AN AUTO-REGRESSIVE MODEL FOR CHECKBOARD SPLITTING BASED WYNER-ZIV CODING

Yongbing Zhang[1], Debin Zhao[1], Siwei Ma[2], Ronggang Wang[3], Wen Gao[2]
[1]Harbin Institute of Technology, Harbin, China
[2]Peking University, Beijing, China
[3]France Telecom R&D, Beijing, Co., Ltd
{ybzhang, dbzhao, swma, wgao}@jdl.ac.cn, ronggang.wang@orange-ftgroup.com

ABSTRACT

An auto-regressive (AR) based side information (SI) generation is proposed in this paper for block based chessboard pattern Wyner-Ziv (WZ) coding, where each WZ frame is split into two sets at encoder and then encoded separately. At the decoder, one set of the WZ frame will be firstly reconstructed, and then proposed AR model is used to generate the SI of the other set, where each pixel is generated as a linear weighted summation of pixels within two square windows in the previous and following reconstructed WZ/key frames along the motion trajectory. To obtain high quality SI for the second set, reconstructed pixels in the four neighboring blocks of the first set are employed to derive accurate AR coefficients. Several experimental results demonstrate that the proposed AR model is able to improve the quality of the SI for the second set of the WZ frame, which leads to the improvement of the rate-distortion performance of the WZ coding.

Index Terms— auto-regressive, side information, chessboard splitting, wyner-ziv

1. INTRODUCTION

Distributed video coding (DVC) has emerged as a complementary of hybrid video coding due to its desirable properties for some applications such as wireless low power video surveillance, video compression and sensor networks. DVC is based on the principles stated by Slepian-Wolf [1] for lossless case and Wyner-Ziv (WZ) [2] for lossy scenario. Slepian and Wolf proved that although two statistical sources X and Y are independently encoded, similar performance can be achieved as long as joint decoding of them is allowed for lossless coding. Wyner and Ziv theory extended the theory to lossy coding with side information (SI) generated at decoder.

One of the most critical aspects in enhancing the compression efficiency of DVC is improving SI quality. According to the Slepian-Wolf theorem [1], the less the conditional entropy $H(X|Y)$ is, the fewer the bits to reconstruct X are required, under the condition that Y can be perfectly reconstructed at the decoder. Intuitively, in practical system, where SI is generated at the decoder side, better SI will result in better performance for the WZ frames.

Currently, many pioneering works have been done to improve the quality of the SI. In A. Aaron's scheme, motion extrapolation or interpolation is applied to generate SI [3], and hash words are also used to aid the motion compensation [4] to further improve the SI quality. A.B.B. Adikari exploits sequential motion estimation to refine SI [5]. J. Ascenso presented a motion compensation method which employed spatial motion smoothing to refine SI [6], and he also presented a motion compensated refinement method [7] to improve SI. In addition, M. Tagliasacchi proposed a novel WZ frame coding approach, where the WZ frames are split into two sets A and B pixel by pixel based on quincunx sampling [8]. Furthermore, H. Liu proposed a block based checkerboard pattern splitting algorithm [9]. All these methods resort to conventional motion estimation to extract motion information from the reconstructed WZ/Key frames at the decoder side. However, the motion estimation method does not always achieve good results, especially for the video sequences with high motion.

To further improve the quality of SI, we propose an auto-regressive (AR) based SI generation in block based checkerboard pattern DVC in this paper. In the proposed AR model, the SI of each pixel in the second set of the WZ frame is generated as a linear weighted summation of pixels within two square windows along the motion trajectory in the previous and following reconstructed WZ/Key frames. To obtain high quality SI, the pixels within the four neighboring reconstructed blocks in the first set of the WZ frame are exploited to derive accurate AR coefficients by the maximum likelihood least method. This allows gains up to 1.5dB for the SI of the second set of the WZ frame.

This paper is organized as follows. Section 2 presents the architecture of the proposed WZ coding system and gives the description of the proposed AR model. In Section 3, the maximum likelihood least square algorithm is described to obtain accurate AR coefficients. Experimental results are presented and discussed in Section 4. Finally, section 5 concludes this paper.

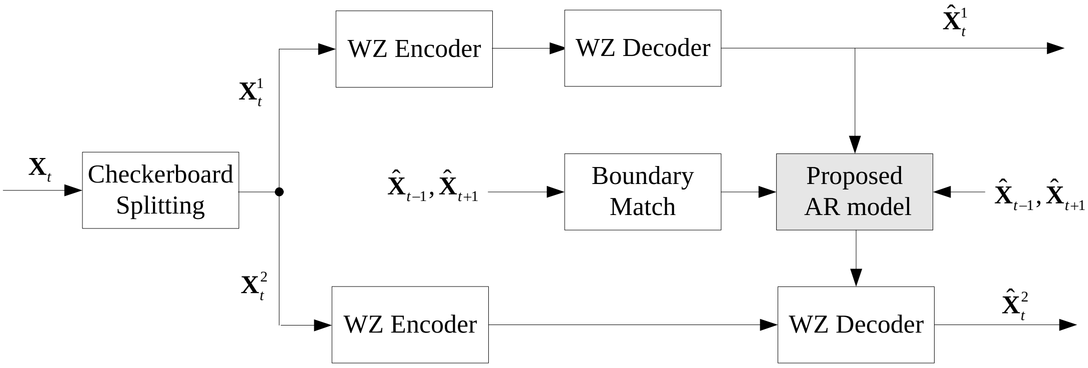

Fig. 1 The architecture of the proposed AR based WZ codec

2. PROPOSED FRAMEWORK AND AR MODEL DESCRIPTION

In this section, we will first give the codec architecture and then the model description will be presented.

2.1. Proposed codec architecture

The codec architecture of the proposed AR model based WZ coding is depicted in Fig. 1. The input sequences are first divided into key (K) frames and WZ frames, which are encoded by H. 264 and Turbo encoder, respectively. Each WZ frame X_t is split into two sets (set 1 and set 2) based on checkerboard pattern, which is depicted in Fig. 2.

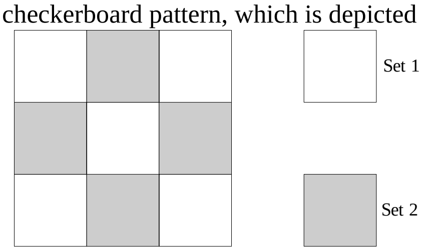

Fig. 2 Block based checkboard splitting

X_t^1 and X_t^2 are used to denote set 1 and set 2 of X_t, and

$$X_t = X_t^1 + X_t^2 . \tag{1}$$

X_t^1 is transformed and quantized first, and then encoded by a Turbo codec. At the decoder side, the reconstruction $\hat{X}_t^1$ of the pixels in the first set within X_t is obtained by Turbo decoder. Together with $\hat{X}_t^1$, the SI Y_t^2 of X_t^2 is interpolated by the proposed AR model. Similar to the first set, $\hat{X}_t^2$ is decoded by Turbo decoder. Finally, $\hat{X}_t^1$ and $\hat{X}_t^2$ are integrated into one frame and generate the ultimate WZ frame $\hat{X}_t$.

2.2. Proposed AR model

In the proposed AR model, the SI of each pixel within set 2 is generated as a linear weighted summation of corresponding pixels within two square windows along the motion trajectory in the previous and following reconstructed WZ/K frames, which is shown in Fig. 3. With the help of reconstructed neighboring pixels within set 1, the integer-pixel accuracy motion vectors of each block in set 2 are first found by boundary match. Then the SI generation is performed along the motion trajectory, which can be expressed as

$$\begin{aligned} Y_t^2(m,n) = & \sum_{-r\le(i,j)\le r} \hat{X}_{t-1}\left(\tilde{m}_p+i,\tilde{n}_p+j\right)\bullet w_p(i,j) \\ & + \sum_{-r\le(i,j)\le r} \hat{X}_{t+1}\left(\tilde{m}_f+i,\tilde{n}_f+j\right)\bullet w_f(i,j)+n_t(m,n) \end{aligned}, \tag{2}$$

where $Y_t^2(m,n)$ represents the SI located at (m,n) in set 2, $\hat{X}_{t-1}$ and $\hat{X}_{t+1}$ represent the previous and following reconstructed WZ/K frames, $(\tilde{m}_p,\tilde{n}_p)$ and $(\tilde{m}_f,\tilde{n}_f)$ represent the corresponding pixel location pointed by the forward and backward motion vectors, $w_p(i,j)$ and $w_f(i,j)$ represent the weighting coefficients corresponding to $\hat{X}_{t-1}$ and $\hat{X}_{t+1}$, and $n_t(m,n)$ represents the white Gaussian noise. Here the variable r in Eq. (2) is defined to be the order of the AR model. Thus according to Eq. (2), the SI of each pixel can be generated as the weighted summation of $(2r+1)\times(2r+1)$ corresponding pixels in the previous reconstructed frame and $(2r+1)\times(2r+1)$ corresponding pixels in the following reconstructed frame.

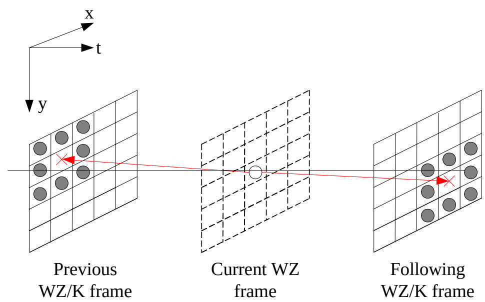

Fig. 3 Proposed AR model

Obviously, the AR coefficients will play a critical role for the quality of the SI. To derive accurate coefficients, we will give a maximum likelihood least square algorithm in the next section to obtain accurate coefficients.

3. MAXIMUM LIKELIHOOD LEAST SQUARE ALGORITHM

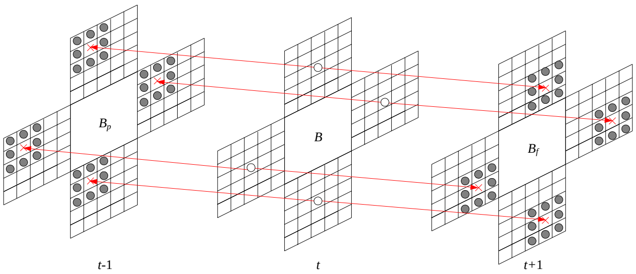

Fig. 4 Pixels involved during the AR coefficient derivation process for the block B in set 2

Since the actual pixels within set 2 are not available at the decoder side, the reconstructed pixels within the neighboring blocks in set 1 are utilized to train the AR coefficients of each block within set 2. As shown in Fig. 4, for each block B of set 2 in the current WZ frame *t*, we first find its best matched blocks (B_p and B_f) by performing boundary match in the previous and following reconstructed frames $\hat{X}_{t-1}$ and $\hat{X}_{t+1}$. Next, we use the AR model to approximate the pixels in the reconstructed neighboring blocks of set 1.

According to Eq. (2), we can rewrite the SI of each pixel within the current block B in set 2 as

$$y = f(\mathbf{x}, \mathbf{w}) + \varepsilon , \tag{3}$$

where $f(\mathbf{x}, \mathbf{w})$ is a deterministic function about the pixel vector, corresponding to the square windows in the previous and following reconstructed WZ/K frames, and the coefficient vector $\mathbf{w}$. Here ε is a zero mean Gaussian random variable with prevision (inverse variance) β . Thus we can write (3) as

$$p(y \mid \mathbf{x}, \mathbf{w}, \beta) = N\left(y \mid f(\mathbf{x}, \mathbf{w}), \beta^{-1}\right). \tag{4}$$

In this paper, we assume $N\left(y \mid f(\mathbf{x}, \mathbf{w}), \beta^{-1}\right)$ obeys the Gaussian distribution, which can be expressed as

$$N\left(x \mid \mu, \sigma^2\right) = \frac{1}{\left(2\pi\sigma^2\right)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}. \tag{5}$$

Now consider a data set of inputs $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ with corresponding target values $y_1, ..., y_N$. We group the target variables $\{y_n\}$, which represents the pixels within the four reconstructed neighboring blocks of set 1, into a column vector denoted by $\mathbf{y}$. Assume these data points are drawn independently from the distribution (4), we obtain the following expression for the likelihood function, which is a function of adjustable parameters $\mathbf{w}$ and β , in the form

$$p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} N\left(y_n \mid \mathbf{w}^T \mathbf{x}_n, \beta^{-1}\right). \tag{6}$$

Taking the logarithm of the likelihood function and making use of the standard form (Eq. 5) of univariate Gaussian, we have

$$\begin{aligned} \ln p(\mathbf{y} \mid \mathbf{w}, \beta) &= \sum_{n=1}^{N} \ln N\left(y_n \mid \mathbf{w}^T \mathbf{x}_n, \beta^{-1}\right) \\ &= \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) - \beta E_D(\mathbf{w}) \end{aligned} , \tag{7}$$

where the sum-of-squares error function is defined by

$$E_D(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \left\{y_n - \mathbf{w}^T \mathbf{x}_n\right\}^2 . \tag{8}$$

The gradient of the log likelihood function (Eq. 7) takes the form

$$\nabla \ln p(\mathbf{y} \mid \mathbf{w}, \beta) = \sum_{n=1}^{N} \left\{y_n - \mathbf{w}^T \mathbf{x}_n\right\} \mathbf{x}_n^{\ T} . \tag{9}$$

Setting this gradient to zero gives

$$0 = \sum_{n=1}^{N} y_n \mathbf{x}_n^{\ T} - \mathbf{w}^T \left(\sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^{\ T}\right). \tag{10}$$

Solving for $\mathbf{w}$ we obtain

$$\mathbf{w}_{ML} = \left(\Phi^T \Phi\right)^{-1} \Phi^T \mathbf{y} , \tag{11}$$

which is known as the normal equation for the least square algorithm. Here Φ is an $N \times M$ matrix, whose elements are given as

$$\Phi = \begin{pmatrix} \mathbf{x}_{0,0} & \mathbf{x}_{1,0} & \cdots & \mathbf{x}_{M-1,0} \\ \mathbf{x}_{0,1} & \mathbf{x}_{1,1} & \cdots & \mathbf{x}_{M-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_{0,N-1} & \mathbf{x}_{1,N-1} & \cdots & \mathbf{x}_{M-1,N-1} \end{pmatrix}. \tag{12}$$

Here N represents the size of the four reconstructed neighboring blocks in set 1, and M represents the number of pixels within the two square windows in the previous and following reconstructed WZ/K frames. Using the coefficients derived by Eq. (11), we can easily compute the SI for the current processing block B according to Eq. (2).

4. EXPERIMENTAL RESULTS

In this section, we report our experimental results on *Foreman*, *News*, and *Flower* in CIF@30HZ, where the odd frames of the first 60 frames in each test sequence are encoded in H. 264 intra frame model, while the even frames are encoded as WZ frames. We compare the proposed AR model with the methods described in [3] and [9]. The parameter sets of the AR model within each test sequence are given in Table 1. The PSNRs of the SI, which are generated by the methods in [3], [9] and the proposed AR model for the pixels within set 2 are provided in Table 2.

Table 1 Parameters of the AR model for each test sequence

Sequence	Block Size	AR order
Foreman	8x8	2
News	8x8	1
Flower	16x16	1

In Table 2, the QP represents the quantization parameters of the reconstructed key frames which are encoded by H.264/AVC reference software jm 98. As shown in Table 2, the proposed AR model has a 0.9~1.8 dB performance improvement than [3] and a 0.2~1.4 dB performance improvement than [9]. Especially for *Flower*, the PSNR gains of the SI generated by the proposed AR model over those generated by [3] and [9] are up to 1.4 dB, when the QP of the K frames are set to be 26 and 28, respectively.

Table 2 PSNRs of the SI in Set 2

Sequence	QP=26			QP=28			QP=30		
	[3]	[9]	AR	[3]	[9]	AR	[3]	[9]	AR
Foreman	34.3	35.2	**35.4**	34.0	34.8	**35.1**	33.4	34.1	**34.3**
News	36.9	38.1	**38.7**	36.3	37.4	**37.8**	35.5	36.5	**36.7**
Flower	30.8	30.9	**32.3**	30.5	30.6	**32.0**	30.3	30.3	**31.6**

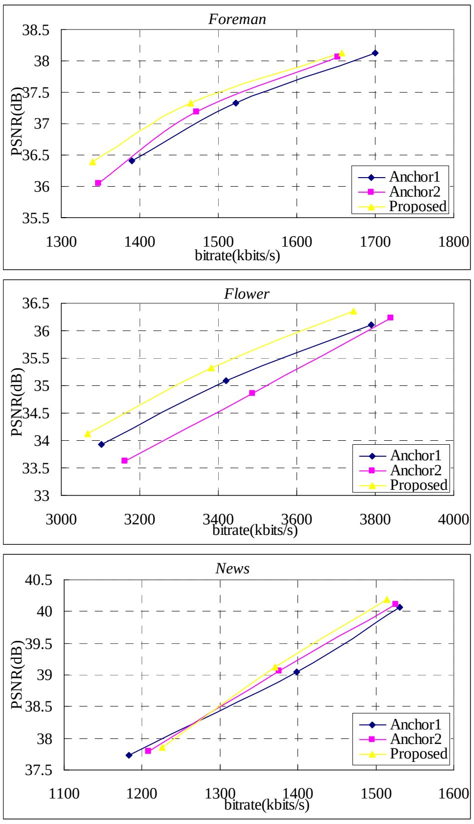

Fig. 5 Rate distortion curves comparisons for *Foreman*, *Flower* and *News*

Fig. 5 shows the average PSNR of the luminance component for both K frames and WZ frames versus the total bit-rate. We compare the proposed AR model with the approach in [3], shown as anchor 1 in Fig. 5, and the approach in [9], shown as anchor 2 in Fig. 5. It can be observed that [9] achieves better performance than [3] does for *Foreman* and *News* at higher bit rates. However, [9] has poor performance than [3] does for *Flower*. On the contrary, the proposed AR model achieves better performance than [3] and [9] for all the test sequences.

5. CONCLUSION

In this paper, a novel AR based SI generation method for block based checkboard pattern WZ coding is proposed. In the proposed method, the SI of each pixel in the second set is generated as the weighted summation of pixels within two square windows along the motion in the previous and following reconstructed WZ/K frames. To obtain accurate coefficients of the AR model for each block in set 2, the four neighboring reconstructed blocks in set 1 are employed to train the coefficients by a maximum likelihood least square algorithm. Experimental results show that the proposed AR model is able to achieve higher performance than traditional SI generation method.

ACKNOWLEGEMENT

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

6. 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] A. Aaron, S. Rane, E. Setton, and B. Girod, “Tansform-domain Wyner-Ziv codec for video,” in Proc. SPIE Visual Communications and Image Processing, San Jose, CA, Jan. 2004.

[4] A. Aaron, S. Rane and B. Girod, “Wyner-Ziv video coding with hash-based motion-compensation at the receiver,” in Proc. IEEE international conference on Image Processing, vol. 5, pp. 24-72, Oct. 2004.

[5] A.B.B. Adikari, W.A.C. Fernando, H. Kodikara Arachchi and W.A.R.J. Weerakkody, “Sequential motion estimation using luminance and chrominance information for distributed video coding of Wyner-Ziv frames,” in Electronics Letters, Volume 42, Issue 7, 30 March 2006 Page(s):398 – 399.

[6] J. Ascenso, C. Brites and F. Pereira, “Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding,” in 5th EURASIP Conf. on Speech and Image Processing, Multimedia Communications and Services, July 2005.

[7] J. Ascenso, C. Brites and F. Pereira, “Motion Compensated Refinement for Low Complexity Pixel Based Distributed Video Coding,” in AVSS 2005. IEEE Conference on Advanced Video and Signal Based Surveillance, 15-16 Sept. 2005 Page(s):593 – 598.

[8] M. Tagliasacchi, A. Trapanese, S. Tubaro, “Exploiting spatial redundancy in pixel domain Wyner-Ziv video coding,” in IEEE International Conference on Image Processing, Oct. 2006 Page(s):253 – 256.

[9] Hongbin Liu, Xiangyang Ji, Debin Zhao, Bo Wu, Wen Gao, “Distributed video coding using block based checkboard pattern splitting algorithm”, the 26th Picture Coding Symposium 2007(PCS2007), Lisbon, Portugal, Nov.2007.