Super-Resolution Reconstruction with Prior Manifold on Primitive Patches for Video Compression

Jingtao Chen #1, Hongkai Xiong #2

Department of Electronic Engineering, Shanghai Jiao Tong University
Shanghai 200240, China
[1] jingtao.ch@sjtu.edu.cn
[2] xionghongkai@sjtu.edu.cn

***Abstract*—This paper proposes a generic video compression framework with low-quality video data and a learning-based approach, which is rooted in sparse representation for the ill-posed problem of video super-resolution reconstruction. It is regularized by the prior manifold only on the "primitive patches", and each primitive patch is modeled by a sparse representation concerning an over-complete dictionary of trained set. Due to low intrinsic dimensionality of primitives, the number of samples in the dictionary can be greatly reduced. Considering the similar geometry of the manifolds of the feature spaces from the low-frequency and the high-frequency primitives, we hypothesize that the low-frequency and its corresponding high-frequency primitive patches share the same sparse representation structure. In this sense, high-resolution frame primitives are divided into low-frequency and high-frequency frame primitives, and high-frequency frame primitive patches can be synthesized from both the high-frequency primitive patch dictionary and the sparse structure of the corresponding low-frequency frame primitive patches. It does not involve with explicit motion estimation and any assistant information, and decomposes the original video sequence into key frames and low-resolution frames with low entropy. The corresponding high-resolution frames would be reconstructed by combining the high-frequency and the low-frequency patches with smoothness constraints and the backprojection process. Experimental results demonstrate the objective and subjective efficiency in comparison with H.264/AVC and existing super-resolution reconstruction approaches.**

I. Introduction

Despite the state-of-the-art H.264/AVC compression engine has achieved a vital efficiency by exploiting pixel-wise redundancy, recent efforts are expected to attain bit rate reduction of 50% at the same subjective quality with the evolution into High Efficiency Video Coding (HEVC) joint project in 2010. Exploiting visual redundancy is also a noticeable way to hallucinate missing contents with good perceptual quality. Motivated by the increasingly sensing low-quality video data, another observation for video coding is to establish a certain correlation between a sparsely sampled low-resolution version and high-resolution contents. Many interpolation algorithms have ever been developed with a statistical correlation between them through learning-based way. In this sense, the super-resolution reconstruction attracts more attention [1], [2].

Freeman et al. [3] brought out an example based learning strategy where the relationship between image (observation) and scene (state) pairs is modeled by a Markov Random Field (MRF). Sun et al. [4] extended this method by using the primal sketch priors, to achieve more satisfying results. However, they share a common hypothesis that each patch in the target high-resolution image comes from only one neighbor in the training set. [5] and [6] adopted locally linear embedding (LLE) to reconstruct high-resolution patches of the target image through k nearest neighbors, namely, a smaller training database. It keeps the number of the nearest neighbors for reconstruction fixed and neglects the non-neighbor patches so as to result in blurring effects. Yang et al. [7], from the perspective of compressed sensing, tried to recover the sparse representation coefficients of each low-resolution patch regarding a dictionary composed of low-resolution patches. The first-order and second-order derivatives are chosen as the features for the image patches, however, the derivatives are incomplete representations of the patches. Moreover, it would result in lots of unnecessary computations in the homogeneous regions and defective reconstructions near the regions of complex textures.

In this paper, we propose a generic video compression scheme with low-quality video data and a learning-based approach via super-resolution reconstruction with sparse representation prior manifold on primitive patches. The video sequence is divided into the sampled key frames and low-quality frames which are obtained by applying the degradation model to the remaining frames. At the decoder, the lost high-frequency details of the downsampled frames can be inferred by the super-resolution reconstruction which is regularized by the prior only on the primitive patches extracted by the primitive filters. Each primitive patch can be modeled by a sparse representation concerning an over-complete dictionary of trained set. Due to the intrinsic low dimensionality of primitives, the number of samples in the dictionary can be greatly reduced. High-resolution primitive patches are divided into low-frequency and high-frequency primitive patches. Considering the similar geometry of the manifolds of the feature spaces from the low-frequency and the high-frequency

978-1-4577-1434-4/11/$26.00 ©2011 IEEE

primitive patches, we hypothesize that the low-frequency and its corresponding high-frequency primitive patches share the same sparse representation structure. In this sense, high-resolution frame primitives are divided into low-frequency and high-frequency frame primitives, and high-frequency frame primitive patches can be synthesized from both the high-frequency primitive patch dictionary and the sparse structure of the corresponding low-frequency frame primitive patches. It does not involve with explicit motion estimation or any assistant information. The corresponding high-resolution frames would be reconstructed by combining the high-frequency and the low-frequency patches with smoothness constraints and the backprojection process. Experimental results demonstrate the objective and subjective efficiency in comparison with H.264/AVC and existing super-resolution reconstruction approaches.

The rest of this paper is organized as follows. Sec. II gives an overview of the entire structure of the proposed video compression scheme via super-resolution reconstruction. Sec. III addresses the sparse representation prior on primitive patches. Sec. IV demonstrates the super-resolution reconstruction in detail. Experimental results are fully evaluated in Sec. V. Sec. VI concludes this paper.

II. Super-Resolution-based Video Compression Framework

As Fig. 1, the sampled key frames $\{Y_n\}$ and the downsampled version $\{X_{nl}\}$ of the remaining frames $\{X_{nh}\}$ are compressed by the hybrid prediction codec H.264/AVC. At the decoder, the decompressed key frames $\{Y'_n\}$ are adopted to train the dictionary pair P. In turn, the high-resolution frames $\{X'_{nh}\}$ are synthesized from the decompressed low-resolution frames $\{X'_{nl}\}$. Finally, the combination of $\{Y'_n\}$ and $\{X'_{nh}\}$ forms the final output. In detail, the reconstruction of $\{X'_{nh}\}$ is decomposed into sub-problems of generating high-resolution patches using their low-resolution version with the help of a learned model. We focus on synthesis of primitive patches, which will be demonstrated in Sec. III. High-resolution primitive patches are divided into low-frequency and high-frequency primitive patches s_l and s_h. High-frequency primitive patches could be synthesized with sparse representation structure α of corresponding low-frequency primitive patches concerning high-frequency primitive dictionary T_h, which can be written as:

$$s_h = T_h \alpha \tag{1}$$

While the sparse coefficient vector α is obtained by solving the optimization problem:

$$\min \ \|\alpha\|_0 \qquad s.t. \quad s_l = T_l \alpha, \tag{2}$$

T_l is low-frequency primitive dictionary. The dictionary pair T_l and T_h will be trained in the learning phase.

Fig. 2 shows the process of the learning phase. Owing to the spatio-temporal correlations among video frames, the low-frequency and the high-frequency primitive patches can be directly extracted from the key frames, and then used to train the dictionary pair P within the framework.

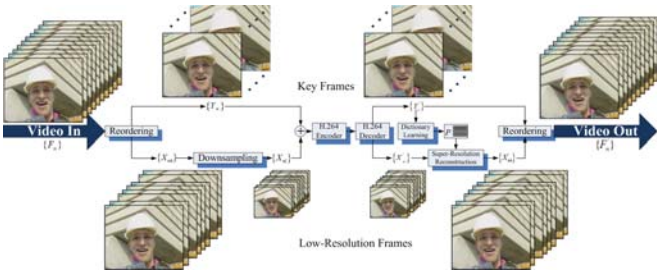


Fig. 1. The proposed video compression framework via the underlying super-resolution reconstruction

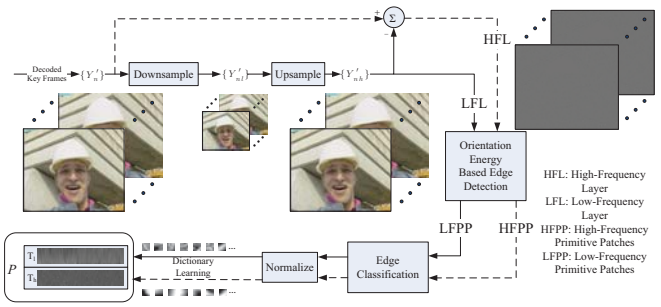


Fig. 2. The learning scheme of the proposed video compression framework

III. Sparse Representation Prior Manifold on Primitive Patches

Image primitives mainly consist of edge segments, bars, blobs, and terminations, reflecting the brightness changes of the image. Homogeneous non-primitive patches could be reconstructed perfectly by lots of interpolation algorithms. Human observers have very limited ability on identifying detail of repeated homogeneous visual patterns. Owing to their low intrinsic dimensionality [8], primitive priors are learned from the training set and used to infer lost high-frequency details of the given degraded images. This representation enables the description of patches with a very small number of training examples, and also facilitates the training of over-complete dictionaries.

A. Reasons of Using Primitive Patches

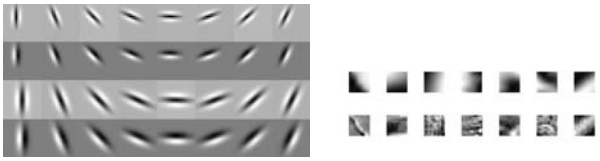
The super resolution based on only "primitive patches" in the primitive layer is mainly derived from three reasons. (1) The primitive patches could be represented by limited samples with the low dimensionality of image primitives [8], and it would make possible to learn small dictionaries for synthesizing video frames. (2) The manifolds formed by the low-frequency and high-frequency image primitives have similar local geometry, and it allows us to infer the representation structure of the high-frequency primitive patch from the low-frequency level, which would be further discussed in Sec. III-D. (3) The missing high-frequency information to be synthesized in super-resolution reconstruction is normally densely distributed over image primitives, as shown in Fig. 3(c) and Fig. 3(d).

B. Extract Image Primitives

There are different methods to extract image primitives from an image. This paper adopts a set of Gaussian derivative filters [9] to extract primitives, e.g. step-edge, ridge, corners, T-junctions, and terminations [4]. Fig. 4(a) illustrates the filter bank, and Fig. 4(b) shows the sampled primitive patch pairs.

(a) (b) (c) (d)

Fig. 3. (a) The high-resolution image; (b) the low-resolution image(interpolated to original size); (c) the filtered response for the low-resolution image (i.e., the extracted primitive image); (d) the lost high-frequency details to be synthesized;



(a) (b)

Fig. 4. (a) Filter bank to extract image primitives; (b) Upper row: the typical low-frequency primitive patches, Lower row: the corresponding high-frequency primitive patches which are extracted;

C. Feature Representation of Primitive Patches

Based on the human vision system, we choose to extract features from the luminance layer of the image. According to Freeman's assumption [3], we could use the middle frequency band to predict the highest frequency band. Furthermore, we improve this process by predicting the highest frequency band with all the other lower frequency bands. The feature representations of the high-frequency and the corresponding low-frequency primitive patches are denoted as f_h and f_l.

D. Similarity Between the Manifolds Formed by the Low-Frequency and High-Frequency Primitive Patches

As shown in Fig. 5, each patch can be considered as samples from its corresponding manifold. x_i, x_j, x_k, and y_i, y_j, y_k are samples from manifolds H and L formed by the high-frequency and low-frequency primitive patches, respectively. To demonstrate the relationship of the manifolds, we define the reconstruction error $e_l = \|y_i^{'} - y_i\|/\|y_i\|$, $e_h = \|x_i^{'} - x_i\|/\|x_i\|$, where $x_i^{'}$, $y_i^{'}$ are the reconstructed version of x_i, y_i. We aim to reconstruct x_i, y_i with a global linear combination in synthesis phase for the samples from manifolds H and L, rather than local linear or k nearest neighbors combination. We make experiments on two datasets which are composed of low-frequency and high-frequency primitive patches. Each dataset contains about 80000 patches. From Fig. 6, it can be observed that the reconstruction errors e_l of 90% low-frequency primitive patches are less than 2%, and the reconstruction errors e_h of 90% high-frequency primitive patches are less than 15%. Obviously the high-frequency primitive patches have a rather low dynamic range

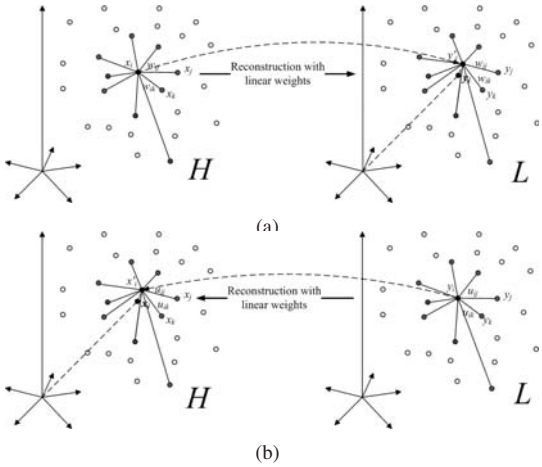
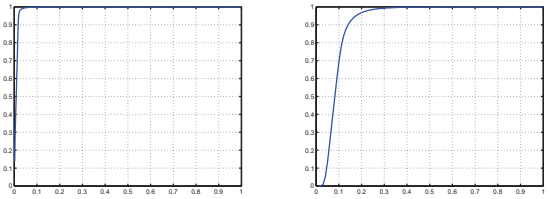


Fig. 5. The similarity between the manifolds formed by the low-frequency and high-frequency image primitives.



(a) (b)

Fig. 6. The X-axis is reconstruction error and the Y-axis is the percentage of the test data whose reconstruction errors are smaller than a given (a) e_l; (b) e_h.

in comparison with low-frequency primitive patches. The reconstructed high-resolution primitive patches are close enough to their corresponding real patches. Clearly, we can recognize that there is a $similiar\text{-}bijection$ relationship between the manifolds formed by the low-frequency and high-frequency primitive patches. Therefore, the low-frequency and high-frequency primitive patches can be modeled to keep nearly the same sparse representations with respect to their corresponding dictionaries.

IV. The Detailed Process of the Video Super-Resolution Reconstruction

The high-resolution key frames and low-resolution version of remaining frames are coded. At the decoder side, decompressed low-resolution frames will be up-scaled by the dictionary pair with online training from decompressed key frames.

A. The Learning Phase

Fig. 6 shows the reconstruction errors on two datasets composed by primitive patches extracted from generic images. Synthesis with dictionary pair trained from correlated frames will result in less errors and more reliable reconstructions. The decompressed high-resolution key frames $\{Y_n'\}$ are decomposed into low-frequency layer G_l and high-frequency layer G_h. In order to keep the relationship between the low-frequency and the high-frequency primitive patches, we need

extract primitive patches from both of G_l and G_h and construct the two training sets F_l and F_h for the synthesis phase.

Although F_l and F_h can be directly used in the synthesis phase, we prefer to learn a more compact dictionary pair. Sparse coding discovers high-level features in the input data and represents the input feature vectors approximately as a weighted linear combination of a small number of basis vectors. The task to learn a pair of compact dictionaries T_l and T_h from the low-frequency and their corresponding high-frequency primitive patches can be formulated as:

$$T = \arg\min_{T,A} \|F - TA\|_F^2 + \lambda \sum_j \|A_{\cdot,j}\|_1. \tag{3}$$

where

$$F = \begin{bmatrix} F_l \\ F_h \end{bmatrix}, \qquad T = \begin{bmatrix} T_l \\ T_h \end{bmatrix} \tag{4}$$

A is the coefficient matrix(each column is a coefficient vector $\alpha^{(i)}$), $A_{\cdot,j}$ is the jth column of A, T is the dictionary we expect to learn. The optimization is convex in T when A is fixed and is convex in A when T is fixed. For example, the objective could be iteratively optimized by alternatingly optimizing T and A when the other is fixed. Lee et al. [10] presented efficient sparse coding algorithms that are based on iteratively solving these two convex optimization problems.

B. The Synthesis Phase

When reconstructing high-resolution frames $\{X'_{nh}\}$ from decompressed low-resolution frames $\{X'_{nl}\}$, the sparse vector α can be exactly recovered by solving the following l_0-$norm$ optimization problem:

$$\min \ \|\alpha\|_0 \qquad s.t. \quad f_{_l} = T_l\alpha, \tag{5}$$

Eq. (5) is NP-hard such that we can not solve it directly. Fortunately, the progress in sparse coding [11] pointed out that a l_1-$norm$ minimization problem can be solved if α is sufficiently sparse:

$$\min \ \|\alpha\|_1 \qquad s.t. \quad \|T_l\alpha - f_{_l}\|_2 \leqslant \varepsilon. \tag{6}$$

where $e = T_l\alpha - f_{_l}$ is a small stochastic error term bounded by $\|e\|_2 \leqslant \varepsilon$. Using Lagrange multipliers, we can reformulate Eq. (6) as:

$$\min \ \lambda\|\alpha\|_1 + \|T_l\alpha - f_{_l}\|_2. \tag{7}$$

This is a non-linear convex optimization problem and can be solved efficiently by various methods. Once obtaining α by solving Eq. (5), the high resolution patch $s_{_h}$ can be acquired by the linear combination of columns in T_h using α as the coefficient: $f_{_h} = T_h\alpha$.

When all the high-frequency primitive patches are generated, by enforcing the local compatibility and smoothness constraints within overlapped patches, we can get the high-frequency frames $\{X'_h\}$. By combining the high-frequency details $\{X'_h\}$ and the upsampled version of $\{X'_{nl}\}$, we can get high-resolution frames $\{X'_H\}$ which does not necessarily meet the reconstruction constraint. Therefore, we use an iterative gradient-based method called backprojection process [12] to enforce the reconstruction constraint on each frame of $\{X'_H\}$:

$$X'^{\,t+1}_H = X'^{\,t}_H + (((X'^{\,t}_H * h) \downarrow s - X'_{nl}) \uparrow s) * p, \tag{8}$$

where $X'^{\,t}_H$ denotes the high-resolution frame at iteration t, X'_{nl} is the original input low-resolution frame, h is a convolution filter for down-sampling, and p is a backprojection filter for error correction. X'_H acts as the starting point in the iteration process. The complete video super-resolution reconstruction process can be summarized as Algorithm 1.

Algorithm 1 Video Super-Resolution Reconstruction

Notations:
$\{Y'_n\}$: Decoded high-resolution key frames
$\{X'_{nl}\}$: Decoded low-resolution frames
$\{F'_n\}$: Final high-resolution output frames
$\{X'_l\}$: Upsampled version of $\{X'_{nl}\}$
$\{X'_h\}$: High-frequency primitive frames
$\{X'_H\}$: Combination of $\{X'_l\}$ and $\{X'_h\}$
$\{X'_{nh}\}$: Reconstructed high-resolution frames

Input: Decoded high-resolution key frames $\{Y'_n\}$ and low-resolution frames $\{X'_{nl}\}$.

Output: Final high-resolution output frames $\{F'_n\}$.

Begin:

1. Train dictionars T_l and T_h according to Eq. (4) from feature vector sets F_l and F_h extracted from $\{Y'_n\}$.
2. Interpolate each frame of $\{X'_{nl}\}$, we obtain $\{X'_l\}$, which could be represented by small overlapped low-frequency primitive patches. For each low-frequency primitive patch
   (a) Solve Eq. (5) to obtain the sparse linear combination coefficient α;
   (b) Linearly combine the columns in T_h with the combination coefficient α, we obtain a high-frequency primitive patch.
3. Construct the high-frequency frames $\{X'_h\}$ by enforcing the local compatibility and smoothness constraints between overlapped high-frequency primitive patch obtained in step 1.
4. Combine $\{X'_l\}$ and $\{X'_h\}$ we get high-resolution frames $\{X'_H\}$, on which we use the backprojection algorithm to enforce reconstruction constraint, resulting in the final high-resolution frames $\{X'_{nh}\}$.
5. Reorder $\{Y'_n\}$ and $\{X'_{nh}\}$, we obtain the final high-resolution output frames $\{F'_n\}$.

End

V. Experimental Results

Without loss of generality, we keep one key frame from every 10 successive frames and downsample other 9 frames given an input video sequence $(F_1, F_2, \cdots, F_n, \cdots)$. At the decoder side, each time 3 successive key frames are used to learn the dictionary pair. The feature representations of low-frequency and high-frequency primitive patches are directly used to synthesize other 3×9 high-resolution frames. The experiments are

performed on test sequences of CIF (352×288) resolution. We keep the overall bit-rate of proposed approach consistent with the bit-rate of the standard-codec H.264/AVC. The proposed scheme is also compared with the bicubic interpolation and the LLE super-resolution algorithm [6]. Fig. 7 shows that the standard-codec H.264/AVC and the bicubic interpolation introduce more blurs and jaggies, while the LLE reconstruction and the proposed scheme achieve much better visual qualities. Fig. 8 and Fig. 9 show the YPSNR and the SSIM values of "Foreman", "Akiyo", and "Highway" sequences. Specifically, the proposed scheme outperforms other methods when processing sequences with regular motions. In comparison, the standard-codec H.264/AVC and the bicubic interpolation have introduced blurs in the regions of the woman's face and cloth; the proposed scheme obviously improves the reconstructed quality.

VI. Conclusion

In this paper, we propose a generic video compression scheme via super-resolution reconstruction with sparse representation prior manifold on primitive patches. It does not estimate the motion, and decomposes the original video sequence into key frames and low-quality version of remaining frames. Owing to low intrinsic dimensionality of primitive patches and the relationship between low-frequency and high-frequency primitive layers, we could train dictionary pair from decompressed key frames and synthesize reliable high-resolution frames from decompressed low-resolution frames. The propose approach can favor the video compression in a more optimal sense.

References

[1] M. Protter and M. Elad, "Super ResolutionWith Probabilistic Motion Estimation," *IEEE Trans. on Image Processing*, vol. 18, no. 8, pp. 1899-1904, Aug. 2009.

[2] 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 Processing*, vol. 20, no. 7, July 2011

[3] W. T. Freeman, E. T. Pasztor, and O. T. Carmichael, "Learning low-level vision," *IEEE International Journal on Computer Vision*, vol. 40, no. 1, pp. 25-47, 2000.

[4] J. Sun, N. Zheng, H. Tao, H. Y. Shum, "Image hallucination with primal sketch priors," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 729-736, Coronto, Canada, June 2003.

[5] H. Chang, D. Y. Yeung, Y. Xiong, C. W. Bay, and H. Kong, "Super-resolution through neighbor embedding," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 275-282, Washington, USA, July 2004.

[6] W. Fan and D. Y. Yeung, "Image Hallucination Using Neighbor Embedding over Visual Primitive Manifolds," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-7, Minneapolis, USA, June 2007.

[7] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image Super-Resolution via Sparse Representation," *IEEE Trans. on Image Processing*, vol. 19, no. 11, pp. 2861-2873, Nov. 2010.

[8] A. B. Lee, K. S. Pedersen, and D. Mumford, "The Nonlinear Statistics of High-Contrast Patches in Natural Images," *IEEE International Journal on Computer Vision*, 2003.

[9] P. Perona and J. Malik, "Detecting and localizing edges composed of steps, peaks and roofs," in *Proc. of International Conference on Computer Vision*, pp. 52-57, Dec. 1990.

[10] H. Lee and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. of Advances in neural information processing systems*, 2007.

(a) Frame 24 (b) Frame 46

(c) PSNR = 34.3dB SSIM = 0.93 (d) PSNR = 34.2dB SSIM = 0.93

(e) PSNR = 33.1dB SSIM = 0.93 (f) PSNR = 33.3dB SSIM = 0.93

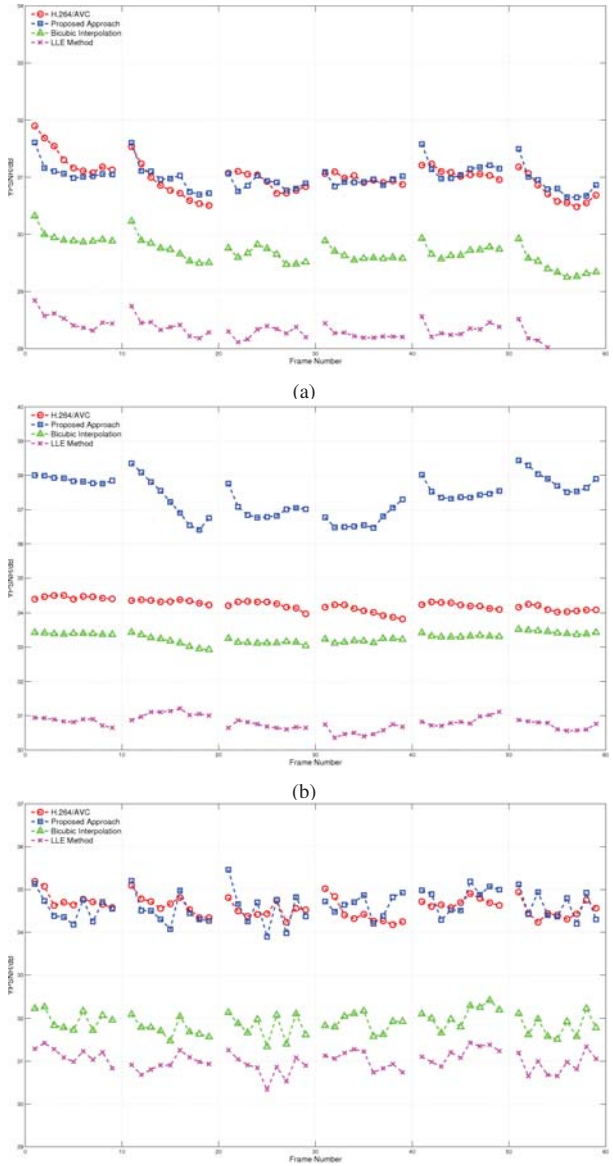
(g) PSNR = 30.8dB SSIM = 0.92 (h) PSNR = 30.8dB SSIM = 0.92

(i) PSNR = 36.8dB, SSIM = 0.95 (j) PSNR = 37.4dB, SSIM = 0.95

Fig. 7. (a) and (b) are the original frames from " Akiyo" sequence; (c) and (d) are standard-codec H.264/AVC frames; (e) and (f) are bicubic interpolated frames; (g) and (h) are LLE frames; (i) and (j) are the proposed video frames. Each example is compared under the same bit-rate = 240kbps.

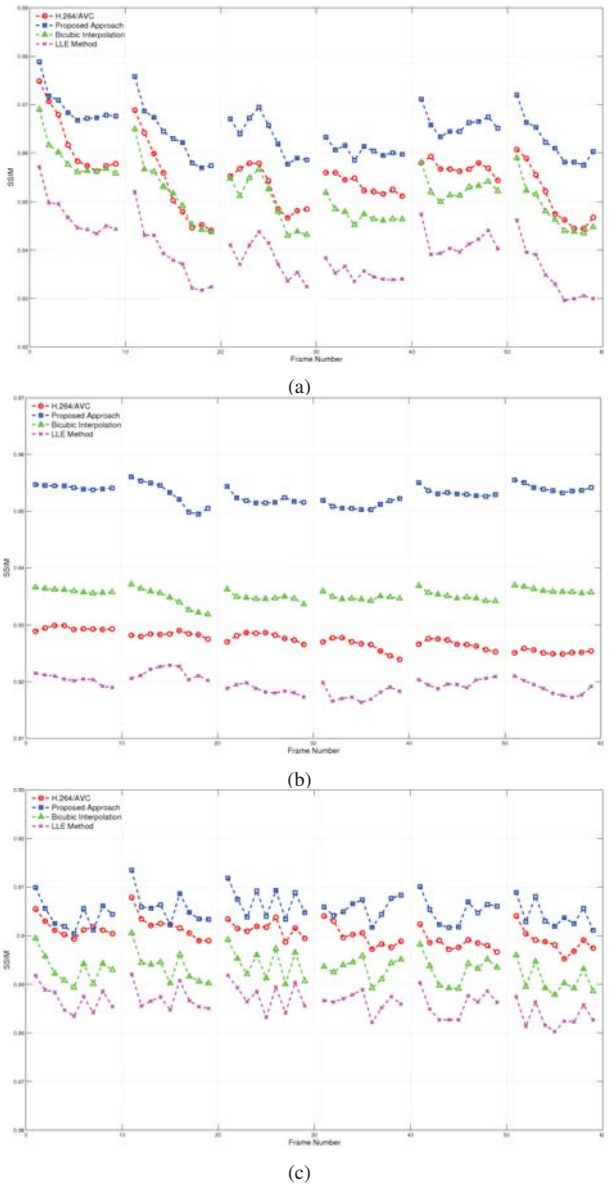
[11] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via l minimization," in *Proc. of the National Academy of Sciences of the United States of America*, vol. 100, pp. 2197-2202, 2003.

[12] M. Irani, "Motion Analysis for Image Enhancement: Resolution, Occlusion, and Transparency," *Journal of Visual Communication and Image Representation*, vol. 4, no. 4, pp. 324-335, Dec. 1993.



(a) (b) (c)

Fig. 8. The experimental results of "Foreman", "Akiyo", and "Highway" sequences. Note that we skip the key frames. YPSNR values of (a) "Foreman" with bit-rate = 310kbps; (b) "Akiyo" with bit-rate = 240kbps; (c) "Highway" with bit-rate = 220kbps.



(a) (b) (c)

Fig. 9. The experimental results of "Foreman", "Akiyo", and "Highway" sequences. Note that we skip the key frames. SSIM index of (a) "Foreman" with bit-rate = 310kbps; (b) "Akiyo" with bit-rate = 240kbps; (c) "Highway" with bit-rate = 220kbps..