of L corresponding to the smallest t eigenvalues, the patches set X is thereby segmented to t clusters $X_i, i = 1, \dots, t$. Each cluster corresponds to a linear low dimension subspace $\mathcal{S}_i^*$. Further, each patch x of the image lives in this union of data-driven subspaces (UoDS) as

$$x \in \mathcal{U}^* \triangleq \bigcup \mathcal{S}_i^* \tag{6}$$

which is depicted in Fig. 3.

After SSC, the basis of $\mathcal{U}^*$ can be derived by training set $X = [X_1, X_2, \dots, X_t]$. We perform PCA on each cluster X_i to learn corresponding linear subspace independently. Therefore, $\mathcal{U}^* = \{\mathcal{S}_i^*\}_{i=1}^t$ is a union of t low rank linear subspaces. $\Psi^* = [\Psi_1^*, \Psi_2^*, \dots, \Psi_t^*]$ is the set of bases for $\mathcal{U}^*$. $\Psi_i^* \in \mathbb{R}^{n\times d_i}$ spans its corresponding d_i-dimensional subspace $\mathcal{S}_i^*$. The basis Ψ^* of the UoDS model is adaptive and nonlocal because it is learned from an entire image instead of a fix searching window. Note that the dimension of each subspace may be different, depending on the type of structure embodied by patches in each cluster. From Fig. 3, patches from plain background should lie on a lower dimensional subspace than patches from area with dramatic change of texture. Besides, the basis $\Psi^* \in \mathbb{R}^{n\times r}$ should be over-complete thereby $r = \sum_{i=1}^t d_i > n$.

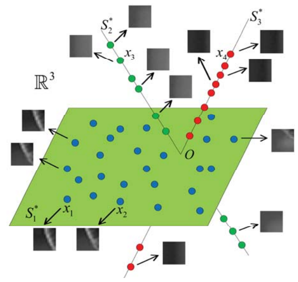

Figure 3: Patches based on the UoDS model.

Ψ_i^* is the solution provided by singular value decomposition (SVD) of X_i which can be shown as $X_i = \Psi_i^* \Sigma V^T$:

$$\Psi_i^* = \arg\max_{\Psi_i} \|\Psi_i^T X_i\|_2, \quad \text{subject to } \Psi_i^T \Psi_i = I_{d_i}, \tag{7}$$

To obtain a subspace which is not only robust to outliers but also invariant to rotations, the PCA-L1 algorithm [11] can be used for each cluster instead:

$$w_{iq}^* = \arg\max_{w_{iq}} \|w_{iq}^T X_i\|_1 = \arg\max_{w_{iq}} \sum_{j=1}^{l_i} |w_{iq} x_{ij}|, \quad \text{subject to } w_{iq}^T w_{iq} = 1, \tag{8}$$

where $q = 1, \ldots, d_i$ is the order of vectors in the basis $\Psi_i^* = [w_{i1}^*, w_{i2}^*, \ldots, w_{id_i}^*]$, l_i is the number of patches in the cluster X_i. By the greedy algorithm to get the basis Ψ_i^*, Fig. 4 shows the representation vectors of the same signal lying on the UoDS spanned from PCA and PCA-L1 basis, respectively.

3.2 Stable Recovery

Based on the UoDS model, the linear sampling can be described as

$$\begin{aligned} y_{m\times 1} &= \Phi_{m\times n} x_{n\times 1} \\ &= \Phi_{m\times n} \Psi_{n\times r}^* c_{r\times 1}^* = A^* c^* \end{aligned} \tag{9}$$

where Φ is an i.i.d. random matrix, $r = \sum_{i=1}^{t} d_i$, $A^* = [A_1^*, \ldots, A_t^*]$.

The UoDS model inherits the merit of UoS model. Therefore, signal of interest will have a block-sparse representation over Ψ^*. Ideally, if these subspaces are disjoint or independent, c^* will be a 1-block-sparse vector which is more sparser than c in Eq.(3). From Fig. 4, it is obvious that the sparsity of the proposed scheme exists in only one block of c^*. As follows, the uniqueness and stability conditions are given for a self-contained description.

Suppose $\mathcal{S}_{ij}^*$ is the convex hull of the set of two different data-driven subspaces $\mathcal{S}_i^* \bigcup \mathcal{S}_j^*$, the maximum dimension of $\mathcal{S}_{ij}^*$ can be defined as $k_{max} = \max_{i\neq j} \dim(\mathcal{S}_{ij}^*)$. Therefore, we have the following sampling requirement.

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

Though $\mathcal{U}^*$ is data-driven, the sampling still follows the property similar with that of predefined UoS. The proof is similar with that of Proposition 3 in [2]. Proposition 1 tells us the minimum number of samples needed to guarantee a stable reconstruction.

If Φ is replaced by A^* and x is replaced by k-block sparse vector c^*, and set $u = c_1^* - c_2^*$, then Proposition 2 can be shown as:

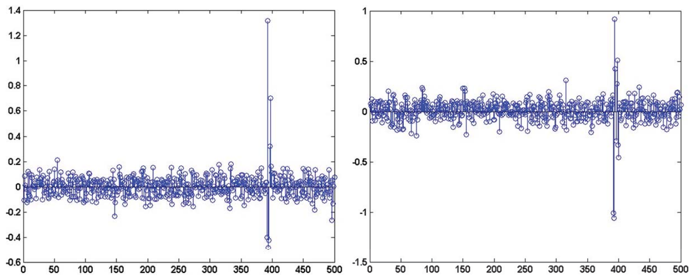

Figure 4: Representation on PCA and PCA-L1 basis. There are 50 subspaces and the dimension of each subspace is 10. Easy to find that the signal is lying on the 40th subspace.

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

$$C_1\|u\|_2^2 \le \|A^*u\|_2^2 \le C_2\|u\|_2^2 \tag{10}$$

Proof: First, $A^* = \Phi\Psi^*$ in terms of Eq. (9). The basis Ψ_i^* is an orthonormal basis for each subspace S_i^* obtained by PCA or PCA-L1. Besides, Φ is an i.i.d. random matrix. According to Proposition 4 and Proposition 5 in [2], we can easily prove Proposition 2. ■

If A^* satisfies the block-RIP condition with $\delta_{2k} \le \sqrt{2} - 1$, the vector c^* of Eq. (11) can be determined according to a convex second-order cone program (SOCP)[3].

$$\min_{c^*} \|c^*\|_{2,1} \quad \textit{subject to } y = A^*c^* \tag{11}$$

We can reconstruct c^* by group-BP algorithm[12].

3.3 UoDS for video sampling and recovery

Fig. 5 depicts the procedure of the proposed model for video sampling and recovery. Given a video sequence with group of pictures (GOP), it is decomposed into a set of key frames (KFs) and the remaining non-key frames (NKFs). We first fully sample KF while blocks of non-key frame are sampled with low sampling rate. Then, the recovered KF is decomposed into several data-driven groups to form a UoDS $\mathcal{U}^*$ by SSC. The corresponding base Ψ^* can be derived by LSL. Due to the spatio-temporal consistency in video sequence, each non-overlapping block x in NKFs lies on $\mathcal{U}^*$. Finally, the block-sparse vector c^* is recovered according to Eq. (11), then we can recover these blocks by $x = \Psi^*c^*$, and immediately assemble them back to form the decoded non-key frames.

4 Experimental Results and Discussion

In this section, experiments are conducted on a variety of video sequences with CIF (352×288) resolution (i.e., *Bike, Bus, Football, NBA*) and DVD (720×480) resolution (i.e. *Driving, Whale Show*). The size of each non-overlapping block is 8×8, 16×16 thereby $n = 64$ and 256, respectively. 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 $SR \in \{0.1, 0.2, \ldots, 0.6\}$. Each GOP contains 10 grayscale frames. Without loss of

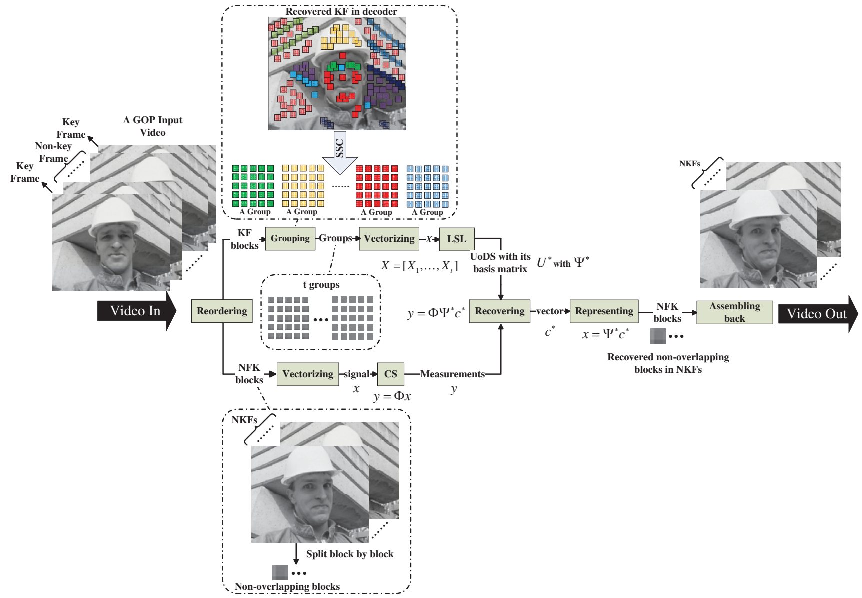

Figure 5: The whole proposed frame for compressive video sampling

generality, the first frame of each GOP is set as the key frame and the remaining nine frames as the non-key frames. Take *Football* sequence for example. For SSC [1], the blocks set is formed by overlapping blocks with $\frac{1}{2}\sqrt{n}$ pixels per step in both rows and columns. When block size is 16×16 and step is 8, the blocks set contains 1505 blocks, thereby the sparse representation matrix $C \in \mathbb{R}^{1505\times1505}$. Later, the blocks of key frame are partitioned into 50 clusters by applying spectral clustering.

The bases of UoDS is learned from each cluster by PCA and PCA-L1. For simplicity, the dimension of each subspace d_i is the same chosen from $\{2, 4, 6, 8, 10, 20\}$. Through the SPGL1 Matlab solver [2] [12], the proposed schemes (UoDS-PCA and UoDS-PCAL1) are compared with four compressive video sampling methods: BCS-DCT [5], BCS-KLT [9], MC-BCS-SPL [8], BCS-UoS-DCT. It is worth mentioning that BCS-DCT and MC-BCS-SPL are based on single subspace model with predefine basis, while BCS-KLT uses adaptive basis. BCS-UoS-DCT method is based on UoS model with fix basis, and MC-BCS-SPL involves with motion compensation. The experimental environment: MATLAB in a workstation with 3.2-GHz CPU and 12-GB RAM.

Fig. 6 shows the subjective visual quality of the recovered frames. Table. 1 and Table. 2 provide the overall and averaged R-D performance. Fig. 7 depicts the performance of the proposed scheme with different subspace dimension d. From Table. 1 and Table. 2, it can be observed that the proposed scheme behaves better than other methods in general, especially for complex or large-motion scenes, such as *Bike, Bus, Football, NBA, Whale show* sequences. Comparing with BCS-DCT and BCS-KLT, the proposed model generates structured sparsity which leads to fewer necessary measurements to be sampled for a desire recovery quality. MC-BCS-SPL outperforms UoDS at a low sampling rate because of motion compensation. For small-motion scene

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

[2] Available at http://www.cs.ubc.ca/∼mpf/spgl1/

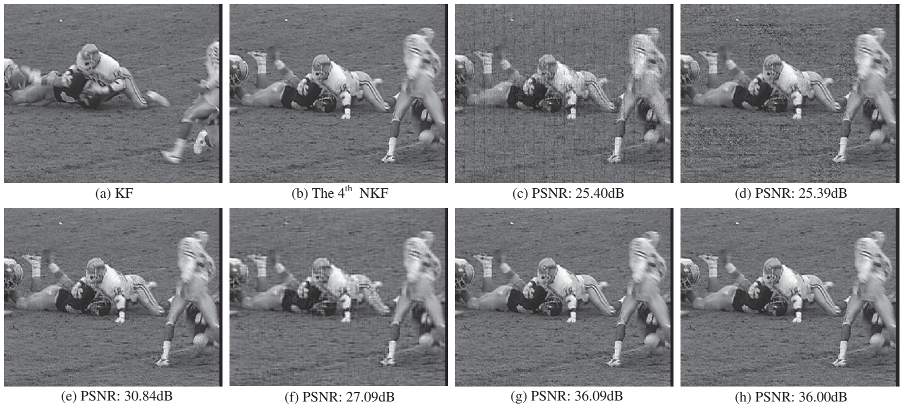

Figure 6: The experimental results on *Football* with 16×16 blocks, SR=0.6 and $d = 10$. (a): the key frame; (b): the 4th non-key frame; (c): BCS-DCT [5]; (d): BCS-KLT [9]; (e): MC-BCS-SPL [8]; (f): BCS-UoS-DCT; (g): UoDS-PCA; (h): UoDS-PCAL1.

Table 1: Average PSNR in dB for several video sequences with 8×8 block, $d = 6$.

	Sampling Rate (m/n)					
Algorithm	0.1	0.2	0.3	0.4	0.5	0.6
Bike						
UoDS-PCA	11.57	16.27	20.67	25.39	28.33	**31.05**
UoDS-PCAL1	11.81	16.23	20.65	25.35	**28.37**	31.04
BCS-DCT	9.57	12.65	15.56	17.27	19.44	21.25
BCS-KLT	8.27	13.79	15.12	18.35	18.97	19.00
MC-BCS-SPL	**19.25**	**21.93**	**23.18**	**25.64**	27.83	29.03
BCS-UoS-DCT	10.69	14.30	17.86	22.38	23.89	25.68
Bus						
UoDS-PCA	13.21	17.69	22.12	26.45	**29.49**	31.92
UoDS-PCAL1	15.25	16.29	22.52	25.85	29.01	**32.30**
BCS-DCT	10.60	13.92	16.16	18.47	20.22	21.92
BCS-KLT	10.29	13.11	16.23	19.22	21.55	24.17
MC-BCS-SPL	**20.72**	**22.95**	**25.79**	**26.71**	29.15	30.90
BCS-UoS-DCT	11.78	15.38	19.34	23.80	24.18	26.43
Football						
UoDS-PCA	9.96	15.87	21.93	28.02	31.49	34.25
UoDS-PCAL1	10.01	15.93	21.98	28.09	**31.75**	**34.53**
BCS-DCT	7.68	11.51	14.82	16.96	19.03	21.05
BCS-KLT	6.41	8.85	10.50	13.16	17.08	20.15
MC-BCS-SPL	**21.77**	**25.15**	**27.03**	**29.85**	30.73	33.41
BCS-UoS-DCT	8.99	13.14	17.77	24.27	26.69	29.55
NBA						
UoDS-PCA	10.38	18.19	20.89	24.43	26.89	29.73
UoDS-PCAL1	8.32	15.26	20.25	**24.49**	**27.65**	**29.99**
BCS-DCT	8.13	11.60	13.85	16.13	17.87	19.61
BCS-KLT	7.92	10.93	13.31	16.12	18.73	21.26
MC-BCS-SPL	**17.57**	**18.90**	**20.96**	23.73	26.12	28.02
BCS-UoS-DCT	9.45	13.20	17.23	20.88	22.43	25.33

(e.g. Driving), MC-BCS-SPL outperforms UoDS overall. However, for large-motion scene, motion compensation sometimes degrades the reconstruction performance. No matter how large the motion is, the moving objects are still contained in the key frame. The nonlocal basis of UoDS is learned from the whole key frame. Therefore, our proposed model can work well for large-motion scenes.

Additionally, from Table. 1 and Table. 2, performance of 16×16 situation is much better than that of 8×8. Because when n is bigger, the reconstruction algorithm can reconstruct more accurate c^* with higher probability. In addition, the dimension of each subspace d also affects the performance of the proposed method as showed in Fig. 7. When d is small at a low SR (e.g. $d = 2, SR = 0.1$), the PSNR value is bigger than that of larger d. Because the decoded KF is decomposed into 50 clusters, from Eq. (9), $t = 50, d = 2, SR = 0.1, n = 64$, then $m = 6, r = 100$, while $d = 4, SR = 0.1, n = 64$, then $m = 6, r = 200$, therefore, c^* is more accurate when

Table 2: Average PSNR in dB for several video sequences with 16×16 block, $d = 10$.

Algorithm	Sampling Rate (m/n)					
	0.1	0.2	0.3	0.4	0.5	0.6
Bike						
UoDS-PCA	16.56	21.41	24.33	26.60	**29.41**	**32.30**
UoDS-PCAL1	16.81	21.45	24.41	26.66	29.33	**32.30**
BCS-DCT	13.53	16.59	18.53	20.19	22.01	24.12
BCS-KLT	12.13	14.90	17.22	20.67	23.20	26.23
MC-BCS-SPL	**19.77**	**22.29**	**24.58**	**26.77**	28.25	30.27
BCS-UoS-DCT	17.12	18.63	20.89	22.04	22.76	23.15
Bus						
UoDS-PCA	18.96	23.24	25.68	27.95	29.99	**32.86**
UoDS-PCAL1	19.37	23.06	25.55	27.84	**30.25**	32.81
BCS-DCT	13.96	16.87	19.10	20.73	22.73	25.02
BCS-KLT	13.82	17.65	20.61	22.72	25.19	28.05
MC-BCS-SPL	**20.61**	**23.94**	**26.72**	**28.03**	29.86	32.10
BCS-UoS-DCT	18.49	20.47	23.00	24.02	24.50	24.79
Football						
UoDS-PCA	17.00	24.35	27.84	30.44	**33.15**	**36.09**
UoDS-PCAL1	16.97	24.37	**28.06**	**30.53**	33.08	35.99
BCS-DCT	13.20	16.83	19.11	21.18	23.29	25.40
BCS-KLT	9.49	15.20	16.21	18.94	22.84	25.39
MC-BCS-SPL	**22.95**	**25.92**	27.99	30.15	31.88	34.10
BCS-UoS-DCT	18.47	21.58	25.04	26.23	26.79	27.09
NBA						
UoDS-PCA	13.83	19.87	22.92	25.32	28.23	31.54
UoDS-PCAL1	16.85	20.03	22.68	**25.43**	**28.41**	**31.63**
BCS-DCT	12.13	14.75	16.66	18.36	20.39	22.43
BCS-KLT	11.71	14.63	17.20	19.38	21.91	24.67
MC-BCS-SPL	**17.15**	**20.23**	**23.10**	25.05	27.00	28.76
BCS-UoS-DCT	15.12	15.96	19.26	20.23	20.71	21.04
Driving						
UoDS-PCA	17.57	25.51	28.95	31.00	33.07	35.18
UoDS-PCAL1	17.83	25.84	29.10	31.08	33.18	35.21
BCS-DCT	13.18	16.91	19.32	21.46	23.56	25.90
BCS-KLT	13.52	18.69	22.15	25.07	27.82	30.54
MC-BCS-SPL	**23.47**	**28.66**	**30.80**	**32.22**	**33.94**	**35.61**
BCS-UoS-DCT	19.49	23.97	26.85	27.73	28.18	28.48
Whale Show						
UoDS-PCA	19.22	25.46	28.20	30.55	**32.89**	35.41
UoDS-PCAL1	19.01	25.39	**28.29**	**30.56**	32.86	**35.42**
BCS-DCT	12.71	16.24	18.69	20.80	22.93	25.32
BCS-KLT	12.15	16.99	20.75	23.99	27.01	29.98
MC-BCS-SPL	**22.53**	**25.51**	27.94	29.62	31.55	33.71
BCS-UoS-DCT	18.79	23.32	25.19	25.86	26.20	26.41

$d = 2$ than that when $d = 4$. Besides, from Fig. 7 (b),(d), bigger d at a high SR can achieve better results. Because $t = 50, d = 20, SR = 0.6, n = 1024$, then $r = 1000, m = 614$ provides enough measurements. At the same time, bigger d can provide more information for each subspace. However, from Fig. 7 (a),(c), when $d = 20, SR = 0.6, n = 64$, then $r = 1000, m = 38$ can not provide enough necessary measurements to recover c^* so that it fails.

5 Conclusions

This paper proposes an union of data-driven subspaces (UoDS) model. 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 sparsity representation. With the proof of stable reconstruction, the proposed scheme is fulfilled in video acquisition where the UoDS is learned from the decoded key frames. Experimental results show that the proposed model gets better performance in comparison to the other compressive video sampling methods. In our future work, we would like to investigate other subspace clustering methods and dictionary learning methods to improve the performance of the proposed method.

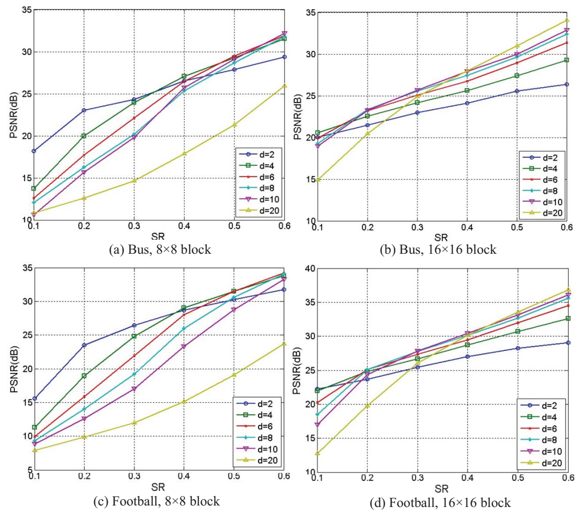

Figure 7: The performance of the proposed scheme with different dimension d. (a): *Bus*, 8×8; (b): *Bus*, 16×16; (c): *Football*, 8×8; (d): *Football*, 16×16.

References

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

[2] Yue M. Lu and Minh 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.

[3] Yonina C. Eldar, and Moshe Mishali, "Robust Recovery of Signals From a Structured Union of Subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5302-5316, Nov. 2009.

[4] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R. Baraniuk, "Compressive imaging for video representation and coding," in *Proc. Picture Coding Symp.*, Beijing, China, April 2006.

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

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

[7] 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.

[8] S. Mun and J. E. Fowler, "Residual reconstruction for block-based compressed sensing of video," in *Proceedings of the IEEE Data Compression Conference*, Snowbird, USA, pp. 183-192, March 2011.

[9] Ying Liu, Ming Li, and Dimitris A. Pados, "Motion-Aware Decoding of Compressed-Sensed Video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 3, pp. 438-444, Mar. 2013.

[10] E. Elhamifar and R. Vidal. "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765-2781, Mar. 2013.

[11] Nojun Kwak, "Principal Component Analysis Based on L1-Norm Maximization," *IEEE Trans. Pattern Analysis and Machine Intelligence.*, vol. 30, no. 9, pp. 1672-1680, Sep. 2008.

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