An Auto-Regressive Model for Improved Low Delay Distributed Video Coding

Yongbing Zhang*[a], Debin Zhao[a], Siwei Ma[b], Ronggang Wang[c], Wen Gao[b]
[a]Department of Computer Science, Harbin Institute of Technology, Harbin, 150001, China
[b]Institute of Digital Media, Peking University, Beijing, 100871, China
[c]France Telecom R&D, Beijing, Co., Ltd
e-mail: ybzhang@jdl.ac.cn

ABSTRACT

In this paper, an auto-regressive (AR) model is proposed to generate the side information for low-delay distributed video coding (DVC). The side information generation of current Wyner-Ziv (WZ) frame t consists of two forward AR interpolations. First, each pixel within the rebuilt frame $t-1$ is approximated as a linear combination of pixels within a spatial neighborhood along the motion trajectory within the rebuilt frame $t-2$. Applying the least mean square algorithm, the coefficient of the first forward AR model is derived. Secondly, the pixels within the rebuilt frame $t-2$ are approximated by the corresponding pixels within rebuilt frame $t-1$. And then the geometric symmetric property of the AR model is exploited to derive the coefficient of the second forward AR model. Finally, the side information is generated as the average of the interpolations obtained by the two forward AR interpolations. The experimental results have demonstrated that the proposed AR model can significantly improve the PSNR of the side information compared to existing motion extrapolation based approaches.

Keywords: AR, DVC, side information, geometric symmetric

1. INTRODUCTION

Distributed video coding (DVC) has received more and more attentions in recent years due to its desirable properties for some applications such as wireless low power video surveillance and video camera phone. DVC is based on the principle of distributed source coding stated by the Slepian-Wolf [1] theorem for the lossless case and later extended by the Wyner-Ziv (WZ) [2] theorem to the lossy scenario. Different from traditional video coding systems, in DVC the high complexity operations, i.e. motion estimation, is performed at the decoder side. In the encoder side of DVC, the *key frames* of the vide sequence are compressed using a conventional intra-frame codec, and the remaining WZ frames are encoded using a WZ coder. At the decoder side, error correcting decoding is performed by concatenating the parity bits with the side information (SI), which can also be regarded as an estimate of the WZ frame or a noisy version of the WZ frame.

One of the most critical aspects in enhancing the compression efficiency of DVC is the improvement of the (SI) [3]. Since the better the SI is, the less error energy between WZ frame and SI is and consequently fewer parity bits are needed to correct the error. However, the construction of efficient SI is difficult in DVC due to the fact that the original video sequence is not available at the decoder. Many pioneering works have been finished to improve the quality of SI, which can be broadly classified into two categories: applying new model to enhance the SI quality [4] [5], transmitting the supplementary information to the decoder to generate higher quality SI [6] [7]. In [4], a camera moving in a 3-D static environment is captured for extracting the SI in DVC. It exploits the benefits of the structure-from-motion paradigm for distributed coding of this type of video content, and thus can generate SI with higher quality. However, this method is constrained to certain scenarios, such as static scenes. In [5], the SI is constructed based on the idea of universal prediction. It exploits the source statistics of the reconstructed video sequence and does not assume an underlying model of the input sequence. Nevertheless, the PSNR of the SI derived by this method drops a lot relative to the traditional motion extrapolation method, though the computational complexity is lowered. In [6] and [7], the FMO of H.264 and hash motion estimation are employed, respectively, to guide the SI generation at the decoder. Both of them

achieve better results, however, the improvement of PSNR values is at the expense of increasing the overhead incurred by transmitting the supplementary information.

In this paper, we propose an auto-regressive (AR) model to improve the quality of the SI for the low delay DVC where future frames are not used for generating the SI. The SI generation of current Wyner-Ziv (WZ) frame t is composed of two forward AR interpolations. First, each pixel within the rebuilt frame $t-1$ is approximated as a linear combination of pixels within a spatial neighborhood along the motion trajectory within the rebuilt frame $t-2$. Such a linear combination is called forward AR interpolation, since each pixel is approximated as the corresponding pixels within the forward frame. The Least-Mean-Square (LMS) algorithm is then employed to derive the AR coefficients. Secondly, the pixels within the rebuilt frame $t-2$ are approximated by corresponding pixels within the rebuilt frame $t-1$. Since each pixel is approximated as the corresponding pixels within the backward frame, this linear combination is called backward AR interpolation. And then the second forward AR coefficient is derived by exploiting the geometric symmetric property of the backward AR model. Finally, the SI is generated as the average of the interpolations obtained by the two forward AR models.

The remainder of this paper is organized as follows. Section 2 presents the detail description of the proposed AR model for the low delay DVC. Section 3 gives the experimental results, and finally this paper is concluded in Section 4.

2. THE PROPOSED AR MODEL AND THE GENERATION OF THE SIDE INFORMATION

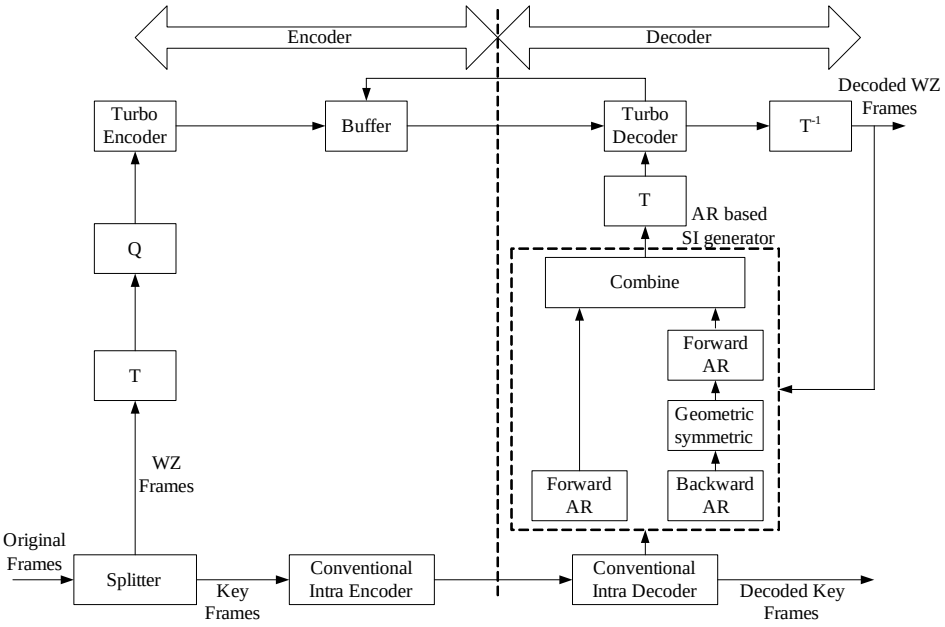

Fig. 1: Architecture of the proposed AR based SI generator in WZ coding

The proposed AR based SI generator in WZ coding aims to enhance the quality of the SI for the low delay DVC, and thus consequently improve the overall compression of WZ coding. Figure 1 illustrates the architecture of the proposed AR based SI generator in WZ coding. The modules enclosed by the dashed line correspond to the proposed AR based SI generator. At the decoder side, the SI is generated based on the observation of two recently rebuilt frames (including key frames and WZ frames). The SI generation is composed of two modules: the forward AR, the backward AR followed by the geometric symmetric rearranging process, whose purpose is to derive another AR forward AR coefficient corresponding to the derived backward AR coefficient. Then the SI generator combines the two forward AR models, and interpolates the ultimate SI of the current WZ frame. The forward and backward AR models will be given in the following two sub-sections.

2.1 Forward AR Model

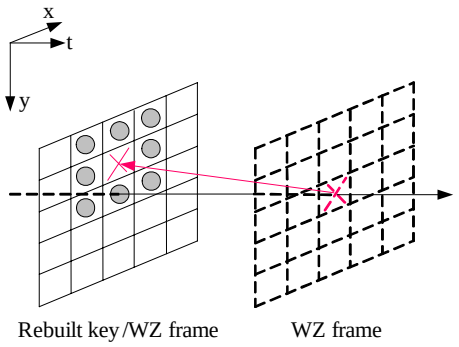

Fig. 2: The forward AR model

The proposed SI generation process is performed block by block. Let $\mathbf{Y}$ be a rectangular block of pixels in the current WZ frame, whose SI will be predicted based on its history observations in the two forward reference frames (rebuilt key/WZ frames). Let $\mathbf{X}$ be the similar block in the forward reference frame along the motion trajectories of $\mathbf{Y}$. As shown in Fig.2, for each pixel in the WZ frame, we find the corresponding pixel in the forward reference frame along the motion trajectory, depicted as the red arrow in Fig.2. Next, we find a square spatial neighborhood, centered on the corresponding pixel in the forward reference frame. And then the SI of the pixel in the current WZ frame is approximated as a linear combination of the pixels in the corresponding spatial neighborhood (the pixels indicated by the circles and the red arrow in Fig.2). The interpolation process can be expressed as

$$\hat{Y}_t(m,n) = \sum_{-r\le(i,j)\le r} X_{t-1}(\tilde{m}+i,\tilde{n}+j)\bullet\alpha_{i,j} + n_t(m,n) \tag{1}$$

where $\hat{Y}_t(m,n)$ represents the SI of the pixel located at (m,n) in the current WZ frame t, X_{t-1} represents the forward reference frame, $(\tilde{m},\tilde{n})$ represents the corresponding integer position in the forward reference frame pointed by the motion vector of $\hat{Y}_t(m,n)$, $\alpha_{i,j}$ is the forward AR coefficient, and $n_t(m,n)$ is the additive Gaussian while noise. Here r is defined to be the radius of the AR filter, and thus the size of the AR filter is $(2r+1)\times(2r+1)$.

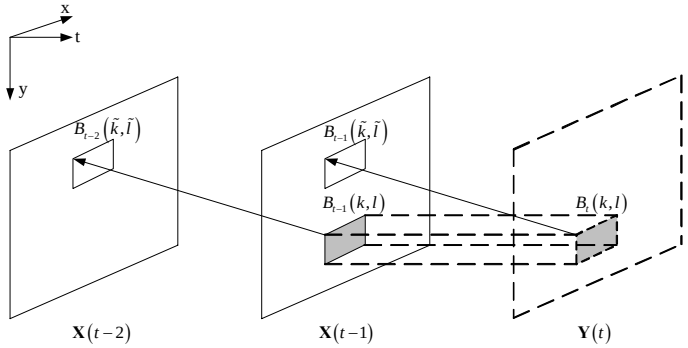

Fig. 3: Corresponding blocks between adjacent frames when deriving AR coefficients

It's clear that the AR coefficient plays a critical role for the accuracy of the SI. Since the actual pixel in the current WZ frame is not available at the decoder, we derive the AR coefficient in the two closest rebuilt key/WZ frames. As shown

in Fig.3, the derivation of AR coefficients for each block in the current WZ frame $\mathbf{Y}(t)$ involves two frames, which are the forward reference frame $\mathbf{X}(t-1)$, and the frame prior to the forward reference frame $\mathbf{X}(t-2)$. For each block $B_t(k,l)$ in the current WZ frame $\mathbf{Y}(t)$, we first find its collocated block $B_{t-1}(k,l)$ in the forward reference frame $\mathbf{X}(t-1)$. And then we find a most similar block $B_{t-2}(\tilde{k},\tilde{l})$ in frame $\mathbf{X}(t-2)$, and the displacement between $B_{t-1}(k,l)$ and $B_{t-2}(\tilde{k},\tilde{l})$ is denoted as $v_{t-1,t-2}(k,l)$. Here, $v_{t-1,t-2}(k,l)$ is of integer accuracy, since the AR interpolation is processed on the integer position pixels. After that, we assume the motion vector $v_{t,t-1}(k,l)$ of $B_t(k,l)$ is the same as that of its collocated block $B_{t-1}(k,l)$. And consequently, we can find the corresponding block $B_{t-1}(\tilde{k},\tilde{l})$ in frame $\mathbf{X}(t-1)$ for block $B_t(k,l)$. We assume each pixel in $B_{t-1}(k,l)$ is approximated as a linear combination of a square spatial neighborhood in block $B_{t-2}(\tilde{k},\tilde{l})$, centered on the corresponding pixel pointed by the motion vector $v_{t-1,t-2}(k,l)$. In other words, each pixel in $B_{t-1}(k,l)$ can be interpolated as

$$\hat{X}_{t-1}(m,n) = \sum_{-r\le(i,j)\le r} X_{t-2}(\tilde{m}+i,\tilde{n}+j)\bullet\alpha_{i,j} + n_{t-1}(m,n) \tag{2}$$

Due to the piecewise stationary characteristics of pixel value, we assume all the pixels within block $B_{t-1}(k,l)$ share the same AR coefficients. The forward AR coefficient should be chosen to be the "best" in some sense. Here we use the most common measure of performance of a predictor: the mean squared error (MSE). Define the resulting mean squared error by

$$\varepsilon_f{}^2(k,l) = \sum_{(m,n)\in B_{t-1}(k,l)}\sum E\left(\left\|X_{t-1}(m,n)-\hat{X}_{t-1}(m,n)\right\|^2\right) \tag{3}$$

The MSE as a performance criterion can be viewed as a measure of how much the energy of the signal is reduced by removing the predictable information based on the observables from it. Since the goal of a predictor is to remove this predictable information, a better predictor corresponds to a smaller MSE. We then use the LMS method to derive the optimum AR coefficients. Since there is a high similarities among the same objects within adjacent frames, we assume the AR coefficients, used to predict $B_{t-1}(k,l)$ as the linear combination of pixels in $B_{t-2}(\tilde{k},\tilde{l})$ are the same with the AR coefficients, used to predict $B_t(k,l)$ as the linear combination of pixels in $B_{t-1}(\tilde{k},\tilde{l})$. Assume the optimum AR coefficients derived by using Eq (3) is $\boldsymbol{\alpha}$, we then use $\boldsymbol{\alpha}$ to obtain the SI of $B_t(k,l)$ as

$$\hat{Y}_t(m,n) = \sum_{-r\le(i,j)\le r} X_{t-1}(\tilde{m}+i,\tilde{n}+j)\bullet\alpha_{i,j} + n_t(m,n) \tag{4}$$

2.2 Backward AR Model and the Geometric Symmetric Rearranging Process

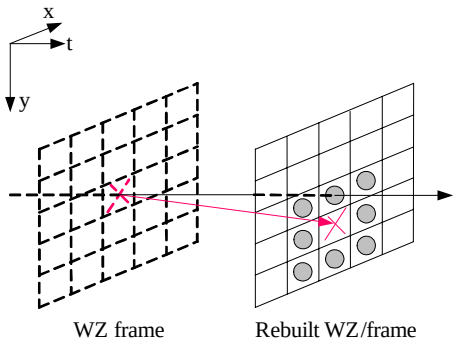

Fig. 4: The backward AR model

If the following rebuilt frame is available, the SI of the current WZ frame can also be interpolated by the backward AR interpolation, which is depicted as Fig. 4. In other words, each pixel in $B_{t-2}\left(\tilde{k},\tilde{l}\right)$ can be approximated as

$$\hat{X}_{t-2}\left(\tilde{m},\tilde{n}\right)=\sum_{-r\le(i,j)\le r} X_{t-1}\left(m+i,n+j\right)\bullet\beta_{i,j}+n_{t-2}\left(\tilde{m},\tilde{n}\right), \tag{5}$$

where $\beta_{i,j}$ is the backward AR coefficient. We assume the forward and backward AR coefficients of the same block are symmetric relative to the center of the AR filter, which is depicted in Figure 5. That is to say, given the backward AR coefficient, the corresponding forward AR coefficient can be computed as

$$\beta'_{i,j}=\beta_{-i,-j}, \tag{6}$$

where $\beta'_{i,j}$ is the corresponding forward AR coefficient. This can be easily explained in geometry. For example, if the backward motion vector is along a certain direction, which is embodied by the relatively larger AR coefficients along the corresponding direction [8]. And if we rearrange it in a symmetric way relative to the center of the AR filter, the AR coefficients along the reverse direction become relatively larger, which thus in turn embodied by the reverse motion vector as the former one.

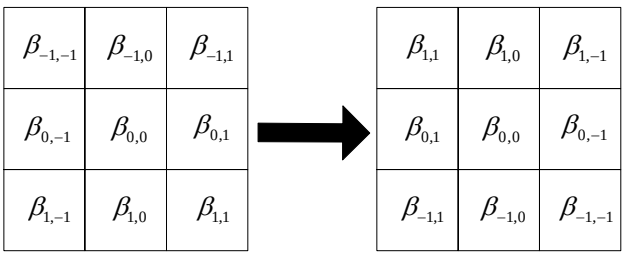

Fig. 5: The geometric symmetric relation between the forward and backward AR coefficients

Similarly, the backward AR coefficients can be derived by minimizing

$$\varepsilon_b{}^2\left(k,l\right)=\sum_{(\tilde{m},\tilde{n})\in B_{t-2}(k,l)}\sum E\left(\left\|X_{t-2}\left(\tilde{m},\tilde{n}\right)-\hat{X}_{t-2}\left(\tilde{m},\tilde{n}\right)\right\|^2\right) \tag{7}$$

And then according to Eq (6), the corresponding forward AR coefficient $\beta'_{i,j}$ is computed, based on which the approximation of the SI can also be approximated as

$$\hat{Y}_t(\tilde{m},\tilde{n}) = \sum_{-r\le(i,j)\le r} X_{t-1}(m+i,n+j)\bullet \beta'_{i,j} + n_{t-1}(\tilde{m},\tilde{n}). \quad (8)$$

Consequently, we derive two AR coefficients: one is derived by Eq. (3), and the other is derived indirectly by the geometric symmetric property of the AR model, which is obtained by Eq. (6). And then, the ultimate SI of each pixel can be computed as

$$\hat{Y}_t(m,n) = \left(\sum_{-r\le(i,j)\le r} X_{t-1}(\tilde{m}+i,\tilde{n}+j)\bullet \alpha_{i,j} + \sum_{-r\le(i,j)\le r} X_{t-1}(\tilde{m}+i,\tilde{n}+j)\bullet \beta'_{i,j} \right) \Big/ 2 + n_t(m,n). \quad (9)$$

In the proposed AR model, the interpolation is processed on the integer pixels; however, it can achieve fractional pixel accuracy. This is because, in the typical quarter-pixel interpolation method, each sub-pixel is interpolated by a fixed filter tap along the horizontal and vertical directions using the integer pixels closest to the sub-pixel to be interpolated. And then each block finds a most similar one in the interpolated fractional resolution frame. While in the proposed AR model, for each pixel, we first find the corresponding integer pixel in the reference frame, and then interpolate the pixel as the weighted linear combination of the integer pixels in a squared neighborhood centered at the corresponding integer pixel. If we choose proper AR coefficients, it achieves the same result as the traditional fractional pixel interpolation method. However, in the real image, the interpolation only along the horizontal and vertical direction may not always accurate enough in the case of complex region. Besides, in the traditional method, the interpolation tap and filter are always fixed, which further limits the accuracy of the interpolation results. Whereas, in the proposed AR model, the interpolation can be along any direction, i.e. the coefficients along a specific direction are significantly larger than the other directions, rather than restricted to the horizontal or the vertical direction. Furthermore, the AR coefficients can be adaptively tuned according to the characteristics of pixels in the adjacent spatial neighborhood.

3. EXPERIMENTAL RESULTS

To verify the superior performance of the proposed AR model, 50 frames of “foreman”, “mobile”, and “city” sequences at QCIF resolution have been encoded and compared to an approach similar to that described in [9]. An RTCP turbo encoder with two identical 1/2 rate constituent convolution encoders and a generator matrix of $\begin{bmatrix} 1 & \frac{1+D+D^3+D^4}{1+D^3+D^4} \end{bmatrix}$ were used [10]. A random puncturing pattern of period 32 was used with a maximum of 18 iterations. The acceptable bit error rate threshold was set to 10^{-3}.

The results presented in Fig. 6 and Fig. 7 show the average PSNR of the SI and rebuilt WZ frames vs the bit rate for all the coded WZ frames. The proposed method (denoted as one AR model and two AR models in the figures) results in a significant improvement compared with the traditional motion extrapolation method for both the SI and rebuilt WZ frames. Here, one AR model refers to applying only one forward AR model for the SI generation, and two AR models indicate that besides the one direct forward AR model, another forward AR model, derived by rearranging the backward AR coefficients, is also applied. The radius of AR model for Mobile and Foreman is set to be 2, and the radius of AR model for City is set to be 1.

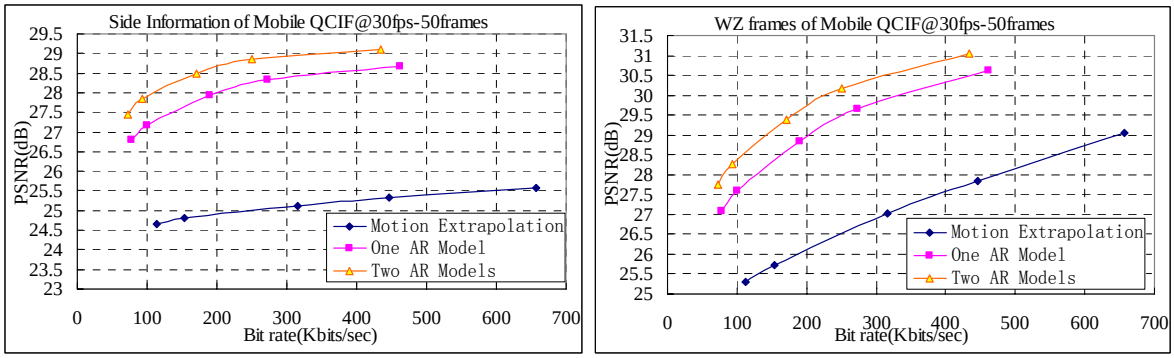

Fig. 6: Comparison of the SI and WZ frames of Mobile sequence

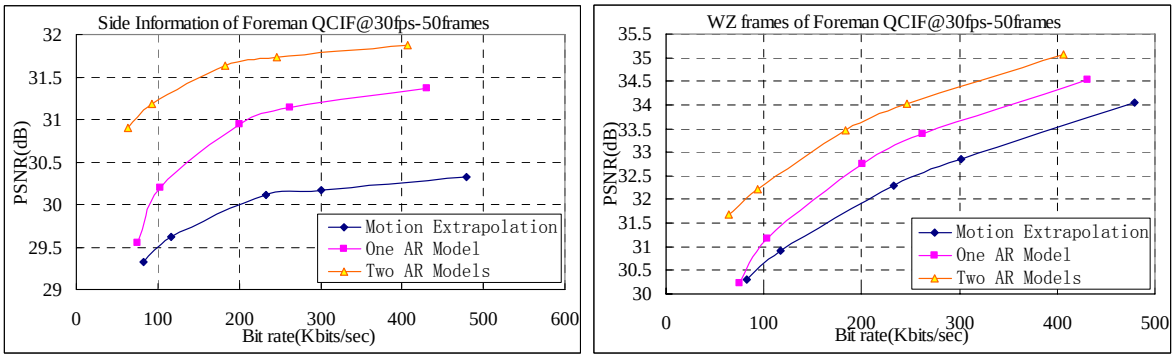

Fig. 7: Comparison of the SI and WZ frames of Foreman sequence

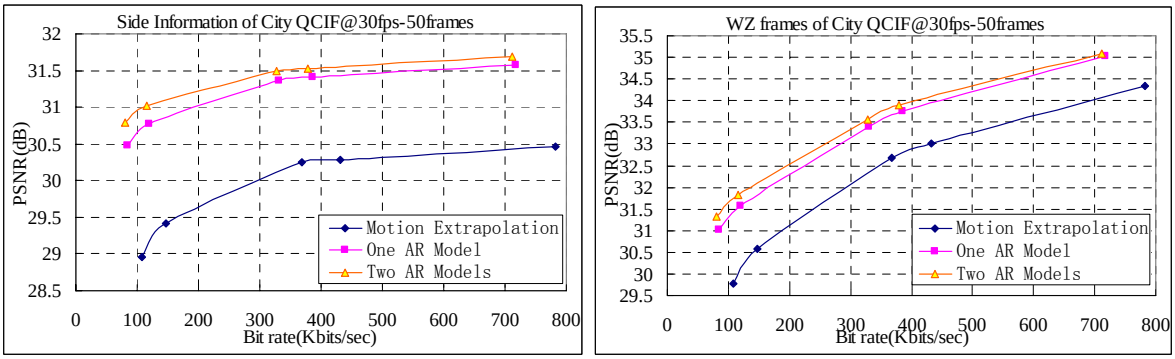

Fig. 8: Comparison of the SI and WZ frames of City sequence

For Mobile sequence, the gains, compared with motion extrapolation method, for the SI are 3 and 3.5dB for the one AR model and two AR models, respectively. Similar gains can also be achieved for the WZ frames in Mobile sequence. For Foreman sequence, the gain for the SI can be 1dB and 1.5dB for the one AR model and two AR models. The high improvement for the SI results in the improvement for the rebuilt WZ frames. The AR model outperforms the traditional extrapolation method greatly attributes to its superior ability to tune to coefficients according to the characteristics of pixels within a spatial neighborhood in the adjacent frame. Two AR models achieve a higher performance than just one AR model is mainly due to the reason that it elegantly exploits the geometric symmetric property of the AR filter and thus can further enhance the accuracy of the performance. For City sequence, the gain for the SI can be 1.5 and 2dB for one AR and two AR models, and similar gains can be achieved for WZ frames. We have additionally plotted the frame by frame PSNR (Mobile) obtained with methods of traditional extrapolation, one AR model, two AR models in Fig. 9. It can be seen that the PSNR obtained by one AR model is significantly larger than that of extrapolation method for every frame. And the two AR models achieve an even higher PSNR values than that of the one AR model. Furthermore, the PSNR fluctuations of the proposed method (one AR model and two AR model) are much smaller compared to motion extrapolation.

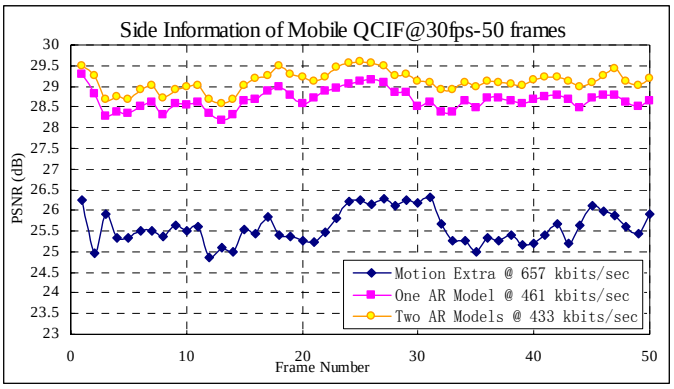

Fig. 9: Comparison of PSNR of each SI of Mobile sequence

4. CONCLUSION

We have presented an AR model for the SI generation for low delay DVC. Based on the two recently decoded frames, we derive two forward AR coefficients, one is derived directly based on LMS, and the other is derived by exploiting the geometric symmetric property of the AR coefficients of the corresponding backward AR model. The side information is generated as the average of two AR interpolations, where each pixel in the current WZ frame is interpolated as a linear combination of pixels within a spatial neighborhood along the motion trajectory in the latest decoded frame. The experimental results presented show that the proposed method can significantly enhance the PSNR of the side information relative to the existing motion extrapolation based approaches.

5. ACKNOWLEDGEMENTS

This work was supported in part by National Science Foundation (60672088 and 60736043) and Major State Basic Research Development Program of China (973 Program, 2009CB320905).

REFERENCES

[1] D. Slepian and J.K.Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, Vol. IT-19, pp.471-480, Jul.1973.

[2] A.D.Wyner and J. Ziv, "The rate distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, Vol. IT-22, no.1, pp.1-10, Jan.1976.

[3] Z. Li, L. Liu, "Rate distortion analysis of motion side estimation in Wyner-Ziv video coding," IEEE Trans. on Image process, Vol. 16, No. 1, pp.98-113, May. 2007

[4] M.Maitre, C. Guillemot, and L. Morin, "3-D Model-based frame interpolation for distributed video coding of static scenes," IEEE Trans. on Image process., Vol. 7, No. 16, pp.1246-1257, May. 2007

[5] Z. Li, L. Liu, and E. J. Delp, "Wyner-Ziv video coding with universal prediction," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 16, No. 11, pp.1430-1436, Nov. 2006

[6] D. Agrafiotis, P. Ferre, and D.R. Bul, "Hybrid key/wyner-Ziv frames with flexible macroblock ordering for improved low delay distributed video coding," Proc. SPIE Vol. 6508, Visual Communications and image processing 2007

[7] J. Ascenso, F. Pereira, "Adaptive hash-based side information exploitation for efficient Wyner-Ziv Video coding," Proc. ICIP, pp. 29-32, Sept. 2007

[8] X. Li, "Least-square prediction for backward adaptive video coding," EURASIP Journal on Applied Signal Processing, 2006, special Issue on H.264 and Beyond.

[9] L. Natario, C. Brites, J. Ascenco and F. Pereira, "Extrapolating Side Information for Low Delay pixel Domain Distributed Video Coding," Int. Workshop on Very Low Bitrate Video Coding, Sept. 2005

[10] Aaron, and B. Girod, "Compression with Side Information using Turbo codes," IEEE Data Compression Conf., Apr. 2002 .