Copyright ⓒ 2014 KSII

Distributed Video Compressive Sensing Reconstruction by Adaptive PCA Sparse Basis and Nonlocal Similarity

WU Minghu[1,2], **ZHU Xiuchang**[3]
[1] School of Electrical and Electronic Engineering, Hubei University of Technology, Wuhan, China
[2]Hubei Collaborative Innovation Center for High-efficiency Utilization of Solar Energy, Hubei University of technology, Wuhan, China
[e-mail: wuxx1005@mail.hbut.edu.cn]
[3]Jiangsu Province Key Lab on Image Processing & Image Communication, Nanjing University of Posts and Telecommunications, Nanjing, China
[e-mail: zhuxc@njupt.edu.cn]

Received January 21, 2014; revised May 2, 2014; accepted June 18, 2014; published August 29, 2014

Abstract

To improve the rate-distortion performance of distributed video compressive sensing (DVCS), the adaptive sparse basis and nonlocal similarity of video are proposed to jointly reconstruct the video signal in this paper. Due to the lack of motion information between frames and the appearance of some noises in the reference frames, the sparse dictionary, which is constructed using the examples directly extracted from the reference frames, has already not better obtained the sparse representation of the interpolated block. This paper proposes a method to construct the sparse dictionary. Firstly, the example-based data matrix is constructed by using the motion information between frames, and then the principle components analysis (PCA) is used to compute some significant principle components of data matrix. Finally, the sparse dictionary is constructed by these significant principle components. The merit of the proposed sparse dictionary is that it can not only adaptively change in terms of the spatial-temporal characteristics, but also has ability to suppress noises. Besides, considering that the sparse priors cannot preserve the edges and textures of video frames well, the nonlocal similarity regularization term has also been introduced into reconstruction model. Experimental results show that the proposed algorithm can improve the objective and subjective quality of video frame, and achieve the better rate-distortion performance of DVCS system at the cost of a certain computational complexity.

Keywords: Compressive Sensing, distributed video compressive sensing, principle components analysis, adaptive sparse basis, nonlocal similarity

This research was supported by National Natural Science Foundation of China (61201160, 61271240 and 61172118), Technology Research Program of Hubei Provincial Department of Education (D20141406), the Natural Science Foundation of Hubei Province (2013CFB022) and the Natural Science Foundation of Jiangsu Province (BK20131377)

http://dx.doi.org/10.3837/tiis.2014.08.016

1. Introduction

The basic idea of Compressive Sensing (CS) is to sample the signal by the way of direct dimensionality reduction while compressing the signal and then recover the original signal by exploiting the sparse prior of signal. Due to its ability to sample signal at the sub-Nyquist rate, the theory of CS has been widely applied into the various fields of image and video processing [1],[2]. The measurement approach of CS is realized by the linear inner-product and thus it has a low computational complexity, however, it requires the high computational costs to non-linearly reconstruct signal. This feature of light coding and heavy decoding makes CS theory easily be combined into the Distributed Video Coding (DVC) [3], which produces a new video compression technology -- Distributed Video Compressive Sensing (DVCS) [4]-[6].

In the DVCS system, the primary problem is the requirement of the huge memory burden in CS measurement. Currently there are two schemes to effectively resolve this problem. The first method is to use the Structurally Radom Matrices (SRMs) [7],[8] to achieve the measurement data. The SRMs use the fast orthogonal transformation to realize CS measurement, and thus avoid to construct the measurement matrix requiring lots of memory. The another method is to perform CS measurement by the Block Compressed Sensing (BCS) [9]. This approach can not only realize a low-memory CS measurement but also measure and transmit the video block one by one, and therefore it is very appropriate for the real-time applications and widely used in various DVCS systems [10]-[11]. The DVCS firstly divides the video stream into the key frames and non-key frames. The key frame can realize codec by either the traditional video coding technology (e.g., H.264) or measuring video frame at a higher measurement rate and using still-image CS reconstruction algorithm [12]-[14] to recover the original video frame. Due to the low measurement rate of non-key frames, its reconstruction requires to combine intra and inter frame correlation. Ref. [5] uses the previous and following frames to interpolate the Side Information (SI) of non-key frame by motion compensation and then regards the SI as the initial solution of GPSR algorithm [15] to construct the final interpolated frame. Ref. [6] uses the temporal-neighboring blocks to construct the sparse dictionary of each interpolated block in the non-key frame and then performs the appropriate minimum l_1-norm algorithm to predict the SI, and finally reconstructs the residual frame between the SI and original frame by using the still-image CS reconstruction algorithm. Ref. [16] firstly uses CS reconstruction algorithm to independently perform intra-frame recovery and then utilizes the previous and following frames to predict the SI by motion estimation and motion compensation, and finally recovers the residual. Ref. [17] uses the Multiple Hypotheses (MH) concept in the traditional video coding to construct the candidate set of each interpolated block, and then replaces the sparse regularization item in the way of l_1-norm with the Tikhonov regularization item in the way of l_2-norm to predict the SI of non-key frame, and this method can effectively improve the predictive precision and reconstruction speed.

Although the above methods can obtain the better reconstructed quality of non-key frame, there are still the two defects: (a) the sparse dictionary cannot adaptively change in terms of the reconstructed quality of reference frame and remove the noise; (b) they only use the sparse prior and overlook the other prior knowledge of video frame. Aim to the first defect, an adaptive construction of sparse dictionary is proposed in this paper. Firstly, it uses the motion information between frames to find the best-matching block in reference frames of each interpolated block and extracts its temporal-neighboring blocks to produce the data matrix. Due to the noises existing in the reference frames, the Principle Components Analysis (PCA) is then used to compute the significant principle components, and finally these significant principle components are used to construct the sparse dictionary. The PCA-based sparse dictionary has a big correlation with the interpolated block, and therefore it can exploit the sparse property of non-key frame to improve the accuracy of reconstruction. For the second defect, this paper uses the Non-local Similarity (NL) of video frame to model the regularization item and combines the sparse prior knowledge to generate the joint CS reconstruction model, and finally an appropriate reconstruction algorithm is designed to solve the joint model. Since the NL is help for preserving edge details and suppressing noises, the proposed joint model can improve the performance of CS reconstruction algorithm. Experimental results show that the proposed joint reconstruction algorithm can effectively improve the rate-distortion performance of DVCS system and achieve the better objective and subjective quality of reconstructed non-key frame.

2. Framework of Proposed DVCS System

The framework of proposed DVCS system is shown in **Fig. 1**. The original video stream is firstly divided into key frames and non-key frames, and they are measured by the BCS proposed by Ref. [9]. An $I_c \times I_r$ video frame $\boldsymbol{x}_t$ with $N = I_c \times I_r$ pixels in total is divided into L small blocks with size of $B \times B$. Let $\boldsymbol{x}_{t,n}$ represents the vectorized signal of the n-th block though raster scanning, and each block $\boldsymbol{x}_{t,n}$ is measured by using the same Gaussian random measurement matrix $\boldsymbol{\Phi}_{\mathrm{B}}$, and the corresponding output CS vector $\boldsymbol{y}_{t,n}$ with the length M_{B} can be obtained. The above process can be described as

$$\boldsymbol{y}_{t,n} = \boldsymbol{\Phi}_{\mathrm{B}} \cdot \boldsymbol{x}_{t,n},\ n = 1, 2, \cdots, L. \tag{1}$$

The measurement rate is defined as $S = M_{\mathrm{B}}/\mathrm{B}^2$. When the non-key frame is reconstructed jointly, the reconstruction quality of previous and following frame can affect seriously the performance of joint reconstruction model. Therefore, the measurement rate S_{K} of key frame should be higher than the measurement rate S_{NK} of non-key frame. The high measurement rate of key frame guarantees also the better reconstruction quality by only using the still-image CS reconstruction algorithm to independently reconstruct key frame, and therefore the key frame is also called as I frame.

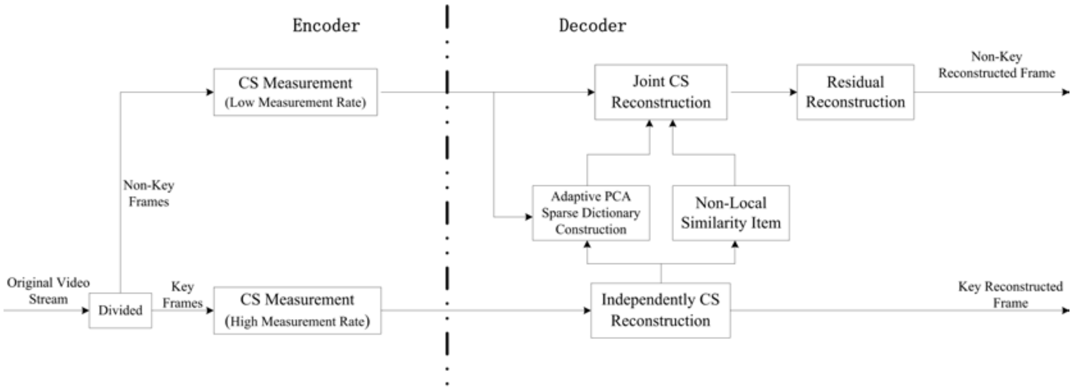


Fig. 1. Framework of proposed DVCS system

Since the non-key frame is measured at a low measurement rate, the sufficient employment of inter-frames and spatial correlation can just guarantee the high quality of reconstructed non-key frame. If the previous key frame is only used, then the current non-key frame is called as P frame. If the previous and following frame are both used, then the current non-key frame is called as B frame. The adaptive PCA sparse dictionary and non-local similarity can be generated by using the neighboring reference frames and the current non-key frame, and then they are used to construct joint reconstruction model, and then the corresponding algorithm is performed to solve the SI $\boldsymbol{x}_{\mathrm{SI}}$ of current non-key frame. To further improve the reconstruction quality of non-key frame, the residual between SI and original frame is reconstructed , and the steps are described as follows,

Step 1) Initialization: $\boldsymbol{x}_t^{(0)} = \boldsymbol{x}_{\mathrm{SI}}$, the initial iteration k is set to 0, the maximum number iterations *maxiter* is set to 5.

Step 2) The CS measurement of residual between SI and original frame can be calculated as

$$\boldsymbol{y}_{r,n} = \boldsymbol{y}_{t,n} - \boldsymbol{\Phi}_{\mathrm{B}} \cdot \boldsymbol{x}_{t,n}^{(k)} = \boldsymbol{\Phi}_{\mathrm{B}} (\boldsymbol{x}_{t,n} - \boldsymbol{x}_{t,n}^{(k)}) = \boldsymbol{\Phi}_{\mathrm{B}} \boldsymbol{r}_{t,n}^{(k)}, \; n = 1, 2, \cdots, L\,. \tag{2}$$

Step 3) The residual frame $\boldsymbol{r}_{t,n}^{(k)}$ is computed by using BCS-SPL-DCT algorithm proposed by Ref. [13], and the k+1 iteration solution $\boldsymbol{x}_t^{(k+1)}$ can be get as follows,

$$\boldsymbol{x}_t^{(k+1)} = \boldsymbol{x}_t^{(k)} + \boldsymbol{r}_t^{(k)}\ . \tag{3}$$

Step 4) $k = k+1$, if $k \le maxiter$ and $\|\boldsymbol{r}_{t,n}^{(k)}\|_2 \ge 10^{-4} \cdot N$, then go back to Step 2) and continue to the process of iteration, otherwise stopping the iteration.

3. Proposed Joint CS Reconstruction

3.1 Construction of Adaptive PCA Sparse Dictionary

Since the statistic characteristic of video frame is non-stationary, there is not the best fixed sparse dictionary (e.g., DCT dictionary, wavelet dictionary, etc.). To exploit the sparse property of video frame, the adaptive sparse dictionary correlated with the content of video frame should be constructed. Ref. [6] and Ref. [7] use directly the temporal-neighboring blocks to construct sparse dictionary, however, although this dictionary can adaptively be adjusted with the variational statistic characteristic of video frame, it cannot always keep the high correlation with the interpolated block. The main reasons of this problem have the following two points: (a) the motion information between frames; (b) the reconstructed key frames contain some noises. To overcome the above defects, we firstly use the CS measurement of the interpolated block to do motion estimation and find its best matching block in the reference frame, and then the spatial neighboring blocks of the best matching block in the reference frame are extracted to generate the data matrix. However, the data matrix contains a certain noises, and therefore the PCA is used to compute the principle components of data matrix, and then we select the significant principle components to construct the final sparse dictionary to suppress the noises. Take the situation of P frame as a example, the concrete construction steps of proposed sparse dictionary are described as follows:

Step 1) Suppose the CS measurement of the interpolated block $\boldsymbol{x}_{t,n}$ is $\boldsymbol{y}_{t,n}$. Due to the Restricted Isometry Property (RIP) [18] of Gaussian measurement matrix, the matching error between $\boldsymbol{x}_{t,n}$ and the candidate matching block $\boldsymbol{x}_{c,j}$ retains approximately unchanged, i.e.,

$$\left\| \boldsymbol{x}_{t,n} - \boldsymbol{x}_{c,j} \right\|_2 \approx \left\| \boldsymbol{y}_{t,n} - \boldsymbol{\Phi}_{\mathrm{B}} \boldsymbol{x}_{c,j} \right\|_2 . \tag{4}$$

Therefore, the block-matching based motion estimation can be performed in the measurement domain as follows:

$$\boldsymbol{x}_{\mathrm{b},n} = \arg \min_{\boldsymbol{x}_{c,j} \in \boldsymbol{S}_1} \left\| \boldsymbol{y}_{t,n} - \boldsymbol{\Phi}_{\mathrm{B}} \boldsymbol{x}_{c,j} \right\|_2 , \tag{5}$$

where $\boldsymbol{S}_1$ is the search window with size of $2S_1 \times 2S_1$. As shown in **Fig. 2**, we extract the blocks $\boldsymbol{x}_{\mathrm{p},k}$ with size of $B \times B$ pixel-by-pixel in the search window with the centre $\boldsymbol{x}_{\mathrm{b},n}$, and then each extracted block is converted into the vector by raster scanning, and all extracted blocks are combinend into the data matrix $\boldsymbol{X}_{\mathrm{p}} = [\boldsymbol{x}_{\mathrm{p},1}, \boldsymbol{x}_{\mathrm{p},2}, ..., \boldsymbol{x}_{\mathrm{p},K}]$ in which $K = 2S_2 \times 2S_2$.

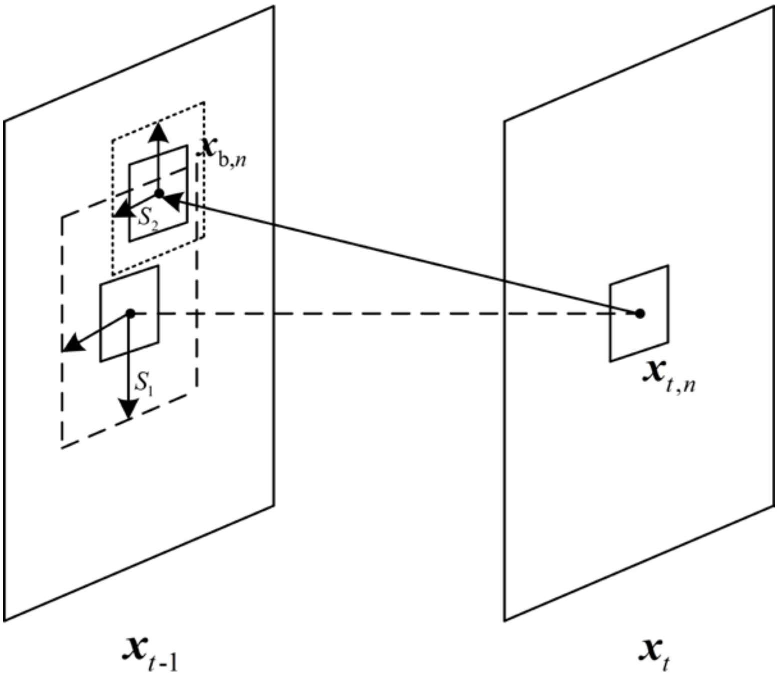

$\boldsymbol{x}_{\mathrm{b},n}$
S_2
S_1
$\boldsymbol{x}_{t,n}$
$\boldsymbol{x}_{t\text{-}1}$
$\boldsymbol{x}_t$

Fig. 2. Illustration of data matrix $\boldsymbol{X}_{\mathrm{p}}$ construction

Step 2) Each block $\boldsymbol{x}_{\mathrm{p},k}$ in data matrix $\boldsymbol{X}_{\mathrm{p}}$ contains noises, and thus it is not the best scheme that $\boldsymbol{X}_{\mathrm{p}}$ is directly regarded as the sparse dictionary. The PCA can compute the orthogonal transformation matrix $\boldsymbol{P}$ which can remove the redundant information between pixels in $\boldsymbol{x}_{\mathrm{p},k}$. If $\boldsymbol{P}$ is used to transform image blocks, and the useful information and noises of $\boldsymbol{X}_{\mathrm{p}}$ can be effectively divided. Firstly, the covariance matrix $\boldsymbol{\Omega}_{\mathrm{p}}$ with size of $d \times d$ ($d = B^2$) corresponding to $\boldsymbol{X}_{\mathrm{p}}$ can be calculated as follows,

$$\boldsymbol{\Omega}_{\mathrm{p}} = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{x}_{\mathrm{p},k} \boldsymbol{x}_{\mathrm{p},k}^{\mathrm{T}} - \overline{\boldsymbol{x}}_{\mathrm{p},k} \overline{\boldsymbol{x}}_{\mathrm{p},k}^{\mathrm{T}} , \tag{6}$$

$$\bar{\boldsymbol{x}}_{\mathrm{p},k} = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{x}_{\mathrm{p},k} \ , \tag{7}$$

and then we can compute d eigenvalues $\eta_1 \geq \ \eta_2 \geq \ \ldots \geq \ \eta_d$ of the covariance matrix $\boldsymbol{\Omega}_{\mathrm{p}}$ and their corresponding normalized eigenvectors (principle components) $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_d$, and finally we can construct the orthogonal transformation matrix $\boldsymbol{P} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_d]$.

Step 3) To effectively divide noises and useful information in the data matrix $\boldsymbol{X}_{\mathrm{p}}$, we should be to find the sparse dictionary $\boldsymbol{D}_n$ which can sparsely represent all blocks in $\boldsymbol{X}_{\mathrm{p}}$ as far as possible, i.e., the $\boldsymbol{D}_n$ should satisfy the following formula,

$$(\hat{\boldsymbol{D}}_n, \hat{\boldsymbol{\Lambda}}_n) = \arg\min_{\boldsymbol{D}_n, \boldsymbol{\Lambda}_n}\{\left\|\boldsymbol{X}_{\mathrm{p}} - \boldsymbol{D}_n\boldsymbol{\Lambda}_n\right\|_{\mathrm{F}}^2 + \mu\left\|\boldsymbol{\Lambda}_n\right\|_1\}, \tag{8}$$

where $\boldsymbol{\Lambda}_n$ is coefficient matrix of $\boldsymbol{X}_{\mathrm{p}}$, $||\cdot||_{\mathrm{F}}$ is Frobenius norm. The r significant principle components in $\boldsymbol{P}$ are used to generate the dictionary $\boldsymbol{D}_{n,r} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_r]$, and the coefficient matrix $\boldsymbol{\Lambda}_n$ can be simply calculated by $\boldsymbol{\Lambda}_{n,r} = \boldsymbol{D}_{n,r}^{\mathrm{T}} \cdot \boldsymbol{X}_{\mathrm{p}}$. The reconstruction error $||\boldsymbol{X}_{\mathrm{p}} - \boldsymbol{D}_{n,r}\boldsymbol{\Lambda}_{n,r}||_{\mathrm{F}}^2$ in Eq. (8) will decrease as r increases, and the item $||\boldsymbol{\Lambda}_{n,r}||_1$ is otherwise increasing. Therefore, the best value r^* of r can be selected by the following formula,

$$r^* = \arg\min_{r}\{\left\|\boldsymbol{X}_{\mathrm{p}} - \boldsymbol{D}_{n,r}\boldsymbol{\Lambda}_{n,r}\right\|_F^2 + \mu\left\|\boldsymbol{\Lambda}_{n,r}\right\|_1\}. \tag{9}$$

Finally, the sparse dictionary $\boldsymbol{D}_n = [\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_{r^*}]$ of the interpolated block $\boldsymbol{x}_{t,n}$ can be achieved.

Step 4) The CS reconstruction model can be constructed by using the $\boldsymbol{D}_n$ from PCA training as follows,

$$\hat{\boldsymbol{\alpha}}_{t,n} = \arg\min_{\boldsymbol{\alpha}_{t,n}}\{\left\|\boldsymbol{y}_{t,n} - \boldsymbol{D}_n\boldsymbol{\alpha}_{t,n}\right\|_2^2 + \lambda_1\left\|\boldsymbol{\alpha}_{t,n}\right\|_1\} \tag{10}$$

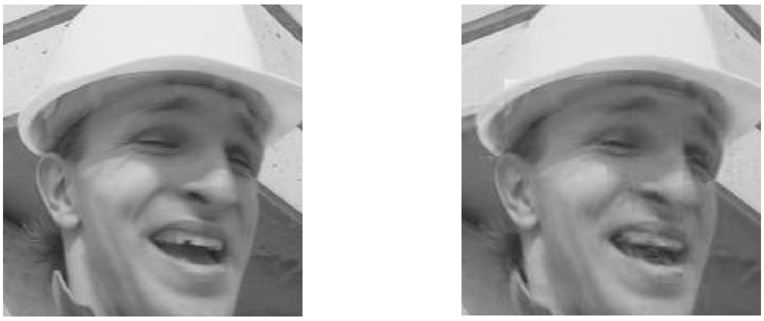
The sparse representation $\boldsymbol{\alpha}_{t,n}$ of $\boldsymbol{x}_{t,n}$ is obtained by using GPSR algorithm to solve Eq. (10), and finally the interpolated block is reconstructed by

$$\hat{\boldsymbol{x}}_{\mathrm{SI},n} = \boldsymbol{D}_n \cdot \hat{\boldsymbol{\alpha}}_{t,n}\,. \tag{11}$$

3.2 Non-local Similarity Regularization Item

Although the adaptive PCA sparse dictionary can exploit the sparse property of video frame, it cannot preserve edge and texture features well since the edge and texture features have a low sparse degree. **Fig. 3** shows that the reconstructed *Foreman* 13-th frame (P situation) when the measurement rate S_{NK} is 0.1 and block size B is 16. It can be observed that edge and texture regions appear the obvious blurring and blocking artifacts. Therefore, to retain the clear edge and texture details, in addition to using the sparse priori knowledge, the other priori knowledge requires also to be added.

For image and video, the pixel is not isolated but jointly describe the image features with its neighboring pixels. The window with center pixel (it is also called as patch) can usually present details of a pixel. The center of patch is corresponding to a pixel of image, then an image can be represented by the over-complete set composed by all patches. In the edge and texture regions usually exist lots of periodical repetitive patterns and they have a high self-similarity, and therefore the patches locating at the different positions have a strong similarity. This property of image and video is called as non-local similarity [19]-[21]. The non-local similarity of video presents that patches have not only spatial correlation but also temporal correlation. As shown in **Fig. 4**, the patch labeled by red color and the patch labeled by blue color can find the similar patches in spatial and temporal neighboring regions. The non-local similarity is very helpful to improve the quality of reconstructed frame, especially for preserving edge and texture structure features, and therefore this property can become a priori knowledge to mix into Eq. (10) and effectively remove the blurring and blocking artifacts in edge and texture regions.



(a) Original frame (b) Reconstructed frame, PSNR = 34.13 dB

Fig. 3 Comparison between original frame and the reconstructed frame from adaptive PCA sparse dictionary for *Foreman* 13-th frame.

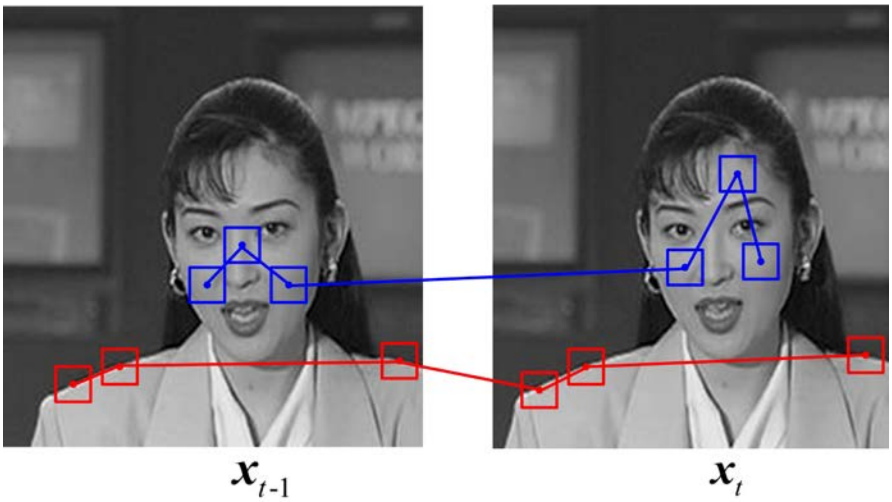


Fig. 4. Non-local similarity of video

Take the P situation as an example, the following content describes the construction of non-local similarity regularization item in details. Any pixel in $\boldsymbol{x}_{t,n}$ is denoted as $x_{t,n}(i)$, i = 1,2, …,d, and $\boldsymbol{x}_{t,n}(i)$ denotes the patch whose center and radius are $x_{t,n}(i)$ and b respectively. For each patch $\boldsymbol{x}_{t,n}(i)$, we find its similar patches in the current block $\boldsymbol{x}_{t,n}$ and the best-matching block $\boldsymbol{x}_{b,n}$ in the previous frame, and each patch $\boldsymbol{x}_{t,n}{}^{m}(i)$ should satisfy $\boldsymbol{e}_i^m = \| \boldsymbol{x}_{t,n}(i) \text{ -} \boldsymbol{x}_{t,n}{}^{m}(i)\|_2 \leq t$, therefore $x_{t,n}(i)$ can be predicted by

$$x_{t,n}(i)=\sum_{m=1}^{M}\beta_i^m x_{t,n}^m(i)+n_i\,, \tag{12}$$

$$\beta_i^m=\frac{\exp(-e_i^m/c)}{\sum_{m=1}^{M}\exp(-e_i^m/c)}\,, \tag{13}$$

where n_i is the additional noise item. Suppose $\boldsymbol{\beta}_i$ is the vector containing all elements β_i^m, $x_{t,n}^m(i)$ corresponding to β_i^m can be generated as $\boldsymbol{g}_i$, and thus Eq. (12) can be equal to

$$x_{t,n}(i)=\boldsymbol{\beta}_i^{\mathrm{T}}\cdot\boldsymbol{g}_i+n_i\,. \tag{14}$$

Considering the non-local similarity of video, the predictive error $||x_{t,n}(i)\text{-}\ \boldsymbol{\beta}_i^{\mathrm{T}}\cdot\boldsymbol{g}_i||_2$ should be smaller, and thus it can be regarded as the regularization item to mix Eq.(10) as follows,

$$\hat{\boldsymbol{\alpha}}_{t,n}=\arg\min_{\boldsymbol{\alpha}_{t,n}}\{\left\|\boldsymbol{y}_{t,n}-\boldsymbol{D}_n\boldsymbol{\alpha}_{t,n}\right\|_2^2+\lambda_1\left\|\boldsymbol{\alpha}_{t,n}\right\|_1+\lambda_2\sum_{i=1}^{d}\left\|x_{t,n}(i)-\boldsymbol{\beta}_i^{\mathrm{T}}\cdot\boldsymbol{g}_i\right\|_2^2\}\,, \tag{15}$$

where λ_2 is the regularization factor used to balance the non-local similarity item. Eq. (15) can be equal to

$$\hat{\boldsymbol{\alpha}}_{t,n}=\arg\min_{\boldsymbol{\alpha}_{t,n}}\{\left\|\boldsymbol{y}_{t,n}-\boldsymbol{D}_n\boldsymbol{\alpha}_{t,n}\right\|_2^2+\lambda_1\left\|\boldsymbol{\alpha}_{t,n}\right\|_1+\lambda_2\left\|(\boldsymbol{I}-\boldsymbol{H}_{n,1})\boldsymbol{D}_n\boldsymbol{\alpha}_{t,n}-\boldsymbol{H}_{n,2}\boldsymbol{x}_{\mathrm{b},n}\right\|_2^2, \tag{16}$$

where $\boldsymbol{I}$ is the identify matrix, $\boldsymbol{H}_{n,1}$ and $\boldsymbol{H}_{n,2}$ satisfy

$$\boldsymbol{H}_{n,1}(i,m)=\begin{cases}\beta_i^m, & x_{t,n}^m(i)\in\boldsymbol{g}_i\ \&\ x_{t,n}^m(i)\in\boldsymbol{x}_{t,n}\\ 0\ , & \text{otherwise}\end{cases}\,, \tag{17}$$

$$\boldsymbol{H}_{n,2}(i,m)=\begin{cases}\beta_i^m, & x_{t,n}^m(i)\in\boldsymbol{g}_i\ \&\ x_{t,n}^m(i)\in\boldsymbol{x}_{\mathrm{b},n}\\ 0\ , & \text{otherwise}\end{cases}\,. \tag{18}$$

To solve Eq. (16), it can be further simplified as the following l_1-l_2 norm minimum model,

$$\hat{\boldsymbol{\alpha}}_{t,n}=\arg\min_{\boldsymbol{\alpha}_{t,n}}\{\left\|\tilde{\boldsymbol{y}}_{t,n}-\boldsymbol{\Phi}_n\boldsymbol{D}_n\boldsymbol{\alpha}_{t,n}\right\|_2^2+\lambda_1\left\|\boldsymbol{\alpha}_{t,n}\right\|_1\}\,, \tag{19}$$

where

$$\tilde{\boldsymbol{y}}_{t,n}=\begin{bmatrix}\boldsymbol{y}_{t,n}\\ \sqrt{\lambda_2}\boldsymbol{H}_{n,2}\boldsymbol{x}_{\mathrm{b},n}\end{bmatrix},\ \boldsymbol{\Phi}_n=\begin{bmatrix}\boldsymbol{\Phi}_{\mathrm{B}}\\ \sqrt{\lambda_2}(\boldsymbol{I}-\boldsymbol{H}_{n,1})\end{bmatrix} \tag{20}$$

Since the construction of $\boldsymbol{H}_{n,1}$ and $\boldsymbol{H}_{n,2}$ requires the interpolated block $\boldsymbol{x}_{t,n}$, however $\boldsymbol{x}_{t,n}$ is unavailable in the process of reconstruction. Therefore, $\boldsymbol{H}_{n,1}$ and $\boldsymbol{H}_{n,2}$ will be updated using the iteration solution in the process of solving Eq. (19). The steps of solving Eq. (19) are described as follows,

Step 1) Initialization:

a) the initial solution $\boldsymbol{x}_t^{(0)}$ is firstly acquired by using Eq. (10) and Eq. (11);

b) $\boldsymbol{H}^{(0)}_{n,1}$ and $\boldsymbol{H}^{(0)}_{n,2}$ are constructed by using the initial solution $\boldsymbol{x}_t^{(0)}$ in term of Eq. (17) and Eq. (18), and then we use them to generate $\tilde{\boldsymbol{y}}^{(0)}_{t,n}$ and $\boldsymbol{\Phi}^{(0)}_n$;

c) the number of iteration k is set to 0, and the maximum number of iteration *maxiter* is set to 10.

Step 2) Combining $\tilde{\boldsymbol{y}}^{(k)}_{t,n}$ and $\boldsymbol{\Phi}^{(k)}_n$ into Eq. (19), and GPSR algorithm is used to compute the sparse representation coefficients $\boldsymbol{\alpha}^{(k)}_{t,n}$, and then we use Eq. (11) to obtain the (k+1)-th iteration solution $\boldsymbol{x}^{(k+1)}_{t,n}$ of each block. Finally, all the interpolated blocks are combined into the estimation $\boldsymbol{x}_t^{(k+1)}$ of current frame.

Step 3) $k = k+1$, if $k \le maxiter$ and $\| \boldsymbol{x}_t^{(k+1)} - \boldsymbol{x}_t^{(k)} \|_2 \ge 10^{-4} \cdot N$, then $\boldsymbol{H}^{(k)}_{n,1}$, $\boldsymbol{H}^{(k)}_{n,2}$, $\tilde{\boldsymbol{y}}^{(k)}_{t,n}$ and $\boldsymbol{\Phi}^{(k)}_n$ can be updated as $\boldsymbol{H}^{(k+1)}_{n,1}$, $\boldsymbol{H}^{(k+1)}_{n,2}$、 $\tilde{\boldsymbol{y}}^{(k+1)}_{t,n}$ and $\boldsymbol{\Phi}^{(k+1)}_n$ by using $\boldsymbol{x}_t^{(k+1)}$ and the iteration goes back to Step 2), otherwise the algorithm will be stopped.

The predict frame $\boldsymbol{x}_{\mathrm{SI}}$ can be obtained by using CS joint reconstruction after the above steps perform several iterations, and finally the reconstruction of residual frame is performed to achieve final non-key reconstructed frame $\hat{\boldsymbol{x}}_t$.

4. Simulation results and analysis

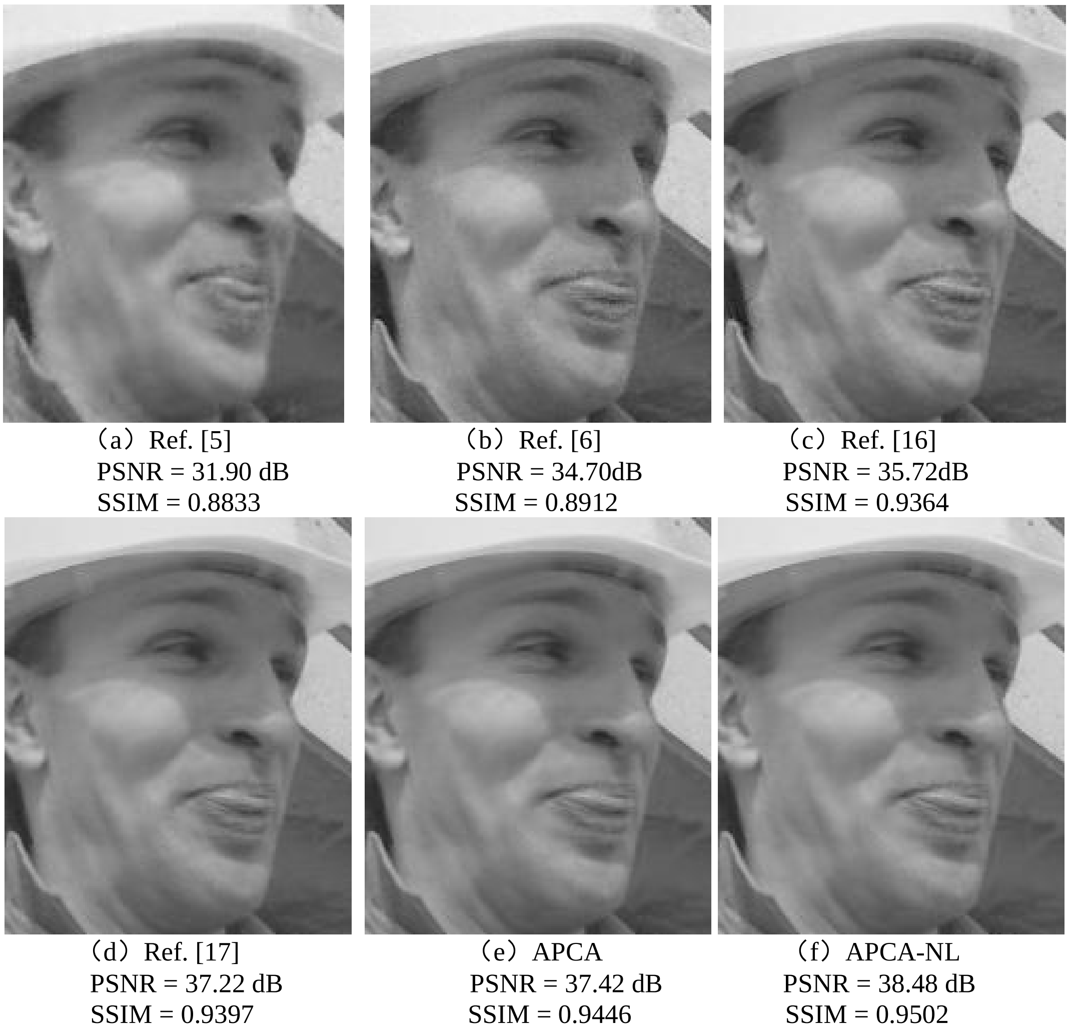
The proposed algorithm is evaluated by using the first 61 frames of four test sequences with CIF formant including *Foreman*, *Mobile*, *Bus* and *News*. The key frame is the odd frame (I frame), and the non-key frame is the even frame (P or B frame). In terms of the style of non-key frame, the proposed algorithm is performed under the two different predictive model, i.e., I-P-I model and I-B-I model. The key frame is independently reconstructed by the MH-BCS-SPL algorithm proposed by Ref. [14], and the non-key frame is reconstructed by the proposed algorithm and the four compared algorithms proposed by Ref. [5], Ref. [6], Ref. [16] and Ref. [17] respectively. The proposed algorithm is divided into two parts to do the comparison experiments: the algorithm uses only adaptive PCA sparse dictionary (i.e., reconstruction model (10)), and it is named as APCA; the algorithm uses adaptive PCA sparse dictionary and non-local similarity regularization item (i.e., reconstruction model (16)), and it is named as APCA-NL. The block size B in all algorithms is set to 16, the measurement rate S_{K} of key frame is set to 0.7, the range of the measurement rate S_{NK} of key frame is [0.1, 0.5]. The parameter setting of proposed algorithm is as follows: the radiuses S_1 and S_2 of search window are both set to B; the radius b of patch is set to 3; the threshold t selecting patch is set to 20; the regularization factors λ_1 and λ_2 are set to 0.2 and 0.5/k respectively; the other parameter c is 10.

The objective quality of reconstructed frame is evaluated by using the Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity (SSIM) [22], and the reconstruction time reveals the computational complexity. The hardware platform of experiments is a PC with 3.20 GHz CPU and 8 GB RAM, and the software platform is the MATLAB 7.6 under the system Windows 7 64 bits.

Table 1 presents the average PSNR and SSIM of all reconstructed non-key frames at the different measurement rate when the predictive model is I-P-I. It can be observed that the proposed algorithms APCA and APCA-NL have the higher PSNR and SSIM than the other compared algorithm at any measurement rate. Comparing APCA algorithm with APCA-NL algorithm , it can be seen that the performance of APCA-NL algorithm outperforms the APCA algorithm at the high measurement rate (S_{NK} is 0.4 or 0.5), e.g., when S_{NK} is 0.5, the APCA-NL algorithm obtains the PSNR gain 2.09 dB and SSIM gain 0.0058 than APCA algorithm for all test sequences, and but APCA-NL algorithm acquires a little performance improvement at the low measurement rate since the inaccurate motion estimation in measurement domain and lots of noises in the initial solution result in the fact that the added regularization item cannot better describe the non-local similarity of video. Besides, since the edge and texture regions of *Mobile* and *Bus* sequences have the complex structural features and do not contain lots of periodical predictive patterns, and their non-local similarity is low, which causes that APCA-NL algorithm cannot effectively improve performance for *Mobile* and *Bus* sequences in the basis of APCA algorithm and even degrades the quality of the reconstructed frame. **Fig. 5** shows the subjective visual quality of the reconstructed *Foreman* 8-th frame for various algorithm when S_{NK} is 0.3. It can be seen that the proposed algorithm can remove the blurring and blocking artifacts around lap, and the better subjective visual quality is obtained.

Table 1. Average PSNR (dB) and SSIM of test sequences for the proposed and existing algorithms under I-P-I Model

Squence	Reconstruction Algorithm					
	[5]	[6]	[16]	[17]	APCA	APCA-NL
	S_{NK} = 0.1					
Foreman	26.25, 0.7609	32.86, 0.8892	31.83, 0.8926	34.60, 0.9270	35.06, 0.9153	**35.10**, **0.9153**
Mobile	17.52, 0.3764	22.52, 0.7365	23.04, 0.7900	23.90, 0.8016	**24.15**, **0.8020**	24.13, 0.8016
Bus	20.23, 0.5116	23.98, 0.7542	20.57, 0.5896	26.10, 0.8461	26.96, 0.8307	**26.98**, **0.8308**
News	22.74, 0.7454	35.21, 0.9516	33.47, 0.9606	37.08, 0.9718	37.91, 0.9711	**37.96**, **0.9711**
Avg.	21.69, 0.5986	28.64, 0.8329	27.22, 0.8082	30.42, 0.8866	31.02, 0.8798	**31.04**, **0.8797**
	S_{NK} = 0.2					
Foreman	29.04, 0.8275	34.10, 0.8992	34.64, 0.9266	36.64, 0.8465	37.00, 0.9360	**37.42**, **0.9377**
Mobile	19.04, 0.4897	24.74, 0.7899	25.64, 0.8527	26.37, **0.8657**	26.66, 0.8639	**26.69**, 0.8633
Bus	22.26, 0.6366	26.07, 0.8062	25.08, 0.8129	28.58, 0.9032	29.66, 0.8906	**29.91**, **0.8922**
News	25.98, 0.8333	36.59, 0.9531	36.52, 0.9727	38.43, 0.9770	39.40, 0.9770	**39.79**, **0.9776**
Avg.	24.09, 0.6968	30.38, 08621	30.47, 08912	32.51, 08981	33.18, 09169	**33.45**, **0.9177**
	S_{NK} = 0.3					
Foreman	31.28, 0.8703	35.26, 0.9116	36.39, 0.9446	37.79, 0.9561	39.00, 0.9480	**39.15**, **0.9520**
Mobile	20.41, 0.5774	25.90, 0.8123	27.60, 0.8884	27.92, 0.8657	**29.52**, **0.8920**	28.40, 0.8917
Bus	23.97, 0.7215	27.63, 0.8382	28.86, 0.9100	30.33, 0.9032	**32.69**, 0.9199	32.19, **0.9238**
News	28.22, 0.8807	38.04, 0.9598	38.38, 0.9779	39.14, 0.9770	41.38, 0.9802	**41.73**, **0.9817**
Avg.	25.97, 0.7625	31.71, 0.8805	32.81, 0.9302	33.80, 0.9255	**35.65**, 0.9350	35.37, **0.9373**
	S_{NK} = 0.4					
Foreman	33.12, 0.8993	36.48, 0.9259	37.74, 0.9562	38.53, 0.9620	38.95, 0.9557	**40.59**, **0.9619**
Mobile	21.73, 0.6499	27.13, 0.8367	28.97, 0.9084	29.08, 0.9150	29.52, 0.9106	**29.87**, **0.9111**
Bus	25.68, 0.7889	29.25, 0.8703	31.41, 0.9402	31.66, 0.9460	32.69, 0.9362	**34.15**, **0.9425**
News	30.65, 0.9144	39.60, 0.9678	39.22, 0.9806	39.54, 0.9815	41.38, 0.9827	**43.33**, **0.9851**
Avg.	27.80, 0.8131	33.12, 0.9002	34.34, 0.9464	34.70, 0.9511	35.64, 0.9463	**36.99**, **0.9502**
	S_{NK} = 0.5					
Foreman	34.93, 0.9234	37.81, 0.9402	38.97, 0.9650	39.10, 0.9665	39.57, 0.9617	**41.94**, **0.9700**
Mobile	23.18, 0.7135	28.44, 0.8616	30.20, 0.9235	30.08, 0.9294	30.67, 0.9255	**31.48**, **0.9277**
Bus	27.40, 0.8412	30.94, 0.8997	33.35, 0.9583	32.79, 0.9570	33.69, 0.9484	**36.18**, **0.9574**
News	32.85, 0.9388	41.27, 0.9754	39.78, 0.9826	39.83, 0.9827	41.86, 0.9845	**44.54**, **0.9879**
Avg.	29.59, 0.8542	34.62, 0.9192	35.58, 0.9573	35.45, 0.9589	36.45, 0.9550	**38.54**, **0.9608**



（a）Ref. [5]
PSNR = 31.90 dB
SSIM = 0.8833

（b）Ref. [6]
PSNR = 34.70dB
SSIM = 0.8912

（c）Ref. [16]
PSNR = 35.72dB
SSIM = 0.9364

（d）Ref. [17]
PSNR = 37.22 dB
SSIM = 0.9397

（e）APCA
PSNR = 37.42 dB
SSIM = 0.9446

（f）APCA-NL
PSNR = 38.48 dB
SSIM = 0.9502

Fig. 5. When S_{NK} is 0.3, the comparison of subjective visual quality on *Foreman* 8-th frame for various algorithms under I-P-I model.

Table 2 presents the average PSNR and SSIM of all reconstructed non-key frames at the different measurement rate when the predictive model is I-B-I. Firstly, when compared with I-P-I model, the reconstructed quality of all test sequences is effectively improved, and this is because that the situation of B frame uses not only the information on the previous reconstructed frame but also performs the information on the following reconstructed frame. The performance variances of different algorithms are similar to the those of I-P-I model, the performance of proposed algorithms APCA and APCA-NL outperforms the other compared algorithm, and the APCA-NL algorithm can effectively improve the quality of reconstructed video frame at the high measurement rate. **Fig. 6** shows the subjective quality of the *Mobile* 4-th frame for various algorithms, and it can be seen that the proposed algorithms obtain the better subjective visual quality.

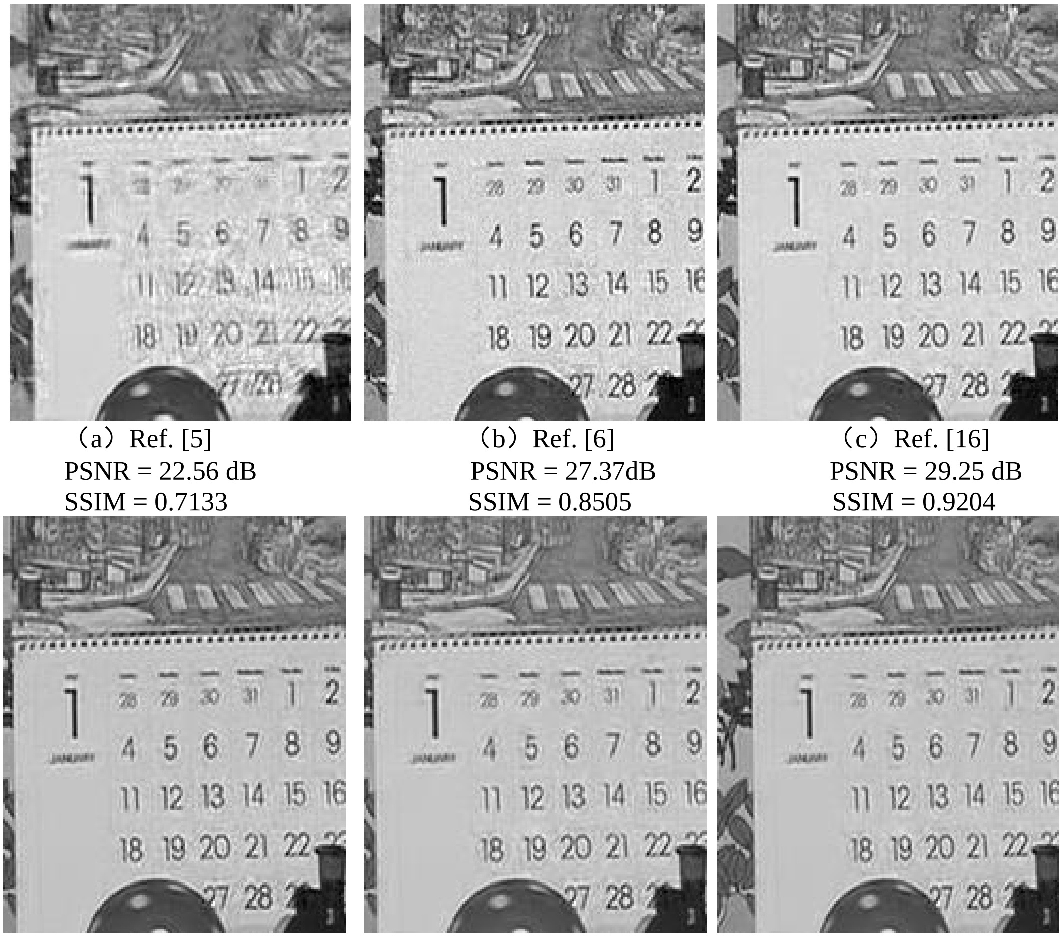
Table 3 presents the average reconstruction time (s/frame) of various algorithms. It can be observed that the reconstruction time under I-P-I model is lower than that of I-B-I model, which presents that the reconstructed quality is improved at the cost of the increasing computational complexity under I-B-I model. Besides, the proposed two algorithms increase the computational complexity and obtain the improvement of reconstructed quality, which presents that the better performance of proposed algorithms are achieved at the cost of the high computational complexity.

Table 2. Average PSNR (dB) and SSIM of test sequences for the proposed and existing algorithms under I-B-I Model

Squence	Reconstruction Algorithm					
	[5]	[6]	[16]	[17]	APCA	APCA-NL
	$S_{\rm NK}$ = 0.1					
Foreman	28.08, 0.8263	34.12, 0.9099	34.17, 0.9258	36.11, 0.9435	36.27, **0.9306**	**36.27**, 0.9305
Mobile	17.81, 0.4072	24.19, 0.8013	26.38, 0.8799	25.67, 0.8632	**25.89**, **0.8610**	25.88, 0.8607
Bus	20.50, 0.5571	25.75, 0.8198	21.78, 0.6564	26.41, 0.8536	27.29, 0.8409	**27.29**, **0.8411**
News	23.73, 0.8090	36.64, 0.9611	36.70, 0.9735	37.93, 0.9772	38.99, 0.9766	**39.00**, **0.9766**
Avg.	22.53, 0.6499	30.18, 0.8730	29.76, 0.8589	31.53, 0.9093	**32.11**, **0.9023**	32.11, 0.9022
	$S_{\rm NK}$ = 0.2					
Foreman	31.88, 0.8925	35.61, 0.9225	36.37, 0.9472	38.09, 0.9588	38.22, 0.9484	**38.49**, **0.9493**
Mobile	20.34, 0.6018	26.82, 0.8485	28.33, 0.9077	28.30, 0.9100	**28.59**, **0.9074**	28.58, 0.9068
Bus	24.04, 0.7510	28.22, 0.8661	26.59, 0.8600	29.94, 0.9073	30.19, 0.8982	**30.37**, **0.8998**
News	27.93, 0.9034	38.01, 0.9633	38.48, 0.9787	39.40, 0.9806	40.03, 0.9802	**40.33**, **0.9805**
Avg.	26.05, 0.7872	32.17, 0.9001	32.44, 0.9234	33.93, 0.9392	34.26, 0.9336	**34.44**, **09341**
	$S_{\rm NK}$ = 0.3					
Foreman	33.91, 0.9212	36.76, 0.9331	37.89, 0.9593	39.18, 0.9661	39.44, 0.9587	**40.16**, **0.9610**
Mobile	22.44, 0.7161	27.86, 0.8626	29.62, 0.9230	29.82, 0.9292	30.08, **0.9260**	**30.17**, 0.9255
Bus	26.45, 0.8375	29.80, 0.8895	30.28, 0.9311	30.78, 0.9334	32.10, 0.9254	**32.70**, **0.9288**
News	30.79, 0.9364	39.25, 0.9679	39.30, 0.9813	39.87, 0.9822	40.03, 0.9822	**41.88**, **0.9832**
Avg.	28.40, 0.8528	33.42, 0.9133	34.27, 0.9487	34.91, 0.9527	35.41, 0.9481	**36.23**, **0.9463**
	$S_{\rm NK}$ = 0.4					
Foreman	35.46, 0.9393	37.92, 0.9439	39.10, 0.9673	39.89,0.9705	40.31, 0.9652	**41.57**, **0.9691**
Mobile	24.30, 0.7972	29.08, 0.8814	30.65, 0.9335	30.85, 0.9406	31.19, 0.9375	**31.46**, **0.9376**
Bus	28.27, 0.8844	31.31, 0.9112	32.93, 0.9562	32.21, 0.9489	33.47, 0.9409	**34.73**, **0.9463**
News	33.61, 0.9570	40.62, 0.9737	39.85, 0.9830	40.13, 0.9832	41.80, 0.9842	**43.61**, **0.9861**
Avg.	30.41, 0.8945	34.73, 0.9276	35.63, 0.9600	35.77, 0.9608	36.69, 0.9570	**37.84**, **0.9600**
	$S_{\rm NK}$ = 0.5					
Foreman	36.95, 0.9533	39.14, 0.9544	40.23, 0.9736	40.42, 0.9739	41.00, 0.9703	**42.89**, **0.9756**
Mobile	26.33, 0.8559	30.37, 0.9005	31.70, 0.9433	31.67, 0.9487	32.31, 0.9470	**33.00**, **0.9484**
Bus	30.33, 0.9215	32.97, 0.9320	34.98, 0.9695	33.43, 0.9598	34.56, 0.9523	**36.78**, **0.9599**
News	36.11, 0.9700	42.14, 0.9794	40.28, 0.9844	39.75, 0.9835	42.25, 0.9856	**44.74**, **0.9884**
Avg.	32.43, 0.9252	36.16, 0.9416	36.80, 0.9677	36.32, 0.9665	37.53, 0.9638	**39.35**, **0.9681**

Table 3. Average reconstruction time (s/frame) comparison of various algorithms

I-P-I Model		I-B-I Model	
Algorithm	Time (s/frame)	Algorithm	Time(s/frame)
[5]	5.55	[5]	8.94
[6]	11.04	[6]	14.82
[16]	35.83	[16]	60.00
[17]	6.43	[17]	10.25
APCA	31.96	APCA	37.85
APCA-NL	46.32	APCA-NL	55.81



（a）Ref. [5]
PSNR = 22.56 dB
SSIM = 0.7133

（b）Ref. [6]
PSNR = 27.37dB
SSIM = 0.8505

（c）Ref. [16]
PSNR = 29.25 dB
SSIM = 0.9204

（d）Ref. [17]
PSNR = 29.62 dB
SSIM = 0.9231

（e）APCA
PSNR = 29.84dB
SSIM = 0.9236

（f）APCA-NL
PSNR = 29.93dB
SSIM = 0.9273

Fig. 6. When S_{NK} is 0.3, the comparison of subjective visual quality on *Mobile* 4-th frame for various algorithms under I-B-I model.

5. Conclusions

This paper combines the adaptive PCA sparse dictionary constructed by the correlation between frames and the regularization item constructed by the non-local similarity to propose a joint reconstruction algorithm for improving the rate-distortion performance of DVCS system. With the various temporal-spatial statistic characteristic, the fixed sparse dictionary cannot effectively exploit the sparse property of video frame, and although the sparse dictionary extracted from neighboring frames can change as the content of video frame is change, it is not the best one, this is because that the example-based sparse dictionary lacks the motion estimation between frames and the reference frame contains noises. The proposed construction of sparse dictionary firstly uses the CS measurements of current non-key frame to perform motion estimation in measurement domain, and then uses the motion information between frames to extract the example to produce the data matrix, and finally uses PCA to compute the significant principle components of data matrix for constructing the sparse dictionary. The sparse priori knowledge cannot still recover the edge and texture details of video frame well. To improve the quality of edge and texture regions, the non-local similarity of video frame is used to construct the regularization item, and the regularization item is mixed into the joint CS reconstruction model to remove the blurring and blocking artifacts in edge and texture regions. Experimental results show that the proposed algorithm can effectively improve the rate-distortion performance of DVCS system at the cost of a certain computational complexity, and achieve the better subjective and objective reconstructed quality.

References

[1] R. G. Baraniuk, "Compressive Sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118-121, 2007. Article (CrossRef Link)

[2] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge: Cambridge University Press, 2012: 1-5. Article (CrossRef Link)

[3] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero. "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no.1, pp.71-83, 2005. Article (CrossRef Link)

[4] D. Baron, M. F. Duarte, M. B. Wakin, S. Sarvotham, and R. G. Baraniuk. "Distributed compressive sensing," 2009, [Online], Available: www. arxiv.org/abs/0901.3403, 2009. 1. Article (CrossRef Link)

[5] L. W. Kang and C. S. Lu. "Distributed video compressive sensing," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp.1169-1172, April 2009. Article (CrossRef Link)

[6] T. T. Do, Y. Chen, D. T. Nguyen, N. Nguyen, L. Gan and T. D. Tran. "Distributed compressed video sensing," in *Proc. of IEEE International Conference on Image Processing*, pp.1393-1396, November 2009. Article (CrossRef Link)

[7] T. Do, L. Gan, N. Nguyen, and T. D. Tran. "Fast and efficient compressive sensing using structurally random matrices," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp.139-154, 2012. Article (CrossRef Link)

[8] K. Li, L. Gan and C. Ling. "Convolutional compressed sensing using deterministic sequences," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 740-752, 2013. Article (CrossRef Link)

[9] L. Gan. "Block compressed sensing of natural images," in *Proc. of International Conference on Digital Siagnal Processing*, pp. 403-406, July 2007. Article (CrossRef Link)

[10] G. Oechard, J. Zhang, Y. Suo, M. Dao, D. T. Nguyen, C. Sang, C. Posch, T. D. Tran and R. Etienne-Cummings. "Real time compressive sensing video reconstruction in hardware," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, no. 3, pp. 604-614, 2013. Article (CrossRef Link)

[11] J. Holloway, A. C. Sankaranarayanan, A. Veeraraghavan, and S. Tambe. "Flutter shutter video camera for compressive sensing of videos," *IEEE International Conference on Computational Photography*, pp. 1-9, April 2012. Article (CrossRef Link)

[12] X. L. Wu, W. S. Dong, X. J. Zhang, G. M. Shi. "Model-assisted adaptive recovery of compressed sensing with imaging application," *IEEE Transaction on Image Processing*, vol. 21, no. 2, pp. 451-458, 2012.Article (CrossRef Link)

[13] S. Mun and J. E. Fowler. "Block compressed sensing of images using directional transforms," in *Proc. of International Conference on Image Processing*, pp. 3021-3024, Cario, Egypt, November 2009. Article (CrossRef Link)

[14] C. Chen, E. W. Tramel, and J. E. Fowler. "Compressed sensing recovery of images and video using multihypothesis predictions," in *Proc. of Conference Record of the Forty Fifth Asilomar Conference*, pp. 1193-1198, November 2011. Article (CrossRef Link)

[15] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright. "Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems," *IEEE Journal Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586-597, 2007. Article (CrossRef Link)

[16] S. Mun and J. E. Fowler. "Residual reconstruction for block-based compressed sensing of video," in *Proc. of Data Compression Conference*, pp. 183-192, March 2011. Article (CrossRef Link)

[17] E. W. Tramel and J. E. Fowler. "Video compressed sensing with multihypothesis," in *Proc. of Data Compression Conference*, pp. 193-202, March 2011. Article (CrossRef Link)

[18] E. Candes, M. Wakin. "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21-30, 2008. Article (CrossRef Link)

[19] J. Ren, Y. Zhuo, J. Y. Liu, Z. Guo. "Illumination-invariant non-local means based video denoising," in *Proc. of IEEE International Conference on Image Processing*, pp. 1185-1188, September 2012. Article (CrossRef Link)

[20] Y. C. Shen, P. S. Wang and J. L. Wu. "Progressive side information refinement with non-local means based denoising process for Wyner-Ziv video coding," in *Proc. of Data Compression Conference*, April 2012: 219-226. Article (CrossRef Link)

[21] D. Kim, B. Keum, H. Ahn, and H. Lee. "Empirical non-local algorithm for image and video denoising," in *Proc. of IEEE International Conference on Consumer Electronics*, pp. 498-499, January 2013. Article (CrossRef Link)

[22] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli. "Image quality assessment : from error visibility to structural similarity," *IEEE Transaction on Image Processing*, vol. 13, no. 4, pp. 600-611, 2004. Article (CrossRef Link)

WU Minghu received the M.S. degree from Huazhong University of Science and Technology, and Ph.D. degree from Nanjing University of Posts and Telecommunications in 2002 and.2013, respectively. He is currently an associate professor in the School of Electrical and Electronic Engineering at Hubei University of Technology. His major research interests include signal processing, video coding and compressive sensing.

ZHU Xiuchang received his B.S. and M.S. degrees from Nanjing University of Po sts and Communications in 1982 and 1987, respectively. He has been working in Na njing University of Posts and Communications since 1987. At present, he is a Profess or and the direct of Jiangsu Key Library of Image Processing and Image Communica tions. His current research interests focus on multimedia information, especially on t he collection, processing, transmission and display of image and video.