Sparse Spatio-Temporal Representation with Adaptive Regularized Dictionaries for Super-Resolution based Video Coding

Zhiming Pan[†] Hongkai Xiong[†]

[†]Department of Electronic Engineering, Shanghai Jiao Tong Univ., Shanghai, 200240, China

{pzming, xionghongkai}@sjtu.edu.cn

Abstract

In this paper, we propose a sparse representation learning with adaptive regularized dictionaries and develop a low bit-rate video coding scheme. In a reversed-complexity manner, it select a subset of key frames to encode at original resolution, while the rest are down-sampled and super-resolution reconstructed by a sparse super-resolution estimations using key frames as training set. Since primitive patches are of low dimensionality and can be well learned from the primitive patches across different images, video frame is divided into three layers: a primitive layer, a non-primitive coarse layer, and a non-primitive smooth layer. The non-primitive layer is constructed as volumes to keep consistent along the motion trajectory, which enables sparse representations over a learned 3-D spatio-temporal dictionary. Correspondingly, the target is formulated as an optimization problem by constructing a sparse representation of low-resolution frame patches or volumes over adaptive regularized dictionaries: a set of 2-D sub-dictionary pairs trained from 2-D primitive patches and a 3-D dictionary trained from non-primitive volumes. 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. Experimental results validate the compression efficiency of the proposed scheme *versus* the H.264/AVC in terms of both objective and subjective comparison.

1 Introduction

Noticeably, increasing low-quality visual data from mobile phones, digital cameras and mobile TV, stimulate a huge demand for video analysis and computer vision techniques. It arises a big perspective 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 inter-layer prediction with up-sampling. However, the coding burden is dominated by a a rigid partition between encoder (heavy) and decoder (light). It would constrain the ubiquitous multimedia access for increasingly 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 the requirements of mobile camera phone and wireless sensor networks [3]. Limited by the estimation of correlated side-information,

This work was supported in part by the NSFC, under Grants 60736043, 60928003, 60772099, the Program for New Century Excellent Talents in University (NCET-09-0554), and Shanghai Rising-Star Program (11QA1402600).

1068-0314/12 $26.00 © 2012 IEEE
DOI 10.1109/DCC.2012.22

practical DVC schemes often have a considerable performance loss compared with traditional H.264/AVC. Along the insight, it stimulates us to further 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 inter prediction [4]. As a matter of fact, more prediction methods, e.g. inpainting-based prediction [5], and texture prediction [6], have been noticed to achieve a better performance. It infers a promising potential to synthesize and hallucinate missing texture with good perceptual quality. By now, the attempts to restore the missing information have involved in various assistant side information, e.g. edge [7], and assistant parameters [8]. To maintain a temporal consistency of video, a space-time completion has recently been referred in a global optimization sense [9].

Naturally, more attention has been drawn 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-resolution contents could be estimated in a nonparametric sense. Recently, learning-base approaches have achieved the best reconstruction results in super-resolution task by inferring the lost high-frequency information from a learned co-occurrence prior knowledge [10]. As [11], an example-based learning strategy was proposed where the low-resolution to high-resolution prediction is learned via a Markov Random Field (MRF) solved by belief propagation. Sun et al. [12] 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, [10] considered to recover the sparse representation coefficients of each low-resolution patch base on a dictionary composed of low-resolution patches, then the high-resolution patch is 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 only efficient for input images of similar statistical features.

This paper proposes a low bit-rate video coding scheme where sparse super-resolution estimations over dictionaries provide effective nonparametric approaches to inverse problems. A subset of key frames in a video sequence are encoded at high-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 of an image are of low dimensionality and can be well learned from the primitive patches across different images [12]. Specifically, a video frame is divided into three layers: a primitive layer, a non-primitive coarse layer, and a non-primitive smooth layer. Considering that image primitives may vary significantly across different frames or different patches in a single frame, we propose to learn various sets of low-resolution / high-resolution subdictionary pairs from the primitive patches of the key frames. It is worth mentioning that non-primitive volumes are consistent along the motion trajectory, have little structure information, and have more sparse representations over a learned 3-D spatio-temporal dictionary. It is fulfilled by hierarchical bi-directional motion estimation and adaptive overlapped block motion compensation. Correspondingly, the target is formulated as an optimization problem by constructing a sparse representation of low-resolution frame patches or volumes over adaptive regularized dictionaries: a set of 2-D subdictionary pairs trained from 2-D primitive patches and a 3-D dictionary trained from non-primitive volumes. 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 not only ensure the visual quality, but also be competitive in rate-distortion performance.

2 Super-resolution based Video Coding Scheme

2.1 Incorporating the Sparse-land Prior

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

Inspired by the basic assumption that each image patch can be represented as a linear combination of a small subset of patches (atoms) from a fixed dictionary [10], the super-resolution task can be described as follows:

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

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.

In order to avoid the complexities caused by the different resolutions between Z_l and X_h, we assume hereafter that Z_l is scaled-up by an interpolation operator Q (e.g. bicubic), returning to the size of X_h. The scaled-up image is denoted by Z_{LF}:

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

Z_{LF} is the low-frequency (LF) part of X_h. The goal is to recover $\hat{X}_h$ from Z_{LF}.

The algorithm we proposed 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 $x_{i,j} = R_{i,j}X_h$ be an image patch which can be sparsely represented 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}. Since the operator $L^{all} = QSH$ transforms the complete HR image X_h to 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 being a portion of L^{all}, and $\hat{v}_{i,j}$ is the additive noise. Thus we have:

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

The key observation from the above derivations 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. This implies that if we recover the sparse representation coefficients of each LR patch base on a LR dictionary, then the HR patch is reconstructed using the recovered coefficients in terms of the corresponding HR dictionary.

Different from [10] which use sparse representation prior on arbitrary image patches, we take 2-D sparse representation on patches located only in image primitive layer. However, a prelearned universal dictionary is neither optimal nor efficient in sparsely coding all of the possible image structures. Hence, we consider to learn various sets of LR / HR subdictionary pairs.

2.2 Overview of the super-resolution reconstruction method

The proposed video coding scheme can be described as: Given a HR video sequence F_h, we decompose it into: a selected HR key frames (KFs) X_h and the down-sampled 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 $\hat{Z}_l$; a HR version of $\hat{Z}_l$ (denoted by $\hat{Z}_h$) is recovered from $\hat{X}_h$ after the learning phase and synthesis phase.

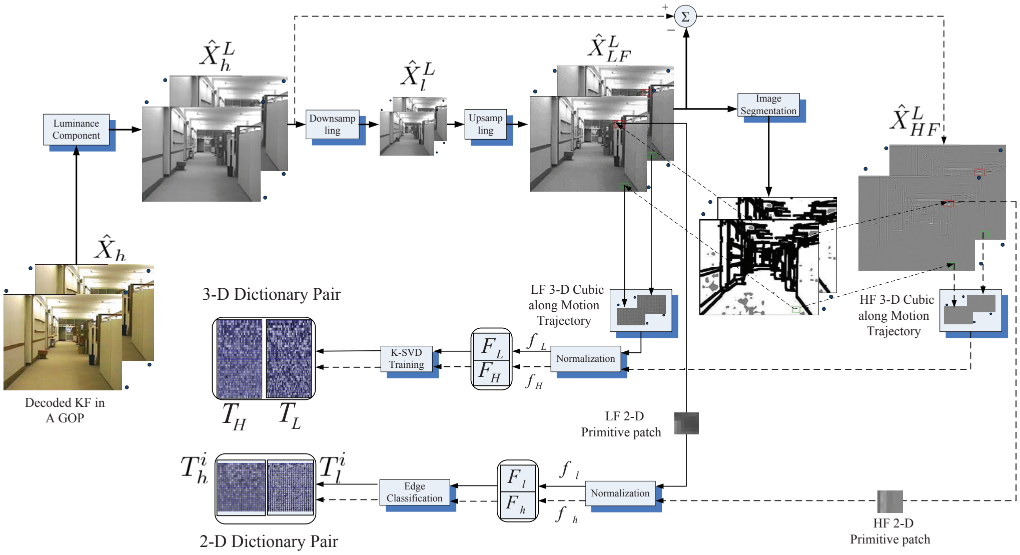


Figure 1: The learning scheme of the proposed video coding framework

2.2.1 Learning phase

Different from existing learning-based super-resolution methods, we generate two kinds of dictionaries: a set of 2-D subdictionary pairs and a 3-D dictionary pair, as shown in Fig. 1. Considering these, we first get the down-sampled version $\hat{X}_l^L$ of the decoded KFs $\hat{X}_h^L$. $\hat{X}_{LF}^L$ are interpolated from $\hat{X}_l^L$, which contain the LF component of $\hat{X}_h^L$. The high-frequency (HF) part are generated from the difference of $\hat{X}_h^L$ and $\hat{X}_{LF}^L$. $\hat{X}_{LF}^L$ are classified into a primitive layer, a non-primitive coarse layer and a non-primitive smooth layer. The training sets of the 2-D subdictionaries are acquired from frame patches located in the primitive layer of the HF and their corresponding LF frames; the patches located in the non-primitive coarse layer are combined with the most matched patches in the neighbor training frames to form 3-D volumes, these volumes would be used to learn the 3-D dictionary pair.

2.2.2 Synthesis phase

In the synthesis phase, the input LR NKFs $\hat{Z}_l$ would be synthesized to generate the final HR frames $\hat{Z}_h$. Initially, $\hat{Z}_l$ are interpolated into $\hat{Z}_{LF}$ with the same size as $\hat{X}_h$. Once selecting the up-sampled frames denoted as $\hat{Z}_{LF_i}$, their luminance component $\hat{Z}_{LF_i}^L$ are also classified into three layers. For a LF patch in the primitive layer of $\hat{Z}_{LF_i}^L$, 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 non-primitive coarse layer are extracted from $\hat{Z}_{LF_i}^L$ along the motion trajectory, and the corresponding HF volumes can be inferred from the 3-D dictionary pair $\{T_L, T_H\}$. When obtained all the HF patches and volumes, we construct the primitive and non-primitive HF frames independently. Hence, the super-resolution HR frames $\hat{Z}_{h_i}^L$ can be obtained by adding HF frames to the corresponding LF ones. Finally, $\hat{Z}_{h_i}$ will be generated by combining the obtained HR luminance frames with the interpolated chroma information.

3 Super-resolution via Adaptive Sparse Representation

3.1 Image Primitives based Segmentation Method

Image primitive mainly consists of edge segments, bars, and terminations, and can reflect the brightness changes of the image [13]. As [12], we adopt a set of Gaussian derivative filters to extract by orientation energy:

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

where $f^{odd}_{\sigma,\theta}$ and $f^{even}_{\sigma,\theta}$ are the first and second Gaussian derivative filters at scale σ and orientation θ.

Once the primitives frame is obtained, the primitive layer is labeled by overlapping frame patches along the primitives sketch. The remaining non-primitive layer is filtered by a high-pass filter. If a block contains sufficient HF energy, it is labeled as a part of non-primitive coarse layer, then the non-primitive coarse layer is distinguished from the non-primitive smooth layer.

3.2 Dictionary learning

1) Learning 2-D Subdictionaries on Primitive Frame Patches

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 $\hat{X}^L_{LF}$ and $\hat{X}^L_{HF}$. We aim to learn K subdictionary pairs $\{T_l^k, T_h^k\}_k$ from $\{F_l, F_h\}$. To this end, 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 edges, corners, and terminations of different orientations and scales, this inspires us to cluster $\{F_l, F_h\}$ into K clusters $\{F_l^k, F_h^k\}_k, k = 1, 2, ..., K$ according to the orientations (16 orientations) and scales (3 scales) of primitive patches. That means the primitive patches with the same orientation and scale would be classified into the same cluster.

Now the remaining problem is how to learn a subdictionary pair $\{T_l^k, T_h^k\}$ from $\{F_l^k, F_h^k\}$. The design of $\{T_l^k, T_h^k\}$ can be intuitively formulated by the following function:

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

where $F_k = [F_l^k, F_h^k]^T, T_k = [T_l^k, T_h^k]^T$. Eq. (5) could be iteratively optimized by alternatingly optimizing Φ_k and T_k when the other is fixed.

However, the K $l_2 - l_1$ joint minimization in Eq. (5) requires much computational cost, so we simply use F_k as the final subdictionary T_k directly based on the following considerations. Firstly, F_k is a subset of $\{F_l, F_h\}$ after clustering, so the computational cost of the sparse coding of a given primitive patch over F_k is small enough. Furthermore, the intrinsic dimensionality of image primitives is very low, thus it is possible to represent all the image primitives well by a small number of primitive examples from the highly correlated training images [13].

2) Learning 3-D Dictionary on Non-primitive Volumes along Motion Trajectory

The non-primitive volumes along the motion trajectory are consistent in the temporal dimension, they are supposed to have more sparse representation structures over a learned 3-D dictionary. Hence, the spatio-temporal consistency can be better obtained by taking a 3-D spatio-temporal dictionary into consideration [14].

According to Eq. (1), we extend it to handle video sequences by considering the temporal dimension. Let X_h and Z_l represent the original HR and down-sampled LR videos, respectively, and an index t in the range [1, T] is added to account for the time dimension, thus we have:

$$f^{All}_{Video}(\{\alpha_{ijt}\}_{ijt}, X_h, D_h) = \arg\min_{X_h,\{\alpha_{ijt}\}} \{\lambda\|SHX_h - Z_l\|_2^2 + \sum_{i,j}\sum_{t=1}^{T}\mu_{ijt}\|\alpha_{ijt}\|_0 + \sum_{i,j}\sum_{t=1}^{T}\|D_h\alpha_{ijt} - R_{ijt}X_h\|_2^2\}. \tag{6}$$

The term $R_{ijt}X$ extracts a patch from X_h in time t and location [i, j] and this patch may be a 3-D volume.

In Eq. (6), all the patches in the sequence are used to train a single dictionary which is then applied to the entire sequence. However, training a single dictionary is problematic: the scene in a video is expected to change rapidly over time; the dimension of the dictionary grows rapidly with the increase of time space.

An alternative approach is to define a locally temporal penalty term. Since the KFs and NKFs are extremely correlated, we learn a 3-D dictionary for a reference frame in the KFs and decompose all the neighbor NKFs in the same GOP with such a dictionary. Hence, Eq. (6) can be rewritten for a reference KF individually as:

$$f^{r}_{Video}(\{\alpha_{ij}\}_{ij}, X^r_h, D^r_h) = \arg\min_{X^r_h,\{\alpha_{ij}\}} \{\lambda\|SHX^r_h - Z^r_l\|_2^2 + \sum_{i,j}\mu_{ij}\|\alpha_{ij}\|_0 + \sum_{i,j}\|D^r_h\alpha_{ij} - R_{ijr}X_h\|_2^2\}. \tag{7}$$

where X^r_h is a reference frame in the KFs of a GOP. The learnt dictionary of the previous GOP is propagated to the next GOP as the initial dictionary, reducing the number of required training iterations.

In order to get a more sparse representation of the sequence and considering the motion influence, the volume is acquired along the motion trajectory from block-matching based motion estimation, thus we have $\tilde{X}_h$ instead of X_h in Eq. (7), where $\tilde{X}_h$ is the motion compensated version of X_h according to a reference frame. $R_{ijr}\tilde{X}_h$ means to extract a volume from X_h along the motion trajectory.

For a reference frame $\{\hat{X}^L_{LF}\}_{RF}$ in $\hat{X}^L_{LF}$, we use a motion-compensated frame interpolation (MCFI) approach to predict the estimated reference frame $\{\hat{X}^L_{LF}\}_{\widetilde{RF}}$ according to its preceding frame $\{\hat{X}^L_{LF}\}_{PF}$ and following frame $\{\hat{X}^L_{LF}\}_{FF}$. The LF volumes are extracted by concatenating the patches located in the non-primitive coarse layer of $\{\hat{X}^L_{LF}\}_{RF}$ and the corresponding patches from $\{\hat{X}^L_{LF}\}_{\widetilde{RF}}$, the HF cubic volumes are generated similarly. Next section will address our MCFI operation.

Once collecting all the LF and HF cubic volumes, we 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 non-primitive layer which have little structure information, we would learn a universal dictionary pair from them. Let $F_L = [f^1_l \; f^2_l \cdots f^p_l]$ be an $n \times P$ matrix of P training sets of length n pixels each, the objective of the K-SVD algorithm [15] is to train an overcomplete dictionary T_L of size $n \times K$ ($P \gg K$ and $K > n$) for a given sparsity level S,

$$min_{T_L,\Theta}\|F_L - T_L\Theta\|_F^2 \quad s.t. \quad \forall i, \|\theta_i\|_0 \leq S \tag{8}$$

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 $T_L = [t^1_l \; t^2_l \cdots t^K_l]$. Using the dictionary learned from the previous GOP as initial dictionary, the K-SVD algorithm progressively improves it in order to optimize the expression.

After the K-SVD dictionary training procedure, we get the LF 3-D dictionary T_L and the corresponding sparse representation coefficients matrix Θ, the next step is to construct the HF 3-D dictionary T_H. Recall from Section 2.1, the target is to recover the HF training sets $F_H = [f_h^1\ f_h^2 \cdots f_h^P]$ by approximating them as $F_H \approx T_H\Theta$. Thus, the dictionary is defined as the one to minimize the mean approximation error, i.e.,

$$T_H = min_{T_H} \|F_H - T_H\Theta\|_F^2. \tag{9}$$

The solution is given by the following Pseudo-Inverse expression (given that Θ has full row rank): $T_H = F_H\Theta^+ = F_H\Theta^T(\Theta\Theta^T)^{-1}$.

The two pairs of corresponding dictionaries $\{T_l^i, T_h^i\}_i$ and $\{T_L, T_H\}$ conclude the training phase of the super-resolution algorithm, which starts with the decoded HR key frames $\hat{X}_h$.

3.3 Motion-Compensated Frame Interpolation Method

Let f_{n-1}, f_n and f_{n+1} denote the preceding frame, the intermediate reference frame, and the following frame, respectively. Similar to [16], we utilizes hierarchical bi-directional motion estimation (ME) to find motion vectors (MVs) and adaptive overlapped block motion compensation (AOBMC) to reduce blocking artifacts.

3.3.1 Hierarchical Motion Estimation

In block matching based ME, we compute the forward MV for a $M \times M$ block $B_{i,j}$ by minimizing the SAD values: $SAD(B_{i,j}, \mathbf{v}) = \sum_{\mathbf{s}\in B_{i,j}} |f_n(\mathbf{s}) - f_{n-1}(\mathbf{s} + \mathbf{v})|$.

When the forward MVs for all the $M \times M$ blocks are selected, we allocate each MV to its subordinate $L \times L$ sub-blocks. A local ME is performed with a smaller search window for each $L \times L$ sub-block around the selected MV. In this way, we can get the forward and backward MVs $\{\boldsymbol{\nu}\}$ for all the $L \times L$ sub-blocks.

3.3.2 Bi-Directional OBMC Using Adaptive Window

OBMC [18] aims to reduce the blocking artifacts from conventional MC method, where the pixel $f_n(\mathbf{s})$ in a $L \times L$ block $S_{i,j}$ is predicted as:

$$f_n(\mathbf{s}) = \sum_{p=-1}^{1}\sum_{q=-1}^{1} w_{p,q}(\mathbf{s}) f_{n-1}(\mathbf{s} + \boldsymbol{\nu}_{i+p,j+q}). \tag{10}$$

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

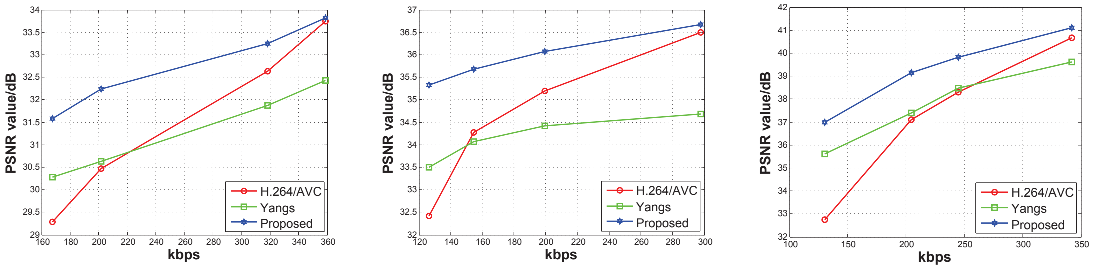
However, if adjacent blocks have substantially different motions, OBMC can yield blurring or over-smoothing artifacts since the weighting coefficients are determined only by the relative distances of the pixels within the block. To overcome this problem, the AOBMC method [17] is adopted to adaptively control the weighting coefficients in terms of the reliability of the neighboring MVs. The reliability of the neighboring MV $\boldsymbol{\nu}_{i+p,j+q}$ for predicting the current block $S_{i,j}$ is defined as:

$$\Phi_{S_{i,j}}(\boldsymbol{\nu}_{i+p,j+q}) = \frac{SAD(S_{i,j}, \boldsymbol{\nu}_{i,j})}{SAD(S_{i,j}, \boldsymbol{\nu}_{i+p,j+q})}. \tag{11}$$

As $\Phi_{S_{i,j}}(\boldsymbol{\nu}_{i+p,j+q})$ gets more closer to 1, $\boldsymbol{\nu}_{i+p,j+q}$ are more reliably to compensate the current block $S_{i,j}$. The weighting coefficients $w_{p,q}(\mathbf{s})$ in Eq. (10) are modified as:

$$\hat{w}_{p,q}(\mathbf{s}) = \frac{\Phi_{S_{i,j}}(\boldsymbol{\nu}_{i+p,j+q}) w_{p,q}(\mathbf{s})}{\sum_{s=-1}^{1}\sum_{t=-1}^{1} \Phi_{S_{i,j}}(\boldsymbol{\nu}_{i+s,j+t}) w_{s,t}(\mathbf{s})} \tag{12}$$

When substituting $\hat{w}_{p,q}(\mathbf{s})$ in Eq. (10), we get the motion-compensated pixels of f_n using the forward MVs from f_{n-1}. Likewise, the backward motion-compensated values of f_n is obtained from 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.



(a) *Foreman* sequence (b) *Highway* sequence (c) *Akiyo* sequence

Figure 2: The rate-distortion performance comparison of the test sequences $foreman\_cif$, $highway\_cif$, and $akiyo\_cif$.

3.4 Synthesis Phase

The overview of the synthesis phase is described in Section 2.2.2. If a LF patch f_l located in the primitive layer of $\hat{Z}^L_{LF_i}$, we get its feature value according to its orientation and scale. The subdictionary pair $\{T^i_l, T^i_h\}$ which is composed of primitive patches of the same orientation and scale as f_l is selected as the optimal one.

To find the sparsest representation of f_l, it can be formulated as:

$$\min \|\alpha\|_0 \quad s.t. \quad \|T^i_l\alpha - f_l\|^2_2 \leq \epsilon. \tag{13}$$

where $T^i_l \in R^{m\times k}$ with $m \ll k$, and $f_l \in R^m$. This NP-hard problem can be solved by solving a l_1-norm minimization problem instead, as long as the α are sufficient sparse [19]:

$$\min \lambda\|\alpha\|_1 + \|T^i_l\alpha - f_l\|^2_2. \tag{14}$$

where λ is a trade-off parameter between sparsity and fidelity. This is a non-linear convex optimization problem and can be solved efficiently by various methods [20].

When α is obtained, the corresponding HR primitive patch f_h can be attained by the linear combination of columns in T^i_h using α as the coefficients: $f_h = T^i_h\alpha$.

If a LF patch located in the non-primitive coarse layer of a reference frame in $\hat{Z}^L_{LF_i}$, we get the corresponding patch from the estimated reference frame acquired by using the aforementioned MCFI algorithm. These two relevant LF patches are concatenated as a LF volume. The procedure of synthesizing the corresponding HF volume is similar to synthesizing a HF primitive patch, except the 3-D dictionary pair $\{T_L, T_H\}$ and volumes instead of 2-D subdictionary pair $\{T^i_l, T^i_h\}$ and primitive patches.

4 Experimental Results

4.1 Implementation

In the experiments, all the test sequences are of the YUV 4:2:0 format, 30HZ frame rate, and a GOP size of 16 frames. Given an original video sequence, we select three successive

Table 1: The coding performance of the proposed method compared to standardized H.264/AVC in PSNR, SSIM and DMOS

(a)

PSNR and SSIM values of *Foreman* CIF					
Metrics	bit-rate (kbps)	167.5	201.5	318.5	358.7
PSNR (dB)	H.264	29.290	30.464	32.629	33.748
	Yang's	30.276	30.632	31.874	32.428
	proposed	31.585	32.245	33.242	33.816
SSIM	H.264	0.824	0.846	0.882	0.897
	Yang's	0.843	0.831	0.876	0.881
	proposed	0.868	0.879	0.894	0.903

PSNR and SSIM values of *Hall* CIF					
Metrics	bit-rate (kbps)	168.3	241.7	302.0	366.0
PSNR (dB)	H.264	30.233	33.401	34.600	35.690
	Yang's	31.221	33.038	33.621	33.736
	proposed	32.500	34.180	34.719	34.937
SSIM	H.264	0.890	0.926	0.935	0.942
	Yang's	0.908	0.923	0.928	0.931
	proposed	0.917	0.930	0.935	0.938

PSNR and SSIM values of *Highway* CIF					
Metrics	bit-rate (kbps)	126.6	154.6	199.5	297.8
PSNR (dB)	H.264	32.419	34.272	35.189	36.491
	Yang's	33.501	34.076	34.421	34.679
	proposed	35.329	35.672	36.066	36.677
SSIM	H.264	0.877	0.896	0.906	0.917
	Yang's	0.883	0.897	0.902	0.911
	proposed	0.912	0.914	0.919	0.925

(b)

PSNR and SSIM values of *Akiyo* CIF					
Metrics	bit-rate (kbps)	130.1	204.5	244.7	342.4
PSNR (dB)	H.264	32.748	37.096	38.308	40.658
	Yang's	35.621	37.392	38.467	39.613
	proposed	36.997	39.145	39.820	41.114
SSIM	H.264	0.910	0.953	0.960	0.971
	Yang's	0.931	0.961	0.963	0.968
	proposed	0.954	0.967	0.970	0.976

PSNR and SSIM values of *News* CIF					
Metrics	bit-rate (kbps)	197.0	248.8	295.5	356.2
PSNR (dB)	H.264	31.385	33.364	34.768	36.036
	Yang's	32.515	33.413	34.217	34.732
	proposed	33.537	35.019	35.466	36.526
SSIM	H.264	0.902	0.928	0.942	0.951
	Yang's	0.921	0.932	0.938	0.944
	proposed	0.933	0.946	0.949	0.957

PSNR and SSIM values of *Waterfall* CIF					
Metrics	bit-rate (kbps)	210.0	302.5	405.2	521.2
PSNR (dB)	H.264	28.136	30.139	31.867	32.700
	Yang's	28.282	29.321	30.232	31.071
	proposed	30.089	31.462	32.409	33.383
SSIM	H.264	0.651	0.761	0.835	0.861
	Yang's	0.713	0.724	0.796	0.831
	proposed	0.765	0.827	0.861	0.891

Table 2: The BD-PSNR and BD-Bitrate comparison of the proposed method *versus* H.264/AVC

Sequences	*Foreman*	*Hall*	*Highway*	*Akiyo*	*News*	*Waterfall*
BD-PSNR (dB)	1.302	0.722	1.112	2.083	1.302	1.135
BD-Bitrate (%)	-24.003	-0.505	-30.708	-29.097	-21.186	-20.124

frames as the KFs in a GOP and down-sample other frames in a ratio 2 as the NKFs. An the decoder, the KFs are used to learn the 2-D dictionary and 3-D dictionary pairs. We keep the overall bit-rate of the proposed approach consist with H.264/AVC. The visual quality, e.g. *Structural Similarity Index Metrics (SSIM)*, and the objective metrics, e.g. *PSNR*, *rate-distortion*, and *BD-Bitrate* are evaluated.

4.2 The Validated Results

Fig. 2 shows the rate-distortion performance of the proposed scheme versus H.264/AVC. In order to validate the proposed scheme superior to the state-of-the-art learning-based super-resolution method [10], we provide the corresponding coding performance. More comprehensive comparisons are shown in Table 1 for six test sequences at different bit-rates. It can be seen that the proposed video compression framework can achieve significant gain in PSNR and better SSIM values at the same bit-rate *versus* H.264/AVC, and the coding gain is more obvious in low bit-rate region.

To evaluate the coding efficiency of the proposed video coding scheme more precisely, the BD-PSNR and BD-Bitrate [21] metrics are evaluated based on the rate-distortion curve fitting. Table 2 provides the comparison results between the proposed scheme and H.264/AVC.

5 Conclusion

In this paper, we proposed a sparse spatio-temporal representation with adaptive regularized dictionaries for super-resolution based video coding scheme. The proposed SR method is used to up-sample the NKFs using the KFs as the training images to build dictionaries. Since image primitives have low dimensionality which means they have more sparse representation over dictionary learned from primitive image patches, we take 2-D sparse representation prior on primitive patches of NKFs. The non-primitive volumes along the motion trajectory are consistent in the temporal dimension, they are supposed to have more sparse representations over a learned 3-D dictionary. With the above considerations, we take 2-D sparse representation prior on primitive patches and 3-D sparse representation prior on non-primitive volumes. Experimental results validate the compression efficiency and restoration performance of the proposed scheme versus the H.264/AVC in terms of both objective and subjective comparison, 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. on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp.1103-1120, Sep. 2007.

[2] M. Shen, P. Xue, C. Wang, “Down-Sampling Based Video Coding Using Super-Resolution Technique,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, no. 6, pp. 755-765, June 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] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, ”Overview of the H.264/AVC video coding standard,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, pp. 560-576, Jul. 2003.

[5] H. Xiong, Y. Xu, Y.F. Zheng, C.W. Chen, “Priority Belief Propagation-Based Inpainting Prediction with Tensor Voting Projected Structure in Video Compression,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, pp. 1115-1129, Aug. 2011.

[6] J. Balle and M. Wien, “Extended texture prediction for H.264/AVC intra coding,” in *Proc. of IEEE International Conference on Image Processing*, San Antonio, TX, Sep. 2007, vol. 6, pp. 93-96.

[7] D. Liu, X. Sun, F. Wu, S. Li, and Y. Zhang, “Image compression with edge-based inpainting,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 10, pp. 1273-1287, Oct. 2007.

[8] Z. Xiong, X. Sun, and F. Wu, “Block-based image compression with parameter-assistant inpainting,” *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1651-1657, Jun. 2010.

[9] Y. Wexler, E. Shechtman, and M. Irani, “Space-time completion of video,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 463-476, Mar. 2007.

[10] J. Yang, J. Wright, T. Huang, and Y. Ma, “Image Super-Resolution via Sparse Representation,” *IEEE Trans. Image Processing*, vol. 19, no. 11, pp. 2861-2873, May. 2010.

[11] W. T. Freeman, and E. C. Pasztor, “Learning low-level vision,” *Proc. of IEEE Intl. Conf. on Computer Vision*, vol. 40, pp. 1182-1189, 1999.

[12] J. Sun, N. Zheng, and H. Shum, “Image Hallucination with Primal Sketch Priors,” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, Jun. 2003.

[13] B. Lee, S. Pedersen, and M. David, ”The Complex Statistics of High-Contrast Patches in Natural Images,” *IEEE Workshop. Statistics and Computational Theories of Vision*, 2001.

[14] M. Protter, and M. Elad, ”Image Sequence Denoising via Sparse and Redundant Representations,” *IEEE Trans. Image Processing*, vol. 18, pp. 27-35, Jan. 2009.

[15] M. Aharon, M. Elad, and A. Bruckstein, ”K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. Signal Processing*, vol. 54, pp. 4311-4322, Nov. 2006.

[16] B. C. Song, S. C. Jeong, and Y. Choi, ”Video Super-Resolution Algorithm Using Bi-Directional Overlapped Block Motion Compensation and On-the-Fly Dictionary Training,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, pp. 274-285, Mar. 2011.

[17] 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 and Systems for Video Technology*, vol. 17, pp. 407-416, 2007.

[18] M. T. Orchard, and G. J. Sullivan, ”Overlapped block motion compensation: An estimation-theoretic approach,” *IEEE Trans. Image Processing*, vol. 3, pp. 693-699, Sept. 1994.

[19] D. L. Donoho, ”For Most Large Underdetermined Systems of Linear Equations the Minimal l1-norm Solution is also the Sparsest Solution,” *Communications on Pure and Applied Mathematics*, vol. 59, pp. 797-829, 2006.

[20] D. L. Donoho, and Y. Tsaig, ”Fast Solution of l1-Norm Minimization Problems When the Solution May Be Sparse,” *Technical Report, Institute for Computational and Mathematical Engineering, Stanford University*, 2006.

[21] G. Bjontegaard, ”Calculation of average psnr differerences between rd-curves (vceg-m33),” *VCEG Meeting (ITU-T SG16 Q.6)*, 2001.