Sparse Spatio-Temporal Representation With Adaptive Regularized Dictionary Learning for Low Bit-Rate Video Coding

Hongkai Xiong, *Senior Member, IEEE,* Zhiming Pan, Xinwei Ye, and Chang Wen Chen, *Fellow, IEEE*

***Abstract*—For promising vision-based video coding on low-quality data, this paper proposes a sparse spatio-temporal representation with adaptive regularized dictionary learning and develops a low bit-rate video coding scheme. In a reversed-complexity Wyner–Ziv coding manner, it selects a subset of key frames to code at original resolution, while the rest are down sampled and reconstructed by a sparse spatio-temporal approximation using key frames as a training dataset. Since primitive patches (geometry) are of low dimensionality and can be well learned from the primitive patches across frames in a scale space, a video frame is divided into three layers: a primitive layer, a nonprimitive coarse layer, and a nonprimitive smooth layer. The multiscale differential feature representations are invertible to facilitate reconstruction with dictionary learning, and the target is formulated as an optimization problem by constructing a sparse representation of 2-D patches and 3-D volumes over adaptive regularized dictionaries, a set of 2-D subdictionary pairs trained from primitive patches, and a 3-D dictionary trained from nonprimitive volumes. Specifically, the nonprimitive layer is constructed as volumes in to order keep it consistent along the motion trajectory, which enables sparse representations over a learned 3-D spatio-temporal dictionary. Through hierarchical bidirectional motion estimation and adaptive overlapped block motion compensation, the 3-D low-frequency and high-frequency dictionary pair is designed by the K-SVD algorithm to update the atoms for optimal sparse representation and convergence. In reconstruction, the lost high-frequency information of the down-sampled frames can be synthesized from the sparse spatio-temporal representation over the adaptive regularized dictionaries. Extensive experiments validate the compression efficiency of the proposed scheme versus H.264/AVC in terms of both objective and subjective comparisons.**

***Index Terms*—Atom decomposition, dictionary learning, primitive patch, sparse representation, video coding.**

I. Introduction

INCREASINGLY, low-quality visual data from mobile phones, digital cameras, and mobile TV noticeably stimulate a huge demand for video analysis and computer vision techniques. There arises a big perspective on whether more disruptive techniques can provide substantial gains. An impressive observation for video coding is to establish a certain correlation between a sampled low-resolution version and high-resolution contents [1], [2]. For example, scalable video coding maintains the spatial capability through down-sampling and interlayer prediction with up-sampling. However, the coding burden is dominated by a rigid partition between the encoder (heavy) and decoder (light). It would constrain the ubiquitous multimedia access for increasing mobile communication. Ever since, distributed video coding (DVC) as a hopeful paradigm motivated by shifting the computationally intensive prediction at the encoder to the decoder accommodates requirements of mobile camera phones and wireless sensor networks [3]. Theoretically, it is inspired by the lossy Wyner–Ziv source coding theory where separate encoding of correlated sources can approach the rate of joint entropy, provided joint decoding is executed with a known correlation [4]. Limited by the estimation of correlated side information, practical DVC schemes often have a considerable performance loss compared to traditional H.264/AVC. It is noticeable that the 3-D sparse representations could explore the spatio-temporal consistency in a video sequence. In view of the fact that primitive patches have more sparse representation structures over dictionary and the nonprimitive volumes are consistent along the motion trajectory in the temporal dimension and have little structural information, it makes more sparse representations over a learned 3-D spatio-temporal dictionary in video coding. Along with the insight, it stimulates us to investigate sparse adaptive inverse reconstruction with advanced regularity in a DVC manner.

Revisiting the traditional video coding schemes, e.g., H.264/AVC and the ongoing High Efficiency Video Coding (HEVC), those focus on exploring redundancy among pixels through intra- and interprediction [5]. As a matter of fact, more prediction methods, e.g., inpainting-based prediction [6] and texture prediction [7], have been noticed to achieve a better performance, which infers a promising potential to synthesize and hallucinate missing image contents with good perceptual quality. A related approach using a texture analysis–synthesis scheme was developed in [8], which reduces the entropy of source information by clustering the homogeneous area into a small patch that contains the epitome content of all associated regions. Those can be treated in a unified manner under the

Manuscript received February 7, 2012; revised June 21, 2012; accepted August 8, 2012. Date of publication October 2, 2012; date of current version April 1, 2013. This work was supported in part by the National Natural Science Foundation of China under Grants U1201255, 61271218, 61271211, and 61228101, the Shanghai Rising-Star Program (11QA1402600). This paper was recommended by Associate Editor L.-P. Chau.

H. Xiong, Z. Pan, and X. Ye are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: xionghongkai@sjtu.edu.cn; pzming51@sjtu.edu.cn; yeahinv@sjtu.edu.cn).

C. W. Chen is with the Department of Computer Science and Engineering, State University of New York at Buffalo, Buffalo, NY 14260 USA (e-mail: chencw@buffalo.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/TCSVT.2012.2221271

framework of Markov random field (MRF) and optimization algorithms, e.g., belief propagation (BP), are concerned as an iterative solution. By now, the attempts to restore the missing information have involved in various assistant side information, e.g., edge [9] and assistant parameters [10]. To maintain the temporal consistency of a video, a space–time completion has ever been developed in a global optimization sense [11], [12]. Despite claims of a bit-rate saving at similar visual quality levels compared with the traditional H.264/AVC codec, these methods failed to ensure pixel-wise fidelity.

Naturally, more attention has been paid to the possibility of video reconstruction with state-of-the-art super-resolution approaches where a correlation between a sparsely sampled low-resolution version and high-frequency contents could be estimated in a nonparametric sense. In essence, it could be formulated as the procedure of reconstruction (interpolation) from incomplete data. It can be broadly classified into three categories: interpolation-based, reconstruction-based, and learning-based. Interpolation-based methods are founded on the assumption that the strong correlations within adjacent pixels tend to blur discontinuities and edges. Reconstruction-based methods introduce the prior knowledge as reconstruction constraints when regularizing the super-resolution image [13], [37], [38]. In [2], Shen *et al.* presented an example-based super-resolution approach. It contains a set that consists of nonadaptive low-resolution/high-resolution patch pairs to enhance the high-frequency detail in the reconstruction. Recently, learning-base approaches have achieved the best reconstruction results in the super-resolution task by inferring lost high-frequency information from a learned co-occurrence prior knowledge [14]. As in [15], an example-based learning strategy was proposed where the low-resolution to high-resolution prediction is learned via an MRF solved by BP. Sun *et al.* [18] extended it by using the primal sketch priors to enhance blurred edges, ridges, and corners. To overcome the deficiency of synthesizing each high-resolution patch from only one neighbor in the training set, [14] considered recovering the sparse representation coefficients of each low-resolution patch based on a dictionary composed of low-resolution patches. The high-resolution patch is then reconstructed using the recovered coefficients in terms of the corresponding high-resolution dictionary. This method adaptively selects the most relevant patches in the dictionary, which leads to a superior performance. However, its dictionary is learned from randomly chosen patches of arbitrary training images, which was efficient only for input images of similar statistical features.

This paper proposes a low bit-rate video coding scheme where sparse spatio-temporal representation over dictionary learning provides effective nonparametric approaches to inverse problems. In a reversed-complexity Wyner–Ziv coding manner, a subset of key frames are encoded at the original resolution and serve as a set of training data at the decoder side, while the remaining frames are coded at low resolution from down-sampling. It is recognized that the primitive patches (geometry) are of low dimensionality and can be well learned from the primitive patches across frames in a scale space. The multiscale differential feature representations are invertible to facilitate reconstruction with dictionary learning. Specifically, a video frame is divided into three layers: a primitive layer, a nonprimitive coarse layer, and a nonprimitive smooth layer. The target is formulated as an optimization problem by constructing a sparse representation of 2-D patches and 3-D volumes over adaptive regularized dictionary learning: a set of 2-D subdictionary pairs trained from primitive patches and a 3-D dictionary trained from nonprimitive volumes. It is worth mentioning that primitives may vary significantly across frames or patches in a frame. We would learn various sets of low-resolution or high-resolution subdictionary pairs from the primitive patches of the key frames. The nonprimitive volumes are supposed to be consistent along the motion trajectory with little structure and could bear more sparse representations over a learned 3-D spatio-temporal dictionary. Through hierarchical bidirectional motion estimation and adaptive overlapped block motion compensation, the 3-D low-frequency (LF) and high-frequency (HF) dictionary pair is designed by the K-SVD algorithm to update the atoms for optimal sparse representation and convergence. In reconstruction, the lost high-frequency information of the non-key frames can be synthesized from the sparse spatio-temporal representation over the adaptive regularized dictionaries. The final high-resolution frames can be acquired by combining all the high-frequency frames and low-frequency frames. Compared to H.264/AVC and other super-resolution-based schemes, experimental results validate that the proposed algorithm would not only ensures the visual quality but also is competitive in rate-distortion performance, e.g., rate-distortion and BD bitrate.

The remainder of this paper is organized as follows. We summarize all the abbreviations in Table I. Section II gives the proposed learning framework for video coding. In Section III, the design of the sparse spatio-temporal representation on adaptive regularized dictionaries is presented with sufficient discussion. Extensive experimental results are validated in Section IV on both objective and visual quality. Conclusions are drawn in Section V.

II. Learning Framework

The proposed scheme is depicted in Fig. 1. Given a video sequence F_h with group of pictures (GOP), it is decomposed into selected high-resolution (HR) key frames (KFs) X_h and the down-sampled low-resolution (LR) non-key frames (NKFs) Z_l. The KFs and NKFs are both encoded and decoded by a standardized H.264/AVC codec as $\hat{X}_h$ and $\hat{Z}_l$. An HR version of $\hat{Z}_l$ (denoted by $\hat{Z}_h$) is recovered from $\hat{X}_h$, using the learning-based super-resolution reconstruction.

As in [19] and [20], a frame is separated into two distinct frequency bands: LF band and HF band in Fig. 2. The low-frequency band is obtained by the down- and up-sampling operators, and the high-frequency band is complementary to the frequency response of the low-pass filter.

A. Incorporating the Sparse-Land Prior

The observed LR frame Z_l is a blurred and down-sampled version of the HR frame X_h: $Z_l = SHX_h$. Here, H represents a blurring filter and S the down-sampling operator.

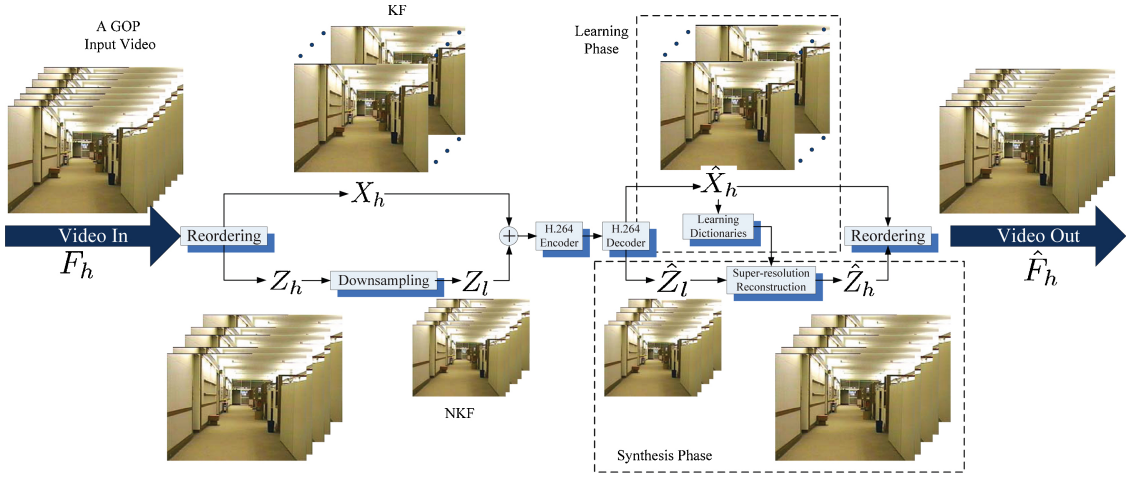


Fig. 1. Proposed learning framework of sparse spatio-temporal representation for video coding.

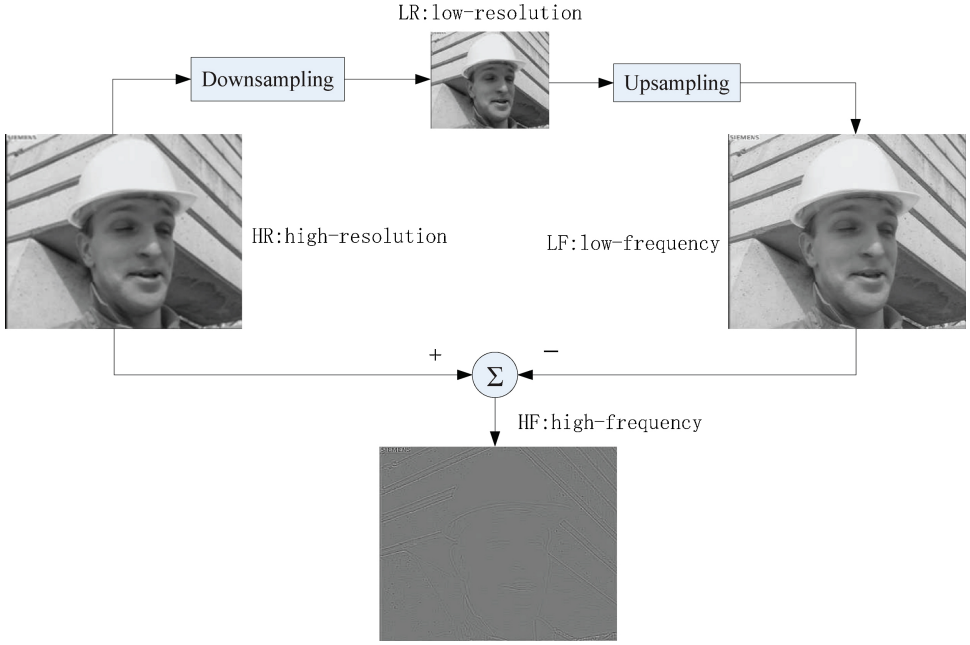


Fig. 2. Illustration of the frequency bands division of a given frame.

Inspired by the basic assumption that each patch can be represented as a linear combination of a small subset of patches (atoms) from a fixed dictionary [14], the super-resolution reconstruction can be described as an energy minimization as

$$f_{Image}(\{\alpha_{i,j}\}_{i,j}, X_h) = \arg\min_{X_h,\{\alpha_{i,j}\}} \{\lambda\|SHX_h - Z_l\|_2^2 + \sum_{i,j}\mu_{i,j}\|\alpha_{ij}\|_0 + \sum_{i,j}\|D_h\alpha_{i,j} - R_{i,j}X_h\|_2^2\} \quad (1)$$

where $R_{i,j}$ is a projection matrix that selects the (i, j)th patch from X_h and $\alpha_{i,j}$ is the sparse coefficient of the patch. The first term demands a proximity between the measured image, Z_l, and its super-resolution (unknown) version X_h. The sparse representation $\boldsymbol{\alpha}$ appeals l_0-norm optimization to keep the number of coefficients for a patch small and the parameter $\boldsymbol{\mu}$ is to control the tradeoff between the representation error and its sparsity. The third term desires that each patch from the reconstructed frame (denoted by $R_{i,j}X_h$) can be represented up to a bounded error by an overcomplete dictionary D_h, with coefficients $\alpha_{i,j}$. To minimize this function with respect to its unknowns yields the super-resolution reconstruction algorithm.

In order to avoid complexities caused by different resolutions between Z_l and X_h, we assume hereafter that Z_l is scaled up by a simple interpolation operator Q (e.g., bicubic interpolation) that fills in the missing rows or columns, returning to the size of X_h. The scaled-up frame is denoted by Z_{LF} and it satisfies the relation

$$Z_{LF} = QZ_l = QSHX_h = L^{all}X_h. \quad (2)$$

As shown in Fig. 2, the target is to recover $\hat{X}_h$ from Z_{LF}, which should be as close as possible to the original HR frame X_h.

The proposed algorithm operates on patches extracted from Z_{LF}, aiming to estimate the corresponding patch from X_h. Let $D_h \in R^{n\times K}$ be an overcomplete dictionary of K bases ($K > n$) and suppose $x_{i,j} = R_{i,j}X_h$ is a 2-D patch (of size $\sqrt{n}\times\sqrt{n}$ pixels) ordered lexicographically as a column vector. It is assumed that $x_{i,j}$ can be represented sparsely over the

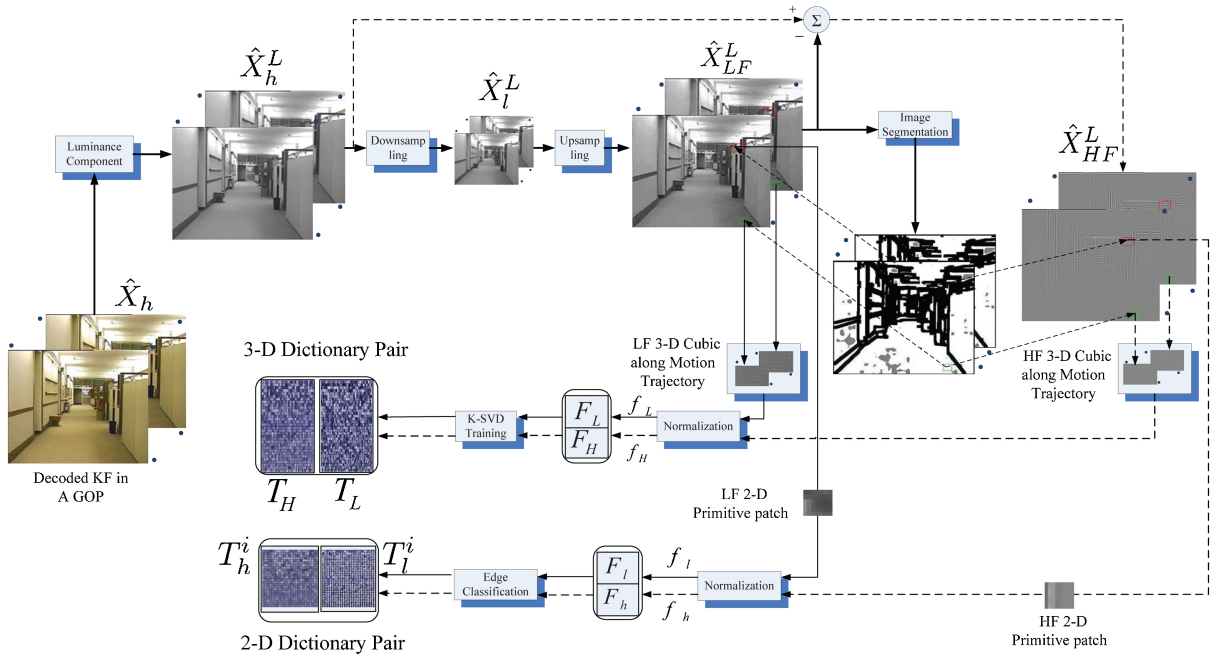


Fig. 3. Learning phase of the proposed framework.

TABLE I
ABBREVIATION TABLE

Summary of Abbreviations and Model Parameters	
HF	High-frequency
LF	Low-frequency
HR	High-resolution
LR	Low-resolution
KF	Key frame
NKF	Non-key frame
MCFI	Motion-compensated frame interpolation
AOBMC	Adaptive overlapped block motion compensation
X_h	Key frame with high-resolution, $\hat{X}_h$ is a coded version
X_{LF}, X_{HF}	Scaled-up frame with LF and HF from X_h, respectively
Z_l	Nonkey frame with low-resolution, $\hat{Z}_l$ is a coded version
Z_{LF}	Scaled-up frame with high-resolution from Z_l
$\boldsymbol{\alpha} = \{\alpha_{i,j}\}$	The sparse representation
D_h	The overcomplete dictionary of a high-resolution frame
$x_{i,j} = R_{i,j}X_h$	A 2-D (frame) patch ordered as a column vector
$R_{i,j,t}X_h$	A 3-D volume in time t and location $[i, j]$
$\{T_l^i, T_h^i\}_i$	The i th 2-D subdictionary pairs
$\{T_L, T_H\}$	The 3-D dictionary pair
$\{\hat{X}_{LF}^L\}_R$	A reference frame for learning 3-D dictionary
$\{\hat{X}_{LF}^L\}_{\bar{R}}$	The predictor of a reference frame from MCFI

dictionary as $x_{i,j} = D_h\alpha_{i,j}$, where $\|\alpha_{i,j}\|_0 \ll n$. Consider the corresponding LR patch $z_{i,j} = R_{i,j}Z_{LF} = R_{i,j}L^{all}X_h$ extracted from Z_{LF} in the same location. Since the operator $L^{all} = QSH$ transforms the complete HR frame X_h into the LR one Z_{LF}, it can be assumed that $z_{i,j} = Lx_{i,j} + \hat{v}_{i,j}$, where L is a local operator that is a portion of L^{all} and $\hat{v}_{i,j}$ is the additive noise. From $x_{i,j} = D_h\alpha_{i,j}$, we get $Lx_{i,j} = LD_h\alpha_{i,j} = z_{i,j} - \hat{v}_{i,j}$, implying that

$$\|z_{i,j} - LD_h\alpha_{i,j}\|_2^2 \leq \epsilon. \tag{3}$$

The key observation from the derivation is that the LR patch $z_{i,j}$ can be represented by the same sparse vector $\alpha_{i,j}$ over the effective dictionary $D_l = LD_h$, within a controlled error ϵ. It implies that if we recover the sparse representation coefficients of each LR patch base on an LR dictionary, the HR patch is reconstructed using the recovered coefficients in terms of the corresponding HR dictionary.

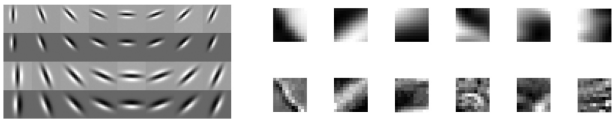


Fig. 5. Left: filter bank used for extracting frame primitives. Top right: typical LF primitives extracted. Bottom right: corresponding HF primitives extracted.

Although efforts have been made to learn a universal dictionary from a set of training images, learning various sets of LR/HR subdictionary pairs is recognized to be optimal and efficient for sparsely coding all the possible patterns [13]. Different from [14] (SR with sparse representation) that uses sparse representation prior on arbitrary image patches, we further utilize 2-D sparse representation on patches located only in a primitive layer.

B. Learning-Based Video Coding With Reconstruction

The proposed learning-based video coding scheme consists of two critical phases: learning and synthesis.

1) *Learning Phase:* Different from the existing learning-based super-resolution methods, two kinds of dictionaries, a set of 2-D subdictionary pairs on primitive frame patches and a 3-D dictionary on nonprimitive cubic volumes, are generated as shown in Fig. 3. Taking the luminance component as an illustration, the down-sampled version $\hat{X}_l^L$ of the decoded KFs $\hat{X}_h^L$ is initially achieved. $\hat{X}_{LF}^L$ that contain the LF component of $\hat{X}_h^L$ are attained by interpolating (e.g., bicubic interpolation) from $\hat{X}_l^L$ and classified into three layers: the primitive layer, the nonprimitive coarse layer, and the nonprimitive smooth layer. The HF part is acquired from the difference of $\hat{X}_h^L$ and $\hat{X}_{LF}^L$.

The training sets of the 2-D subdictionaries are generated from frame patches located in the primitive layer of the HF

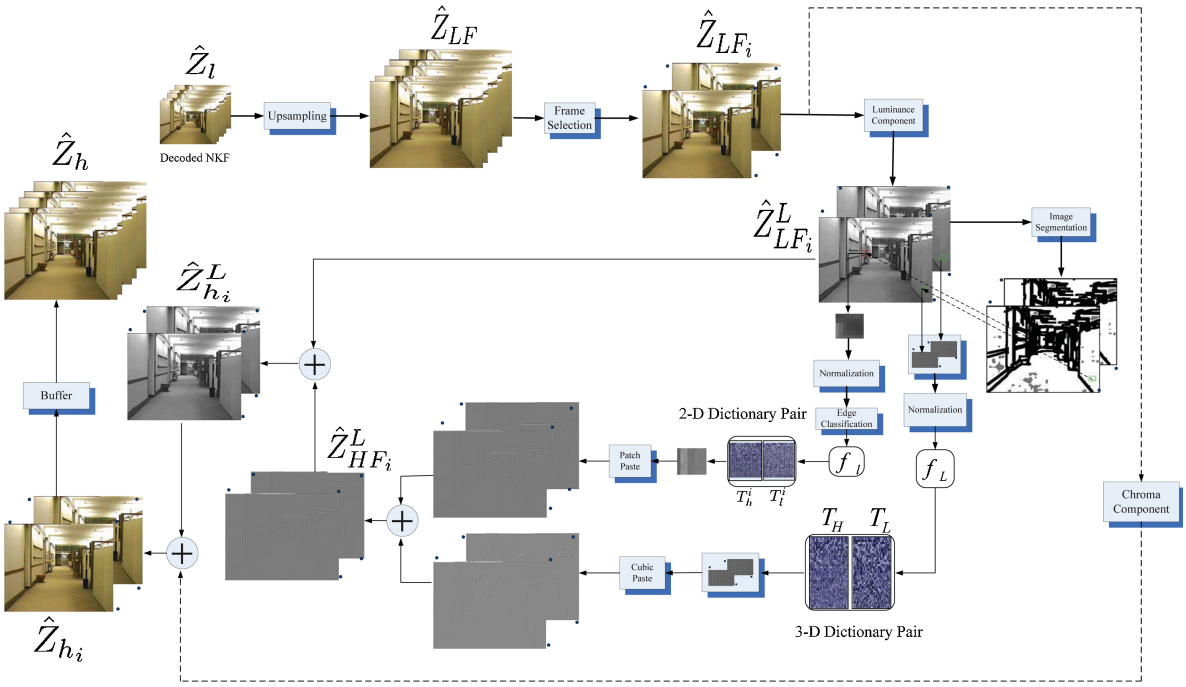


Fig. 4. Reconstruction phase of the proposed framework.

and their corresponding LF frames, and these primitive patches are clustered into subsets to learn a set of subdictionary pairs $\{T_l^i, T_h^i\}_i, i = 1, 2, ..., K$. The patches located in the nonprimitive coarse layer are combined with the most matched patches in the neighbor training frames to form 3-D cubic volumes, and these cubic volumes would be adopted to learn the 3-D dictionary pair $\{T_L, T_H\}$.

2) *Synthesis Phase:* As shown in Fig. 4, the input LR NKFs $\hat{Z}_l$ would be synthesized to generate the final HR frames $\hat{Z}_h$ in terms of both the 2-D subdictionaries ($\{T_l^i, T_h^i\}_i$) and the 3-D dictionaries ($\{T_L, T_H\}$). Initially, $\hat{Z}_l$ is interpolated into $\hat{Z}_{LF}$ that has the same size as $\hat{X}_h$. Once selecting the upsampled frame as $\hat{Z}_{LF_i}$, its luminance component $\hat{Z}^L_{LF_i}$ is also classified into three layers. For an LF patch in the primitive layer of $\hat{Z}^L_{LF_i}$, we synthesize its corresponding HF patch with the aid of the optimal 2-D subdictionary pair $\{T_l^i, T_h^i\}$. The LF volumes located in the nonprimitive coarse layer are extracted from $\hat{Z}^L_{LF_i}$ along the motion trajectory and the corresponding HF volumes can be inferred from the 3-D dictionary pair $\{T_L, T_H\}$. When all the HF patches and volumes are obtained, both the primitive and the nonprimitive HF frames can be generated, respectively. Hence, the HF frames $\hat{Z}^L_{HF_i}$ are formed by combining the primitive and nonprimitive HF frames. In sequence, the HR frames $\hat{Z}^L_{h_i}$ can be obtained by adding HF frames to the corresponding LF ones. Finally, $\hat{Z}_{h_i}$ will be generated by combining the derived HR luminance frames with the interpolated chrominance component.

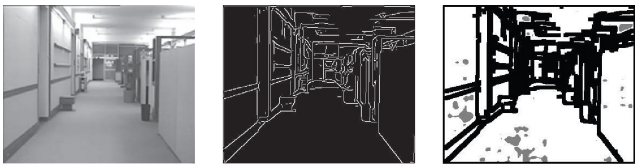
III. Sparse Spatio-Temporal Representation on Adaptive Regularized Dictionaries

As mentioned, the LF frames are classified into a primitive layer, a nonprimitive coarse layer, and a nonprimitive smoothness layer. Based on the decomposition, different kinds of training data would be collected to learn two kinds of dictionaries in alignment with adaptive reconstruction of the HF frames.

A. Image Primitives-Based Decomposition

Revisiting the philosophy of vision perception, primitive grouping is a critical operation to form meaningful entities from an unstructured data set (gray-level values). In essence, the visual system imposes organization on data and perceives structures (e.g., edge) in images. In this context, a multiscale representation is constructed to generate a family of derived signals where the fine-scale information could be successively suppressed. Moreover, artifacts cannot be introduced by the smoothing transformation when going from a finer to a coarser scale. For the scale-space structure, the notion of causality is introduced to show that no new level curves could be created when the scale parameter is increased. In other words, it should always be possible to trace a gray-level value existing at a certain level of scale to a similar gray-level at any finer level of scale [16].

The majority of literature on sparse representation can be categorized into two basic approaches: the analytic approach and the learning-based approach. The analytic construction is developed by dictionaries that are highly structured, referred to implicit dictionaries, e.g., wavelets, curvelets, contourlets, and bandelets. The learning-based approaches infer the dictionary from a set of examples, where the dictionary is represented as an explicit matrix. A training algorithm would be exploited to adapt the matrix coefficients to the examples, e.g., principal component analysis and the K-SVD [23], and it is in favor of the much finer-tuned dictionaries. Image primitives mainly consist of edge segments, bars, blobs, and



(a) (b) (c)

Fig. 6. (a) LF frame (interpolated from LR frame). (b) Extracted primitive frame. (c) Different layers after decomposition (the primitive layer, the nonprimitive coarse layer, and the nonprimitive smooth layer are indicated by black, white, and gray, respectively).

terminations, and can reflect the brightness changes [18], [20], [21]. The geometry of images is usually characterized by these features in a multiscale sense. It has not been clear whether images can be recovered from these multiscale differential components [17]; however, such representations are invertible to facilitate reconstruction with dictionary learning. The maximum response (MR) filter bank [39], Leung and Malik filter bank [40], Cula and Dana filter bank [41], and Schmid filter bank [42] are among the most studied and best known recent texture analysis techniques. It was demonstrated that the rotationally invariant multiscale maximum response filter bank yields better results than all the others [43]. The MR filter bank contains not only the Gaussian derivative filters to detect edge, bar, and spot at multiple scales and orientations, but also Gaussian kernel and Laplacian of Gaussian kernel filters to detect and localize the smooth texture. The Gaussian derivative filters are widely used to detect and localize the curved edges. In this paper, the primitive patches are recognized as part of a subclass, e.g., an edge, a ridge, or a corner with different orientations and scales. All of these geometries are of low dimensionality and can be well learned from patches across frames in a scale-space. As in [18], we adopt the Gaussian derivative filters [22] and record the maximum filter response

$$OE_{\sigma,\theta} = (I * f_{\sigma,\theta}^{odd})^2 + (I * f_{\sigma,\theta}^{even})^2 \tag{4}$$

where $f_{\sigma,\theta}^{odd}$ and $f_{\sigma,\theta}^{even}$ are the first and second Gaussian derivative filters at scale σ and orientation θ. As a result, a set of linear Gaussian derivative filter outputs is obtained, then the primitive layer is labeled by overlapping frame patches along the primitives sketch and the remaining nonprimitive layer is filtered by a high-pass filter. Fig. 5 illustrates some typical primitive patch pairs, while the filter bank consists of two scales and 16 orientations.

Fig. 6(b) shows the primitives extracted from the LF frame in Fig. 6(a). Once the primitives frame is obtained, the primitive layer is labeled by overlapping frame patches along the primitives sketch. The remaining nonprimitive layer is filtered by a high-pass filter. If a block contains sufficient HF energy, it is labeled as a part of a nonprimitive coarse layer. As Fig. 6(c), the nonprimitive coarse layer is distinguished from the nonprimitive smooth layer.

B. Dictionary Learning

The nature image statistics [21] show that the intrinsic dimensionality of its primitives is low, which implies that image primitives have more sparse representation structures over dictionary learned from primitive patches and makes it possible to represent each primitive in natural images by a small subdictionary of the similar structure feature.

1) *Learning 2-D Subdictionaries on Primitive Frame Patches:* In order to learn a series of 2-D subdictionary pairs $\{T_l^i, T_h^i\}_i$, $i = 1, 2, ..., K$, to represent the various local structures, a straightforward way is to concatenate the luminance values of all pixels inside a primitive patch directly from $\hat{X}_h^L$ and $\hat{X}_l^L$ to form an HR and a corresponding LR training set. In this paper, we aim to strengthen the training on characterizing the relationship between the LR patches (LF) and the primitive elements within the corresponding HR patches (HF).

Suppose that M primitive patch pairs $F_l = [f_l^1, f_l^2, ..., f_l^M]$ and $F_h = [f_h^1, f_h^2, ..., f_h^M]$ are collected from the primitive layer of $\hat{X}_{LF}^L$ and $\hat{X}_{HF}^L$. The target is to learn K subdictionary pairs $\{T_l^k, T_h^k\}_k$ from $\{F_l, F_h\}$ so that, for each given LF primitive patch, the corresponding HF primitive patch can be reconstructed from the selected most suitable subdictionary pair. To achieve this, we cluster the dataset $\{F_l, F_h\}$ into K clusters and learn a subdictionary pair from each of the K clusters. Apparently, the K clusters are expected to represent the distinctive patterns in $\{F_l, F_h\}$. Considering that image primitive mainly consists of step-edges, bars, corners, and terminations of different orientations and scales, $\{F_l, F_h\}$ is divided into K clusters $\{F_l^k, F_h^k\}_k$, $k = 1, 2, ..., K$ in terms of the orientations (16 orientations) and scales (three scales) of primitive patches. The primitive patches with the same orientation and scale would be classified into the same cluster.

Currently, the problem is how to learn a subdictionary pair $\{T_l^k, T_h^k\}$ from the cluster $\{F_l^k, F_h^k\}$ such that all the atoms in $\{F_l^k, F_h^k\}$ can be sparsely and faithfully represented by $\{T_l^k, T_h^k\}$. The design of $\{T_l^k, T_h^k\}$ can be intuitively formulated by

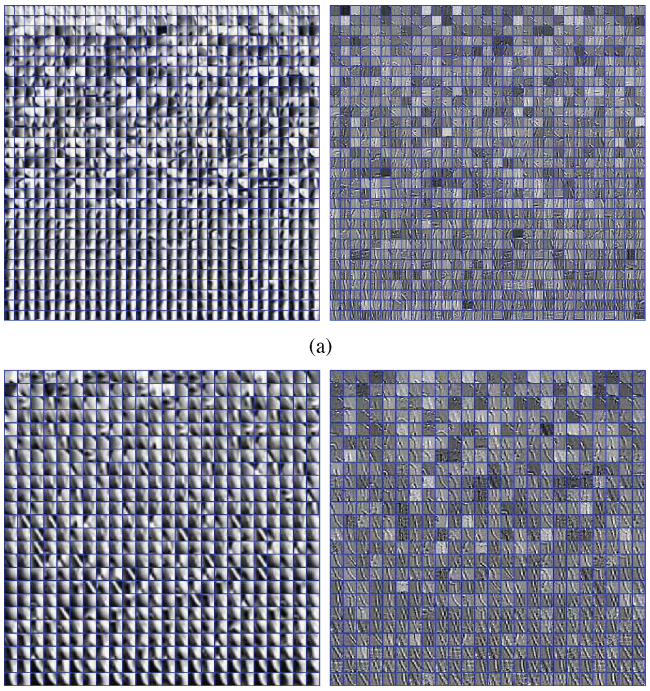
$$(T_k, \Phi_k) = \arg\min_{T_k, \Phi_k} \{\|F_k - T_k\Phi_k\|_2^2 + \lambda\|\Phi_k\|_1\} \tag{5}$$

where

$$F_k = \begin{bmatrix} F_l^k \\ F_h^k \end{bmatrix} \qquad T_k = \begin{bmatrix} T_l^k \\ T_h^k \end{bmatrix} \tag{6}$$

and Φ_k is the representation coefficient matrix of F_k over T_k. Equation (5) is an l_1-regularized least-squares problem over Φ_k and an l_2-constrained least-squares problem over T_k. This $l_2 - l_1$ joint minimization problem of Φ_k and T_k could be iteratively optimized by alternatingly optimizing Φ_k and T_k when the other is fixed [23], [24].

In view of the fact that the K $(l_2 - l_1)$ joint minimization in (5) requires much computational cost, F_k might be regarded as the final subdictionary T_k from the following considerations. First, the computational cost of the sparse coding of a given primitive patch over F_k is small enough because F_k is a subset of $\{F_l, F_h\}$. Second, the intrinsic dimensionality of image primitives is very low, so it is possible to represent all the image primitives well by a small number of primitive examples from the highly correlated training images [21]. Furthermore, the manifolds of the feature spaces formed by the LF and the HF primitive patches have similar local geometry, which allow us to infer the HF primitive patch efficiently from the corresponding LF one. It will be further validated in Section III-E. Fig. 7 shows two examples of the learned 2-D subdictionary



(a) (b)

Fig. 7. Examples of the learned 2-D subdictionaries. The left column shows two of the learned 2-D low-frequency subdictionaries and the right column shows the corresponding 2-D high-frequency subdictionaries.

pairs and it can be obviously seen that different subdictionaries consist of primitive patches at different scales and orientations.

2) *Learning 3-D Dictionary on Nonprimitive Volumes Along Motion Trajectory:* Considering that the nonprimitive volumes along the motion trajectory are consistent in the temporal dimension, they are supposed to bear more sparse representation structures over a learned 3-D dictionary. Hence, the spatio-temporal consistency from incomplete visual patterns can be better obtained by taking a 3-D spatio-temporal dictionary into consideration [25].

Revisiting Section II-A, we extend it to a video sequence by exploiting the temporal dimension. Let X_h and Z_l represent the original HR and down-sampled LR sequence, respectively. With the index t in the range [1, T] for the time dimension, it leads to a minimum error energy function as follows:

$$f^{All}_{Video}(\{\alpha_{i,j,t}\}_{i,j,t}, X_h, D_h) = \arg\min_{X_h,\{\alpha_{i,j,t}\}} \{\lambda\|SHX_h - Z_l\|_2^2 + \sum_{i,j}\sum_{t=1}^{T}\mu_{i,j,t}\|\alpha_{i,j,t}\|_0 + \sum_{i,j}\sum_{t=1}^{T}\|D_h\alpha_{i,j,t} - R_{i,j,t}X_h\|_2^2\}. \quad (7)$$

The term $R_{i,j,t}X_h$ extracts a patch of a fixed size from the volume X_h in time t and location $[i, j]$. However, training a single dictionary to the entire sequence as Section III-B is problematic: the scene in a video is supposed to change rapidly over time and the dimension of the dictionary grows rapidly with the increase of time space. An alternative solution is to define a locally temporal penalty term. Given the extremely correlated KFs and NKFs, we could learn a 3-D dictionary from the KFs and decompose all the neighbor NKFs with such a dictionary. Hence, Section III-B can be rewritten for a reference KF as follows:

1) Initialize $\hat{T}_L$.
2) Repeat until convergence (stopping rule):
- **Sparse coding:** for $\hat{T}_L$ fixed, solve Eq. (11) using any pursuit algorithm to obtain $\hat{\Theta}$, i.e. the Orthogonal matching pursuit algorithm.
- **Dictionary Update:**
  for each $j = 1$ to K
  - Compute w_j.
  - Compute the overall representation error matrix E_j, by

$$E_j := F_L - \sum_{i\neq j} t_l^i \delta_i^T. \quad (10)$$

  - Restrict E_j and δ_j^T by choosing only the columns corresponding to w_j, and obtain E_j^R, $\delta_{j,R}^T$.
  - Obtain the largest singular value of E_j^R and the corresponding singular vectors.
  - Update $\hat{T}_L$ and $\hat{\Theta}$.
  end

Fig. 8. K-SVD algorithm.

$$f^r_{Video}(\{\alpha_{i,j}\}_{i,j}, X^r_h, D^r_h) = \arg\min_{X^r_h,\{\alpha_{i,j}\}} \{\lambda\|SHX^r_h - Z^r_l\|_2^2 + \sum_{i,j}\mu_{i,j}\|\alpha_{i,j}\|_0 + \sum_{i,j}\|D^r_h\alpha_{i,j} - R_{i,j,r}X_h\|_2^2\} \quad (8)$$

where X^r_h is a reference KF of a GOP. The learned dictionary of the previous GOP can be propagated to the subsequent GOP as the initial dictionary to reduce the number of training iterations.

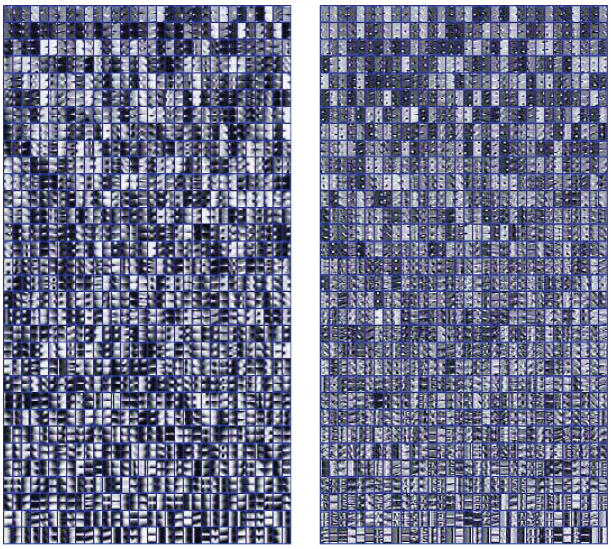
In order to get a more sparse representation of the sequence, the volume is further generated along the motion trajectory by block-matching based motion estimation

$$f^r_{Video}(\{\alpha_{i,j}\}_{i,j}, X^r_h, D^r_h) = \arg\min_{X^r_h,\{\alpha_{i,j}\}} \{\lambda\|SHX^r_h - Z^r_l\|_2^2 + \sum_{i,j}\mu_{i,j}\|\alpha_{i,j}\|_0 + \sum_{i,j}\|D^r_h\alpha_{ij} - R_{i,j,r}\tilde{X}_h\|_2^2\} \quad (9)$$

where $\tilde{X}_h$ is the motion-compensated estimator of X_h in alignment with a reference key frame and $R_{ijr}\tilde{X}_h$ denotes an extracted volume from X_h along the estimated motion trajectory.

For every reference frame $\{\hat{X}^L_{LF}\}_R$, we adopt a motion-compensated frame interpolation (MCFI) approach to attain the estimated reference frame (predictor) $\{\hat{X}^L_{LF}\}_{\tilde{R}}$ according to its preceding frames $\{\hat{X}^L_{LF}\}_P$ and following frames $\{\hat{X}^L_{LF}\}_B$. The LF volumes are extracted by concatenating the patches located in the nonprimitive coarse layer of $\{\hat{X}^L_{LF}\}_R$ and the corresponding patches from $\{\hat{X}^L_{LF}\}_{\tilde{R}}$, and thus the HF volumes are similarly obtained. The detailed MCFI procedure will be described in the next subsection.

Collecting all the LF and HF volumes could get the training sets $\{F_L, F_H\}$ of the 3-D dictionary pair $\{T_L, T_H\}$. Since $\{F_L, F_H\}$ are constructed by volumes in the nonprimitive



(a) (b)

Fig. 9. Learned 3-D dictionary pair, where $K = 1024$ atoms in the dictionary, $S = 6$ atoms for each vector, and the iteration number 30 is set in the K-SVD algorithm. (a) Learned LF 3-D dictionary. (b) Corresponding HF 3-D dictionary.

layer that have little structure information, we would learn a universal dictionary pair from them. Let $F_L = [f_l^1 \ f_l^2 \cdots f_l^p]$ be an $n \times P$ matrix of P training sets of length n pixels each, we aim to train an overcomplete dictionary T_L of size $n \times K$ ($P \gg K$ and $K > n$) for a given sparsity level S through the K-SVD algorithm [23]

$$min_{T_L,\Theta} \|F_L - T_L\Theta\|_F^2 \quad \text{s.t.} \quad \forall i, \|\theta_i\|_0 \leq S \tag{10}$$

where $\Theta = [\theta_1 \ \theta_2 \cdots \theta_p]$ and θ_i is the sparse vector of coefficients representing the ith volume in terms of the columns of the dictionary $T_L = [t_l^1 \ t_l^2 \cdots t_l^K]$. It is worth mentioning that the Frobenius matrix norm $\|.\|_F$ formulates a general definition of a matrix norm and is compatible with the euclidean vector norm. Using the dictionary learned from the previous GOP as an initial dictionary, the K-SVD algorithm in Fig. 8 is adopted to progressively optimize the expression.

Once obtaining the LF 3-D dictionary T_L and the corresponding sparse representation coefficients matrix Θ, the next task is to construct the HF 3-D dictionary T_H. Recall from Section II-A that recovering the HF training sets $F_H = [f_h^1 \ f_h^2 \cdots f_h^P]$ is approximated by $F_H \approx T_H\Theta$. In other words, the sparse representation vector for a LF volume is multiplied by the HF dictionary to recover the corresponding HF volume. In this sense, the dictionary T_H is constructed such that this approximation is as exact as possible. Hence, this dictionary is defined by minimizing the mean approximation error, that is

$$T_H = min_{T_H} \|F_H - T_H\Theta\|_F^2. \tag{11}$$

The solution is given by the following pseudo-inverse expression (given that Θ has full row rank):

$$T_H = F_H\Theta^+ = F_H\Theta^T(\Theta\Theta^T)^{-1}. \tag{12}$$

Compared to the joint dictionary learning [14] and the coupled dictionary learning [45], the 3-D dictionary learning in the proposed scheme will enable a fast low-scale computation with one half dimensionality. Fig. 9 shows an example of the learned 3-D dictionary pairs.

C. Motion-Compensated Frame Interpolation

As depicted in the previous section, MCFI is adopted to predict an estimated reference frame $\{\hat{X}_{LF}^L\}_{\widetilde{R}}$ during the 3-D dictionaries learning procedure. In MCFI, the quality of an interpolated frame is mainly dependent on how to estimate accurate motion trajectories through the interpolated frame and how to merge motion compensated blocks without unnatural artifacts.

Let f_{n-1}, f_n, and f_{n+1} denote the preceding frame, the intermediate reference frame, and the following frame, respectively. To find motion vectors, hierarchical bidirectional motion estimation (ME) [26] is employed in alignment with adaptive overlapped block motion compensation (AOBMC) that can control the weights of overlapping windows according to the reliability of the neighboring motion vectors [27], as shown in Fig. 10.

1) *Hierarchical Motion Estimation:* Let $\mathbf{s}$ be a 2-D vector to represent a pixel location, and let $\mathbf{v}$ denote a candidate forward motion vector for an $M \times M$ matching block $B_{i,j}$. For each candidate motion vector $\mathbf{v}$ in block matching based ME, we compute the sum of absolute differences (SAD). When the forward MVs for all the $M \times M$ blocks are attained, we allocate each MV for the $M \times M$ block to its subordinate $L \times L$ sub-blocks ($M = 16$ and $L = 4$ in this paper). A local motion estimation is performed with a smaller search window for each $L \times L$ sub-block around the selected MV. In this way, the forward MVs for all the $L \times L$ sub-blocks $\{\boldsymbol{v}\}$ are obtained through hierarchical ME. Similarly, the backward MVs could be acquired.

2) *Bidirectional OBMC Using Adaptive Window:* To reduce an artifact, OBMC [28] was developed where a block MV is applied to its neighboring blocks by a weighting window. To be concrete, the pixel $f_n(\mathbf{s})$ in an $L \times L$ block $S_{i,j}$ is predicted using the MVs of the block itself, the top neighbor block $S_{i-1,j}$, the bottom neighbor block $S_{i+1,j}$, the left neighbor block $S_{i,j-1}$, the right neighbor block $S_{i,j+1}$

$$f_n(\mathbf{s}) = \sum_{p=-1}^{1}\sum_{q=-1}^{1} w_{p,q}(\mathbf{s}) f_{n-1}(\mathbf{s} + \boldsymbol{v}_{i+p,j+q}) \tag{13}$$

where $\boldsymbol{v}_{i+p,j+q}$ denotes the MV of block $S_{i+p,j+q}$ and $w_{p,q}(\mathbf{s})$ is the corresponding weighting coefficient that satisfies $\sum_{p=-1}^{1}\sum_{q=-1}^{1} w_{p,q}(\mathbf{s}) = 1$. $w_{p,q}(\mathbf{s})$ is determined by the relative position of $\mathbf{s}$ within $S_{i,j}$.

If adjacent blocks exist in substantially different motions, OBMC would yield blurring or over-smoothing artifacts since the weighting coefficients are determined only by the relative distances of the pixels within the block. To overcome the disadvantage, the reliability of the neighboring MV $\boldsymbol{v}_{i+p,j+q}$ for predicting the current block $S_{i,j}$ is defined as

$$\Phi_{S_{i,j}}(\boldsymbol{v}_{i+p,j+q}) = \frac{SAD(S_{i,j}, \boldsymbol{v}_{i,j})}{SAD(S_{i,j}, \boldsymbol{v}_{i+p,j+q})}. \tag{14}$$

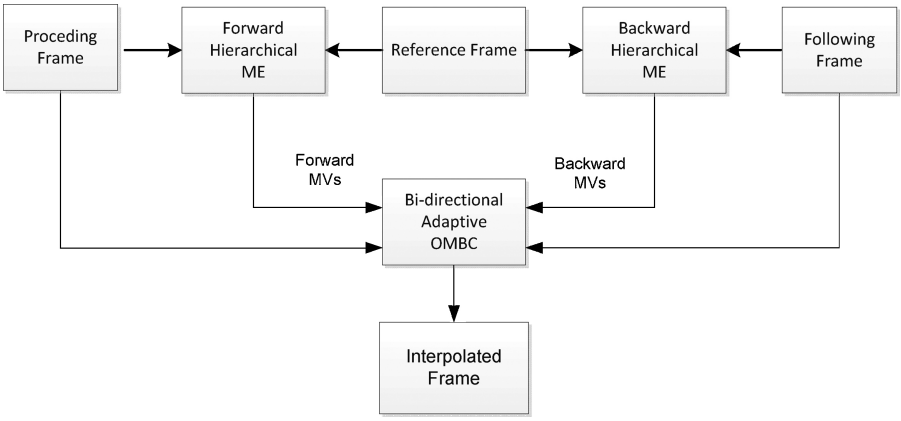


Fig. 10. Overview of the MCFI operation.

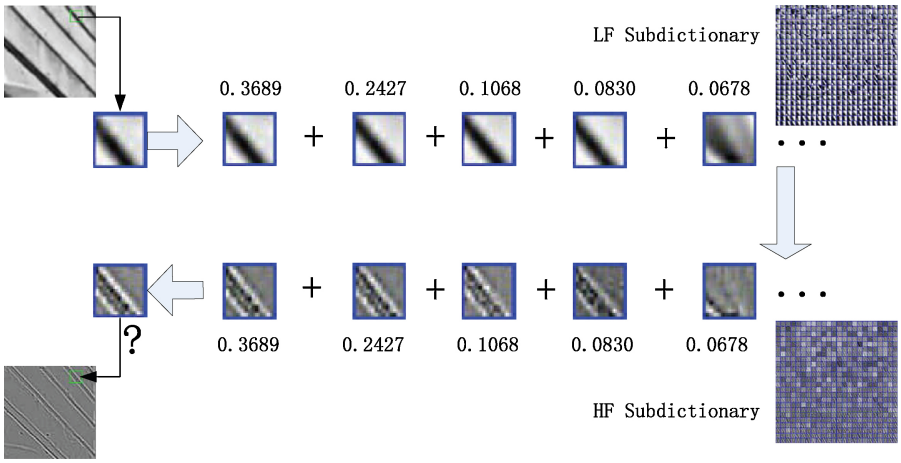


Fig. 11. Illustration showing how a high-frequency primitive patch is generated.

Note that $\boldsymbol{v}_{i,j}$ is the estimated MV to minimize $SAD(S_{i,j}, \boldsymbol{v})$; therefore, we have $0 \leq \Phi_{S_{i,j}}(\boldsymbol{v}_{i+p,j+q}) \leq 1$. As $\Phi_{S_{i,j}}(\boldsymbol{v}_{i+p,j+q})$ gets more closer to 1, the neighboring MV $\boldsymbol{v}_{i+p,j+q}$ are more reliable to compensate the current block $S_{i,j}$.

Using the reliability, the weighting coefficients $w_{p,q}(\mathbf{s})$ in (13) are modified as

$$\hat{w}_{p,q}(\mathbf{s}) = \frac{\Phi_{S_{i,j}}(\boldsymbol{v}_{i+p,j+q}) w_{p,q}(\mathbf{s})}{\sum_{s=-1}^{1}\sum_{t=-1}^{1}\Phi_{S_{i,j}}(\boldsymbol{v}_{i+s,j+t}) w_{s,t}(\mathbf{s})}. \quad (15)$$

The coefficients are proportional to the reliability of corresponding neighbor motion vector $\Phi_{S_{i,j}}(\boldsymbol{v}_{i+p,j+q})$.

When substituting $\hat{w}_{p,q}(\mathbf{s})$ in (13), we can get the motion-compensated pixels of f_n from the forward MVs from f_{n-1}. Likewise, the backward motion-compensated values of f_n can be acquired by the backward MVs of f_{n+1}. Among the two directional MVs of each block $S_{i,j}$, the one for a smaller SAD is chosen as the optimum for the block.

D. Synthesis Phase

By a set of 2-D subdictionary primitive pairs $\{T_l^i, T_h^i\}_i$ and 3-D nonprimitive volume dictionary pair $\{T_L, T_H\}$ after the learning phase, the synthesis phase could be enabled.

If an LF patch f_l is located in the primitive layer of $\hat{Z}_{LF_i}^L$, its feature value could be obtained by the orientation and scale. The optimal subdictionary pair $\{T_l^i, T_h^i\}$ that is most relevant to the primitive patch f_l would be selected. Here, the selected subdictionary could be thought of as composed of primitive patches of the same orientation and scale as f_l. In turn, synthesizing the corresponding HF primitive patch f_h is addressed and illustrated in Fig. 11. First, the sparsest representation of f_l should be found by the formulation

$$\min \|\alpha\|_0 \quad \text{s.t.} \quad \|T_l^i \alpha - f_l\|_2^2 \leq \epsilon \quad (16)$$

where $T_l^i \in R^{m \times k}$ with $m \ll k$, and $f_l \in R^m$. It is an NP-hard problem and can be fortunately recovered by solving an l_1-norm minimization problem as long as the desired coefficients α are sufficient sparse [29]

$$\min \|\alpha\|_1 \quad \text{s.t.} \quad \|T_l^i \alpha - f_l\|_2^2 \leq \epsilon. \quad (17)$$

Using lagrange multipliers, we can reformulate (17) as

$$\min \lambda\|\alpha\|_1 + \|T_l^i \alpha - f_l\|_2^2. \quad (18)$$

Theoretically, (18) is an l_1-regularized least-squares (LS) problem called LASSO, where $\lambda > 0$ is a tradeoff parameter between sparsity and fidelity. This optimization always has a solution, but it need not be unique. As $\lambda \to 0$, the limiting point has the minimum l_1 norm among all points that satisfy $\|T_l^i \alpha - f_l\|_2^2 = 0$, which leads to high fidelity. Moreover, as $\lambda \to \infty$, $\lambda\|\alpha\|_1$ is the dominant item in (18),

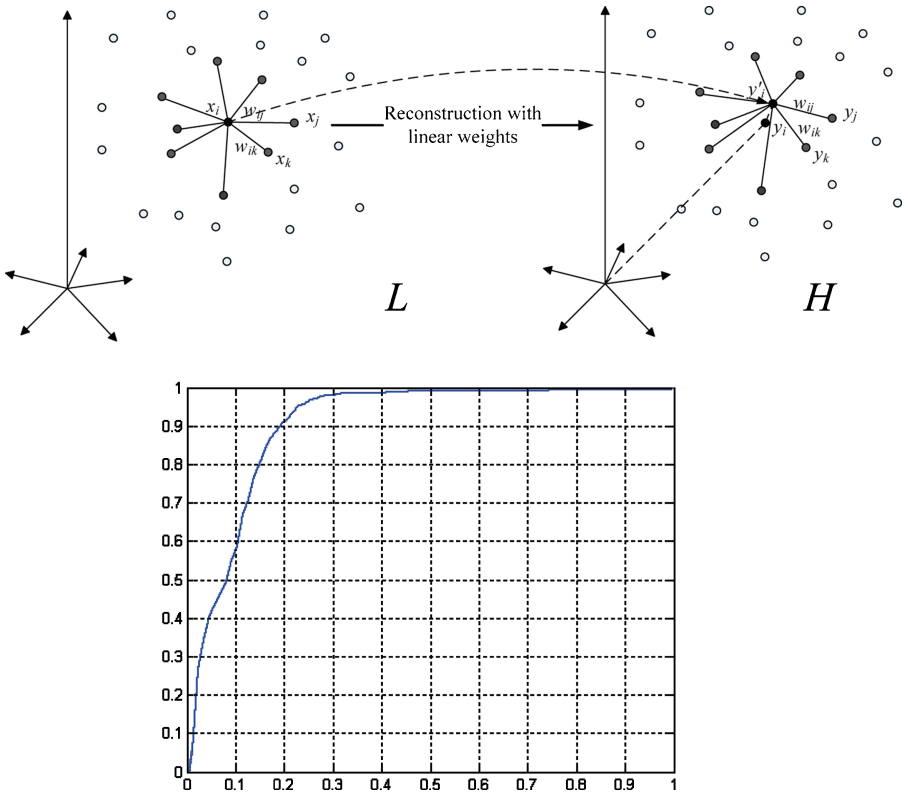
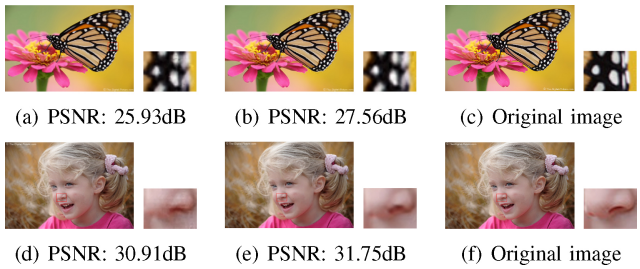


Fig. 12. (a) Manifolds formed by LF and HF primitive patches. (b) x-axis is the reconstruction error and the y-axis is the percentage of the test data whose reconstruction errors are smaller than a given e_h.



(a) PSNR: 25.93dB (b) PSNR: 27.56dB (c) Original image

(d) PSNR: 30.91dB (e) PSNR: 31.75dB (f) Original image

Fig. 13. Experimental results on *Butterfly* and *Girl* with a magnification factor 3. (a), (d) Yang's method [14]. (b), (e) Proposed scheme. (c), (f) Original high-resolution image. (a) PSNR: 25.93 dB. (b) PSNR: 27.56 dB. (c) Original image. (d) PSNR: 30.91 dB. (e) PSNR: 31.75 dB. (f) Original image.

it requires more consideration on the sparsity of α. However, the convergence occurs for a finite value of $\lambda_{\max} = \|2T_l^{i\,T} f_l\|_\infty$, which means that the optimal solution of α is $\vec{\mathbf{0}}$ for any $\lambda \geq \lambda_{\max}$. To simultaneously achieve sparse representation and ensure fidelity, the regularization parameter λ is always set in $[0.01\lambda_{\max}, 0.1\lambda_{\max}]$. This nonlinear convex optimization problem can be solved efficiently by a variety of methods [30], [31]. In the following experiments, a software packet SparseLab [32] is adopted to find the coefficients α.

If attaining optimal α by solving (18), the corresponding HR primitive patch f_h can be acquired by the linear combination of columns in T_h^i using α as the coefficients: $f_h = T_h^i \alpha$.

If an LF patch in $\hat{Z}_{LF_i}^L$ is located in the nonprimitive coarse layer, we get the corresponding patch of the estimated reference frame from the MCFI algorithm. These two relevant LF patches would be concatenated as a LF volume. Synthesizing the corresponding HF volume is similar to synthesizing a HF primitive patch, but with the 3-D dictionary pair $\{T_L, T_H\}$ and volumes instead of 2-D subdictionary pair $\{T_l^i, T_h^i\}$ and primitive patches.

E. Discussion on Sparse Representation Priors

As shown before, we classify each frame into three layers and different sparse representation priors are used in different layers. In detail, the HR patch corresponding to an LR patch in the nonprimitive smooth layer is obtained by a linear interpolation method (e.g., bicubic), the HR primitive patch corresponding to an LR primitive patch is synthesized via the adaptive 2-D sparse representation prior, and the remaining part in the nonprimitive coarse layer with more energy is generated with 3-D spatio-temporal sparse representation along the elegant motion trajectory.

To validate it over primitive prior, we illustrate the manifolds H and L formed by the HF and LF primitive patches through experiments. As shown in Fig. 12(a), each patch can be considered as a sample from its corresponding manifold where x_i, x_j, x_k are samples from L and their corresponding samples in H are y_i, y_j, y_k. It is noted that x_i can be represented as a linear weights of the samples in L, that is : $x_i = \sum_{x_m \in L} w_{im} x_m$. The reconstruction of y_i could be written as $y_i' = \sum_{y_m \in H} w_{im} y_m$. To demonstrate how well the HF primitive patch is inferred from the corresponding LF patch, we define the reconstruction error: $e_h = \|y_i' - y_i\| / \|y_i\|$. The experiments are made on 10 video sequences of 16 frames with three key frames and 1000 test primitive samples are randomly sampled from each key frame. After the learning phase, we

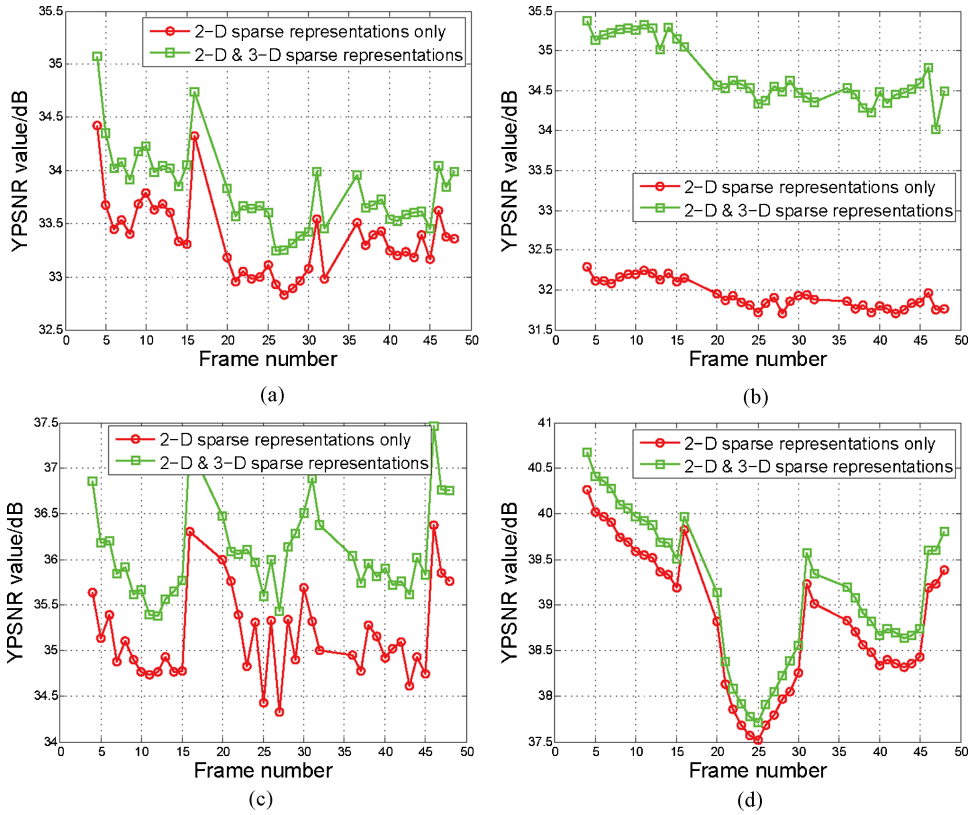


Fig. 14. Performance comparison between the proposed scheme and 2-D sparse representations only. (a) *Foreman*. (b) *Hall*. (c) *Highway*. (d) *Akiyo*.

get a set of subdictionaries. The final curve as the average result of 10 test sequences is plotted in Fig. 12. It can be observed that the reconstruction errors e_h of 90% HF primitive patches are less than 20%. It can be comprehended that the reconstructed patches are close enough to their corresponding real patches, and the HR frame primitive patches can be efficiently synthesized from their corresponding LR ones via the 2-D sparse representation prior.

To further validate the efficiency of sparse representation on primitive patches over adaptively selected subdictionary, Fig. 13 gives the visual comparison of the proposed scheme versus the recent approach [14] where randomly selected images patches are collected to learn a universal LR/HR dictionary pair. In the proposed scheme, only the primitive patches are used to learn a set of subdictionary pairs. It can be observed that the reconstructed edges by [14] are relatively smooth and some structures are not recovered.

To validate the efficiency of 3-D sparse representation on the nonprimitive coarse layer, we make experiments via both the proposed scheme and only 2-D sparse representations on primitive patches. Fig. 14 shows that the performance gain achieved is very noticeable because the 3-D sparse representations could explore the spatio-temporal consistency in video sequence.

On the other hand, previous research on nature image statistics [21] shows that the intrinsic dimensionality of image primitives is very low. This means that primitive patches have more sparse representation structures over dictionary and make it possible to represent each primitive by a small subdictionary of the same structure feature. The nonprimitive volumes are consistent along the motion trajectory in the temporal dimension and have little structure information, which makes more sparse representations over a learned 3-D spatio-temporal dictionary.

IV. Experimental Results

In the experiments, all the test sequences are with the YUV 4:2:0 format, 30-Hz frame rate, a GOP size of 16 frames. They are composed of CIF (352 × 288) and SDTV (832 × 480) resolution with a variety of motion activities: *Foreman*, *Hall*, *Highway*, *Akiyo*, *Waterfall*, *News*, *Mobile*, and *Basketball*. Given an original sequence, we select successive frames as the KFs in a GOP and down-sample other frames in a ratio 2 as the NKFs. The KFs would be coded by the H.264/AVC engine as the "$IPP\cdots$" order and the remaining NKFs are coded as the "$IBBPBBP\cdots$" order in a low bit-rate range. The KFs in both current GOP and next GOP are used to learn dictionary pairs for the continuity of successive frames. With the decoded low-resolution NKFs and corresponding 2-D and 3-D dictionary pairs, the high-resolution version of NKFs would be recovered by the proposed learning-based super-resolution algorithm. The 13 × 13 patch size in 2-D sparse representation and 7×7 volume size (spatial dimension) in 3-D sparse representation are enabled. At the decoder, the KFs are used to learn both the 2-D and 3-D dictionary pairs. The overall bit rate of the proposed scheme is kept consist with H.264/AVC and we validate both the objective (PSNR, rate-distortion, BD-bitrate) and the visual quality, e.g., structural

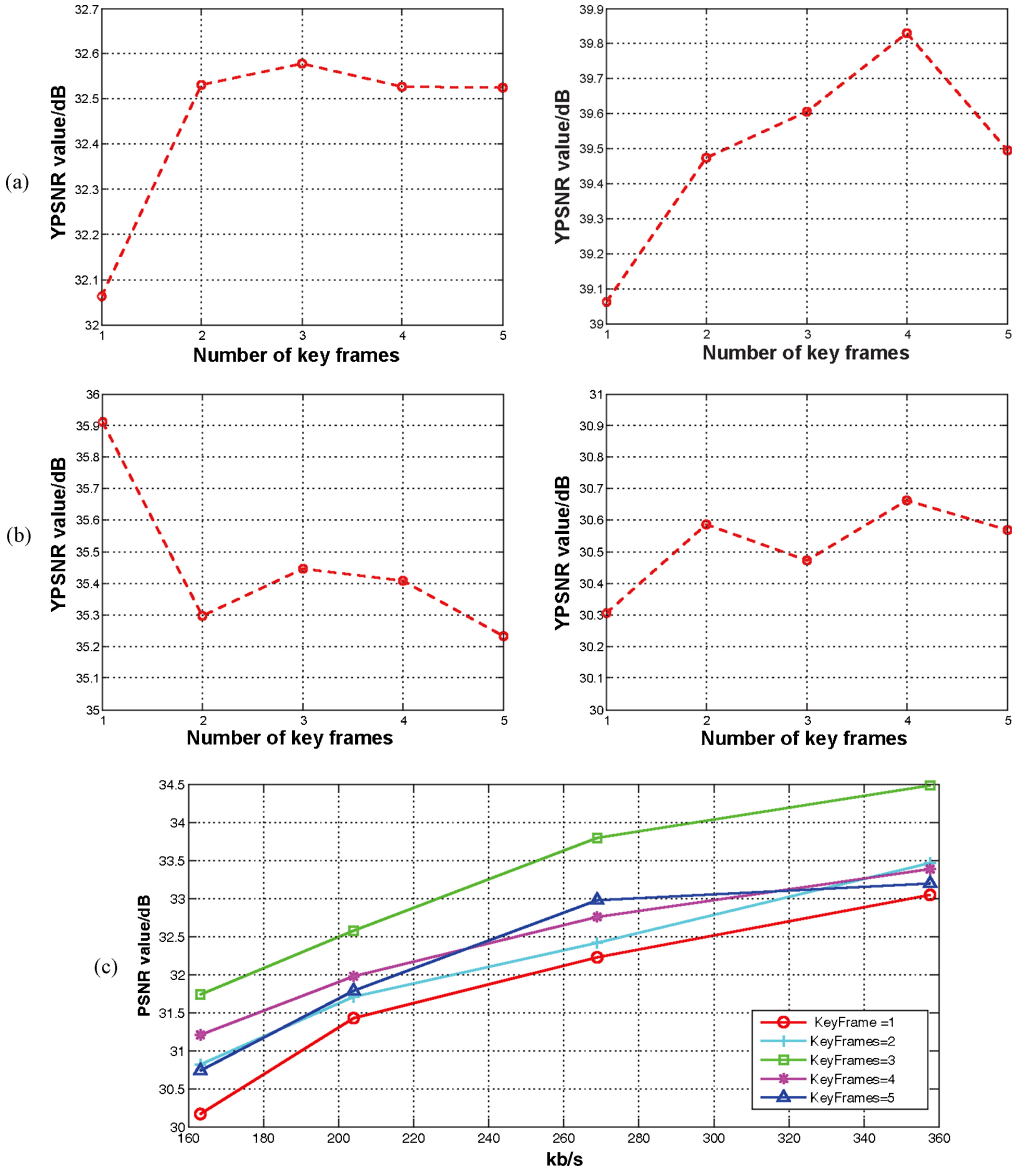


Fig. 15. PSNR behavior of the 3-D sparse representations with different numbers of key frames. (a) PSNR comparison of *Foreman* and *Akiyo*. (b) PSNR comparison of *News* and *Waterfall*. (c) R-D performance of 3-D sparse representation with different numbers of key frames over the *Foreman* sequence.

similarity index metrics (SSIM) and difference mean opinion score (DMOS).

Figs. 15 and 16 describe the both *PSNR* (R-D performance) and *SSIM* impact on 3-D sparse representation with different depths of 3-D atoms (the number of key frames). When using different number of key frames, the estimated reference frame is interpolated from the different number of neighbor frames. When there exist more key frames in a GOP, the high-resolution key frames can only be encoded at a low bit rate because the total bit rate is fixed. Thus, it would degrade the synthesis reconstruction. On the other hand, when the number of key frames is reduced, the estimated reference frame would be distorted. Thus, it would lead to low-quality dictionaries. We can conclude that the number of key frames in the proposed scheme would maintain a tradeoff between reliable dictionary and high-fidelity encoding. It can be seen that the optimal point is commonly located at three key frames in a GOP, and it would be set in the following experiments.

Figs. 17 and 18, respectively, compare *PSNR* and *SSIM* between the proposed video compression scheme and H.264/AVC, where the overall bit rate is 201.5 kb/s for the *Foreman* sequence, 204.5 kb/s for the *Akiyo* sequence, 248.8 kb/s for the *News* sequence, and 213.9 kb/s for the *Waterfall* sequence, respectively.

The SSIM metric has been shown to be able to catch the local statistical characteristic of signals and to tune with human perception better than PSNR under lots of situations. It concentrates on the distortion of structural content and is very sensitive to any artifacts introduced into the distorted image [33]. Hence, it is suitable for evaluating the smoothness of the reconstructed sequence and detecting the artifacts that may cause spatial-temporal inconsistency. SSIM is calculated block by block and defined as

$$SSIM = \frac{(2\mu_{I_1}\mu_{I_2} + C_1)(2\sigma_{I_1 I_2} + C_2)}{(\mu_{I_1}^2 + \mu_{I_2}^2 + C_1)(\sigma_{I_1}^2 + \sigma_{I_2}^2 + C_2)} \tag{19}$$

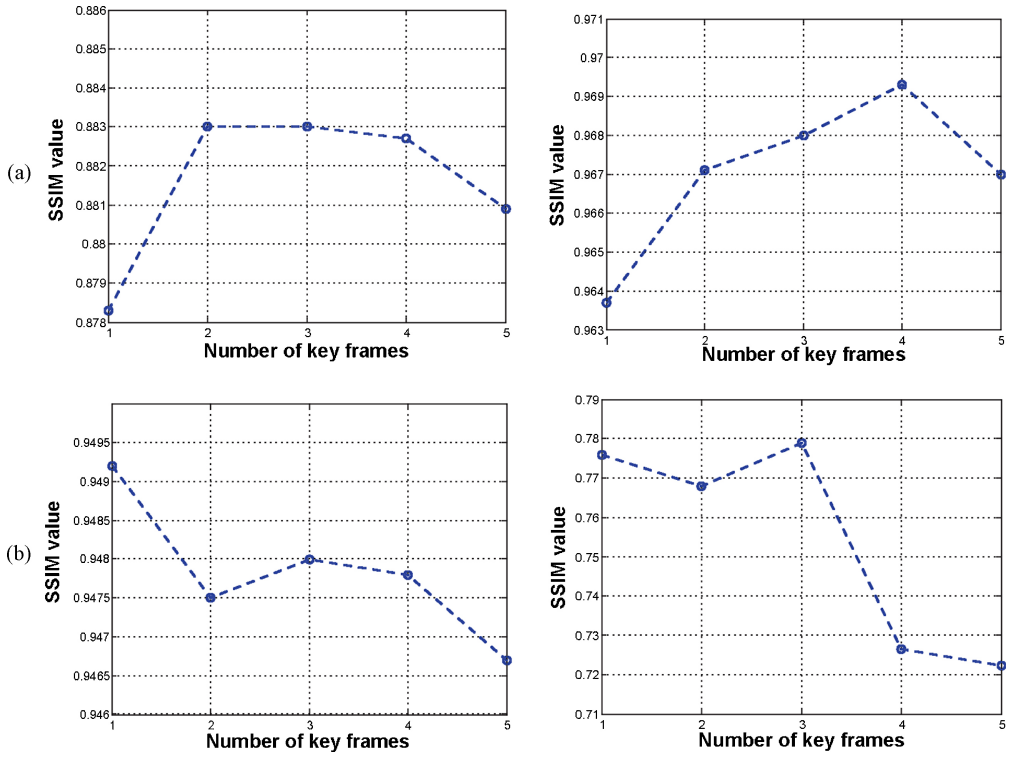


Fig. 16. SSIM behavior of the 3-D sparse representations with different numbers of key frames. (a) SSIM comparison of *Foreman* and *Akiyo*. (b) SSIM comparison of *News* and *Waterfall*.

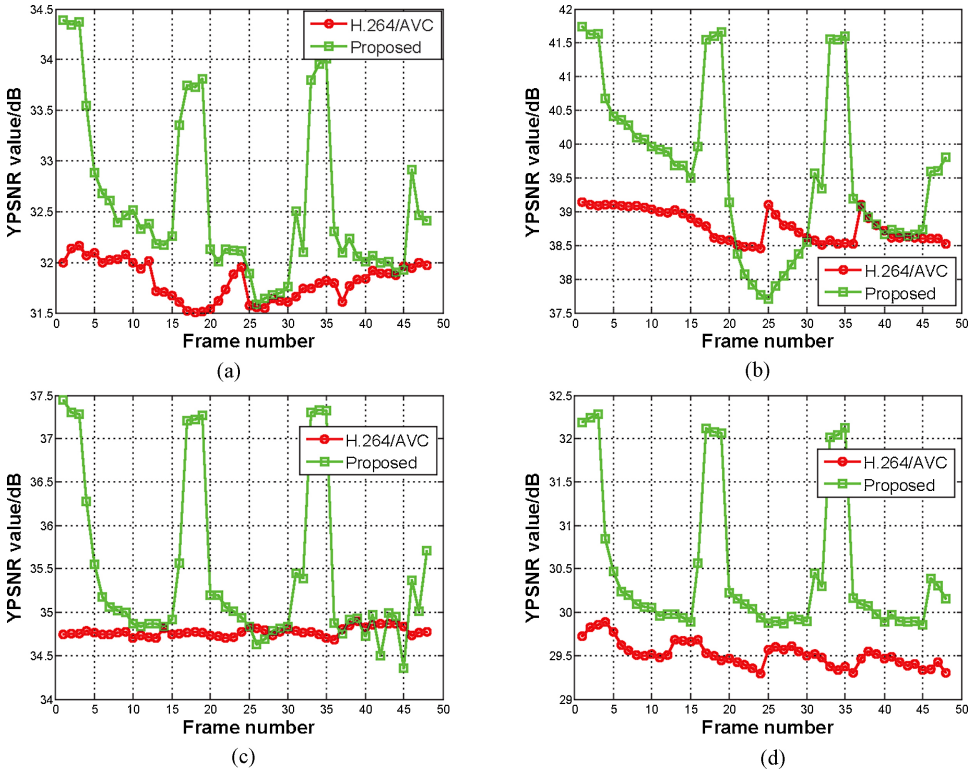


Fig. 17. *PSNR* results of the test sequences. (a) *Foreman*: 201.5 kb/s. (b) *Akiyo*: 204.5 kb/s. (c) *News*: 248.8kb/s. (d) *Waterfall*: 213.9kb/s.

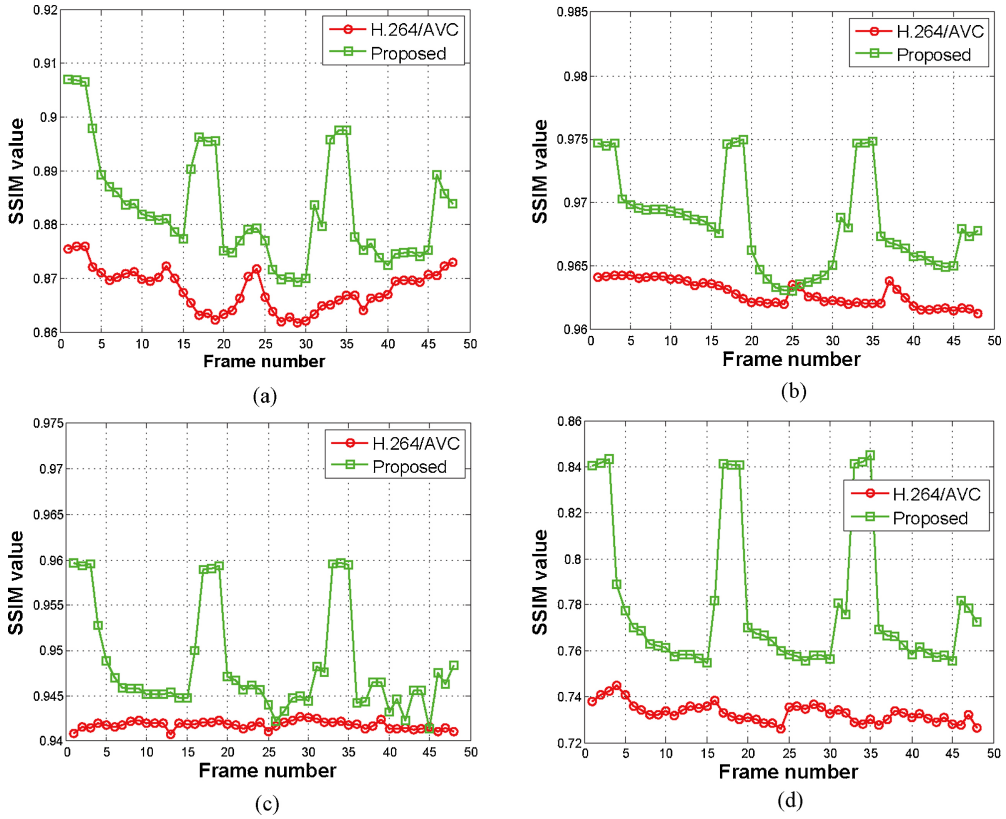


Fig. 18. *SSIM* results of the test sequences. (a) *Foreman*: 201.5 kb/s. (b) *Akiyo*: 204.5 kb/s. (c) *News*: 248.8 kb/s. (d) *Waterfall*: 213.9 kb/s.

where I_1 and I_2 denote the two images to compare, μ is the mean value and σ denotes the variance or covariance, while C_1 and C_2 are two variables to stabilize the division with weak denominator. From Figs. 17 and 18, we find that the proposed scheme not only ensures the visual quality of the decoded result, but also is competitive in pixel-wise reconstruction precision. We further test the *Mobile* sequence with intensive motion and the *Basketball* sequence at 832×480 resolution in Fig. 19. As a result, we can infer that the proposed coding scheme benefits more when the test sequence owns a smooth temporal structure. From Fig. 19(a) where the test *Mobile* sequence features scene movement it could be seen that the average quality (PSNR) of the proposed scheme and H.264/AVC, respectively, is 25.71dB and 25.55dB. More evaluations on different resolutions (*Basketball* sequence with 832×480 resolution) and longer sequences (*Foreman* sequence with 144 frames) have been provided in Fig. 19(b) and (c).

We evaluate the proposed scheme with the state-of-the-art video super-resolution (SR) algorithm, e.g., down-sampling based video coding algorithm (DBC) [2]. The image-based super-resolution algorithm [14] is absorbed in the video SR with 2-D dictionaries for comparison. Fig. 20 and Table II show that the proposed scheme could achieve up to a 2-dB coding gain in PSNR and SSIM within low bit-rate region.

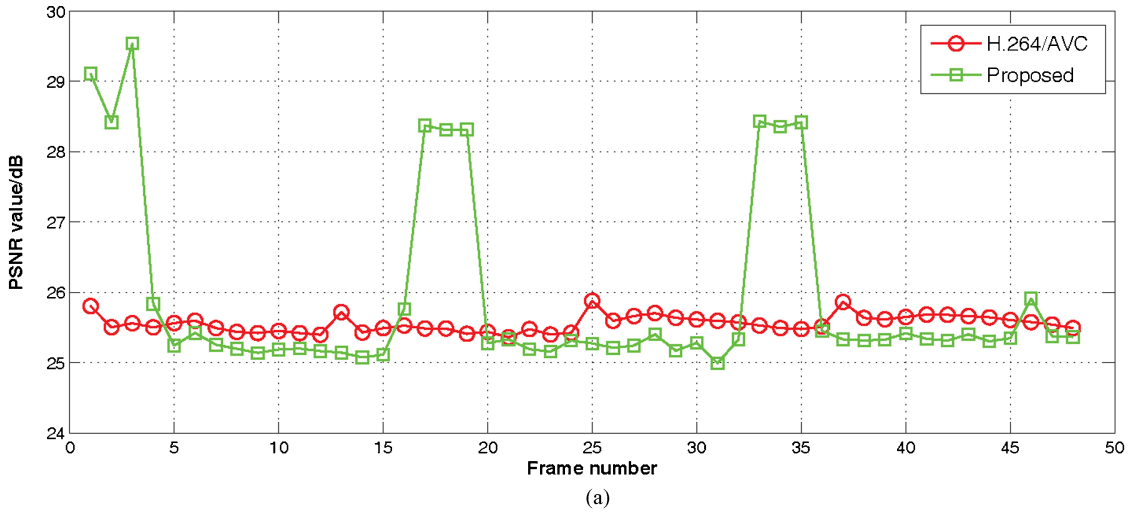
Fig. 21 shows the visual effects comparison of the decoded results between H.264/AVC and the proposed approach. The overall bit rate of the six CIF sequences *Foreman*, *Hall*, *Highway*, *Akiyo*, *News*, and *Waterfall* are 169.4 kb/s, 168.3 kb/s, 126.6 kb/s, 132.9 kb/s, 197.0 kb/s, and 213.9 kb/s, respectively. It is obvious that the proposed compression framework can obtain better subjective performance than the traditional H.264/AVC approach. We also adopt the DMOS matrix as a measurement of visual quality. The fitted curves of objective metrics (PSNR and SSIM) versus DMOS can be illustrated in the form of logistic function [34]. Human observers were asked to provide their perception of quality on a continuous linear scale and a nonlinear mapping between the objective and subjective scores is fitted with the form of logistic function. Smaller DMOS scores indicate that the processed images have better visual perception

$$\begin{aligned} Quality(x) &= \beta_1 \text{logistic}(\beta_2, (x - \beta_3)) + \beta_4 x + \beta_5 \\ \text{logistic}(\tau, x) &= \frac{1}{2} - \frac{1}{1 + \mathrm{e}^{\tau x}} \end{aligned} \tag{20}$$

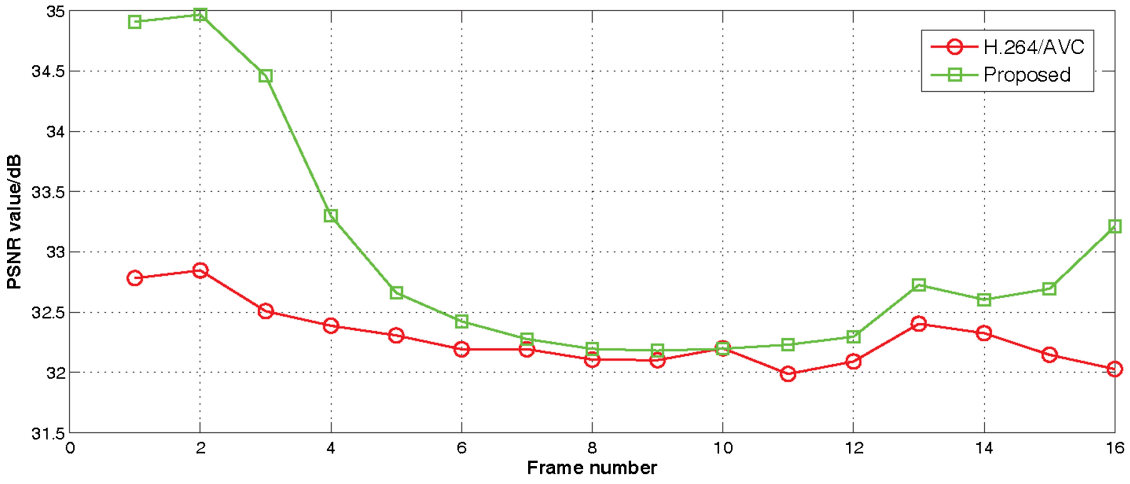
where $Quality(\cdot)$ could describe the DMOS scores and x denotes either SSIM or PSNR. The database of subjective quality assessment is acquired from [35] and we can obtain the parameters of mapping functions. In alignment with the PSNR and SSIM values of the test sequences, the corresponding DMOS scores are also shown in Table II.

To evaluate the coding efficiency of the proposed video compression scheme more precisely, the BD-PSNR [36] is adopted based on the rate-distortion curve fitting. The BD-PSNR indicates the average PSNR difference in decibels over the whole range of test bit rates, which is calculated as

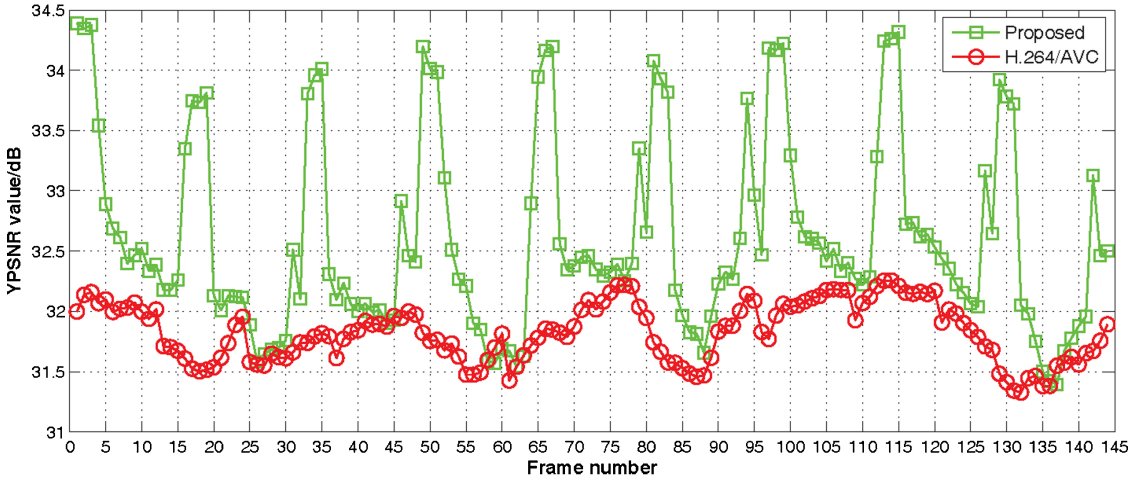
$$BD - PSNR = \left(\int_{r_L}^{r_H} (D_2(r) - D_1(r))\, \mathrm{d}r \right) / (r_H - r_L) \tag{21}$$



(a)



(b)



(c)

Fig. 19. PSNR behavior of the proposed scheme over test sequences with more conditions. (a) *Mobile* (352 × 288): 382.9 kb/s. (b) *Basketball*(832 × 480): 897.5 kb/s. (c) *Foreman:* (352 × 288): 201.5 kb/s.

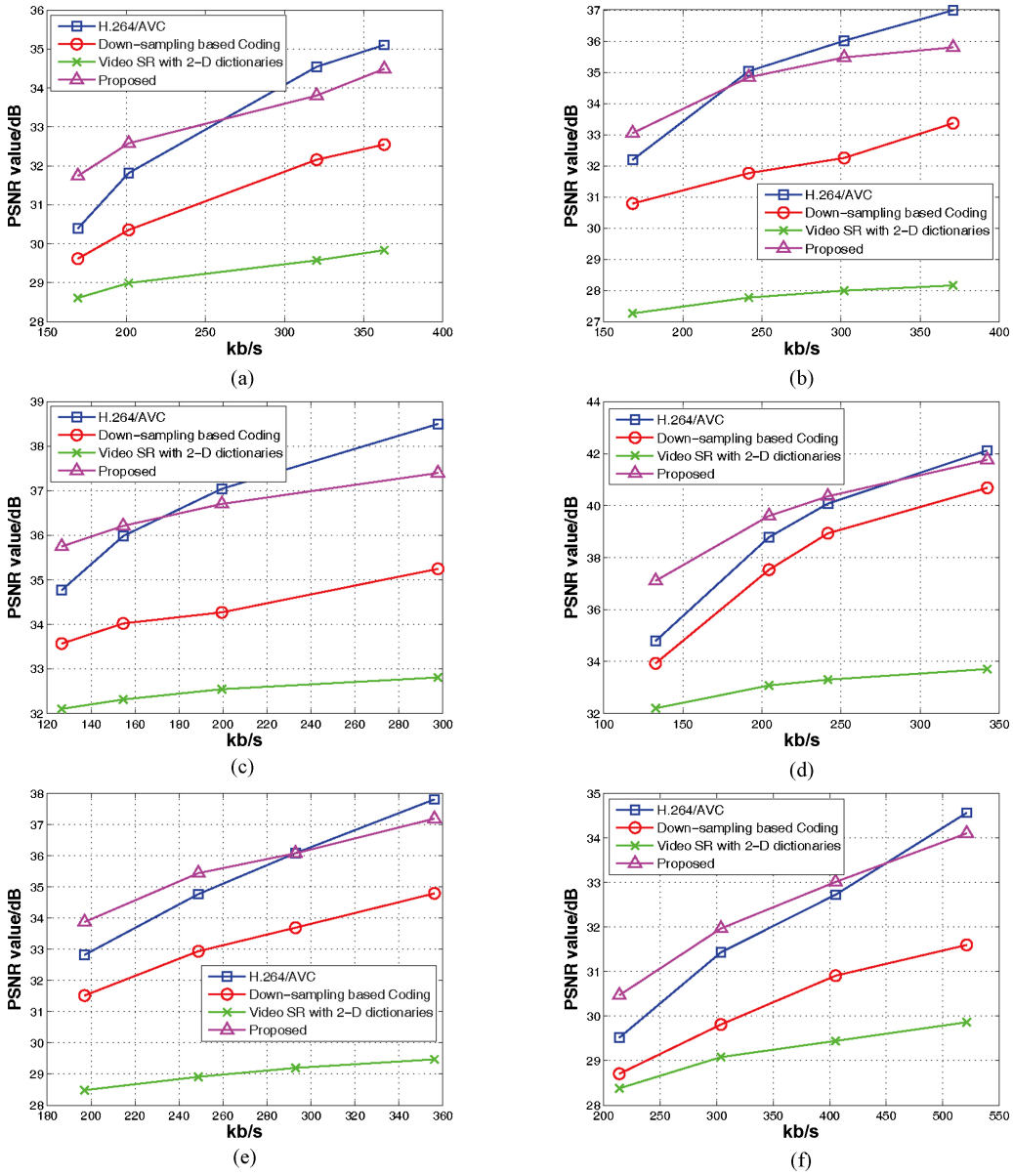


Fig. 20. Rate-distortion performance comparison of the test sequences *foreman_cif*, *hall_cif*, *highway_cif*, *akiyo_cif*, *news_cif*, and *waterfall_cif*. (a) *Foreman* sequence. (b) *Hall* sequence. (c) *Highway* sequence. (d) *Akiyo* sequence. (e) *News* sequence. (f) *Waterfall* sequence.

where $D_1(r)$ and $D_2(r)$ are two R-D curves that, respectively, represent the reconstructed distortions (PSNR) of the reference and the test video sequences approximated by a third-order logarithmic polynomial as

$$D_{PSNR} = D(r) = a_0 r^3 + a_1 r^2 + a_2 r + a_3 \tag{22}$$

and $r = \log R$ represents the logarithm of the output bit rate, so that r_H and r_L in (21) are the logarithmic forms of R_H and R_L that bound the bit-rate range of coding results of two fitted R–D curves, $a_0, \cdots, a_3$ are the parameters of fitting logarithmic polynomial R–D curves. An inverse process that fits the interpolation to find bit rate as a function of PSNR as $D^{\dagger}_{bit-rate} = D^{\dagger}(PSNR) = a_0 PSNR^3 + a_1 PSNR^2 + a_2 PSNR + a_3$ can be used to find BD-Bitrate. The BD-PSNR and BD-Bitrate comparison of the proposed scheme versus H.264/AVC are shown in Table III, as well as the average complexity and the runtime. The experiments are operated by MATLAB on a PC with 3.0GHz dual-core CPU and 4G RAM, and it is evaluated by the decoding pixels per second(pel/s). The total bit rate declines and the entire complexity could be tuned to an acceptable level by a further optimization.

V. Conclusion

In this paper, we proposed a sparse spatio-temporal representation with adaptive regularized dictionary learning for low bit-rate video coding. It acted in a reversed-complexity Wyner–Ziv coding manner, where a subset of key frames serve as a training dataset at the decoder side, while the remaining frames are coded at low resolution. Specifically, a video frame was divided into three layers: a primitive layer, a nonprimitive

TABLE II

CODING PERFORMANCE OF THE PROPOSED SCHEME VERSUS OTHER ALGORITHMS IN PSNR, SSIM, AND DMOS

PSNR, SSIM and DMOS of *Foreman* CIF					
Metrics	Bit Rate (kb/s)	169.4	201.5	320.3	363.0
PSNR (dB)	H.264	30.393	31.815	34.542	35.104
	DBC [2]	29.617	30.354	32.157	32.547
	2-D [14]	28.609	28.991	29.572	29.833
	Proposed	31.739	32.578	33.803	34.489
SSIM	H.264	0.845	0.868	0.908	0.915
	DBC [2]	0.835	0.850	0.880	0.889
	2-D [14]	0.827	0.837	0.852	0.858
	Proposed	0.869	0.883	0.900	0.910
DMOS	H.264	40.6720	37.0609	30.8850	29.8757
	DBC [2]	41.9702	40.6769	35.2771	33.8610
	2-D [14]	43.3724	41.8917	39.5851	38.6430
	Proposed	36.9024	34.6917	32.0713	30.5936
PSNR, SSIM and DMOS of *Hall* CIF					
Metrics	Bit Rate (kb/s)	168.3	241.7	302.0	370.9
PSNR (dB)	H.264	32.189	35.035	36.012	36.992
	DBC [2]	30.787	31.766	32.251	33.369
	2-D [14]	27.264	27.769	27.993	28.159
	Proposed	33.051	34.841	35.476	35.801
SSIM	H.264	0.914	0.939	0.944	0.948
	DBC [2]	0.910	0.925	0.932	0.935
	2-D [14]	0.856	0.863	0.866	0.869
	Proposed	0.921	0.934	0.939	0.942
DMOS	H.264	30.0180	26.6762	26.0664	25.5936
	DBC [2]	33.0658	31.9552	30.9193	30.3895
	2-D [14]	38.9579	37.8531	37.3779	36.9024
	Proposed	29.0360	27.3064	26.6762	26.3078
PSNR, SSIM and DMOS of *Highway* CIF					
Metrics	Bit Rate (kb/s)	126.6	154.6	199.5	297.8
PSNR (dB)	H.264	34.768	35.986	37.045	38.502
	DBC [2]	33.569	34.021	34.271	35.247
	2-D [14]	32.102	32.312	32.544	32.808
	Proposed	35.748	36.211	36.705	37.400
SSIM	H.264	0.903	0.914	0.922	0.934
	DBC [2]	0.891	0.897	0.902	0.913
	2-D [14]	0.883	0.884	0.889	0.894
	Proposed	0.915	0.917	0.923	0.930
DMOS	H.264	31.6236	30.0180	28.8985	27.3064
	DBC [2]	32.6783	31.9484	31.2902	30.2167
	2-D [14]	34.6917	34.5350	33.7558	32.9849
	Proposed	29.8757	29.5931	28.7617	27.8248
PSNR, SSIM and DMOS of *Akiyo* CIF					
Metrics	Bit Rate (kb/s)	132.9	204.5	241.7	342.4
PSNR (dB)	H.264	34.791	38.784	40.078	42.125
	DBC [2]	33.928	37.531	38.932	40.684
	2-D [14]	32.196	33.077	33.300	33.709
	Proposed	37.106	39.607	40.358	41.764
SSIM	H.264	0.934	0.963	0.969	0.976
	DBC [2]	0.923	0.955	0.962	0.971
	2-D [14]	0.913	0.924	0.925	0.930
	Proposed	0.954	0.968	0.971	0.977
DMOS	H.264	27.3064	23.9409	23.3331	22.6620
	DBC [2]	28.1860	26.0653	24.5751	23.2112
	2-D [14]	30.1610	28.6256	28.4903	27.8248
	Proposed	24.9096	23.4323	23.1372	22.5694
PSNR, SSIM and DMOS of *News* CIF					
Metrics	Bit Rate (kb/s)	197.0	248.8	293.0	356.2
PSNR (dB)	H.264	32.822	34.773	36.086	37.812
	DBC [2]	31.520	32.939	33.695	34.797
	2-D [14]	28.481	28.910	29.197	29.471
	Proposed	33.879	35.446	36.083	37.193
SSIM	H.264	0.922	0.942	0.952	0.963
	DBC [2]	0.901	0.921	0.929	0.942
	2-D [14]	0.881	0.889	0.891	0.895
	Proposed	0.936	0.948	0.953	0.960
DMOS	H.264	28.8985	26.3078	25.1342	23.9409
	DBC [2]	29.7194	28.5056	26.9817	26.2895
	2-D [14]	35.0057	33.7558	33.4464	32.8318
	Proposed	27.0519	25.5936	25.0215	24.2561
PSNR, SSIM and DMOS of *Waterfall* CIF					
Metrics	Bit Rate (kb/s)	213.9	303.8	405.2	521.2
PSNR (dB)	H.264	29.516	31.434	32.728	34.567
	DBC [2]	28.705	29.811	30.908	31.598
	2-D [14]	28.379	29.081	29.443	29.863
	Proposed	30.471	31.969	33.014	34.106
SSIM	H.264	0.733	0.816	0.862	0.911
	DBC [2]	0.693	0.749	0.807	0.837
	2-D [14]	0.748	0.794	0.813	0.832
	Proposed	0.779	0.841	0.873	0.902
DMOS	H.264	53.6157	44.9314	38.0113	30.4488
	DBC [2]	56.2183	52.3926	46.0231	41.2370
	2-D [14]	52.4575	47.7841	45.3422	42.6388
	Proposed	49.5028	41.2853	36.2686	31.7730

TABLE III

BD-PSNR AND BD-BIT RATE COMPARISON OF THE PROPOSED SCHEME VERSUS H.264/AVC, AS WELL AS THE AVERAGE COMPLEXITY (PEL/S) AND THE RUNTIME OF THE PROPOSED SCHEME

Sequences	*Foreman*	*Hall*	*Highway*	*Akiyo*	*News*	*Waterfall*
BD-PSNR (dB)	0.052	−0.215	−0.215	0.777	0.356	0.405
BD-Bit Rate (%)	3.019	10.696	7.320	−7.952	−4.742	−7.193
Complexity (pel/s)	253.44	235.76	307.20	239.09	266.78	72.41
Runtime (s/frame)	400.3	429.8	327.7	424.0	377.2	1401.1

Fig. 21. Visual comparison between the proposed scheme and H.264/AVC. From left to right: original, H.264/AVC, and the propose scheme.

coarse layer, and a nonprimitive smooth layer. To make the invertible reconstruction possible in a differential scale-space, it was optimized by constructing a sparse representation of 2-D patches and 3-D volumes over adaptive regularized dictionary learning: a set of 2-D subdictionary pairs trained from primitive patches and a 3-D dictionary trained from nonprimitive volumes. Various sets of low-resolution or high-resolution subdictionary pairs from the primitive patches were learned, while the 3-D LF and HF dictionary pairs were generated from motion-compensated frame interpolation and the K-SVD algorithm for optimal sparse representation and convergence. The lost high-frequency information of the non-key frames

can be synthesized over the adaptive regularized dictionaries. The final high-resolution frames can be acquired by combining all the high-frequency frames and low-frequency frames. It is superior to H.264 in terms of both objective and subjective comparisons, especially in low bit-rate regions.

REFERENCES

[1] H. Schwarz, D. Marpe, and T. Wiegand., "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[2] M. Shen, P. Xue, and C. Wang, "Down-sampling based video coding using super-resolution technique," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 6, pp. 755–765, Jun. 2011.

[3] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[4] R. Zamir, "The rate loss in the Wyner-Ziv problem," *IEEE Trans. Inform. Theory*, vol. 42, no. 6, pp. 2073–2084, Jun. 1996.

[5] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[6] H. Xiong, Y. Xu, Y. F. Zheng, and C. W. Chen, "Priority belief propagation-based inpainting prediction with tensor voting projected structure in video compression," *IEEE Trans. Circuits Systems Video Technol.*, vol. 21, no. 8, pp. 1115–1129, Aug. 2011.

[7] J. Balle and M. Wien, "Extended texture prediction for H.264/AVC intra coding," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2007, pp. 93–96.

[8] A. Dumitras and B. G. Haskel, "An encoder-decoder texture replacement method with application to content-based movie coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 6, pp. 825–840, Jun. 2004.

[9] D. Liu, X. Sun, F. Wu, S. Li, and Y. Zhang, "Image compression with edge-based inpainting," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 10, pp. 1273–1287, Oct. 2007.

[10] Z. Xiong, X. Sun, and F. Wu, "Block-based image compression with parameter-assistant inpainting," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1651–1657, Jun. 2010.

[11] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," *IEEE Trans. Pattern Analysis Mach. Intell.*, vol. 29, no. 3, pp. 463–476, Mar. 2007.

[12] Z. Yuan, H. Xiong, and Y. F. Zheng, "A generic video coding framework based on anisotropic diffusion and spatio-temporal completion," in *Proc. IEEE Int. Conf. Acoustics, Speech Signal Process.*, Mar. 2010, pp. 926–929.

[13] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1838–1857, Jul. 2011.

[14] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[15] W. Freeman and E. Pasztor, "Learning low-level vision," in *Proc. IEEE Int. Conf. Comput. Vision*, Oct. 1999, pp. 1182–1189.

[16] T. Lindeberg, "Scale-space theory: A basic tool for analysising structures at different scales," *J. Appl. Statist.*, vol. 21, no. 2, pp. 224–270, 1994.

[17] Y. Wang, "Image representations using multiscale differential operators," *IEEE Trans. Image Process.*, vol. 8, no. 12, pp. 1757–1771, Dec. 1999.

[18] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Image hallucination with primal sketch priors," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Oct. 2003, pp. 729–736.

[19] H. Chang, D. Yeung, Y. Xiong, C. Bay, and H. Kong, "Super-resolution through neighbor embedding," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2004, pp. 275–282.

[20] W. Fan and D. Yeung, "Image hallucination using neighbor embedding over visual primitive manifolds," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Jun. 2007, pp. 1–7.

[21] A. B. Lee, K. S. Pedersen, and D. Mumford, "The nonlinear statistics of high-contrast patches in natural images," *Int. J. Comput. Vision*, vol. 54, no. 1, pp. 83–103, 2003.

[22] P. Perona and J. Malik, "Detecting and localizing edges composed of steps, peaks and roofs," in *Proc. IEEE Int. Conf. Comput. Vision*, Dec. 1990, pp. 52–57.

[23] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[24] H. Lee and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. Advances Neural Inform. Process. Syst.*, 2007, pp. 801–808.

[25] M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 27–35, Jan. 2009.

[26] B. C. Song, S.-C. Jeong, and Y. Choi, "Video super-resolution algorithm using bidirectional overlapped block motion compensation and on-the-fly dictionary training," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 3, pp. 274–285, Mar. 2011.

[27] B.-D. Choi, J.-W. Han, C.-S. Kim, and S.-J. Ko, "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 4, pp. 407–416, Apr. 2007.

[28] M. T. Orchard and G. J. Sullivan, "Overlapped block motion compensation: An estimation-theoretic approach," *IEEE Trans. Image Process.*, vol. 3, no. 9, pp. 693–699, Sep. 1994.

[29] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, 2004.

[30] D. L. Donoho and Y. Tsaig, "Fast solution of l1-norm minimization problems when the solution may be sparse," *IEEE Trans. Inform. Theory*, vol. 54, no. 11, pp. 4789–4812, Nov. 2008.

[31] G. Davis and M. Avellaneda, "Adaptive greedy approximations," *Constructive Approximation*, vol. 13, no. 1, pp. 57–98, Jan. 1997.

[32] D. Donoho, I. Drori, V. Stodden, Y. Tsaig, and M. Shahram. (2007). *Sparselab* [Online]. Available: http://sparselab.stanford.edu/

[33] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[34] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.

[35] H. R. Sheikh, Z. Wang, A. C. Bovik, and L. K. Cormack. (2012, Feb.). *Image and Video Quality Assessment Research at LIVE* [Online]. Available: http://live.ece.utexas.edu/research/quality/

[36] G. Bjontegaard, *Calculation of Average PSNR Differences Between rd-Curves (VCEG-M33)*, VCEG Meeting (ITU-T SG16 Q.6), Austin, TX, Apr. 2001.

[37] W. Dong, L. Zhang, and G. Shi, "Centralized sparse representation for image restoration," in *Proc. IEEE Int. Conf. Comput. Vision*, Nov. 2011, pp. 1259–1266.

[38] W. Dong, X. Li, L. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2011, pp. 457–464.

[39] M. Varma and A. Zisserman, "Classifying images of materials: Achieving viewpoint and illumination independence," in *Proc. Eur. Conf. Comput. Vision*, vol. 3. May 2002, pp. 255–271.

[40] T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *Int. J. Comput. Vision*, vol. 43, no. 1, pp. 29–44, Jun. 2001.

[41] O. G. Cula and K. J. Dana, "Compact representation of bidirectional texture functions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Dec. 2001, pp. 1041–1047.

[42] C. Schmid, "Constructing models for content-based image retrieval," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, vol. 2. Dec. 2001, pp. 39–45.

[43] M. Varma and A. Zisserman, "Texture classification: Are filter banks necessary," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, vol. 2. Dec. 2003, pp. 691–698.

[44] C. Liu and D. Sun, "A Bayesian approach to adaptive video super resolution," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2011, pp. 209–216.

[45] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3467–3478, Aug. 2012.

[46] Y. Li, X. Sun, H. Xiong, and F. Wu, "Incorporating primal sketch based learning low bit-rate image compression," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3. Sep. 2007, pp. 173–176.

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

Since then, he has been with the Department of Electronic Engineering, SJTU, where he is currently a Professor. From December 2007 to December 2008, he was with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, as a Research Scholar. From 2011 to 2012, he was a Scientist with the Division of Biomedical Informatics, University of California, San Diego. He has published over 100 refereed journal and conference papers. In SJTU, he directs the Image, Video, and Multimedia Communications Laboratory and Multimedia Communication Area in the Key Laboratory of the Ministry of Education of China–Intelligent Computing and Intelligent System that is also cogranted by Microsoft Research, Beijing, China. His current research interests include source coding/network information theory, signal processing, computer vision and graphics, and statistical machine learning.

Dr. Xiong was a recipient of the Top 10% Paper Award for Super-Resolution Reconstruction With Prior Manifold on Primitive Patches for Video Compression at the 2011 IEEE International Workshop on Multimedia Signal Processing. He received the First Prize of the Shanghai Technological Innovation Award in 2011, the SMC-A Excellent Young Faculty Award of Shanghai Jiao Tong University in 2010, and the New Century Excellent Talents Award in University, Ministry of Education of China, in 2009. He serves as a Technical Program Committee Member or the Session Chair for a number of international conferences.

Zhiming Pan received the B.S. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2009, and the M.S. degree from the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2012.

His current research interests include video compression and signal processing.

Xinwei Ye received the B.S. degree in mathematics from Nanjing University, Nanjing, China, in 2009. He is currently pursuing the M.S. Degree with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China.

His current research interests include compressive sensing, video compression, and signal processing.

Chang Wen Chen (F'04) received the B.S. degree from the University of Science and Technology of China, Hefei, Anhui, China, in 1983, the M.S.E.E. degree from the University of Southern California, Los Angeles, in 1986, and the Ph.D. degree from the University of Illinois at Urbana-Champaign, Urbana, in 1992.

He has been a Professor of computer science and engineering with the Department of Computer Science and Engineering, State University of New York at Buffalo, Buffalo, since 2008. Previously, he was an Allen S. Henry Distinguished Professor with the Department of Electrical and Computer Engineering, Florida Institute of Technology, Melbourne, from 2003 to 2007. He was on the Faculty of Electrical and Computer Engineering, University of Missouri-Columbia, Columbia, from 1996 to 2003, and with the University of Rochester, Rochester, NY, from 1992 to 1996. From 2000 to 2002, he served as the Head of the Interactive Media Group, David Sarnoff Research Laboratories, Princeton, NJ. He has also been a Consultant with Kodak Research Laboratories, Microsoft Research, Beijing, China, Mitsubishi Electric Research Laboratories, Cambridge, MA, NASA Goddard Space Flight Center, Greenbelt, MD, and the U.S. Air Force Rome Laboratories, Rome, NY.

Dr. Chen was the Editor-in-Chief for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from January 2006 to December 2009. He has served as an editor for the PROCEEDINGS OF THE IEEE, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE MULTIMEDIA, the *Journal of Wireless Communication and Mobile Computing*, the *EURASIP Journal of Signal Processing: Image Communications*, and the *Journal of Visual Communication and Image Representation*. He has also chaired and served on numerous technical program committees for the IEEE and other international conferences. He was elected fellow of the SPIE for his contributions to electronic imaging and visual communications.