Distributed Video Coding for Resource Critical Applocations

Huang, Xin; Forchhammer, Søren

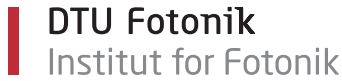
Publication date:
2009

Document Version
Publisher's PDF, also known as Version of record

DISTRIBUTED VIDEO CODING FOR RESOURCE CRITICAL APPLICATIONS

Xin Huang

30th April 2009
(revised, 16th September 2009)



DTU Fotonik
Technical University of Denmark

Preface

To my family

Abstract

In a number of emerging applications, it is required to have a low complexity video encoder in terms of physical size and power consumption. Distributed Video Coding (DVC) is such a video coding paradigm, which exploits the source statistic partially or totally at the decoder based on the availability of some decoder side information. Thereby computational power is shifted from encoder to decoder. In this work, one of the best available DVC codec, feedback channel based transform domain Wyner-Ziv video codec, is reviewed and implemented. Although Rate-Distortion (RD) performance of Wyner-Ziv video codec is promising, there is still a significant coding gap compared to conventional video codec like H.264/AVC. In order to further improve the RD performance of state-of-the-art Wyner-Ziv video codec, an Overlapped Block Motion Compensation (OBMC) based side information generation method, an improved virtual channel noise model and a novel multiple side information based Wyner-Ziv decoder are proposed. The proposed algorithms have clearly improved the coding efficiency of state-of-the-art Wyner-Ziv video codec. Therefore this work is a valuable contribution for designing future DVC codecs.

Resumé

I en række nye applikationer, ønskes en lav kompleksitet i video encodere i form af fysisk størrelse og strømforbrug. Distributed Video Coding (DVC) er et sådant video kodnings paradigme, der udnytter kildens statistik helt eller delvist på dekoder siden baseret på tilgængelig dekoder side-oplysninger. Derved flyttes kompleksitet fra kodeenhed til dekoder. I dette arbejde, et er DVC codec, et feedback kanal baseret frekvens domæne Wyner-Ziv video-codec, gennemgået og gennemført. Selv om Rate-Distortion (RD) af Wyner-Ziv video codec er lovende, er der en stadig betydelige kløft i forhold til konventionelle video codec som H.264/AVC. For yderligere at forbedre RD af state-of-the-art Wyner-Ziv video-codecs, er en overlappende blok bevægelseskompensation (OBMC) baseret side-oplysninger generations metode, en forbedret virtuel kanal støj model og flere side-oplysninger baseret Wyner-Ziv dekoder foreslået. De foreslåede algoritmer har klart forbedret kode effektivitet af state-of-the-art Wyner-Ziv video-codec. Derfor er dette arbejde er et værdifuldt bidrag til udformningen af fremtidige DVC codec.

Acknowledgements

First of all, I would to thank Professor Søren Forchhammer, supervisor of my PhD study. He gave me a great inspiration and patient guidance throughout the project. I am grateful to Professor Fernando Pereira, Assistant Professor João Ascenso and PhD student Catarina Brites from Instituto Superior Técnico - Instituto de Telecomunicações, Portugal. Thanks for their invaluable help during my visit. The most special thanks give to my family. I appreciate considerate support from my wife, Hui Diao. I am grateful in depth to my parents Chunping He and Zhizhong Huang for their infinite support throughout my years in Denmark.

Ph.D. Publications

The following publications have resulted from this Ph.D. project. The conference contributions [A], [B], [C] and [D] are reported in Appendix.

[A] X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer. "Distributed video coding with multiple side information", *Picture Coding Symposium (PCS)*, May 2009

[B] X. Huang and S. Forchhammer. "Improved virtual channel noise model for transform domain wyner-ziv video coding", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2009

[C] X. Huang and S. Forchhammer. "Improved side information generation for distributed video coding", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 223–228, Oct. 2008

[D] X. Huang, H. Li, and S. Forchhammer. "A multi-frame based post-processing approach to improve decoding of h.264/avc", *IEEE International Conference on Image Processing (ICIP)*, pp. 381–384, Sept. 2007

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Digital video coding is a vital element in many video applications today including high-definition TV, DVD, mobile video/TV (broadcasting), and video on demand etc. High efficient digital video coding paradigms, represented by ISO MPEG-x [1] and ITU-T H.26x [2] [3] standards, are based on a hybrid coding approach by combining Discrete Cosine Transform (DCT) and interframe predictive coding. In the hybrid coding framework, the encoder compress a video sequence by reducing the existing spatial and temporal redundancy, which requires a higher computational complexity because of motion estimation. The decoder reconstructs the video sequence simply by following the instruction of received information. Thus the complexity of the decoder is typically 5 to 10 times less than the encoder [4]. The asymmetric architecture in terms of complexity, typically having one complex encoder and many simpler decoders, is well-suited for broadcast or down-link applications where the video sequence is compressed once and decoded many times.

1.1 Motivation

In a number of emerging applications e.g. wireless video surveillance, wireless camera, mobile camera etc (as in Fig 1.1), the complex encoder is disadvantageous in terms of physical size and power consumption. The asymmetry of the conventional video coding paradigm should be reversed or balanced to have simple and efficient video encoders, but possibly highly complex decoders. The simple solution to perform a

video coding solution with low complexity encoder is to fully or partly remove motion estimation algorithm from conventional video coding, i.e. by using intraframe coding or predictive coding with "zero" motion estimation. However, it will degrade the coding efficiency compared with the conventional hybrid predictive video coding.

Figure 1.1: (Left) ordinary wireless camera and (right) wearable wireless webcam imitates surveillance cameras common in casinos and department stores [5]

The Slepian-Wolf [6] theorem proves that independent encoding but joint decoding of two statistically dependent signals cost the same rate as for typical joint encoding and decoding. The Wyner-Ziv theorem [7] extends the Slepian-Wolf theorem to the lossy case. It suggests that a novel video coding system, which encodes individual frames independently, but decodes them jointly, might achieve low complexity encoding with the similar coding efficiency as conventional hybrid predictive video coding. With the theoretical support, it becomes realistic to design a Distributed Video Coding (DVC) system [8], which encodes a video sequence requiring only intraframe processing computation power and decodes it by exploiting the statistical dependence between frames, thus demanding much more complex interframe processing computation power.

1.2 Objectives

In the literature, there are essentially two preliminary DVC architectures based on Slepian-Wolf and Wyner-Ziv theorems, which are feed-

back channel based frame level DVC [9] and an encoder side rate controller based block level DVC (PRISM) [10] [11]. Since practical efforts towards DVC solutions are just starting, the performance of these DVC paradigms have not yet reached the compression efficiency of the conventional hybrid predictive video coding paradigm, sometimes even worse than the Intra coding and the no motion estimation Inter coding.

This thesis is mainly focusing on the feedback channel based DVC, since it gives a better coding performance than PRISM [12]. The main objectives of this thesis are:

- Evaluate the solution of conventional hybrid predictive video coding structure with low complexity encoder (i.e. Intra coding and no motion estimation Inter coding), develop a postprocessing method to improve the quality of decoded sequences.
- Review and evaluate the architecture of state-of-the-art feedback channel based DVC codec, compare the performance with conventional hybrid predictive video coding.
- Develop some novel and efficient modules in state-of-the-art DVC codec. Improve the coding efficiency and reduce the performance gap when compared to conventional hybrid predictive video coding paradigm.

1.3 Main Contributions

The main contributions of this project are:

- A multi-frame based postprocessing method [13] is proposed to improve the quality of H.264/AVC coded sequences. The algorithm applies an adaptive filter along motion trajectories at the decoder side utilizing an estimated quality of the pixel on each trajectory. The improvements of the proposed postprocessing method are stable in a wide range, the Rate-Distortion (RD) gain is up to 0.6 dB for low motion sequences.
- One of the state-of-the-art distributed video coding approaches, transform domain Wyner-Ziv video codec, is implemented. The coding performance is comparable with the best available DVC

codec (executable DISCOVER codec [14]). The implemented codec is seen as a baseline to be combined with the subsequently proposed modules.

- An improved side information generation method [15] is proposed in DVC codec, which consists of an Y, U and V based variable block size motion estimation algorithm and an adaptive weighted Overlapped Block Motion Compensation (OBMC) method. With proposed algorithm, coding efficiency is improved up to 1 dB for DVC coded frames.

- A virtual channel noise model module is improved. The proposed method [16] utilizes cross band correlation and two different estimators to predict more accurate Laplacian parameter for noise modeling. Compared with best available noise model in [17], the improved noise model can improve coding efficiency up to 1 dB for DVC coded frames.

- A novel multiple side information based DVC decoder [18] is designed. The multiple side information frames are generated by interpolation and extrapolation, respectively. With multiple observations, the proposed decoder can select or combine the available side information estimations to decrease the amount of 'correlation noise' and thus to reduce misleading soft inputs. Compared with the single side information solution, the RD performance can be improved up to 0.4 dB for DVC coded frames.

1.4 Outline of the Thesis

This thesis mainly describes and develops the possible video coding solutions for encoding resource critical applications. The structure of this thesis is organized as follows: A brief introduction of H.264/AVC with Intra coding and no motion estimation Inter coding are given in Chapter 2. In order to improve the quality of decoded sequences, a multi-frame based postprocessing method is described. Chapter 3 starts by introducing the theory basis of DVC. As one approach to DVC, feedback channel based transform domain Wyner-Ziv video codec is described in detail afterwards.

In Chapter 4, an OBMC based side information generation method is proposed, which is compared with a number of different side information generation methods. Then the impact of different side information generation methods on DVC coding performance is discussed. Virtual channel noise models of DVC within different granularity levels are discussed in Chatper 5. Furthermore, an improved noise model is proposed to enhance the coding efficiency of DVC.

In Chapter 6, a novel multiple side information based DVC is described. Its coding performance is evaluated and compared with the single side information based DVC. Finally, the achievements of this thesis are summarized in Chapter 7. Possible directions for the future work are specified as well.

References to Chapter 1

[1] "Information technology: generic coding of moving pictures and associated audio (mpeg-2)", *ISO/IEC 13818*, 1995.

[2] "Video codec for audiovisual serivices at p x 64 kbit/s", *ITU-T Recommendation H.261*, March 1993.

[3] "Coding of audiovisual objects-part 10: Advanced video coding", *ISO/IEC 14496-10*, 2003.

[4] T. Wiegand, G. Sullivan, G. B. ntegaard, and A. Luthra. "Overview of the h.264/avc video coding standard", *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 13, no. 7, July 2003.

[5] F. Pereira, L. Torres, C. Guillemot, T. Ebrahimi, R. Leonardi, and S. Klomp. "Distributed video coding: Selecting the most promising application scenarios", *Signal Processing: Image Communication*, vol. 23, pp. 339–352, 2008.

[6] D. Slepian and J. Wolf. "Noiseless coding of correlated information sources", *IEEE Trans. Inform. Theory*, vol. 19, pp. 471–480, July 1973.

[7] A. Wyner and J. Ziv. "The rate-distortion function for source coding with side information at the decoder", *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, Jan 1976.

[8] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero. "Distributed video coding", *Proc. of IEEE, Special issue on advances in video coding and delivery*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[9] A. Aaron, S. Rane, E. Setton, and B. Girod. "Transform domain wyner-ziv codec for video", *Proc. SPIE VCIP*, pp. 520–528, Jan 2004.

[10] R. Puri and K. Ramchandran. "Prism: A new robust video coding architecutre based on distributed compression principles", *40th Allerton Conference on Communication, Control and Computing*, Oct. 2002.

[11] R. Puri and K. Ramchandran. "Prism: A video coding paradigm with motion estimation at the decoder", *IEEE Trans. on Image Process.*, vol. 16, no. 10, pp. 2436–2448, Oct. 2007.

[12] F. Pereira, C. Brites, J. Ascenso, and M. Tagliasacchi. "Wyner-ziv video coding: a review of the early architectures and further developments", *IEEE International Conference on Multimedia and Expo*, pp. 625–628, June 2008.

[13] X. Huang, H. Li, and S. Forchhammer. "A multi-frame based post-processing approach to improve decoding of h.264/avc", *IEEE International Conference on Image Processing (ICIP)*, pp. 381–384, Sept. 2007.

[14] DISCOVER Project: www.discoverdvc.org, Dec 2007.

[15] X. Huang and S. Forchhammer. "Improved side information generation for distributed video coding", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 223–228, Oct. 2008.

[16] X. Huang and S. Forchhammer. "Improved virtual channel noise model for transform domain wyner-ziv video coding", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2009.

[17] C. Brites and F. Pereira. "Correlation noise modelling for efficient pixel and transform domain wyner-ziv video coding", *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 18, no. 9, Sept. 2008.

[18] X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer. "Distributed video coding with multiple side information", *Picture Coding Symposium (PCS)*, May 2009.

Chapter 2

H.264/AVC with Low Complexity Encoder

Conventional video coding techniques such as H.264/AVC [1] are based on a hybrid predictive video coding structure. Each macroblock (block of size 16×16) is coded either in Intra mode or Inter mode. In Intra mode, predicted block is formed from the samples of current slice that have previously been encoded and reconstructed. In Inter mode, the predicted block is obtained by motion-compensated prediction from one or more reference frame(s). The predicted block is subtracted from current block to produce a residue, which is transformed, quantized and entropy coded afterwards. Generally, the compression performance of the Inter mode is more efficient than the Intra mode. However, motion estimation in Inter mode requires relevant high computation power at the encoder which is a limitation for some resource critical applications. In order to have a low complexity video encoding scheme, Intra mode and no motion estimation Inter mode in conventional video coding come as two natural solutions. However, if the decoder is kept unchanged, it will lose coding efficiency by removing high complexity motion estimation from the encoder. Post-processing schemes are usually utilized to improve the quality of the decoded sequences. Applying a post-processing scheme on Intra and no motion Inter coded sequences can be seen as an optional video coding solution matching low complexity encoder but relative high complexity decoder scenario.

The objective of this chapter is to introduce two low complexity video

encoding solutions, i.e. Intra coding and no motion Inter coding, which are simply derived from conventional video coding scheme H.264/AVC. Coding performance of these solutions are evaluated and compared. In order to improve the quality of decoded sequences, a multi-frame based post-processing scheme is proposed and applied on H.264/AVC Intra, H.264/AVC no motion Inter and H.264/AVC Inter coded sequences, respectively.

The structure of this chapter is organized as follows: H.264/AVC Intra coding is introduced in Section 2.1. In order to improve the coding performance of H.264/AVC Intra without introducing too much computation into encoder, H.264/AVC no motion estimation Inter coding (B picture) is described in Section 2.2. Then the multi-frame based post-processing scheme for H.264/AVC coded sequences is proposed in Section 2.3. Test conditions and corresponding results are presented in Section 2.4.

2.1 H.264/AVC Intra Coding

In H.264/AVC Intra mode, a predicted block is formed based on previously encoded and reconstructed blocks. For luminance components, there are two different types for Intra prediction, which are Intra_4×4 with nine prediction modes on 4×4 blocks and Intra_16×16 with four modes on 16×16 blocks. As shown in Fig. 2.1, a predicted 4×4 block is obtained by the means of the samples A...D and I...L in Intra_4×4 mode 2. The rest of the modes in type Intra_4×4 predict the 4×4 block by directional interpolation or extrapolation, i.e. vertical, horizontal, diagonal down-left, diagonal down-right, vertical-right, horizontal-down, vertical-left and horizontal-up, respectively. As an alternative to type Intra_4×4, type Intra_16×16 predicts the entire macroblock in one operation with four modes. The four modes are vertical extrapolation, horizontal extrapolation, DC prediction and plane prediction, respectively. Since the chrominance signals are very smooth in most cases, four modes intra prediction on each 8×8 block is performed in a similar way to Intra_16×16. A detailed description of all the Intra prediction modes can be found in [1]. The prediction mode which provides the minimized difference between predicted block and current block is selected. Then predicted residue is obtained by subtracting the predicted

block from the current block prior to transform coding, quantization and entropy coding.

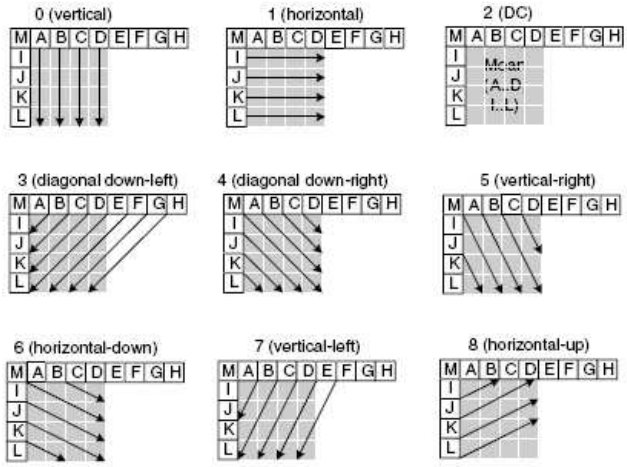


Figure 2.1: Intra 4×4 prediction modes for luminance components [2]

2.2 H.264/AVC Inter Coding without Motion Estimation

In the normal H.264/AVC Inter prediction mode, a predicted frame is formed from one or more previously encoded reference frames based on the block based motion estimation and motion compensation. Subsequently, the predicted frame is subtracted from the current frame to produce a motion compensated residue. The motion estimation in H.264/AVC supports a range of block size from 16×16, 16×8, 8×16 down to 8×8 for luminance samples. An 8×8 block in P-slice, may be further divided into partitions with block sizes of 8×4, 4×8, or 4×4 [3]. Two types of Inter predictions, P picture and B picture, are based on unidirectional motion compensation and bi-directional motion compensation, respectively. Bi-directional motion compensation (B picture) as depicted in Fig. 2.2 utilizes two super-imposed motion compensated sig-

nals from previous and next reference frames to improve the motion compensation accuracy and coding efficiency.

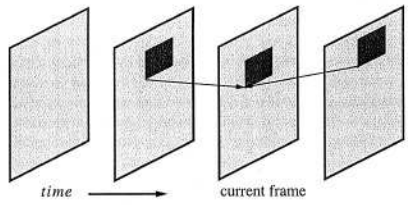


Figure 2.2: Bidirectional motion compensation [4]

In order to avoid complex motion estimation in H.264/AVC Inter mode, co-located blocks from reference frames are chosen as the matching blocks in a proposed H.264/AVC no motion Inter mode. Subtracting the average value of the matched blocks from current coded block, the co-located motion compensated residue is obtained. Then, according to the coding procedure of H.264/AVC, compensated residue is transform coded, quantized and entropy coded.

2.3 Post-Processing

Although it is possible to reduce the encoding complexity by removing the motion estimation from the encoder, this inevitably introduces some coding performance loss. Many postprocessing methods [5] [6] based on the video codec like MPEG2 and H.263 etc have been proved efficient on improving the quality of decoded sequences. Therefore, a multi-frame based post-processing scheme is proposed in this section to improve the quality of H.264/AVC decoded sequences. Moreover, this algorithm can also be applied onto H.264/AVC Intra and H.264/AVC no motion Inter coded sequences, which can be seen as an optional video coding solution with low complexity encoder but relative high complexity decoder for encoding resource critical applications.

The basic idea of the proposed post-processing scheme is to apply an adaptive filter along motion trajectories utilizing an estimated quality of the pixel on each trajectory. The process can be divided into quality evaluation, up-sampling and down-sampling as shown in Fig. 2.3. First,

the assumed quality of each pixel in the decoded sequence is estimated based on prediction type (I, B or P picture) and quantization information. Then, a superresolution version (quadruple resolution default) of each directly decoded picture is constructed through temporal and spatial upsampling. Finally, a quality based decimation filter is designed to improve the video quality.

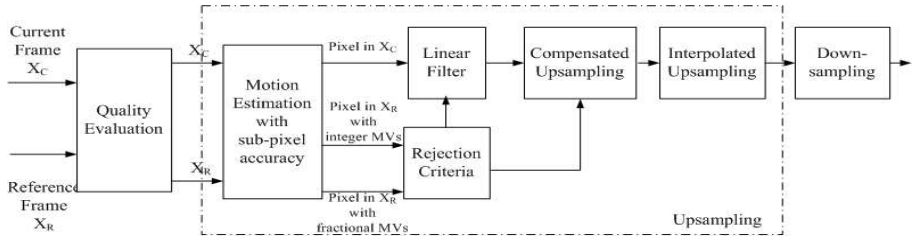


Figure 2.3: The procedure of post-processing scheme

2.3.1 Quality Evaluation

The degradation of a coded video sequence is mainly caused by coarse quantization and inaccurate motion compensation. Macroblocks with different Quantization Parameters (QP) and prediction types (i.e. I, P or B) may introduce different distortion. The Mean Squared Error (MSE) caused by the quantization depends on the distribution of transform coefficients. Due to the different prediction modes, Intra and Inter coded blocks may have different degradation. Based on different picture types, a quality parameter q is defined to reflect the MSE for pixels in I, P and B pictures approximately. The quality parameter is estimated through a functional curve which is obtained by testing the MSE of the luminance components of H.264/AVC decoded sequences. Fig. 2.4 indicates that Intra coded pictures (I) provide the best quality, and unidirectional prediction pictures (P) have better quality than bidirectional prediction pictures (B). These training data are only used to describe relative comparisons between the different coding modes, thus it is not an absolute measure. All the settings and testing in later experiments are based on these functional curves. With this quality parameter, it is feasible to combine pixels with the assumed better quality from neigh-

boring pictures to current picture, and to prevent poor quality pixels degrading better quality pixels.

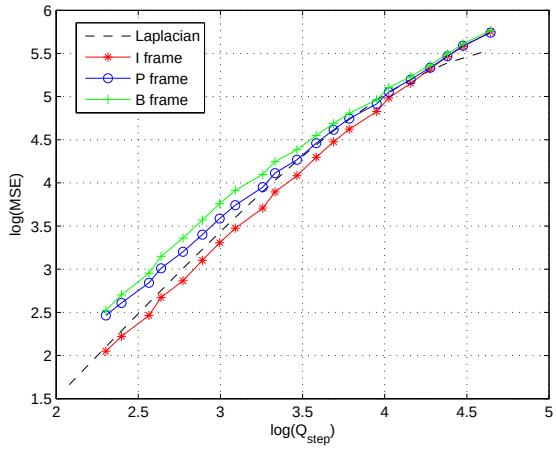


Figure 2.4: Functional curve measured with mobcal(CIF) [7]

2.3.2 Up-sampling

A superresolution picture (default has (V=4) times the number of pixels vertically and (H=4) times number of pixels horizontally) is formed using the information from the current picture and the N_f temporal neighboring pictures in an upsampling module. Compared with directly decoded picture, upsampled high resolution picture contains more information, which is helpful to remove noise and improve the quality of the decoded sequences. The upsampling module starts with sub-pixel accuracy motion estimation to align pixels in current picture with pixels in the reference pictures. The pixels from the reference pictures with integer motion vector are combined with decoded pixel in the current picture using a linear filter. The pixels from the reference pictures with fractional motion vector are motion compensated to corresponding locations in higher resolution pictures

• ***Sub-pixel Accuracy Motion Estimation*** : In order to obtain reliable and homogeneous motion pixels x_r from reference pictures, a hierarchical block-based ME is utilized. The initial searching block size

is set to be 16 × 16, then 4 sub-blocks (8 × 8). This final block size is a compromise between larger blocks for robustness and smaller blocks for accuracy as in [6]. The motion vectors are obtained by searching the best matching 8 × 8 block in reference pictures. It is denoted by $(m + \Delta m, n + \Delta n)$, where (m, n) is the integer part and $(\Delta m, \Delta n)$ is the fractional part of each motion vector. The fractional part is calculated by refining the best matching block in an interpolated sub-pixels $\hat{x}$ region. The interpolated sub-pixels $\hat{x}(m', n')$ are generated by a six tap filter and then a linear filter as in H.264/AVC. Assuming (m_r, n_r) is the absolute coordinates of the best matching pixel, x_r, with integer motion vectors in a reference picture, if interpolated pixels with relative displacement $(\Delta m, \Delta n)$ have minimum Sum of Absolute Difference (SAD) within 8 × 8 block, its corresponding best match pixels x_r with integer motion vectors are perceived as upsampling pixels at position $\big((m_r - m - \Delta m)V, (n_r - n - \Delta n)H\big)$. If more than one reference pixel map to the same position of the current superresolution picture, the pixel is assigned to be the reference pixel with best estimated quality above. If these reference pixels have an equal quality parameter, the superresolution pixel is assigned to be their weighted average.

• ***Linear Filter***: If the reference pixels with integer motion vectors have minimum SAD, they are defined to have the same trajectories with directly decoded pixels in the current picture. These pixels are combined in current superresolution picture by using a linear filter. The linear filter is only implemented if the reference pixels have better estimated quality parameters. Let x_c be a pixel in current decoded picture and x_r a trajectory pixel from a reference picture with integer motion vector. An estimated pixel with expected minimum MSE is obtained by:

$$\hat{x} = h_r x_r + h_c x_c \tag{2.1}$$

the coefficients h_r and h_c are estimated by solving the Wiener-Hopf equations in a training session

$$\begin{pmatrix} E\{X_r X_r\} & E\{X_r X_c\} \\ E\{X_c X_r\} & E\{X_c X_c\} \end{pmatrix} \begin{pmatrix} h_r \\ h_c \end{pmatrix} = \begin{pmatrix} E\{X X_r\} \\ E\{X X_c\} \end{pmatrix} \tag{2.2}$$

where X_r and X_c represent stochastic variables of pixel values in the reference picture and the current picture respectively. X represents a

stochastic variable of original pixel values at the same position in original resolution picture. In order to preserve the mean value, coefficients of this filter should be computed under the constraint $h_r + h_c = 1$. Given enough training data, the second-order mean value in (2.2) should be conditioned on quality of x_r and x_c, the coefficients h_r and h_c are described as [6]:

$$h_r = 1 - (1 - \alpha)^{(q_c/q_r)^\beta} \tag{2.3}$$

$$h_c = 1 - h_r \tag{2.4}$$

the parameter α specifies the *a priori* weight that x_r should carry. The parameter β specifies how much the difference in qualities of x_r and x_c should influence the estimated pixel value. Equation (2.3) is monotonically increasing in ratio q_c/q_r from 0 to 1 and it has the property that for $0 \leq \alpha \leq 1$, $\alpha \geq 0$, $q_r, q_c \geq 0$ and $0 \leq h_r \leq 1$. α, β of the filter (Eq. 2.3) can be estimated by using many frames of different sequences based on Eq. 2.2, (See Fig. 2.5), the curve yields $\alpha = 0.15$ and $\beta = 0.7$. Once this filter is operated on the current picture pixels and reference

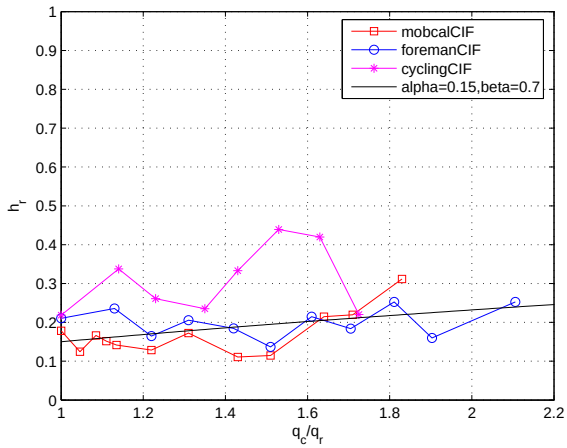


Figure 2.5: Filter coefficient h_r as a function of q_c/q_r [7]

picture pixels, estimated pixels in superresolution picture are assigned with a new quality parameter value as:

$$\hat{q} = h_r q_r + h_c q_c \tag{2.5}$$

• ***Rejection Criteria***: Since block-based motion estimation is not sufficient to guarantee the best match pixels according to true motion. It might introduce errors e.g., at occlusions in the motion compensation process. In order to reduce the risk of errors, a rejection criteria is used for the evaluation of each pixel x_r whether it should be placed in the superresolution picture. The evaluation is based on an intra-prediction as in JPEG-LS [8]

$$\hat{x}_{intra} = \begin{cases} min(a,b) & \text{if } c \geq max(a,b) \\ max(a,b) & \text{if } c \leq min(a,b) \\ a+b-c & \text{otherwise} \end{cases} \tag{2.6}$$

where a, b and c denote the pixel at the left, top and top-left of pixel x_c respectively. These intra-predicted pixels are compared with the best match pixels based on SAD. The pixels x_r with larger SAD over a 8 × 8 block will be rejected.

• ***Interpolated Upsampling***: After the compensated upsampling, an unfinished superresolution picture is formed. In order to complete the current superresolution picture, spatial interpolation is employed to fill the holes left by the compensated upsampling. Cubic spatial interpolation is based on rectangular lattice samples, which can supply true continuity among each segment and produce less jaggy edges. In order to utilize the irregular samples generated by the compensated upsampling, the cubic interpolation method is improved by adding an irregular sample detection process. If there are no irregular samples in the nearest 4×4 pixel region, a normal cubic interpolation is implemented . Otherwise, a modified version is used:

$$\begin{aligned} x_{intp}(m',n') = &\textstyle\sum_i \sum_j x_{re}(i,j) K_1 \beta^3(|m'-i|)\beta^3(|n'-j|) \\ &+ \textstyle\sum_a \sum_b x_{ir}(a,b) K_2 \beta^3(|m'-a|)\beta^3(|n'-b|) \end{aligned} \tag{2.7}$$

where K_1 and K_2 are normalization coefficients,$x_{re}(i,j)$ and $x_{ir}(a,b)$ represent samples at regular and irregular positions respectively, $\beta^3(z)$ is a typical cubic convolution kernel [9]:

$$\beta^3(z) = \begin{cases} \frac{3}{2}|z|^3 - \frac{5}{2}|z|^2 + 1 & \text{if } 0 \leq |z| \leq 1 \\ -\frac{1}{2}|z|^3 + \frac{5}{2}|z|^2 - 4|z| + 2 & \text{if } 1 \leq |z| \leq 2 \\ 0 & \text{if } 2 \leq |z| \end{cases} \tag{2.8}$$

2.3.3 Down-sampling

A superresolution picture for each directly decoded picture is formed after upsampling. In order to improve the quality of decoded frame and get the desired picture resolution, a down-sampling scheme is proposed by applying quality based spatial filter. In order to reduce the risk of blurring edges in the decimation process, the decimation filter is operated in a small 9 × 9 window. A two-dimensional spatial linear filter combined with adaptive quality weights is applied in the vicinity of each sample position (m_0, n_0) to obtain a lower resolution picture.

$$p_l(m_0', n_0') = \sum_{m,n} g(m, n, m_0, n_0) p_h(m, n) = \sum_{m,n} K g_v(|m - m_0| g_h(|n - n_0|) w(m, n) p_h(m, n) \tag{2.9}$$

where $p_l(m_0', n_0')$ represents a downsampled pixel in the lower resolution picture, $p_h(m, n)$ represent the pixels which are adjacent to sample pixel $p_h(m_0, n_0)$ in the superresolution picture. K is normalizing factor $(\sum_g = 1)$. g_v and g_h are 1-D symmetric filters on vertical and horizontal direction respectively. $w(m, n)$ is weight function for each pixel based on its corresponding quality parameter. The 1-D symmetric filters g_v and g_h reflecting the spatial distance are defined [6]:

$$g_2 = (\ldots, 0, a, 1, a, 0, \ldots) \tag{2.10}$$

$$g_4 = g_2 * g_2 = (\ldots, a^2, 2a, 1 + 2a^2, 2a, a^2, \ldots) \tag{2.11}$$

$$g_v = g_h = g_4 * g_4 \tag{2.12}$$

Furthermore, the value of a should be adaptive depending on local characteristics (smooth or texture). Therefore, we calculate standard deviation σ of each downsampling sample $p_h(m_0, n_0)$ within 9 × 9 window to adaptive control a value:

$$a = \begin{cases} 1, & \text{if } \sigma \leq 10 \\ 0.5, & \text{otherwise} \end{cases} \tag{2.13}$$

$w(m, n)$ is a weight function reflecting the qualities of different kinds of pixels. It depends on whether $p_h(m, n)$ and $p_h(m_0, n_0)$ are compensated

upsampling pixels (p_{cu}) or interpolated upsampling pixels (p_{iu}). If both of them are compensated upsampling pixels, their quality parameters are used to determine the weight of $p_h(m,n)$. If one of them is obtained by interpolation, a constant weight value is assigned [6]:

$$w(m,n) = \begin{cases} \frac{w_0}{\gamma}\gamma^{\,q(m,n)/q(m_0,n_0)}, \\ \qquad p_h(m,n), p_h(m_0,n_0) \in p_{cu} \\ 1, \quad p_h(m,n) \in p_{iu}, p_h(m_0,n_0) \in p_{cu} \\ w_0, \ p_h(m,n) \in p_{cu}, p_h(m_0,n_0) \in p_{iu} \end{cases} \tag{2.14}$$

where the parameter w_0 (set to 6) specifies the *a priori* worth of a compensated upsampling (p_{cu}) pixel compared to an interpolated pixel (p_{iu}). The parameter γ (set to 0.3) is a global parameter reflecting the influence introduced by quality ratio.

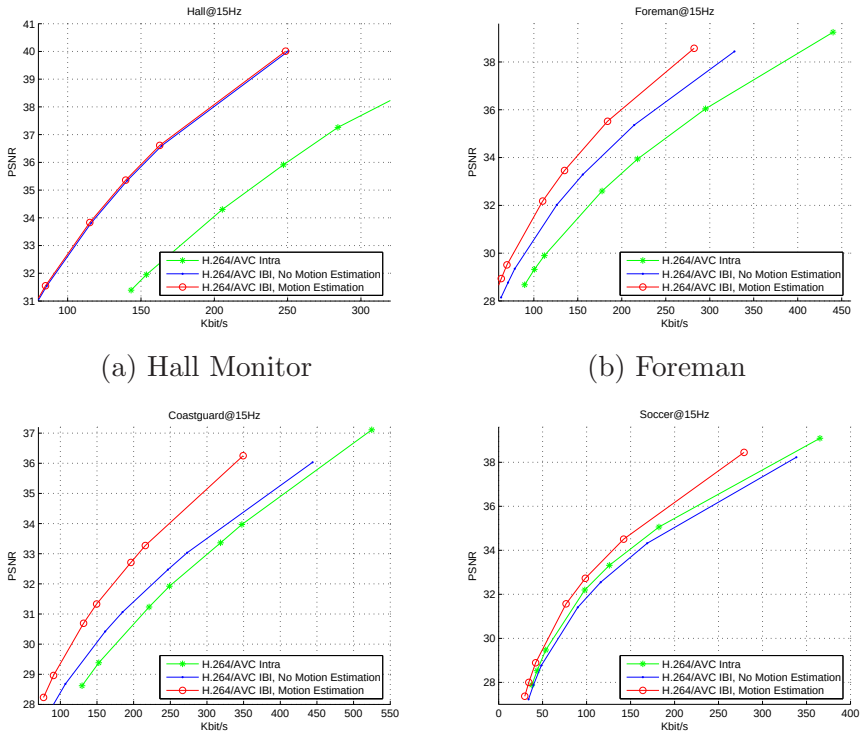
2.4 Experimental Results

The RD performances of H.264/AVC with Intra coding mode, no motion Inter coding mode and bidirectional motion estimation (B picture) based Inter coding mode are compared in Fig. 2.6. The detail settings of H.264/AVC reference codec [10] are reported in Appendix B.

Generally, the motion estimation based Inter coding mode outperform the Intra coding mode and the no motion Inter coding mode. The performance of the no motion estimation Inter coding mode is better than the Intra coding mode for video sequences with low and medium motion, but worse for high motion sequences. For low motion sequence like *Hall Monitor*, due to the dominated static background, the coding performances of Inter coding modes (both with and without motion estimation) are much more efficient than the Intra coding mode, the gain is up to 4 dB for overall RD performance. Meanwhile, the differences between the motion estimation Inter coding mode and the no motion Inter coding mode are less than 0.02dB. It indicates that the coding performance will be not degraded by removing the motion estimation from the encoder for static dominated sequences.

However, along with more motion being included as in sequences *Coastguard* and *Foreman*, the temporal residue caused by co-located block prediction become larger and larger. Therefore, the performance gap between the Inter coding mode and the no motion Inter coding mode

starts to increase. Meanwhile, the coding gain of no motion Inter coding mode keeps decreasing but still better than the Intra coding mode. For high motion sequence like *Soccer*, simply utilizing the co-located blocks for Inter prediction is not efficient for reducing the temporal redundancy, thus the performance of the no motion Inter coding mode becomes quite close to Intra coding mode. While the motion estimation Inter mode is always efficient to give the best coding performance for different sequences.



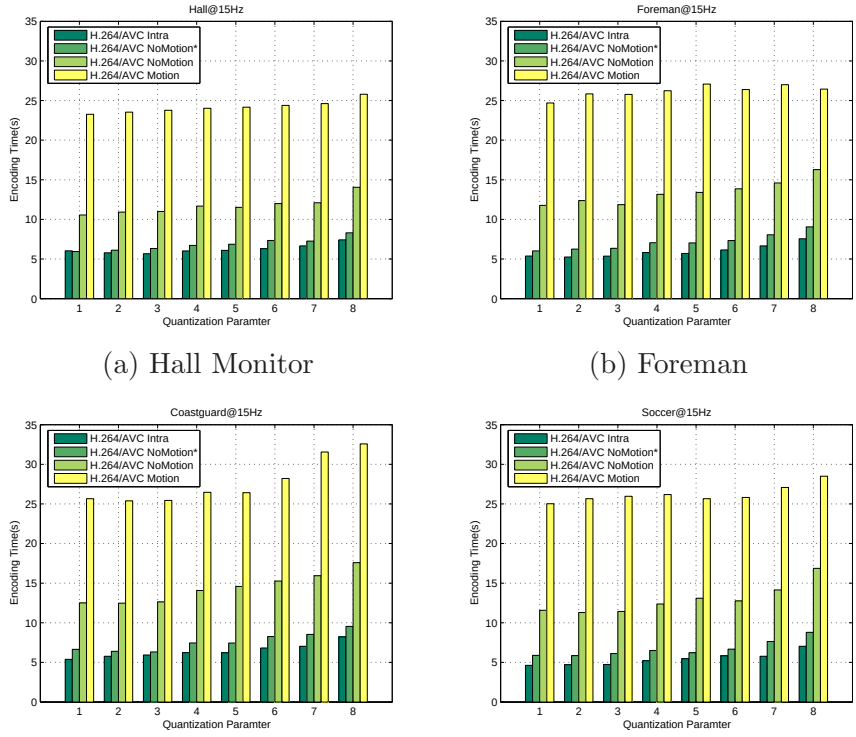
(a) Hall Monitor (b) Foreman

(c) Coastguard (d) Soccer

Figure 2.6: Coding performance of H.264/AVC with low complexity encoder

On the other hand, encoding complexity of H.264/AVC Intra mode, Inter mode and no motion Inter mode are evaluated. The complexity is measured by means of the encoding time of the full sequence on a 3

GHz PC (for relevant Intra, Inter or no motion Inter mode frames only). As shown in Fig. 2.7, H.264/AVC Inter mode always requires the most computation power from the encoder, while H.264/AVC Intra requires the least. Taking both coding efficiency and encoding complexity into account, H.264/AVC no motion Inter mode could be a good balance between the coding efficiency and the encoder complexity if the encoding resource is not very critical.



(a) Hall Monitor (b) Foreman

(c) Coastguard (d) Soccer

Figure 2.7: Encoding complexity comparison between H.264/AVC Intra mode, H.264 no motion Inter mode and H.264 motion Inter mode

In order to evaluate the performance of the proposed post-processing

[0]Without sacrificing the coding efficiency, the encoding complexity of H.264/AVC no motion Inter mode can be optimized (as described in Fig. 2.7, "H.264/AVC NoMotion*") by removing Inter mode decision.

scheme, average Peak Signal-to-Noise Ratio (PSNR) over all the frames of a sequence is used to evaluate the quality of a sequences. Post-processing method is applied on H.264/AVC Intra, H.264/AVC no motion Inter and H.264/AVC Inter coded sequences, respectively. The number of reference frames N_f in post-processing algorithm is set to 5. According to the results in Tables 2.1- 2.3, it is clear that the proposed post-processing algorithm generally improves quality of Intra, no motion Inter and Inter coded sequences.

For low motion sequence like *Hall Monitor*, more temporal dependency can be utilized, therefore post-processing method achieves the most significant gains (up to 0.6 dB). For high motion sequences like *Foreman*, *Coastguard* and *Soccer*, it becomes more difficult to use the temporal correlations at the decoder, thus the gains are not as much as the low motion sequence. Meanwhile, the post-processing scheme has better performance on the decoded sequences with relatively big QP compared to the one with small QP, because the low pass based post-processing algorithm introduces higher risk to oversmooth the high frequency content. For instance, post-processing on sequence *Coastguard* with QP 30 decreases the PSNR value due to over-smoothing effects.

Sequence	QP	Intra (dB)	Postprocessing (dB)	Δ (dB)
Foreman	29	36.04	36.23	+0.19
	34	32.60	32.89	+0.29
	39	29.32	29.68	+0.36
Hall Monitor	29	37.26	37.48	+0.22
	33	34.30	34.81	+0.51
	36	31.95	32.48	+0.53
Coastguard	30	33.97	33.83	-0.14
	34	31.23	31.30	+0.07
	38	28.62	28.75	+0.13
Soccer	31	35.06	35.04	-0.02
	36	32.20	32.26	+0.06
	43	28.53	28.57	+0.04

Table 2.1: Post-processing algorithm evaluation on H.264/AVC Intra coded sequences

Sequence	QP	Intra + No Motion Inter (dB)	Post -processing (dB)	Δ (dB)
Foreman	29	35.36	35.56	+0.20
	34	32.02	32.34	+0.32
	39	29.17	29.51	+0.34
Hall Monitor	29	36.60	36.92	+0.32
	33	33.82	34.35	+0.53
	36	31.54	32.09	+0.55
Coastguard	30	33.03	32.97	-0.06
	34	30.42	30.58	+0.16
	38	27.94	28.12	+0.18
Soccer	31	34.32	34.34	+0.02
	36	31.41	31.50	+0.09
	43	27.88	27.94	+0.06

Table 2.2: Post-processing algorithm evaluation on H.264/AVC no motion Inter coded sequences

2.5 Summary

In this chapter, H.264/AVC Intra coding mode and no motion Inter coding mode are introduced as two optional low complexity encoding solutions. Generally speaking, H.264/AVC no motion Inter coding mode gives better coding performance than Intra coding mode for relative low and medium motion sequences, the gain is up to 2 dB. However, H.264/AVC no motion Inter requires also more the encoding complexity and larger frame buffer, which may not fulfil some critical applications with extreme low complexity encoder.

Compared with H.264/AVC Inter mode, the coding efficiency of H.264/AVC Intra and H.264/AVC no motion Inter modes are degraded significantly especially for some high motion sequences. Therefore, a multi-frame based post-processing scheme is applied to improve the quality of decoded sequences and corresponding RD performances. With the proposed post-processing algorithm, the video quality can be improved up to 0.6 dB. Applying the post-processing algorithm on the H.264/AVC Intra or H.264/AVC no motion Inter coded sequences can be seen as an

Sequence	QP	Intra + Inter (B frame) (dB)	Post -processing (dB)	Δ (dB)
Foreman	29	35.52	35.67	+0.15
	34	32.18	32.57	+0.39
	39	28.94	29.29	+0.35
Hall Monitor	29	36.61	36.95	+0.34
	33	33.83	34.41	+0.58
	36	31.55	32.09	+0.54
Coastguard	30	33.27	33.16	-0.11
	34	30.69	30.77	+0.08
	38	28.23	28.32	+0.09
Soccer	31	32.73	32.72	-0.01
	36	31.57	31.65	+0.08
	43	28.00	28.06	+0.06

Table 2.3: Post-processing algorithm evaluation on H.264/AVC motion Inter coded sequences

optional video coding solution with low complexity encoder but relative high complexity decoder. However, this solution is not very competitive both in the aspects of coding efficiency and encoding complexity. Therefore, it make sense to explore the other efficient video coding solutions with low complexity encoder.

References to Chapter 2

[1] "Coding of audiovisual objects-part 10: Advanced video coding", *ISO/IEC 14496-10*, 2003.

[2] I. Richardson. *H.264 and MPEG-4 Video Compression.* John Wiley & Sons Ltd, 2003.

[3] T. Wedi and H. Musmann. "Motion- and aliasing-compensated prediction for hybrid video coding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 577–587, July 2003.

[4] M. Flierl and B. Girod. *Video Coding with Superimposed Motion-Compensated Signals, Application to H.264 and Beyond.* Kluwer Academic, 2004. ISBN 1-4020-7765-3.

[5] Y.Nie, H.S.Kong, A. Vetro, and K. Barner. "Fast adaptive fuzzy post-filtering for coding artifacts removal in interlaced video", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 993–996, March 2005.

[6] B. Martins and S. Forchhammer. "A unified approach to restoration, deinterlacing and resolution enhancement in decoding mpeg-2 video", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 803–811, Sept 2002.

[7] X. Huang, H. Li, and S. Forchhammer. "A multi-frame based post-processing approach to improve decoding of h.264/avc", *IEEE International Conference on Image Processing (ICIP)*, pp. 381–384, Sept. 2007.

[8] "Lossless and near-lossless coding of continuous tone still images", *JPEG-LS IS 14 495-1*, 1998.

[9] E. Meijering and M. Unser. "A note on cubic convolution interpolation", *IEEE Trans. Image Process.*, vol. 12, pp. 477–479, April 2003.

[10] *Joint Video Team (JVT) Reference Software.*, Available on: http://iphome.hhi.de/suehring/tml/index.htm.

Chapter 3

Distributed Video Coding

Distributed Video Coding (DVC) [1] [2] is a new video coding paradigm which fully or partly exploits the video redundancy at the decoder and not anymore at the encoder as in the predictive video coding, thereby shift computation power from encoder to decoder. According to the Slepian-Wolf theorem [3], it is possible to achieve the same rate by independently encoding but jointly decoding two statistically dependent signals as for typical joint encoding and decoding (with a vanishing error probability). The Wyner-Ziv theorem [4] extends the Slepian-Wolf theorem to the lossy case. It becomes the key theoretical basis for DVC where source is lossy coded based on the availability of some correlated source at the decoder from which the so-called side information is derived.

With the theoretical doors opened, it becomes more realistic to design a practical DVC codec. The objective of this chapter is to review relevant practical implementations of DVC in literature. One of the most popular DVC codec, i.e. feed back channel based transform domain Wyner-Ziv video codec, is described in detail. In order to evaluate the coding efficiency of Wyner-Ziv video codec, RD performance on a set of testing sequences are compared with existing low complexity encoding solutions H.264/AVC Intra and no motion Inter coding. Meanwhile, the best available released DVC codec [5] is used as a benchmark to verify our implementation.

The structure of this chapter is organized as follows: The theory background of DVC is described in Section 3.1. The relevant DVC ar-

chitectures are briefly introduced in Section 3.2. The practical DVC codec with the best available coding performance in literature is described in Section 3.3. In Section 3.4 test conditions are presented first, then the coding efficiency and the encoding complexity of the described DVC codec are evaluated.

3.1 Information Theory Background

Distributed source coding refers to separate encoding and joint decoding of mutually correlated sources. In information theory [6], it is known that a rate $R > H(X)$ is sufficient to encode and decode a source X. Similarly, a rate $R > H(X,Y)$ is sufficient if two correlated sources (X,Y) are jointly encoded and decoded. But what if the X source and the Y source are encoded separately but decoded jointly? By separate encoding X and Y, it seems natural that a rate $R > H(X) + H(Y)$ is sufficient. However, Slepian and Wolf [3] proved that a total rate $R = H(X,Y)$ is sufficient for two correlated sources (X,Y) which are independently and identically distributed (i.i.d). Let R_X and R_Y represent the required rate for coding X source and Y source, respectively. As described in the Slepian-Wolf theorem, for distributed source coding problem, the achievable rate region is given by [3] (See also Fig. 3.1) :

$$R_X \geq H(X|Y) \tag{3.1}$$

$$R_Y \geq H(Y|X) \tag{3.2}$$

$$R_X + R_Y \geq H(X,Y) \tag{3.3}$$

where $H(X|Y)$ is the conditional entropy of X given Y and $H(X,Y)$ is the joint entropy of correlated source (X,Y)

According to the corner points $(H(X|Y), H(Y))$ of the rate region in Slepian-Wolf coding as shown in Fig. 3.1, the coding rate $R_Y = H(Y)$ and $R_X = H(X|Y)$ can be achieved by separately encoding and jointly decoding of correlated sources (X,Y). The corner points $(H(X|Y), H(Y))$ presents a particular case (as shown in Fig. 3.2) which deals with the lossless source coding of X considering source Y as side information available at decoder side only.

Wyner and Ziv [4] have studied the Rate-Distortion function $(R(d))$ of this problem in lossy way. Mathematically, the Wyner-Ziv theorem

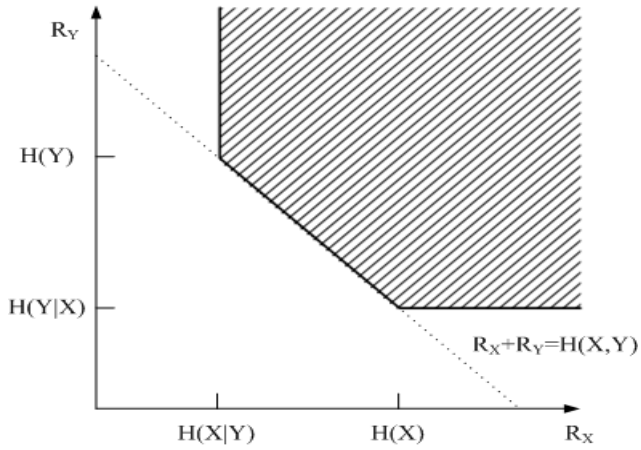


Figure 3.1: Rate region for Slepian-Wolf theorem

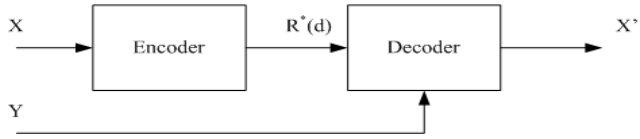


Figure 3.2: Rate distortion with decoder side information [6]

can be described as:

$$R^*(d) \geq R_{X|Y}(d), d \geq 0 \tag{3.4}$$

where $R^*(d)$ represents the minimum rates to encode X within distortion d when side information Y is available at decoder only. $R_{X|Y}(d)$ represents the minimum rates to encode X within distortion d when side information Y is available both at encoder and decoder. When the distortion $d = 0$, the Slepian-Wolf result, i.e. $R^*(0) = R_{X|Y}(0) = H(X|Y)$ is obtained. The Wyner-Ziv theorem [4] extends the Slepian-Wolf theorem to the lossy case, which is well-suited to video coding scenario. Therefore, it becomes the key theoretical basis for DVC where some source (X) is lossy coded based on the availability of the side information (Y) at the decoder.

3.2 Distributed Video Coding Implementations

Although the theoretical foundation for DVC was established in 1970s, the practical DVC codecs were developed around 2002 following important developments in channel coding technology [7]. So far, there are essentially two practical distributed video coding schemes available in the literature, which are pioneered by groups at Berkeley [8] [9] and Stanford [2] [10] respectively.

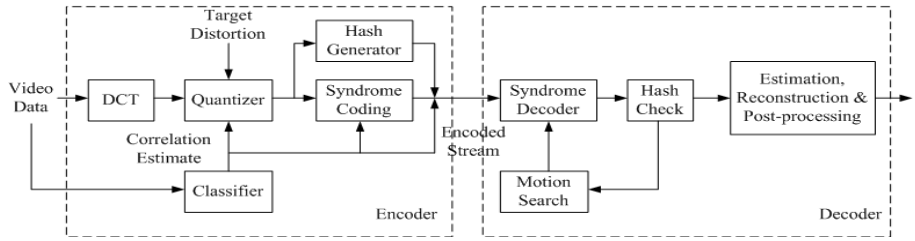


Figure 3.3: PRISM video coding architecture [9]

The Berkeley coding structure named as PRISM (Power-efficient, Robust, hIgh compression Syndrome based Multimedia coding) is shown in Fig 3.3. PRISM video codec is working at block level and characterized by an encoder side rate controller. Each block of current frame is classified into skip class (no coding), Intra coding class and syndrome coding class depending on the estimated temporal correlation [9]. In syndrome coding class, it is assumed that the most significant bits can be predicted from the side information, therefore only the least significant bits of the quantized transformed coefficients in a block are encoded using standard entropy coding principles or a coset channel code. For more details, please refer to [9].

Different from the block level coding and encoder side rate control as in PRISM codec, the Stanford coding structure is working at frame level and characterized by a feedback channel based decoder rate control scheme as shown in Fig. 3.4. The best available distributed video codec based on Stanford architecture is released by European project DISCOVER [5]. Compared with PRISM codec released by Berkeley [9], the RD performance gain of Stanford architecture is significant [7]. There-

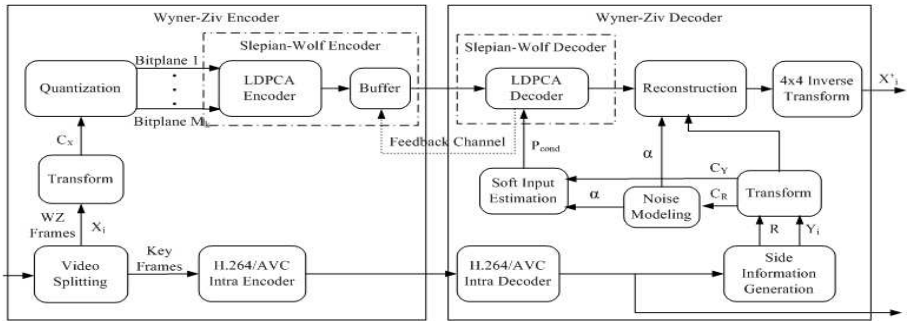


Figure 3.4: Feedback channel based transform domain Wyner-Ziv video codec architecture

fore, the Stanford architecture based DVC becomes one of the most popular solutions in research community. This thesis is focusing on the Stanford architecture based DVC, more details are described in following section.

3.3 Feedback Channel Based Transform Domain Wyner-Ziv Video Coding

Feedback channel based transform domain Wyner-Ziv video coding is one approach to DVC, which was first proposed in [10] by Stanford group, and then improved by many researchers, among others those in the DISCOVER project [5]. The architecture of transform domain Wyner-Ziv video codec is described in Fig. 3.4. In a nutshell, the encoding procedure follows:

1. A fixed Group of Pictures (GOP=N) is adopted to split video sequences into two kinds of frames, i.e. Key frames and Wyner-Ziv frames. Periodically one frame out of N in the video sequence is named as key frame and intermediate frames are WZ frames. The key frames are Intra coded by using a conventional video coding solution such as H.264/AVC Intra [11] while the Wyner-Ziv frames are coded using a Wyner-Ziv video coding approach.

2. Each Wyner-Ziv frame X_i are partitioned into non-overlapped 4×4 blocks and an integer transform [11] is applied to each of them.

3. The transform coefficients within a given band $b_k, k \in \{0...15\}$, are grouped together and then quantized. DC coefficients are uniformly scalar quantized and AC coefficients are dead zone quantized, respectively. Please see the details of quantization in Section 3.3.2.

4. After quantization, the coefficients are binarized. The binary bits with the same significance are formed to a bitplane, which is given to a rate compatible Low Density Parity Check Accumulate (LDPCA) encoder [12]. Starting from the most significant bitplane, each bitplane is independently encoded by the LDPCA encoder, the corresponding accumulated syndrome is stored in a buffer together with an 8-bit Cyclic Redundancy Check (CRC) [13]. The amount of transmitted bits depends on the requests made by the decoder through a feedback channel. More details about the LPDCA encoder is introduced in Section 3.3.3.

The decoding procedure is described as follows:

1. A side information frame Y_i and its corresponding noise residual frame R are created in side information generation module by using previously decoded frames. The side information frame Y_i is seen as a 'noise' version of the encoded Wyner-Ziv frame X_i, the estimated noise residual frame R is utilized to express the correlation noise between the Wyner-Ziv frame X_i and the side information frame Y_i. Different side information generation methods are discussed in Section 3.3.4 and Chapter 4.

2. The estimated noise residual frame R and side information frame Y undergo the integer transform to obtain the coefficients C_R and C_Y. Taking C_R and C_Y as inputs of a noise model module, the noise distribution between corresponding frequency bands of the side information frame Y_i and the Wyner-Ziv frame X_i is modeled. The general procedure of noise model module is described in Section 3.3.5. Different noise models are introduced and evaluated in Chapter 5.

3. Using a modeled noise distribution, the coefficient values of the side information frame C_Y and the previous successfully decoded

bitplanes, soft-input P_{cond} (conditional bit probabilities) for each bitplane is calculated. The calculation procedure is described in Section 3.3.6.

4. With the obtained soft-input P_{cond}, the LDPCA decoder starts to process various bitplanes to correct bit errors. Convergence is tested by the 8-bit CRC sum and the Hamming distance. The Hamming distance is the difference between the received syndrome and the one obtained from the decoded bitplane. For more details please refer to 3.3.7.

5. After successfully LDPCA decoding, the obtained bitplanes are grouped together to form a set of decoded quantization symbols for each band b_k. With the received quantization information, the decoded quantized symbols are used to calculate the correct intervals in which the Wyner-Ziv coefficients are located. Together with side information coefficients C_Y, noise distribution parameter α and the interval information, decoded coefficients within band b_k of the Wyner-Ziv frame are reconstructed. The reconstruction algorithm is described in Section 3.3.8.

6. After all the coefficients bands are reconstructed, 4×4 block inverse transform is performed to obtain the reconstructed Wyner-Ziv frame X'_i.

In the following subsections of this chapter, each module of the feedback channel based transform domain Wyner-Ziv video codec is described in detail.

3.3.1 Transform

In order to remove the spatial redundancy between neighboring pixels, transform coding is employed in Wyner-Ziv video coding. As in [11], the 4×4 block integer transform coding is applied to all 4×4 non-overlapping blocks of a Wyner-Ziv frame. The 4×4 transform matrix H is defined as [14]:

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}. \tag{3.5}$$

The corresponding inverse transform matrix $\tilde{H}_I$ is defined as [14]:

$$\tilde{H}_I = \begin{pmatrix} 1 & 1 & 1 & \frac{1}{2} \\ 1 & \frac{1}{2} & -1 & -1 \\ 1 & -\frac{1}{2} & -1 & 1 \\ 1 & -1 & 1 & -\frac{1}{2} \end{pmatrix}. \tag{3.6}$$

where the tilde indicates that $\tilde{H}_I$ is a scaled inverse of the transform matrix H satisfied:

$$\tilde{H}_I D H = I \tag{3.7}$$

$$D = \begin{pmatrix} \frac{1}{4} & 0 & 0 & 0 \\ 0 & \frac{1}{5} & 0 & 0 \\ 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & \frac{1}{5} \end{pmatrix} \tag{3.8}$$

where I is the identity matrix. The multiplications by $1/2$ in Eq. 3.6 can be implemented by sign-preserving 1-bit right shifts [14]. The transform and the inverse transform coding of an 4×4 block is given by:

$$C_X = H X H^T \tag{3.9}$$

$$X = \tilde{H}_I D C_X D^T \tilde{H}_I^T \tag{3.10}$$

3.3.2 Quantization

After applying the transform coding on each 4×4 block of Wyner-Ziv frame, decorrelated coefficients within 16 different frequency bands $b_k, k \in \{0...15\}$ are obtained. The coefficients in band b_0 contains the lowest frequency information of one 4×4 sample block, which is called DC coefficient. The remaining 15 coefficients in the 4×4 block containing higher frequency information are named as AC coefficients. Each band b_k is quantized with a predefined number of levels ($2^{M_{b_k}}$) as shown in Fig. 3.5 depending on the target quality of the Wyner-Ziv frame. The $2^{M_{b_k}} \in \{0, 4, 8, 16, 32, 64, 128\}$ indicates the number of quantization levels associated to the coefficients band b_k, $2^{M_{b_k}} = 0$ means that no bits are sent for coefficient band b_k and the corresponding side information within this band is directly used for reconstruction.

Since DC coefficients are not negative values, while AC coefficients can either be negative or positive values. DC and AC coefficients are

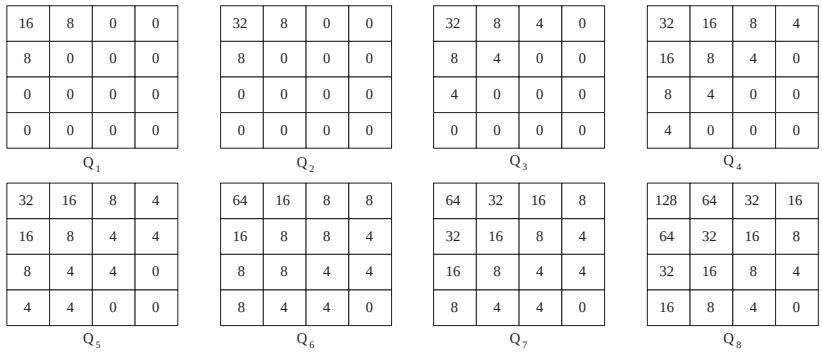


Figure 3.5: Eight quantization matrices regarding to different RD performances [15]

quantized in different ways. DC coefficients are uniform scalar quantized within the data range $[0, 1024]$. The upper bound (1024) of the DC coefficient range is obtained since the transform coding is applied on the 8-bit accuracy data (i.e. from 0 to 255) within 4×4 block. Thus, the quantization step size for DC coefficients is calculated as:

$$Q_{step}^{DC} = \lceil 1024/2^{M_{b_0}} \rceil \tag{3.11}$$

The DC coefficients located in the interval $I_{b_0}^q$ described in Eq. 3.12 are expressed by quantized symbol q.

$$I_{b_0}^q = \left[qQ_{step}^{DC}, (q+1)Q_{step}^{DC} \right) \tag{3.12}$$

For AC coefficients, dead-zone quantization with doubled zero interval is applied. Different from the fixed data range utilized in DC coefficient, a dynamic data range $[-MAX_{b_k}, MAX_{b_k}]$ is defined for each AC coefficient band $b_k, k \geq 1$, where MAX_{b_k} denotes the maximum absolute value of the coefficients within frequency band b_k. Transmitting the dynamic range $[-MAX_{b_k}, MAX_{b_k}]$ to decoder has the advantage to introduce lower distortion after reconstruction. Since all the AC coefficients of a given band b_k are located in the dynamic range, the same number of quantization levels $2^{M_{b_k}}$ are distributed over a shorter range (i.e. $[-MAX_{b_k}, MAX_{b_k}]$). Thus, a smaller quantization step size is utilized, which introduces the lower distortion at the decoder. The corresponding quantization step size for AC coefficients within band b_k is

calculated as:

$$Q_{step}^{b_k} = \left\lceil \frac{2 \cdot MAX_{b_k}}{2^{M_{b_k}}} \right\rceil \tag{3.13}$$

The AC coefficients located in the interval $I_{b_k}^q$ are expressed by quantized symbol q. The intervals are defined as [16]:

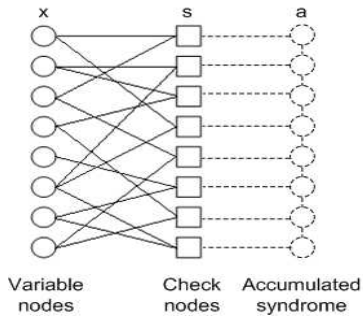
$$I_{b_k}^q = \begin{cases} \left[(q-1)Q_{step}^{b_k}, qQ_{step}^{b_k}\right), & q < 0 \\ \left[-Q_{step}^{b_k}, Q_{step}^{b_k}\right), & q = 0 \\ \left[qQ_{step}^{b_k}, (q+1)Q_{step}^{b_k}\right), & q > 0 \end{cases} \tag{3.14}$$

The quantized symbols q of each frequency band b_k are then organized in M_{b_k} bitplanes and fed to LDPCA encoder.

3.3.3 Slepian-Wolf Encoder

Powerful channel codes like turbo codes [10] [17] and Low-Density Parity Check (LDPC) codes [12] are usually used to encode the bitplane of the quantized coefficients in practical DVC codecs. In order to achieve compression close to Slepian-Wolf bound, parity bits of turbo codes need to be punctured and syndrome bits of LDPC codes need to be accumulated. Compared with punctured turbo codes, LDPC accumulate (LDPCA) codec allows higher compression efficiency [5] [12] in distributed source coding problem and it has been applied in the best available DVC codec [5]. Thus, LDPCA encoder [12] is also chosen as the Slepian-Wolf encoder in this work.

LDPCA encoder consists of an LDPC syndrome generator concatenated with an accumulator as shown in Fig. 3.6. The syndrome bits s of source bits x are calculated according to the graph structure (from variable nodes to check nodes) based on low-density parity check matrix H, i.e. $s = Hx$. The complexity of syndrome based encoding is linear in the number of the edges (1's) in LDPC codes. Since there are low density of 1s in parity check matrix H, the complexity of Slepian-Wolf encoder is kept at low level. In order to make LDPC codes perform incremental rate adaptive decoding, i.e. the additional syndromes bits can be combined with previous sent syndrome bits for decoding, syndrome bits



(a)

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

(b)

Figure 3.6: (a) The example LDPCA encoder [12]. (b) corresponding low-density parity check matrix H

s are in turn accumulated modulo 2, producing the accumulated syndrome bits a. All the accumulated syndromes are stored into a buffer and only transmitted a few syndromes initially. If Slepian-Wolf decoder fails in decoding with the transmitted syndromes, more accumulated syndromes are requested from the buffer using the feedback channel. Following a predefined order, the accumulated syndrome bits are transmitted incrementally until the successful decoding. Furthermore, 8-bits CRC sum [13] [18] with standard polynomial $x^8 + x^2 + x + 1$ of each encoded bit bitplane is transmitted also to aid the decoder detecting errors.

3.3.4 Side Information Generation

Based on the architecture of state-of-the-art transform domain Wyner-Ziv video coding shown in Fig. 3.4, it can be seen that the quality of side information frame has influence on both the soft input estimation module and the reconstruction module. A more accurate side information frame contains fewer errors and consequently requires fewer syndrome bits from the buffer for reconstructing the Wyner-Ziv frame with the same decoding quality. Therefore, a side information frame Y_i and its estimated noise residual frame R can influence the coding efficiency of the transform domain Wyner-Ziv video coding significantly.

Frame interpolation [19] [20] and frame extrapolation [21] [22] based

algorithms are two major schemes employed in Wyner-Ziv video coding. Frame interpolation methods use previous and following decoded frames to generate the side information but introduce some delay, while the extrapolation methods only use previously decoded frames which has benefits for real-time applications due to the low latency. In state-of-the-art transform domain Wyner-Ziv video codec [16] [5], an advanced motion compensated frame interpolation [19] algorithm is employed, which includes forward motion estimation, bi-directional motion estimation, spatial smoothing of motion vectors and bi-directional motion compensation. More details about the side information generation methods are introduced in Chapter 4.

3.3.5 Noise Model

Once the side information frame Y_i is obtained, a virtual channel noise model is utilized at the Wyner-Ziv decoder to estimate the noise distribution between the side information frame Y_i and the original Wyner-Ziv frame X_i. As shown in Fig. 3.4, the estimated noise distribution (parameter α) is consequently used to calculate the soft input P_{cond} which is subsequently fed into LDPCA decoder. The more accurate the noise distribution is, the more precise soft input is fed into LDPCA decoder and thereafter less syndrome bits are required. Therefore the noise model can also influence the coding performance of Wyner-Ziv video coding significantly.

Laplacian distribution is usually employed to model the noise in preliminary works as in [2] [10]. However, accurate estimation of Laplacian parameter α_0 of corresponding noise distribution could be a complex task, since the original frame X_i is never available at the Wyner-Ziv decoder. Therefore, estimated residual frame R created by side information generation is used to estimate Laplacian parameter α approximately.

$$f(X_i - Y_i) = \frac{\alpha_0}{2} e^{-\alpha_0 |X_i - Y_i|} \approx \frac{\alpha}{2} e^{-\alpha |R|} \tag{3.15}$$

In transform domain Wyner-Ziv video codec, noise model can be constructed in different granularity levels, i.e. from band level [23] to coefficient [24] [25] level. In state-of-the-art transform domain Wyner-Ziv video coding [16], an online Laplacian distribution noise model in coefficient level [24] is utilized. With this model, each coefficient within

band b_k is assigned with a Laplacian parameter $\alpha_{b_k}(u,v)$. The value of Laplacian parameter $\alpha_{b_k}(u,v)$ indicates the amount of noise at position (u,v) by taking both spatial and temporal variation into account. More details about different noise models are introduced in Chapter 5.

3.3.6 Soft Input Calculation

Soft input calculation is based on bitplane level, starting with the Most Significant Bitplane (MSB) and ending with the Least Significant Bitplane (LSB) of each band. With a given parameter $\alpha_{b_k}(u,v)$, Laplacian distribution centered around the side information coefficient $C_Y^{b_k}(u,v)$ (within band b_k at position (u,v)) is uniquely defined. With the obtained side information coefficient $C_Y^{b_k}(u,v)$, Laplacian parameter $\alpha_{b_k}(u,v)$ and previously decoded bitplanes (Pbp), soft input P_{cond} at position (u,v) can be calculated. Soft input P_{cond} is defined as the conditional probability of bits equal to 0 or 1, i.e. $P_{cond} = P(x|C_Y^{b_k}(u,v), \alpha_{b_k}(u,v), Pbp)$. After all the soft input P_{cond} in one bitplane are obtained, they are grouped together and fed into LDPCA decoder for iterative decoding.

In order to demonstrate how to calculate the soft input information P_{cond}, a simplified example is used. As shown in Fig. 3.7, it is assumed that the side information coefficients $C_Y^{b_k}(u,v)$ are quantized into 3 bits. With a given distribution $\alpha_{b_k}(u,v)$, the probability having the value 0

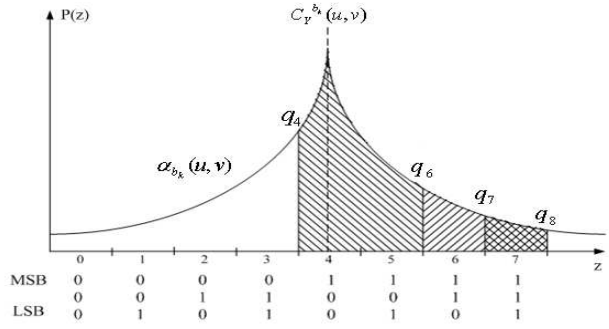


Figure 3.7: Example on soft input calculation, Laplacian distribution $\alpha_{b_k}(u,v)$ is centered on the value of side information $C_Y^{b_k}(u,v)$

or 1 at position (u,v) can be obtained by calculating the integral of

probability density function (pdf) from lower bound to upper bound with value 0 or 1. In Fig. 3.7, the probabilities for the MSB are:

$$P_{cond}(x=1) = \frac{\int_{q_4 Q_{step}}^{q_8 Q_{step}} f(z)dz}{\int_{q_0 Q_{step}}^{q_8 Q_{step}} f(z)dz} \tag{3.16}$$

$$P_{cond}(x=0) = 1 - P_{cond}(x=1) \tag{3.17}$$

where $q_4 Q_{step}$ and $q_8 Q_{step}$ are the lower bound and the upper bound with value 1 in MSB, $q_0 Q_{step}$ is the lower bound with value 0 in MSB. $f(z)$ is the pdf of a given Laplacian distribution. Similarly, assuming that the first two bitplanes are successfully decoded and the corresponding values at position (u, v) are both "1", the probability for the LSB can be calculated as:

$$P_{cond}(x=1) = \frac{\int_{q_7 Q_{step}}^{q_8 Q_{step}} f(z)dz}{\int_{q_6 Q_{step}}^{q_8 Q_{step}} f(z)dz}. \tag{3.18}$$

In order to avoid numerical computation, the integral of the given Laplacian distribution (with parameter α) is converted into different expressions depending on the relation between the bounds of proceeded interval (i.e. the lower bound LB and the upper bound UB) and the value of side information coefficient C_Y.

$$\int_{LB}^{UB} p(z)dz = \begin{cases} 1 - 0.5(e^{\alpha(LB-C_Y)} + e^{\alpha(C_Y-UB)}) & C_Y \in (LB, UB) \\ 0.5(e^{\alpha(UB-C_Y)} - e^{\alpha(LB-C_Y)}) & UB \leq C_Y \\ 0.5(e^{\alpha(C_Y-LB)} - e^{\alpha(C_Y-UB)}) & LB \geq C_Y \end{cases} \tag{3.19}$$

3.3.7 Slepian-Wolf Decoder

For LDPCA decoding, variable nodes are seeded with Log-Likelihood Ratio (LLR) of conditional probability $P_{cond}(x)$ obtained from soft input calculation module.

$$L(x_i) = \log \frac{P_{cond}(x=1)}{P_{cond}(x=0)} \tag{3.20}$$

Then the soft information LLR are passed back and forth between variable nodes and the check nodes according to the log-domain Sum-Product Algorithm (SPA) [26]. Puncturing the syndrome bits of LDPC

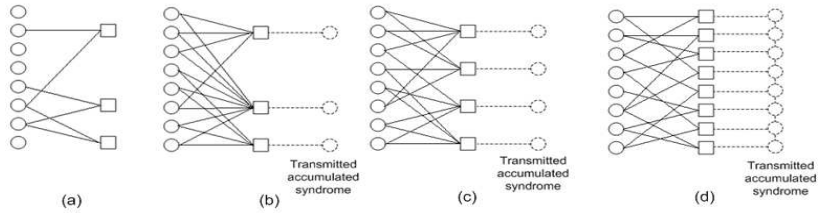


Figure 3.8: LDPC decoding graphs if the encoder transmits (a) Syndrome bits with compression ratio 8:3 (b) Accumulated syndrome bits with compression ratio 8:3 (c) Accumulated syndrome bits with compression ratio 2:1 (d) Accumulated syndrome bits with compression ratio 1:1

(as in Fig. 3.8 (a)) is the simplest method to achieve compression. However, it will degrade the decoding graph which consequently leads to poor performance of decoding. In order to avoid graph degradation, the decoding graphs are constructed by accumulating the check nodes while keeping the number of the edge is constant as shown in Fig. 3.8 (b). The design of rate-adaptive LDPCA is starting with the highest compression ratio graph. Following the predefined incrementally transmission order, other graphs are obtained by successively dividing the syndrome nodes into pairs. The rate-adaptive LDPCA decoder is achieved by modifying its decoding graph each time when it receives an additional increment of the accumulated syndromes [12] as shown in Fig. 3.8 from (b) to (d).

Together with the 8-Bit CRC sum, the decoding bitplane can be tested against the syndrome bits to verify correctness. If the Hamming distance between received accumulated syndromes and the accumulated syndromes of the decoded bits is different from zero or the 8-bits CRC sum is incorrect after a certain amount of iterations, the LDPCA decoder requests more accumulated syndrome bits from the buffer via the feedback channel to correct the potential bit errors. If both the Hamming distance and the CRC sum are satisfied, convergence is declared, guaranteeing a very low error probability for the decoded bitplane. Decoded Wyner-Ziv frame with or without 8-bits CRC sum are compared as in Fig 3.9. It is necessary to notify that when the number of received accumulated syndrome bits a equals the number of source bits x, i.e. the compression ratio is 1:1, it guarantees successful decoding of the source bits x via straightforward linear algebra, i.e. inverse of H matrix, regardless of the quality of the side information.

(a) with CRC (b) without CRC

Figure 3.9: Wyner-Ziv coded frame with and without 8-bits CRC, No. 51 frame in Hall Monitor, GOP=4, (a)PSNR=40.0696 dB, Bits=16735, (b)PSNR=33.8544 dB, Bits=15871

3.3.8 Reconstruction

After a bitplane is successfully decoded, a quantization interval, i.e. $[q_i Q_{step}, q_{i+1} Q_{step})$, can be obtained. It indicates the range of the original Wyner-Ziv coefficient C_X. Hence, C_X can be reconstructed by computing the expectation $E[C_X | C_X \in [q_i Q_{step}, q_{i+1} Q_{step}), C_Y]$ of the random variable C_X given the quantization interval and side information coefficient C_Y [10]:

$$C_{\hat{X}} = \frac{\int_{q_i Q_{step}}^{q_{i+1} Q_{step}} z f(z) dz}{\int_{q_i Q_{step}}^{q_{i+1} Q_{step}} f(z) dz}. \tag{3.21}$$

where $C_{\hat{X}}$ denotes the reconstructed value. q_i denotes the quantization index of C_X, Q_{step} is the corresponding quantization step. $f(z)$ is the pdf of the given Laplacian distribution

To avoid numerical computation of integrals, a closed form expression of Eq. 3.21 with a given Laplacian distribution parameter α is derived [27]:

$$C_{\hat{X}} = \begin{cases} q_i Q_{step} + \frac{1}{\alpha} + \frac{Q_{step}}{1 - e^{\alpha Q_{step}}} & C_Y < q_i Q_{step} \\ C_Y + \frac{(\gamma + \frac{1}{\alpha}) e^{-\alpha\gamma} - (\delta + \frac{1}{\alpha}) e^{-\alpha\delta}}{2 - (e^{-\alpha\gamma} + e^{-\alpha\delta})} & C_Y \in [q_i Q_{step}, q_{i+1} Q_{step}) \\ q_{i+1} Q_{step} - \frac{1}{\alpha} - \frac{Q_{step}}{1 - e^{\alpha Q_{step}}} & C_Y \geq q_{i+1} Q_{step} \end{cases} \tag{3.22}$$

where $\gamma = C_Y - q_i Q_{step}$ and $\delta = q_{i+1} Q_{step} - C_Y$.

3.4 Performance Evaluation

To demonstrate the coding performance of state-of-the-art transform domain Wyner-Ziv video codec, test conditions are precisely described as follows:

- Four different test sequences (available at [5]), *Foreman*, *Soccer*, *Coastguard* and *Hall Monitor*, are adopted for the RD performance test.
- The spatial resolution of the sequences is QCIF, the temporal resolution is 15 frames per second (fps). Commonly used GOP size 2 is chosen, which means every odd frame is key frame and every even frame is Wyner-Ziv frame.
- Key frames are encoded with H.264/AVC Intra (Reference codec JM 9.5 [28]), the setting is reported in Appendix B. The Quantization Parameters (QP) are chosen as in Table 3.1, so that the quality of the WZ frames is similar to the quality of the key frames [5] [29].

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8
Foreman	40	39	38	34	34	32	29	25
Coastguard	38	37	37	34	33	31	30	26
Soccer	44	43	41	36	36	34	31	25
Hall Monitor	37	36	36	33	33	31	29	24

Table 3.1: Quantization Parameter for key frames in different RD points, QCIF@15Hz

- Bitstream of key frame (H.264/AVC Intra bits) and LDPCA syndrome bits for Wyner-Ziv frames are counted as used coding bits. Average Peak Signal-to-Noise Ratio (PSNR) over all the frames of a sequence is used to evaluate the quality of decoded sequences. Only luminance component is coded. Thus the metrics (i.e. coding bits and PSNR) refer only to the luminance component.

- The motion search in side information generation is performed with half-pixel accuracy.

The RD performance of the implemented transform domain Wyner-Ziv video codec is compared with DISCOVER Wyner-Ziv video codec [5], H.264/AVC Intra codec and H.264/AVC no motion codec as in Fig. 3.10. It can be seen from the results that the performance of implemented codec is comparable with the best available transform domain Wyner-Ziv video codec. Compared with H.264/AVC Intra coding, the Wyner-Ziv video coding provides better RD performance for *Coastguard* and *Hall Monitor*, with the gain around 1dB and 3dB, respectively. For sequences with some motion and scene change like *Foreman*, the coding performance is quite close to H.264/AVC Intra coding but not better than it at higher bitrate. The coding gap in higher bitrate is about 0.5 dB. For sequences with more fast and irregular motion like *Soccer*, Wyner-Ziv video coding lose to H.264/AVC Intra coding around 2.5 dB. Compared with H.264/AVC no motion Inter coding, Wyner-Ziv video coding only gives the better coding efficiency for *Coastguard* but worse performance for sequences *foreman*, *Soccer* and *Hall Monitor*.

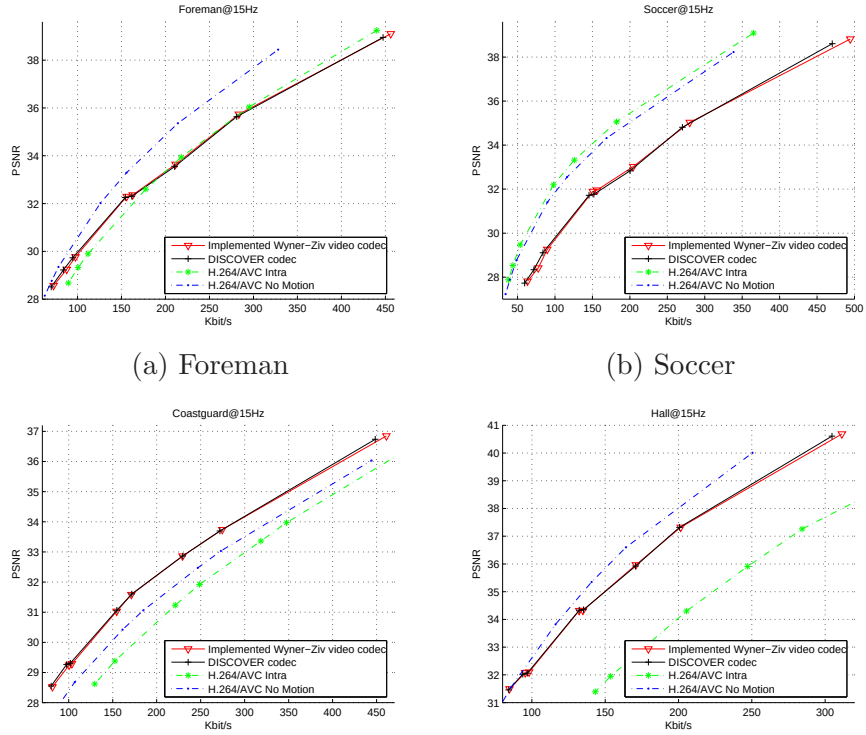
On the other hand, the encoding complexity of H.264/AVC Intra codec, no motion Inter codec and Wyner-Ziv video codec are also compared. The complexity is measured by means of the encoding time for even or Wyner-Ziv frames of the full sequence on a 3 GHz PC. As shown in Fig. 3.11, Wyner-Ziv video codec always requires the least computation power from encoder. Generally, encoding complexity of Wyner-Ziv video codec is around 1/4 of H.264/AVC Intra and 1/8 of H.264/AVC no motion Inter (1/4 of the optimized H.264/AVC no motion Inter).

Taking both the coding performance and the encoding complexity into account, it shows that Wyner-Ziv video codec is a promising video coding solution for critical encoding resource scenario.

3.5 Summary

A transform domain Wyner-Ziv video codec is described in this chapter, which is based on information theory results: the Slepian-Wolf and the

[0]The chosen encoding configurations of H.264/AVC motion and no motion Inter coding give similar coding efficiency results compared to the DISCOVER results [5].

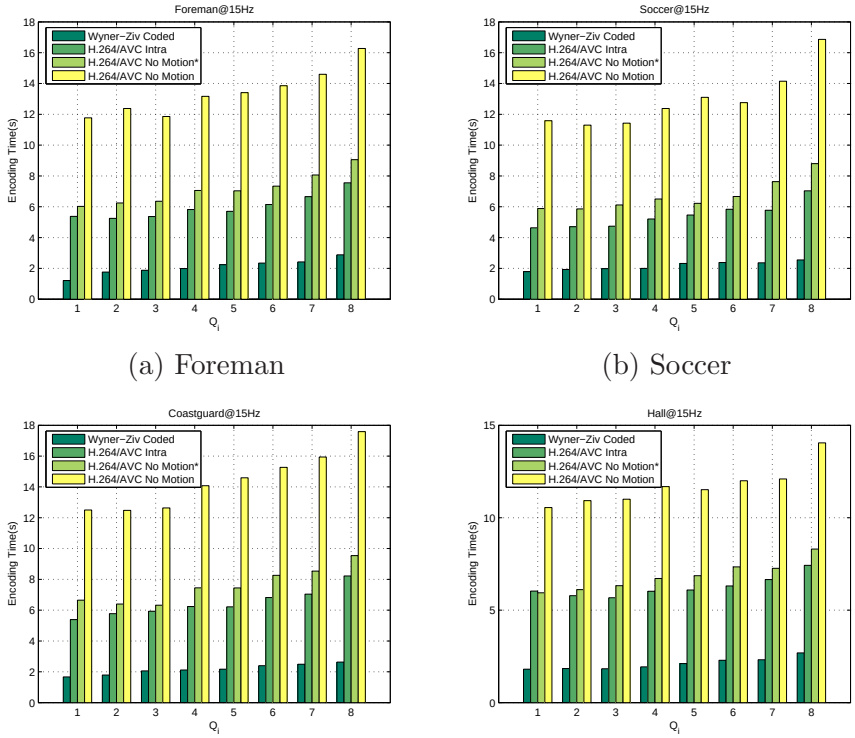


(a) Foreman (b) Soccer

(c) Coastguard (d) Hall Monitor

Figure 3.10: Performance evaluation of transform domain Wyner-Ziv video codec

Wyner-Ziv theorems. It achieves low complexity encoding by removing the motion estimation from the encoder but fully or partly exploiting the video redundancy at the decoder. The RD performance of practical Wyner-Ziv video codec is efficient but not as good as the conventional video codec yet. Compared with H.264/AVC Intra codec and no motion Inter codec, Wyner-Ziv video codec provides a better RD performance for some low motion video sequences like *Coastguard*. However, for relevant high motion video sequences like *Foreman* and *Soccer* or static background sequence like *Hall Monitor*, the performance of Wyner-Ziv video coding can not outperform H.264/AVC no motion coding (but



(a) Foreman (b) Soccer

(c) Coastguard (d) Hall Monitor

Figure 3.11: Encoding complexity comparison between Wyner-Ziv coding, H.264/AVC Intra and No motion

sometimes wins against or closes to H.264/AVC Intra codec).

Considering that the encoding complexity of Wyner-Ziv video codec is only 1/4 of H.264/AVC Intra and 1/8 of H.264/AVC no motion Inter (1/4 of the optimized H.264/AVC no motion Inter), it can be concluded that Wyner-Ziv video codec is a very promising coding solution for critical encoding resource applications. Therefore it is necessary to explore the possibilities of further improving the coding efficiency of the practical Wyner-Ziv video codec.

References to Chapter 3

[1] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero. "Distributed video coding", *Proc. of IEEE, Special issue on advances in video coding and delivery*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[2] A. Aaron, R. Zhang, and B. Girod. "Wyner-ziv coding of motion video", *Proc. Asilomar Conf. on Signals, Systems and Computers*, pp. 240–244, Nov. 2002.

[3] D. Slepian and J. Wolf. "Noiseless coding of correlated information sources", *IEEE Trans. Inform. Theory*, vol. 19, pp. 471–480, July 1973.

[4] A. Wyner and J. Ziv. "The rate-distortion function for source coding with side information at the decoder", *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, Jan 1976.

[5] DISCOVER Project: www.discoverdvc.org, Dec 2007.

[6] T. Cover and J. Thomas. *Elements of Information Theory.* A Wiley-Interscience publication, 1991. ISBN 0-471-06259-6.

[7] F. Pereira, C. Brites, J. Ascenso, and M. Tagliasacchi. "Wyner-ziv video coding: a review of the early architectures and further developments", *IEEE International Conference on Multimedia and Expo*, pp. 625–628, June 2008.

[8] R. Puri and K. Ramchandran. "Prism: A new robust video coding architecutre based on distributed compression principles", *40th Allerton Conference on Communication, Control and Computing*, Oct. 2002.

[9] R. Puri and K. Ramchandran. "Prism: A video coding paradigm with motion estimation at the decoder", *IEEE Trans. on Image Process.*, vol. 16, no. 10, pp. 2436–2448, Oct. 2007.

[10] A. Aaron, S. Rane, E. Setton, and B. Girod. "Transform domain wyner-ziv codec for video", *Proc. SPIE VCIP*, pp. 520–528, Jan 2004.

[11] "Coding of audiovisual objects-part 10: Advanced video coding", *ISO/IEC 14496-10*, 2003.

[12] D. Varodayan, A. Aaron, and B. Girod. "Rate-adaptive distributed source coding using low-density parity-check codes", *EURASIP Signal Process. Journal, Special Section on Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[13] P. Koopman and T. Chakravarty. "Cyclic redundancy code(crc) polynomial selection for embedded networks", *Int'l Conf. ON Dependable Systems and Networks*, June 2004.

[14] H. Malvar, A. Hallapuro, M. Karczewicz, and L. Kerofsky. "Low-complexity transform and quantization in h.264/avc", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 598–603, July 2003.

[15] C. Brites, J. Ascenso, and F. Pereira. "Improving transform domain wyner-ziv video coding performance", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 14–19, May 2006.

[16] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret. "The discover codec: Architecture, techniques and evaluation", *Picture Coding Symposium*, Nov. 2007.

[17] A. Aaron and B. Girod. "Compression with side information using turbo codes", *IEEE Data Compression Conf.*, 2002.

[18] D. Kubasov, K. Lajnef, and C. Guillemot. "A hybrid encoder/decoder rate control for wyner-ziv video coding with a feedback channel", *Int'l Workshop on Multimedia Signal Processing*, Oct. 2007.

[19] J. Ascenso, C. Brites, and F. Pereira. "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding", *5th EURASIP Conf. on Speech and Image Process., Multimedia Commun. and Services*, July 2005.

[20] X. Huang and S. Forchhammer. "Improved side information generation for distributed video coding", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 223–228, Oct. 2008.

[21] L. Natario, C. Brites, J. Ascenso, and F. Pereira. "Extrapolating side information for low-delay pixel-domain distributed video coding", *Int'l Workshop on Very Low Bitrate Video Coding*, Sept. 2005.

[22] S. Borchet, K. Westerlaken, R. Gunnewiek, and R. Lagendijk. "On extrapolating side information in distrbuted video coding", *Picture Coding Symposium*, Nov. 2007.

[23] C. Brites, J. Ascenso, and F. Pereira. "Studying temporal correlation noise modeling for pixel based wyner-ziv video coding", *IEEE International Conference on Image Processing*, Oct. 2006.

[24] C. Brites and F. Pereira. "Correlation noise modelling for efficient pixel and transform domain wyner-ziv video coding", *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 18, no. 9, Sept. 2008.

[25] L. Qing, X. He, and R. Lv. "Distributed video coding with dynamic virtual channel mode estimation", *Int'l Symposium on Data, Privacy and E-Commerce*, pp. 170–173, 2007.

[26] L. Shu and D. Costello. *Error Control Coding.* Pearson Education International, Second edition, 2004. ISBN 0-13-017973-6.

[27] D. Kubasov, J. Nayak, and C. Guillemot. "Optimal reconstruction in wyner-ziv video coding with multiple side information", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 183–186, Oct 2007.

[28] *Joint Video Team (JVT) Reference Software.*, Available on: http://iphome.hhi.de/suehring/tml/index.htm.

[29] C. Brites, J. Ascenso, J. Pedro, and F. Pereira. "Evaluating a feedback channel based transform domain wyner-ziv video codec", *Signal Processing: Image Communication*, vol. 23, no. 4, pp. 269–297, April 2008.

Chapter 4

Side Information Generation

The quality of side information has a major impact on the RD performance of Wyner-Ziv video coding, which is in the same way the quality of the motion compensated prediction had a significant influence in predictive video coding like H.264/AVC. Based on the architecture of state-of-the-art transform domain Wyner-Ziv video coding, it can be seen that the quality of the side information frame not only influences the soft input estimation module but also the reconstruction module. Side information frame is seen as an observation of original Wyner-Ziv frame with an amount of 'noise'. Generally, more accurate side information frame means that there are fewer errors in side information frame and consequently fewer bits are requested from the encoder for the same decoding quality. Therefore, the choice of adopted side information generation scheme can significantly influence the RD performance of Wyner-Ziv video coding.

There are several side information generation schemes proposed in the literature, notably frame interpolation [1] [2] and frame extrapolation [3] [4] based algorithms. Frame interpolation methods use previous and next decoded frames to generate the side information introducing some delay, while extrapolation methods only use previously decoded frames which has benefits for real-time applications due to the lower latency. The main objective of this chapter is to progress the coding efficiency of Wyner-Ziv video coding and reduce the RD performance

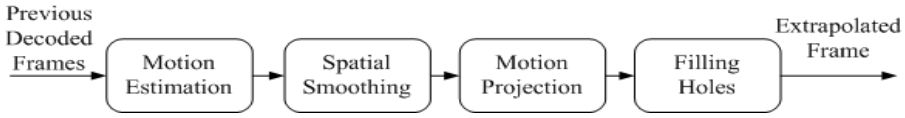


Figure 4.1: The procedure of frame extrapolation

gap regarding conventional video coding solutions, by proposing an improved frame interpolation method. In order to get an impression of the influence given by different side information generation methods, other block based frame extrapolation and frame interpolation methods are also introduced in this chapter.

The structure of this chapter is organized as follows: An extrapolation based side information generation method is described in Section 4.1. Including the proposed improved side information generation method, different interpolation based algorithms are described in Section 4.2. In Section 4.3, the performance of different side information generation methods and their corresponding coding efficiency results are compared and presented.

4.1 Frame Extrapolation

In order to extrapolate a side information frame similar to the Wyner-Ziv frame being decoded, frame extrapolation method estimates the motion field among previously Intra or Wyner-Ziv decoded frames to predict a forthcoming frame. Since the obtained motion field is going to be projected to current Wyner-Ziv frame time instant as a prediction, the motion estimation should be done carefully so that the capture of true motion can be ensured. Therefore, it is necessary to estimate the motion field not only based on temporal correlation but also spatial correlations. Similar to [3], the general process of a block based frame extrapolation method is depicted in Figure 4.1. Without loss of generality, the process is described in the following for GOP size 2, where the previous Intra coded frame X'_{2i-1} and the previous Wyner-Ziv coded frame X'_{2i-2} are used to extrapolate side information frame YE_{2i}. The procedure proceeds as follows:

- ***Motion Estimation***: Frame X'_{2i-1} is split into non-overlapped

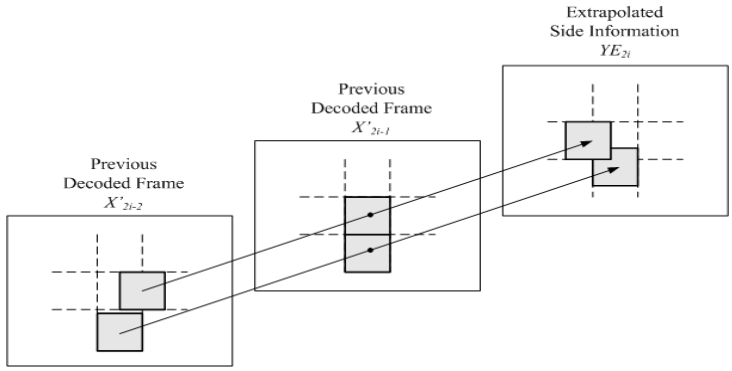


Figure 4.2: Frame projection

8×8 blocks. Block based motion estimation is performed for each block of frame X'_{2i-1} by searching for the best matching block with minimum Mean Squared Error (MSE) in frame X'_{2i-2}.

$$Argmin\{E_{(m_0,n_0)\in block}\{(X'_{2i-1}(m_0,n_0) - X'_{2i-2}(m_0+\Delta m, n_0+\Delta n))^2\} \tag{4.1}$$

where (m_0, n_0) are coordinates of current 8×8 block, E is the expected value over $(m_0, n_0) \in 8 \times 8$ block. $(\Delta m, \Delta n)$ represents the motion vectors.

• ***Spatial Smoothing***: After motion estimation, all blocks in X'_{2i-1} are assigned with motion vectors. However, since the motion estimation is only applied in the temporal domain, the obtained motion vectors have relative low spatial coherence, especially for the blocks belonging to one moving object. Therefore, a weighted vector median filter [5] is applied to smooth the motion vector field, which increases the spatial coherence of different motion vectors and aims to reduce the number of incorrect motion vectors compared to true motion. The weighted vector median filter is defined as in [5]:

$$\sum_{i=1}^{N} w_i ||MV_{vm} - MV_i||_L \leq \sum_{i=1}^{N} w_i ||MV_c - MV_i||_L \tag{4.2}$$

where MV_c and MV_i are motion vectors of current block and its corresponding neighboring blocks, respectively. N is the number of neigh-

boring blocks. MV_{vm} represents the motion vector after the weighted vector median filter, which minimizes the weighted sum of distances among other N neighboring motion vectors in terms of the L_2-norm. The weight parameter w_i is obtained according to the prediction error as in [1]:

$$w_i = \frac{MSE(MV_c, Block_c)}{MSE(MV_i, Block_c)} \tag{4.3}$$

where the $MSE(MV_\bullet, Block_c)$ represents the MSE value between current block $Block_c$ in decoded frame X'_{2i-1} and the block with relevant motion vector $MV_\bullet$ in decoded frame X'_{2i-2}. Motion vectors with and without spatial smoothing are compared in Fig. 4.3, the relevant extrapolated frames are shown in Fig. 4.4.

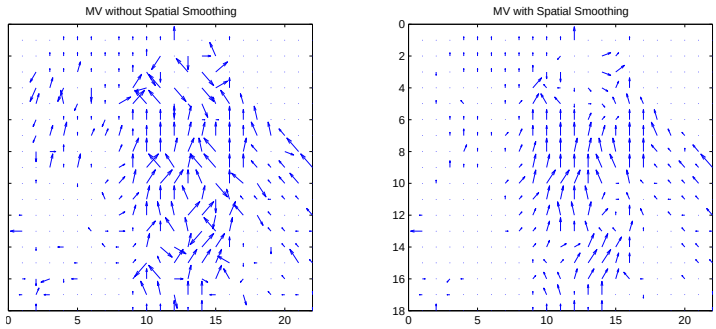
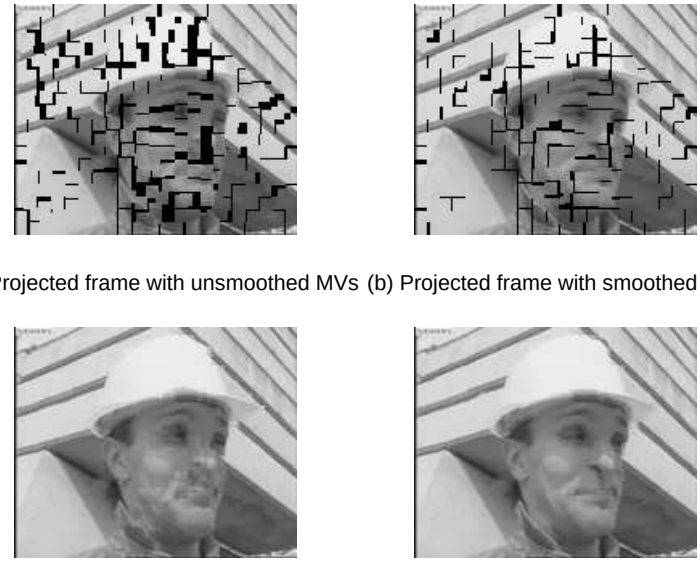


Figure 4.3: Motion Vectors between frame #2 and #3 of foreman, QCIF, left: without spatial smoothing, right: with spatial smoothing

• ***Frame Projection***: To obtain an extrapolated frame for time instant $2i$, the obtained motion vectors between frame X'_{2i-2} and frame X'_{2i-1} are applied between frame X'_{2i-1} and frame YE_{2i} following a linear motion assumption. Then, the pixels in frame X'_{2i-1} are projected onto frame YE_{2i} as shown in Fig. 4.2. If there is more than one pixel in frame X'_{2i-1} projected onto the same position in frame YE_{2i}, an average value between these overlapping pixels is taken. An example frame after frame projection can be found presented in Fig. 4.4.

• ***Filling Holes***: In order to fill the unreferenced pixel areas in frame YE_{2i}, the motion vectors of these unfilled pixels need to be estimated first. With the estimated motion vector of the unreferenced pixels, the

holes are going to be filled with the projected pixels from frame X'_{2i-1}. There are two possible motion vectors which can be utilized, these are the motion vector of the co-located block in frame X'_{2i-1} and the nearest neighboring motion vector of current block in frame YE_{2i}. Therefore, different from the work in [3] [4], the nearest neighboring motion vectors in the spatial domain and the co-located motion vectors in the temporal domain are both used to determine the estimated pixels. An average of these estimations is computed for filling the holes remaining after frame projection process.



(a) Projected frame with unsmoothed MVs (b) Projected frame with smoothed MVs

(c) Filling the holes of (a) (d) Filling the holes of (b)

Figure 4.4: Extrapolated frame (frame No. 4, Foreman, QCIF). (a), Projected frame with the unsmoothed MVs. (b), Projected frame with the spatial smoothed MVs. (c), Frames after holes filling of (a), PSNR=27.9587 dB. (d), Frames after holes filling of (b), PSNR=28.1573 dB.

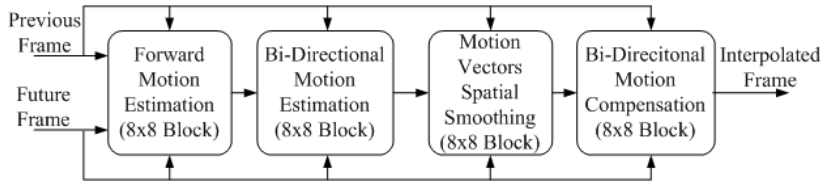


Figure 4.5: The procedure of motion compensated frame interpolation [1]

4.2 Frame Interpolation

Different from frame extrapolation algorithm, frame interpolation utilizes one previous frame and one subsequent frame to predict the frame in between. Although a latency is introduced due to the usage of subsequent frames, more accurate motion vectors according to true motion could be obtained by frame interpolation. Furthermore, interpolated frame are obtained by combining the pixels' value both in previous frame and subsequent frame. It is an advantage compared to the extrapolated frame where the pixel value is copied from previous frame only. In the following sections, a block based motion compensated frame interpolation algorithm [1] adopted in state-of-the-art Wyner-Ziv video codec is introduced first. Then an improved frame interpolation scheme [2] with Overlapped Block Motion Compensation (OBMC) is proposed.

4.2.1 Motion Compensated Frame Interpolation

Similar to frame extrapolation, motion compensated frame interpolation starts with a block based unidirectional motion estimation. Following the linear motion assumption, motion field of each block in interpolated frame is refined by a bidirectional motion estimation and a spatial smoothing filter. Finally, the interpolated frame is generated by averaging the best two matching blocks in previous frame and subsequent frame. The general process of the motion compensated frame interpolation is depicted in Figure 4.5. Without loss of generality, the process is described in the following for GOP size 2, where Intra coded previous frame X'_{2i-1} and subsequent frame X'_{2i+1} are used to generate interpolated side information frame YI_{2i}. The procedure proceeds as follows:

- ***Forward Motion Estimation***: Frame X'_{2i-1} is split into non-

overlapped 8×8 block, then block based motion estimation with half pixel accuracy [6] is performed on each block of frame X'_{2i-1} by searching for the best matching block with minimum MSE in frame X'_{2i+1}. Since the obtained motion vectors of each block represents the movement from frame X'_{2i-1} to frame X'_{2i+1}, the motion vectors are not necessary passing through the center of each non-overlapped block in interpolated frame YI_{2i} as shown in Fig. 4.6. In order to avoid overlapped and unreferenced area as in frame extrapolation, each obtained motion vector passing through the interpolated block is seen as a candidate. As shown in Fig. 4.6, the motion vectors which is closer to the center of interpolated block is finally selected as the best motion vectors. After the selection, each block in interpolated frame is assigned an estimated motion vector.

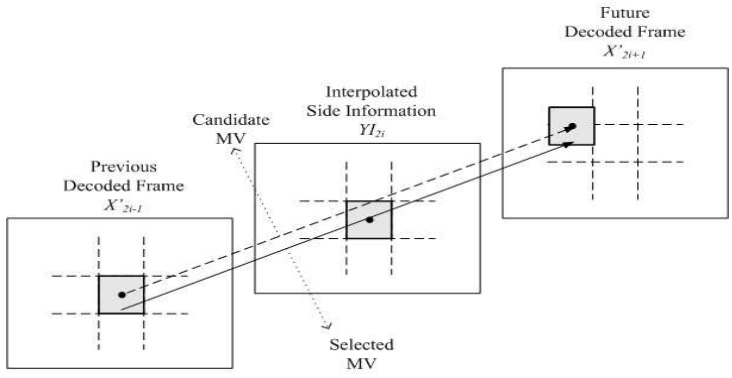


Figure 4.6: Forward motion estimation and motion vector selection

• ***Bi-directional Motion Estimation***: The motion vector obtained from the previous step is based on unidirectional motion estimation. It can be refined by a bidirectional motion estimation [1] scheme. Taking unidirectional motion vectors as an initial point, the bidirectional motion estimation selects a linear trajectory between frame X'_{2i-1} and frame X'_{2i+1} passing through the center of the interpolated blocks. The searching is confined to a small displacement and exact symmetric relative to the interpolated blocks as shown in Fig. 4.7. The

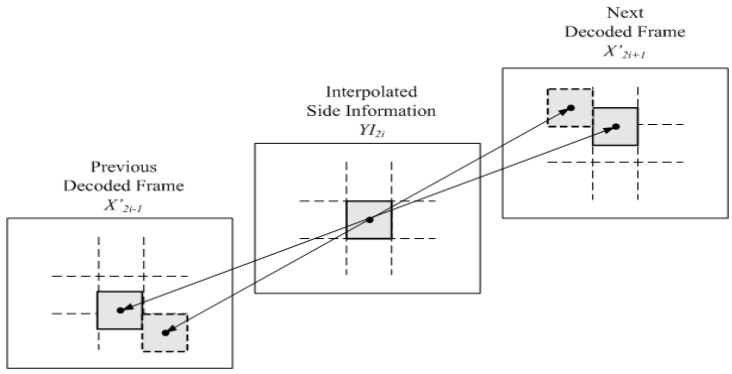


Figure 4.7: Bidirectional Motion Estimation

bi-directional motion estimation can be described as:

$$Argmin\{E_{(m_0,n_0)\in block}\{(X'_{2i-1}(m_0-\Delta m, n_0-\Delta n) \\ -X'_{2i+1}(m_0+\Delta m, n_0+\Delta n))^2\} \quad (4.4)$$

where (m_0, n_0) are coordinates belonging to current interpolated 8×8 block, E is the expected value over $(m_0, n_0) \in 8 \times 8$ block. $(\Delta m, \Delta n)$ represents the symmetric motion vectors.

• ***MV Spatial Smoothing***: After bi-directional motion estimation, each non-overlapped block in frame YI_{2i} is assigned with a motion vector. However, the obtained motion vectors have relative low spatial coherence. In order to increase spatial coherence among different blocks, the same spatial smoothing techniques as described in Section 4.1 is applied.

• ***Bi-directional Motion Compensation***: With the smoothed motion vectors, interpolated frame is generated by a bidirectional motion compensation as defined in standard video coding schemes [7]. Following linear motion assumption, the time interval between frame X'_{2i-1} and frame YI_{2i} equals the time interval between frame YI_{2i} and frame X'_{2i+1}. Therefore the exact same weight is assigned to the best matching blocks for bi-directional motion compensation:

$$YI_{2i}(m_0, n_0) = \frac{1}{2} \times (X'_{2i-1}(m_0-\Delta m, n_0-\Delta n) \\ +X'_{2i+1}(m_0+\Delta m, n_0+\Delta n)) \quad (4.5)$$

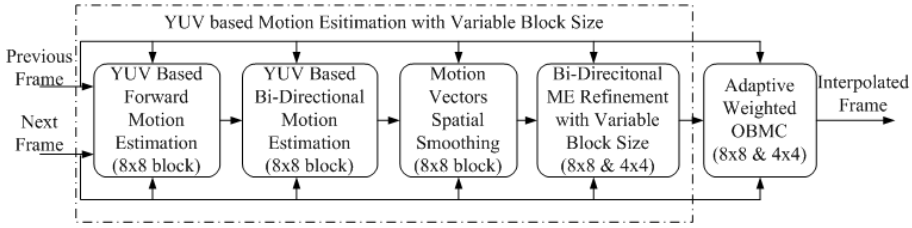


Figure 4.8: The procedure of OBMC based frame interpolation [2]

Motion compensated residue R_{ME} is used to approximately describe the correlation noise between side information frame YI_{2i} and original Wyner-Ziv frame X_{2i}

$$R_{ME}(m_0, n_0) = (X'_{2i-1}(m_0 - \Delta m, n_0 - \Delta n) - X'_{2i+1}(m_0 + \Delta m, n_0 + \Delta n)) \quad (4.6)$$

4.2.2 Overlapped Block Motion Compensation Based Frame Interpolation

Although motion compensated frame interpolation included some sophisticated techniques to optimize motion vector accuracy, there are still some limitations: First of all, it does not utilize all the information which is available at the decoder side, ex. decoded chrominance information. Secondly, the block size used for motion estimation and compensation might not be an optimal choice. Finally, only a simple bidirectional motion compensation is employed. Overcoming these limitations will enhance the quality of side information frame and further improve RD performance of Wyner-Ziv video coding. Therefore, an improved frame interpolation scheme is proposed as shown in Fig. 4.8.

Without loss of generality, the process is described for GOP size 2. The procedure of the improved frame interpolation is divided into two parts: Y, U and V based motion estimation with variable block sizes is applied on two key frame X'_{2i-1} and X'_{2i+1} to get accurate motion vectors at first. Then an adaptive weighted Overlapped Block Motion Compensation (OBMC) is employed to generate better interpolated side information frame YI_{2i}.

YUV Based Motion estimation with variable block size

In order to take advantage of more information available at the decoder, the chroma components (U and V) in Intra decoded key frames are utilized, which are useful to assist luminance component (Y) in motion estimation. Taking forward motion estimation as an example, the YUV based motion estimation is defined as:

$$Argmin\{E_{(m,n)\in block}\{(X^{Y}_{2i-1}(m,n) - X^{Y}_{2i+1}(m+\Delta m, n+\Delta n))^2\} \\ +\lambda E_{(m',n')\in block}\{(X^{UV}_{2i-1}(m',n') - X^{UV}_{2i+1}(m'+\Delta m', n'+\Delta n'))^2\}\} \quad (4.7)$$

where $X^{Y}_{2i-1}(m,n)$ and $X^{UV}_{2i-1}(m',n')$ are the corresponding luma and chroma values at coordinates (m,n) and (m',n') in key frame X_{2i-1}, respectively. $(\Delta m, \Delta n)$ and $(\Delta m', \Delta n')$ represent the motion vectors. For 4:2:0 video sequences, $\Delta m = 2\Delta m', \Delta n = 2\Delta n', m = 2m'$ and $n = 2n'$. λ is a parameter to balance the weight between luma and chroma values.

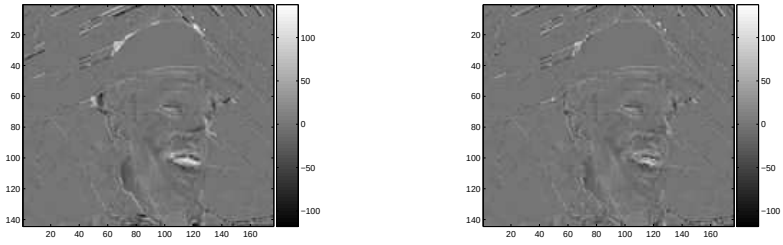
Besides YUV based motion estimation, the first three modules in Fig. 4.8 are similar to the motion compensated frame interpolation scheme described in Fig. 4.5. However, since only 8×8 block based motion estimation is applied in motion compensated frame interpolation, it may not perfectly match the true motion especially around object boundaries. Variable size block based motion estimation is more efficient in representing irregular motion. Therefore, a bi-directional motion estimation with variable block size (8×8 and 4×4) is adopted after the motion vector smoothing module. Selecting two predefined thresholds τ_{mse} and τ_{σ}, each 8×8 block is evaluated to decide whether to divide it into 4×4 sub-blocks based on:

$$MAP_{4\times4} = \begin{cases} True & \text{if } MSE_{8\times8} \geq \tau_{mse} \\ & \text{and } Var(MV) \geq \tau_{\sigma} \\ False & \text{otherwise} \end{cases} \quad (4.8)$$

where $MSE_{8\times8}$ is the YUV based MSE value between X_{2i-1} and X_{2i+1} over the corresponding 8×8 block, $Var(MV)$ is a function to calculate the variance of the relevant motion vectors for the current block in an 3×3 window.

$$Var(MV(m,n)) = \frac{\Sigma^{1}_{i=-1}(MV(m+i,n+i) - \bar{MV})^2}{9} \quad (4.9)$$

where $\bar{MV}$ is the mean value of MVs. If an 8×8 block satisfies the above conditions, its MV is taken as initial MV for each 4×4 sub-blocks and the relevant $MSE_{4\times4}$ are calculated. A small refinement search range ρ is chosen to find the best matching 4×4 sub-block with minimum $MSE_{4\times4}$. With variable block size, the smaller blocks are used to describe irregular motion around the edges of objects, the larger blocks are used for homogeneous motion. As shown in Fig. 4.9, the energy of the motion estimated residual R_{ME} with variable block size is smaller than the one with a fixed 8×8 block. Thus providing an advantage by introducing fewer inaccurate pixels into the side information frame.



(a) (b)

Figure 4.9: Comparison of motion estimated residue R_{ME} with (a): fixed block size and (b): adaptive block size.

Adaptive Weighted Overlapped Block Motion Compensation

Overlapped Block Motion Compensation (OBMC) is usually applied to reduce blocking artifacts and improve subjective quality in frame rate up-conversion. However, it also has a higher risk of over-blurring the interpolated side information frame compared with the simple bi-directional motion compensation used in [1]. Since the MSE value over each block of the YUV based motion estimation approximately reflects the reliability of its relevant motion vectors, an adaptive OBMC [8] weighted by MSE is employed to reduce the interpolated errors and control the blurring. Let $j \in [0, k]$ denote the index of the neighboring blocks. As shown in Fig. 4.10, the value of k is varying due to variable block size adopted.

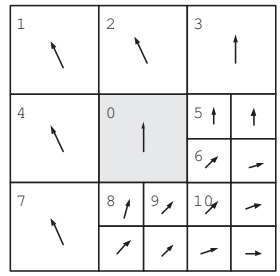


Figure 4.10: Utilized neighboring motion vectors and blocks for adaptive weighted OBMC

$$Y_{2i}(m_0, n_0) = \frac{\Sigma_{j=0}^{k} \omega_j \hat{Y_j}}{\Sigma_{j=0}^{k} \omega_j} \tag{4.10}$$

$$\hat{Y_j} = \frac{1}{2} \times (X_{2i-1}(m_0 - \Delta m_j, n_0 - \Delta n_j) + X_{2i+1}(m_0 + \Delta m_j, n_0 + \Delta n_j)) \tag{4.11}$$

$$R_{ME}(m_0, n_0) = \frac{\Sigma_{j=0}^{k} \omega_j \hat{R}_j}{\Sigma_{j=0}^{k} \omega_j} \tag{4.12}$$

$$\hat{R}_j = (X_{2i-1}(m_0 - \Delta m_j, n_0 - \Delta n_j) - X_{2i+1}(m_0 + \Delta m_j, n_0 + \Delta n_j)) \tag{4.13}$$

where (m_0, n_0) belongs to current block, $(\Delta m_j, \Delta n_j)$ is corresponding symmetric motion vectors of $Block_j$. ω_j is the weight of $Block_j$ obtained by calculating the inverse proportion of the YUV based MSE:

$$\omega_j = (E_{(m_j,n_j)\in Block_j}((X_{2i-1}^{YUV}(m_j - \Delta m_j, n_j - \Delta n_j) - X_{2i+1}^{YUV}(m_j + \Delta m_j, n_j + \Delta n_j))^2))^{-1} \tag{4.14}$$

4.3 Experimental Results

In order to evaluate the performance of different side information generation schemes, average PSNR results over extrapolated/interpolated frames are compared in Table 4.1. The motion search is performed with

half-pixel accuracy [9] for all the different side information generation methods. The methods include: a) A frame extrapolation method described in Section 4.1; b) A bidirectional motion search based method employed in [10]; c) A frame interpolation using variational method [11]; d) A motion compensated frame interpolation (MCFI) method described in Section 4.2.1 and [1]; e) A motion compensated frame interpolation method with YUV motion estimation; f) A motion compensated frame interpolation method with with Variable Block Size (VBS) based YUV motion estimation; g) An adaptive weighted OBMC based frame interpolation (OBMCFI) method with fixed block size (8×8) based YUV motion estimation; h) An OBMCFI method with VBS based Y motion estimation i) An OBMCFI method with VBS based YUV motion estimation described in Section 4.2.2.

Sequence	**Foreman**	**Coastguard**	**Soccer**	**HallMonitor**
Key frames	**QP=25**	**QP=26**	**QP=25**	**QP=24**
a)	25.3215	28.6134	19.3666	33.1699
b)	27.8192	29.7681	20.6988	35.0267
c)	26.9101	30.1105	20.8623	35.3261
d)	28.9047	31.4664	20.8326	36.3338
e)	28.9843	31.4681	20.8483	36.3339
f)	28.9999	31.5371	20.8453	36.3735
g)	29.2358	31.7708	21.2874	36.3331
h)	29.2296	31.8317	21.2961	36.4548
i)	**29.2537**	**31.8340**	**21.2967**	**36.4593**

Table 4.1: The average PSNR results for different methods, key frames are H.264/AVC Intra coded with fixed Quantization Parameter (QPs)

As shown in Table 4.1, the proposed OBMC based frame interpolation method (i) gives the best PSNR performance. Furthermore, it can be seen from the results of method (d)-(i) that each module proposed in Section 4.2.2 generally improves the PSNR results step by step. Visual comparison of different methods is reported in Section C.1.

The complexity for different schemes ((d)-(i)) are evaluated by calculating the average time (on a 3GHz PC) for generating one side information frame. As shown in Fig. 4.11, the proposed method (i) improves

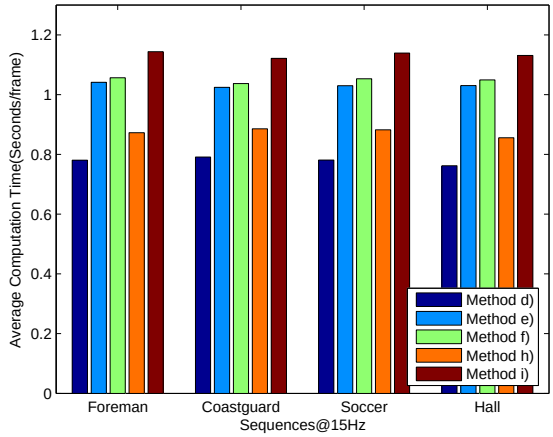
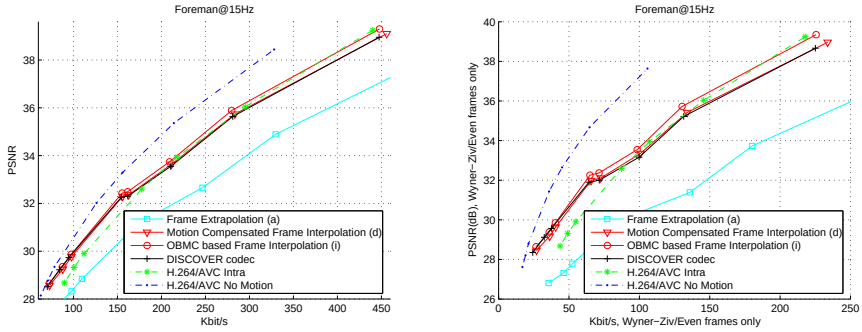


Figure 4.11: Complexity comparison for different side information generation schemes

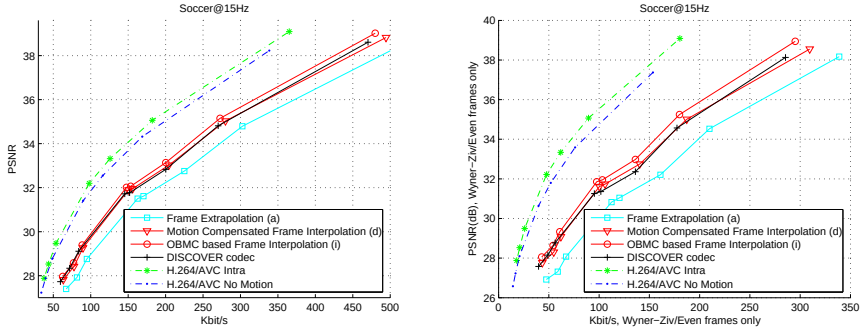
the PSNR result by introducing more complexity. However, the more complex decoder is acceptable in DVC scenario. Even if it is required that the complexity of the decoder should not be significantly increased, the proposed method (h), which removes YUV based motion estimation from method (i), gives a good balance between decoder complexity and PSNR performance.

In order to demonstrate how much influence is given by side information frame on the coding efficiency of Wyner-Ziv video codec, RD performances with frame extrapolation method (a), motion compensated frame interpolation (d) and the OBMC based frame interpolation method (i) are compared. For the sake of fair comparisons, the DISCOVER project [12] test conditions described in Section 3.4 are adopted. The test sequences are *Foreman*, *Soccer*, *Coastguard* and *Hall Monitor*, at QCIF, 15 frames per second (fps); the GOP size is 2. The key frames are encoded using H.264/AVC Intra and the QPs are chosen so that the average PSNR of the WZ frames is similar to the average PSNR of the key frames (as in [12]). The RD performance is evaluated for the luminance component of both the key frames and WZ frames. The benchmark codecs used are the DISCOVER Wyner-Ziv video codec [12], the H.264/AVC Intra codec and the H.264/AVC no motion Inter codec.



(a) (b)

Figure 4.12: GOP2 RD performances comparison for sequence Foreman with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)



(a) (b)

Figure 4.13: GOP2 RD performances comparison for sequence Soccer with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)

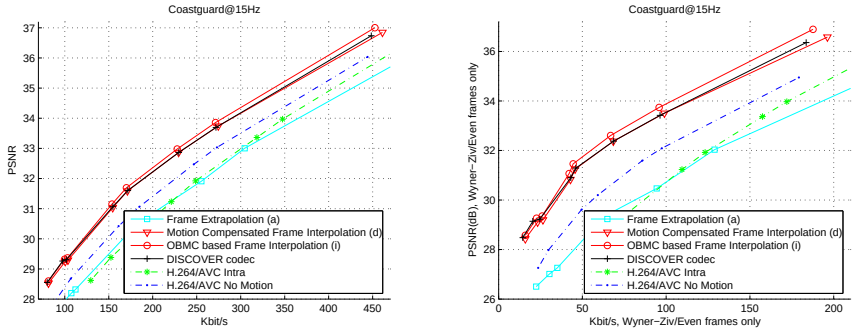


Figure 4.14: RD comparison for sequence Coastguard with different side information generation methods, GOP2. (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)

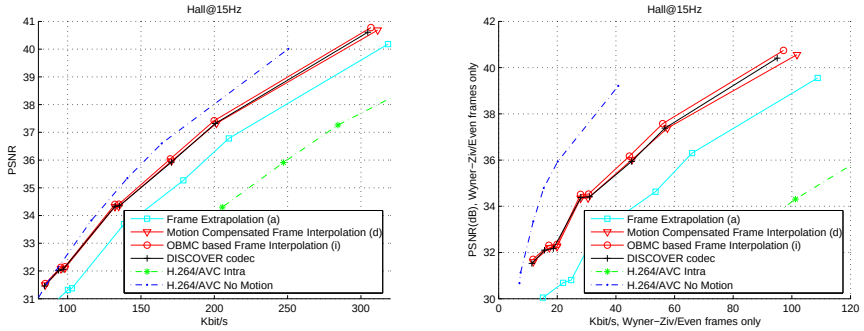


Figure 4.15: GOP2 RD performances comparison for sequence Hall monitor with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)

According to RD performance results shown in Figs. 4.12-4.15, the performance of transform domain Wyner-Ziv video coding with interpolation based side information generation methods (d) and (i) is much better than the one with extrapolation based side information generation. It means that the additional delay involved by interpolation really brings additional RD performance. Compared with the motion compensated frame interpolation method (d) used in [12], employing the proposed OBMC based frame interpolation scheme (i) improves the coding efficiency of transform domain Wyner-Ziv video coding for high bit-rates up to 0.5 dB for the overall RD performance and 1 dB for the Wyner-Ziv frames.

Compared with H.264/AVC Intra coding, Wyner-Ziv video coding with OBMC based frame interpolation gives better RD performance on *Coastguard* and *Hall Monitor*, comparable performance on *Foreman*; For sequence with more irregular motion like *Soccer*, where the decoder frame estimation process is more difficult, the performance gap between H.264/AVC Intra coding and Wyner-Ziv video coding has been reduced. Compared with H.264/AVC no motion Inter coding, Wyner-Ziv video coding still gives worse performance on *Foreman*, *Soccer* and *Hall Monitor*.

For Wyner-Ziv video coding with larger GOP sizes, the RD performance improvements introduced by proposed scheme (i) are even larger compared with GOP size 2. However, winning against H.264/AVC Intra and no motion Inter codec is getting more difficult, since the distance between key frames become far way. The RD performances with larger GOP size (=4) can be found in Figs. C.3-C.6.

4.4 Summary

In this chapter, different side information generation methods from extrapolation to interpolation are introduced. RD performances of Wyner-Ziv video codec with different side information generation methods are evaluated and compared. It shows that the quality of side information frame is one of the most important factors influencing the coding efficiency of Wyner-Ziv video coding. Therefore, for further improving RD performance, an adaptive weighted OBMC based side information generation method is proposed. Experimental results show that the proposed

scheme is efficient on improving the coding efficiency. Compared with the best available scheme employed in DISCOVER executable codec, the proposed scheme can improve coding efficiency of Wyner-Ziv video codec up to 0.5 dB for the overall performance and 1 dB for the Wyner-Ziv frames.

References to Chapter 4

[1] J. Ascenso, C. Brites, and F. Pereira. "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding", *5th EURASIP Conf. on Speech and Image Process., Multimedia Commun. and Services*, July 2005.

[2] X. Huang and S. Forchhammer. "Improved side information generation for distributed video coding", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 223–228, Oct. 2008.

[3] L. Natario, C. Brites, J. Ascenso, and F. Pereira. "Extrapolating side information for low-delay pixel-domain distributed video coding", *Int'l Workshop on Very Low Bitrate Video Coding*, Sept. 2005.

[4] S. Borchet, K. Westerlaken, R. Gunnewiek, and R. Lagendijk. "On extrapolating side information in distrbuted video coding", *Picture Coding Symposium*, Nov. 2007.

[5] L. Alparone, M. Barni, F. Bartolini, and V. Cappellini. "Adaptively weighted vector-median filters for motion fields smoothing", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 1996.

[6] S. Klomp, Y. Vatis, and J. Ostermann. "Side information interpolation with sub-pel motion compensation for wyner-ziv decoder", *International Conference on Signal Processing and Multimedia Applications*, Aug 2006.

[7] I. Richardson. *H.264 and MPEG-4 Video Compression.* John Wiley & Sons Ltd, 2003.

[8] S. Lee, O. Kwon, and R. Park. "Weighted-adaptive motion-compensated frame rate-up conversion", *IEEE Trans. Consum. Electron.*, vol. 49, no. 485-492, Aug 2003.

[9] S. Klomp, Y. Vatis, and J. Ostermann. "Side information interpolation with sub-pel motion compensation for wyner-ziv decoder", *International Conference on Signal Processing and Multimedia Applications*, Aug. 2006.

[10] Z. Li, L. Liu, and E. J. Delp. "Rate distortion analysis of motion side estimation in wyner-ziv video coding", *IEEE Trans. on Image processing*, vol. 16, no. 98-113, Jan 2007.

[11] S. Keller. "Video upscaling using variational methods", *PhD Thesis, The Image Group, Department of Computer Science, University of Copenhagen*, 2007.

[12] DISCOVER Project: www.discoverdvc.org, Dec 2007.

Chapter 5

Noise Model for Transform Domain Wyner-Ziv Video Coding

A virtual channel noise model is utilized in Wyner-Ziv decoder to estimate the noise distribution between side information frame and original Wyner-Ziv frame. Since noise distribution decides accuracy of the soft input, the more accurate the noise distribution is, the more reliable soft input is fed into LDPCA decoder and the less syndromes bits will be required. Therefore, noise model gives significant influence on the coding performance of Wyner-Ziv video codec.

A Laplacian distribution is usually utilized to model the noise distribution in state-of-the-art Wyner-Ziv video coding [1]. Accurately estimating the Laplacian distribution parameter is a complex task, because side information frame is not reconstructed at the encoder side and original frame is not available at the decoder side. In some preliminary work [2] [3] [4], the noise distribution is estimated based on offline processing, where the adopted Laplacian parameter is calculated by using the actual noise difference (between Wyner-Ziv frame and side information frame) at the decoder side [3] or through training data [2] [4]. Compared with offline noise model, online noise model [5] [6] [7] is a more adaptive solution which estimates the Laplacian parameter of decoded frame at Wyner-Ziv decoder side. Recently, different granular level online models have been proposed, i.e. from band (frame) level [5]

to coefficient (pixel) [6] [7] level for transform (pixel) domain Wyner-Ziv video coding. The results indicate that including finer granularity in the noise model improves the Rate-Distortion (RD) performance. Following this indication, the objective of this chapter is to further progress the RD performance of transform domain Wyner-Ziv video codec by improving the noise model. For the sake of evaluating the impact of noise model on the coding efficiency of transform domain Wyner-Ziv video coding, the other granular level noise models are introduced and compared.

The structure of this chapter is organized as follows: Laplacian distribution with online noise estimation is introduced in Section 5.1. Then a band level and two coefficient level noise models are described in Section 5.2 and Section 5.3, respectively. In Section 5.4, an improved noise model are proposed for progressing the coding efficiency of transform domain Wyner-Ziv video coding. In Section 5.5, the RD performance results for different noise models are presented and compared.

5.1 Online Noise Estimation

In order to take advantage of side information for decoding, the Wyner-Ziv decoder needs reliable information describing the noise behavior R_{XY} between original Wyner-Ziv frame and side information frame. Since the side information frame is not reconstructed at the encoder side and the original Wyner-Ziv frame is not available at the decoder side, it is not realistic to use frame difference R_{XY} directly. As an adaptive online noise model described in [5], a motion compensated residue R_{ME} is used to describe the correlation noise between original Wyner-Ziv frame and side information frame approximately. Without loss of generality, taking frame interpolation with GOP size 2 as an example, the motion compensated residues R_{ME} described in Eq. 4.6 or Eq. 4.12 are obtained by calculating the difference between frames X'_{2i-1} and X'_{2i+1} after motion compensation. Laplacian distribution is usually utilized to model the difference between original Wyner-Ziv frame X_{2i} and side information frame Y_{2i} in Wyner-Ziv video coding. Based on obtained online noise estimation residue R_{ME}, the Laplacian distribution can be described approximately as:

$$f(X_{2i}(x,y) - Y_{2i}(x,y)) = \frac{\alpha_0}{2} e^{-\alpha_0 |X_{2i}(x,y) - Y_{2i}(x,y)|} \approx \frac{\alpha}{2} e^{-\alpha |R_{ME}(x,y)|} \quad (5.1)$$

$$\alpha_0 = \sqrt{2/\sigma_0^2} \tag{5.2}$$

$$\sigma_0^2 = E((X_{2i} - Y_{2i})^2) - E((X_{2i} - Y_{2i}))^2 \tag{5.3}$$

$$\alpha = \sqrt{2/\sigma^2} \tag{5.4}$$

$$\sigma^2 = E(R_{ME}^2) - E(R_{ME})^2 \tag{5.5}$$

where f is the probability density function, (x, y) is the coordinate in a frame. α_0 and α are Laplacian parameter calculated based on the variance (σ_0^2 and σ^2) of corresponding residue R_{XY} and R_{ME}.

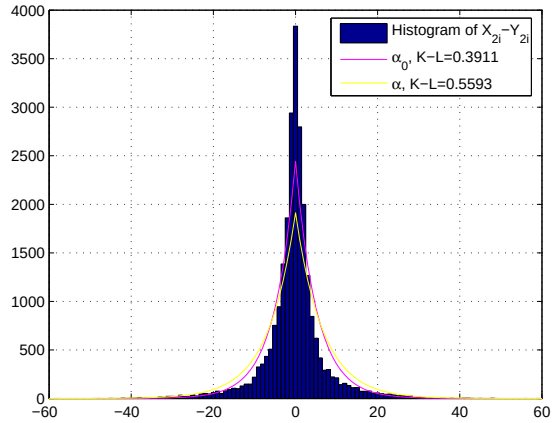


Figure 5.1: Histogram of the actual residue $R_{XY} = X_{2i} - Y_{2i}$ and the estimated Laplacian distributions with residue R_{XY} and R_{ME} (Frame no.4 of Foreman, QCIF).

Fig. 5.1 depicts the histogram of the actual residue $X_{2i} - Y_{2i}$ and the estimated Laplacian distributions with parameter α_0 and α at frame level. Kullback-Leibler distances (KL) [8] are calculated to measure the distance between true histogram and modeling distributions as:

$$D_{KL}(P||Q) = \sum_i P(i) \cdot \log \frac{P(i)}{Q(i)} \tag{5.6}$$

where P and Q are discrete probabilities for the true distributions and the modeled distribution, respectively. It can be seen from Fig. 5.1 that the frame level online noise model is not accurate enough compared with

true histogram. Since estimated Laplacian distribution plays a very important role in converting the side information frame into soft-input information (probabilities), it makes sense to improve the accuracy of online noise model. In the following sections, a band level noise model [5], two coefficient level noise models [6] [7] and an improved noise model [9] will be introduced, respectively.

5.2 Band Level Noise Model

The pixel quality of side information frame is varying not only from frame to frame but also within one frame, thus an accurate noise model should take both temporal and spatial variations into account. Following this intuition, different Laplacian distributions are applied on different frequency bands in transform domain Wyner-Ziv video coding. With the online noise estimation residue R_{ME}, 16 bands of transformed residual coefficients $C_{R_{ME}}^{b_k}, b_k \in \{0...15\}$ are obtained after the 4×4 DCT transform. For a given band b_k, different Laplacian parameters $\alpha_{b_k}^{|\sigma|}$ are utilzied to online model the distribution between transformed coefficients $C_{X_{2i}}^{b_k}$ and $C_{Y_{2i}}^{b_k}$:

$$f(C_{X_{2i}}^{b_k}(u,v) - C_{Y_{2i}}^{b_k}(u,v)) \approx \frac{\alpha_{b_k}^{|\sigma|}}{2} e^{-\alpha_{b_k}^{|\sigma|}|C_{R_{ME}}^{b_k}(u,v)|} \tag{5.7}$$

where f is the probability density function, (u,v) is the coordinate of a block. $\alpha_{b_k}^{|\sigma|}$ are Laplacian parameter calculated based on the variance $\sigma_{|b_k|}^2$.

$$\alpha_{b_k}^{|\sigma|} = \sqrt{2/\sigma_{|b_k|}^2} \tag{5.8}$$

$$\sigma_{|b_k|}^2 = E(|C_{R_{ME}}^{b_k}|^2) - E(|C_{R_{ME}}^{b_k}|)^2 \tag{5.9}$$

where $\sigma_{|b_k|}^2$ is the variance over the absolute value of the transformed motion compensated residue ($|C_{R_{ME}}^{b_k}|$) within band b_k. Different from Eq. 5.5, the absolute value is chosen for Laplacian parameter estimation, since the Laplacian parameter obtained by residue $C_{R_{ME}}^{b_k}$ is generally underestimated (as shown in Fig. C.7 and C.8) compared with the one obtained by residue $C_{R_{XY}}^{b_k} (= C_{X_{2i}}^{b_k} - C_{Y_{2i}}^{b_k})$. It is observed that the distribution with parameter $\alpha_{b_k}^{|\sigma|}$ is closer to the histogram of the actual

residue $C_{R_{XY}}^{b_k}$ especially in the lower frequency band compared with the distribution with the parameter $\alpha_{b_k}^{\sigma}$ obtained by residue $C_{R_{ME}}^{b_k}$ through experiments [6].(See also Fig. 5.2).

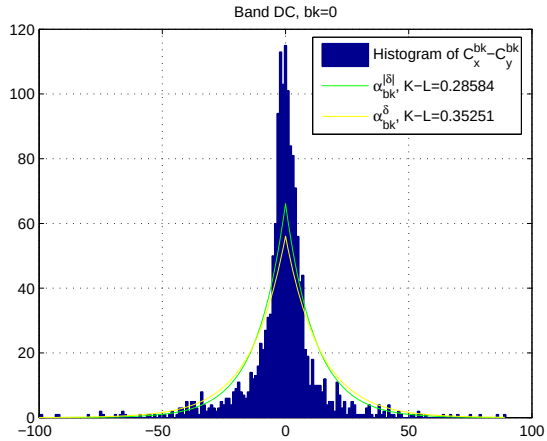


Figure 5.2: Histogram of the actual residue $C_{R_{XY}}^0 = C_{X_{2i}}^0 - C_{Y_{2i}}^0$ and the estimated distributions with $|C_{R_{ME}}^{b_k}|$ and $C_{R_{ME}}^{b_k}$ (DC coefficients, frame no.22 of Foreman, QCIF).

5.3 Coefficient Level Noise Model

In the band level noise model, the same Laplacian parameter $\alpha_{b_k}^{|\sigma|}$ is utilized for all the coefficients within band b_k. The spatial variation between different blocks is not explored yet, thus a coefficient level noise model ($c1$) is proposed in [6] to further exploit spatial variation.

$$\alpha_{b_k}^{c1}(u,v) = \begin{cases} \alpha_{b_k}^{|\sigma|}, & \text{if } (u,v) \in map_{b_k}^{in} \\ \sqrt{2/D(u,v)^2}, & \text{if } (u,v) \in map_{b_k}^{out} \end{cases} \tag{5.10}$$

where

$$map_{b_k}^{in} = \{(u,v) | D(u,v)^2 \leq \sigma_{|b_k|}^2\} \tag{5.11}$$

$$map_{b_k}^{out} = \{(u,v) | D(u,v)^2 > \sigma_{|b_k|}^2\} \tag{5.12}$$

$$D(u,v) = C_{R_{ME}}^{b_k}(u,v) - E(|C_{R_{ME}}^{b_k}|) \tag{5.13}$$

where $\alpha_{b_k}^{c1}(u,v)$ represents the estimated Laplacian parameter for the coefficient located at (u,v) within band b_k. $\alpha_{b_k}^{|\sigma|}$ and $\sigma_{|b_k|}^2$ are estimates of the Laplacian parameter and the variance at band level as described in Eqs. 5.8 and 5.9 . $E(|C_{R_{ME}}^{b_k}|)$ represents the average absolute value of coefficients in band b_k. $C_{R_{ME}}^{b_k}(u,v)$ is the coefficient value at position (u,v) within band b_k. This coefficient level noise model divides coefficients into two categories (inlier $map_{b_k}^{in}$ and outlier $map_{b_k}^{out}$) by comparing D^2 and the variance $\sigma_{|b_k|}^2$. If D^2 is smaller than the variance, the band level Laplacian parameter $\alpha_{b_k}^{|\sigma|}$ is applied. Otherwise, the coefficient level parameter $\sqrt{2/D(u,v)^2}$ is assigned [6].

Alternatively, a pixel level noise model is proposed in [7] for pixel domain Wyner-Ziv video coding, which each pixel is adaptively assigned with a Laplacian parameter based on pixel's reliability. This work is here extended to a weighted coefficient level noise model ($c2$) for transform domain Wyner-Ziv video coding which weights band level and coefficient level statistics.

$$\alpha_{b_k}^{c2}(u,v) = \frac{\beta \cdot E(|C_{R_{ME}}^{b_k}|) \cdot \alpha_{b_k}^{|\sigma|}}{(\beta-1)\cdot|C_{R_{ME}}^{b_k}(u,v)| + E(|C_{R_{ME}}^{b_k}|)} \tag{5.14}$$

where parameter β determines the amplitude of the deviations of $\alpha_{b_k}^{c2}(u,v)$ from $\alpha_{b_k}^{|\sigma|}$. $\beta = 2$ was chosen experimentally [7]. Generally, this noise model assigns Laplacian parameters adaptively based on the absolute magnitude of the transformed motion compensated residue. The larger the absolute transformed residue $|C_{R_{ME}}^{b_k}(u,v)|$ is, the less reliable it is, and therefore a smaller Laplacian parameter $\alpha_{b_k}(u,v)$ is assigned.

5.4 Improved Noise Model

As described in Sections 5.2 and 5.3, the variance $\sigma_{|b_k|}^2$ is utilized to estimate the Laplacian parameter at band level (Eq. 5.9) which in turn influences the estimated coefficient level (Eqs. 5.10 and 5.14). The maximum likehood estimator can also be used to estimate the Laplacian

parameter:

$$\alpha_{b_k}^{|b|} = ((\sum ||C_{R_{ME}}^{b_k}| - E(|C_{R_{ME}}^{b_k}|)|)/N)^{-1} \tag{5.15}$$

Assuming a Laplacian distribution, these two different estimators (Eqs. 5.8 and 5.15) should give the same parameter value (i.e. $\alpha_{b_k}^{|b|} = \alpha_{b_k}^{|\sigma|}$). However, as shown in Fig. 5.3, the experiments indicate that $\alpha_{b_k}^{|b|}$ is generally larger than $\alpha_{b_k}^{|\sigma|}$ especially in lower frequency band. The histogram of the actual residue $C_{R_{XY}}^{b_k}$ is more peaked and has longer tails than the assumed Laplacian distribution. $\alpha_{b_k}^{|b|}$ is closer to the histogram close to zero while the $\alpha_{b_k}^{|\sigma|}$ is closer at the high values. Therefore it is reasonable to classify coefficients into two categories and apply the estimators $\alpha_{b_k}^{|b|}$ (Eq. 5.8) and $\alpha_{b_k}^{|\sigma|}$ (Eq. 5.15) for each category, respectively. Further, these estimators will be based on the coefficients within the respective category.

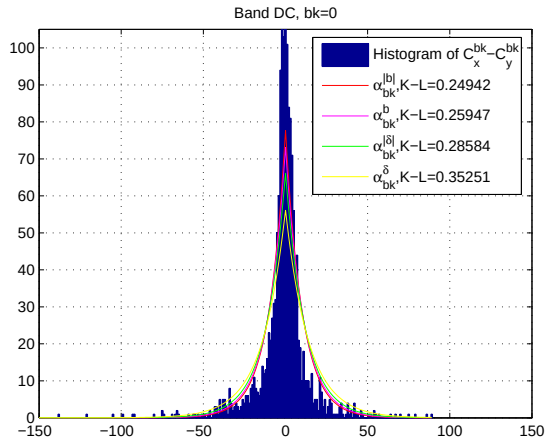


Figure 5.3: Histogram of the actual residue $C_{R_{XY}}^0 = C_{X_{2i}}^0 - C_{Y_{2i}}^0$ and the estimated distributions with different estimators (DC coefficients, frame no.22 of Foreman).

The coefficient level noise model [6] classifies coefficients by comparing $D(u,v)^2$ and the variance $\sigma_{|b_k|}^2$ as shown in Eqs. 5.11 and 5.12. However, this calculation is only based on motion compensated residue

$C_{R_{ME}}^{b_k}$, which may be unreliable in some regions. Only using $C_{R_{ME}}^{b_k}$ (Eqs. 5.10- 5.13) may lead to inaccurate local parameter calculation. The correlation between classifications of different bands is tested in Fig. 5.4 based on comparing $D(u,v)^2$ and $\sigma_{|b_k|}^2$ of the actual residue $C_{R_{XY}}^{b_k}$.The results indicate that there exist some cross-band correlations, which can be utilized for category classification.

Since the Wyner-Ziv frames can be decoded successively band by band, after successfully decoding one (lower frequency) band b_k, an unfinished decoded frame (Z) can be reconstructed. By calculating the coefficients difference between $C_Z^{b_k}$ and $C_{Y_{2i}}^{b_k}$, an updated residue $C_{R_{ZY}}^{b_k}$ in band b_k is obtained, which is closer to the actual residue $C_{R_{XY}}^{b_k}$ than the motion compensated residue $C_{R_{ME}}^{b_k}$. The $\sigma_{|b_k|}^2$ and $D(u,v)^2$ in Eqs. 5.9 and 5.13 are recalculated based on the updated residue $C_{R_{ZY}}^{b_k}$, therefore the updated classification map of band b_k can be obtained by refined values of $\sigma_{|b_k|}^2$ and $D(u,v)^2$ based on Eqs. 5.11 and 5.12.

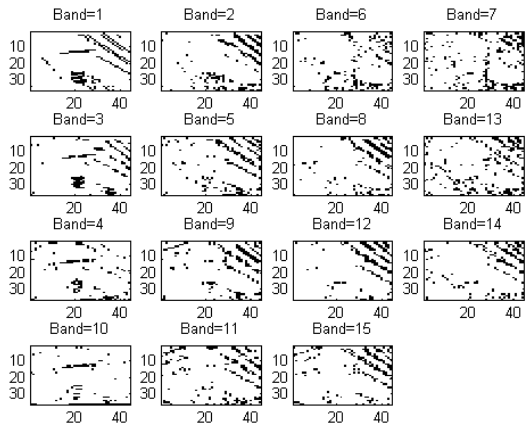


Figure 5.4: Coefficient classification within different bands tested on the actual residue $C_{R_{XY}}^{b_k}$ (Frame no.22 of Foreman)

Due to the existing cross-band correlation as shown in Fig. 5.4, classification map of band b_k can be utilized to estimate the classification map of the next (higher frequency) band $b_l, l > k$. The classification estimation follows the decoding order as shown in Fig. 5.5. The estimation

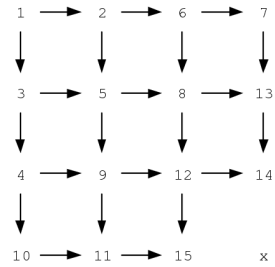


Figure 5.5: The classification estimation from lower frequency band to higher frequency band

function f_{esti} can be denoted as:

$$map^{\bullet}_{b_l} = f_{esti}(map^{\bullet}_{b_k}) \tag{5.16}$$

where $map^{\bullet}_{b_l}$ is the estimated classification map of higher frequency band b_l and $map^{\bullet}_{b_k}$ are updated classification maps based on decoded lower frequency band b_k. The estimation function f_{esti} is simply based on copy or union operations. For instance, after the first band is successfully decoded, the classification map of band 1 (map^{out}_1,map^{in}_1) is obtained as described in Eqs. 5.11 and 5.12. The classification maps of band 2 and band 3 are simply estimated by copying the *map* of the neighboring band 1, i.e. $map^{out}_3 = map^{out}_2 = map^{out}_1$ and $map^{in}_3 = map^{in}_2 = map^{in}_1$. Similarly, the classification map of band 5 is estimated by using band 2 and band 3 by $map^{out}_5 = map^{out}_2 \cup map^{out}_3$ and $map^{in}_5 = map^{in}_2 \cup map^{in}_3$ etc. With the estimated classification, $\alpha^{|b|}_{b_k}$ and $\alpha^{|\sigma|}_{b_k}$ can be calculated within the coefficient sets $map^{in}_{b_k}$ and $map^{out}_{b_k}$, respectively.

$$\alpha^{|b|}_{map^{in}_{b_k}} = ((\sum ||C^{map^{in}_{b_k}}_{R_{ME}}| - E(|C^{map^{in}_{b_k}}_{R_{ME}}|)|)/N)^{-1} \tag{5.17}$$

$$\alpha^{|\sigma|}_{map^{out}_{b_k}} = \sqrt{2/(E(|C^{map^{out}_{b_k}}_{R_{ME}}|^2) - E(|C^{map^{out}_{b_k}}_{R_{ME}}|)^2)} \tag{5.18}$$

In order to combine the advantages of the two coefficient level noise models described in the Section 5.3, the Laplacian parameters for lower frequency bands and higher frequency bands are assigned differently. Let $\alpha^{c2}_{b_k}[(u,v)|C^{map^{\bullet}_{b_k}}_{R_{ME}}, \alpha^{|\sigma|}_{b_k}]$ denote the function in Eq. 5.14. For coefficients

$C_{R_{ME}}^{b_k}, b_k \in \{0,1,2\}$,

$$\alpha_{b_k}(\mathrm{u,v}) = \begin{cases} \alpha_{b_k}^{c2}[(u,v)|C_{R_{ME}}^{map_{b_k}^{in}}, \alpha_{map_{b_k}^{in}}^{|b|}] & (u,v) \in map_{b_k}^{in} \\ \alpha_{b_k}^{c2}[(u,v)|C_{R_{ME}}^{map_{b_k}^{out}}, \alpha_{map_{b_k}^{out}}^{|\sigma|}] & (u,v) \in map_{b_k}^{out} \end{cases} \tag{5.19}$$

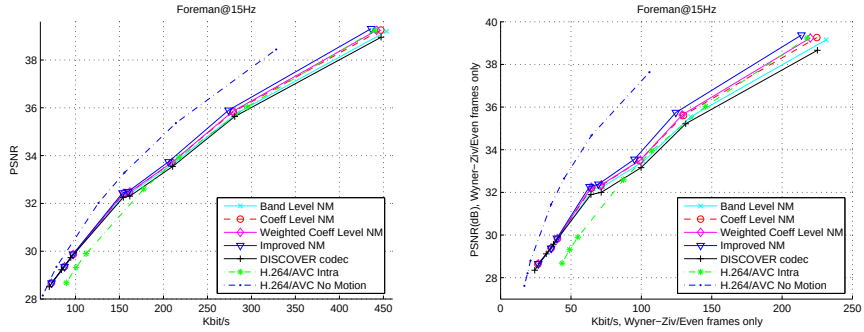
For coefficients $C_{R_{ME}}^{b_k}, b_k \in \{3...15\}$,

$$\alpha_{b_k}(\mathrm{u,v}) = \begin{cases} \alpha_{map_{b_k}^{out}}^{|\sigma|} & \text{if } \sqrt{2/D(u,v)^2} \geq \alpha_{map_{b_k}^{out}}^{|\sigma|} \\ & \cup (u,v) \in map_{b_k}^{out} \\ \alpha_{map_{b_k}^{in}}^{|b|} & \text{if } \sqrt{2/D(u,v)^2} \geq \alpha_{map_{b_k}^{in}}^{|b|} \\ & \cup (u,v) \in map_{b_k}^{in} \\ \sqrt{2/D(u,v)^2}, & \text{otherwise} \end{cases} \tag{5.20}$$

5.5 Experimental Results

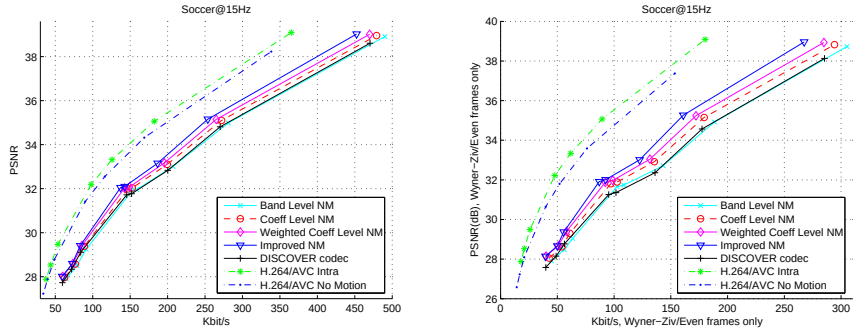
In order to demonstrate the effects introduced by different noise models, RD performances of Wyner-Ziv video coding with the band level noise model, coefficient level noise model, weighted coefficient level noise model and improved noise model are compared. For the sake of fair comparisons, the test conditions adopted are the DISCOVER project [10] test conditions described in Section 3.4. The OBMC based frame interpolation [11] described in Chapter 4 is employed as side information generation method. The test sequences are *Foreman*, *Soccer*, *Coastguard* and *Hall Monitor*, at QCIF, 15 frames per second (fps); the GOP size is 2. The key frames are encoded using H.264/AVC Intra and the QPs are chosen so that the average PSNR of the WZ frames is similar to the average PSNR of the key frames (as in [10]). The RD performance is evaluated for the luminance component of both the key frames and WZ frames. The benchmark codecs used are the DISCOVER Wyner-Ziv video codec [10], the H.264/AVC Intra codec and the H.264/AVC no motion codec.

According to RD results shown in Figs. 5.6-5.9, Wyner-Ziv video coding with the band level noise model is seen as a baseline. The two different coefficient level noise models achieve better RD performance than band level noise model. Compared with the coefficient level model [6]



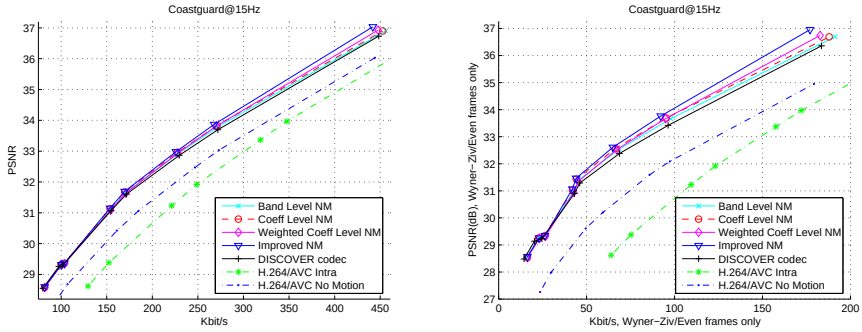
(a) (b)

Figure 5.6: GOP2 RD performances comparison for sequence Foreman with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)



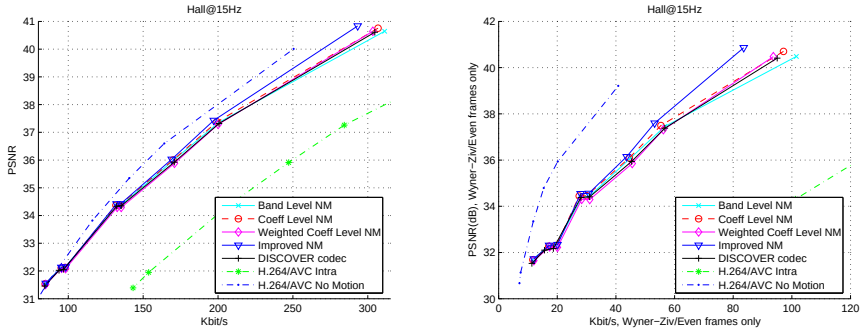
(a) (b)

Figure 5.7: GOP2 RD performances comparison for sequence Soccer with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)



(a) (b)

Figure 5.8: GOP2 RD performances comparison for sequence Coastguard with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)



(a) (b)

Figure 5.9: GOP2 RD performances comparison for sequence Hall monitor with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames. (i.e. half frame rate)

(Eq. 5.10) employed in the DISCOVER codec, the weighted coefficient level model (Eq. 5.14) provides better RD performance results for sequences *Foreman*, *Soccer* and *Coastguard*, but worse RD performance for sequence *Hallmonitor*. The improved noise model achieves better RD performance than all the other noise models. Compared with the coefficient level noise models, the improved noise model is more robust and it improves the coding efficiency of high bit-rates sequences up to 0.5 dB for the overall RD performances and 1 dB for the Wyner-Ziv frames.

With proposed noise model, Wyner-Ziv video coding gives better RD performance than H.264/AVC Intra coding not only for relative low motion sequence *Hall Monitor* but also for the medium motion sequence *Coastguard* and *Foreman*; For sequences with very high motion like *Soccer*, the performance gap between H.264/AVC Intra coding and Wyner-Ziv video coding has been further reduced but not closed yet. Compared with H.264/AVC no motion, Wyner-Ziv video coding provides the better coding performance for *Coastguard* but worse performance for *Foreman*, *Soccer* and *Hall Monitor*. However, the gap between H.264/AVC no motion and Wyner-Ziv video coding has been reduced. For Wyner-Ziv video coding with larger GOP sizes, the proposed noise model can provide similar RD performance improvements. The results are reported in Figs. C.9-C.12. in Appendix C. However, winning against H.264/AVC Intra and no motion Inter codec is getting more difficult, since the distance between key frames become far way.

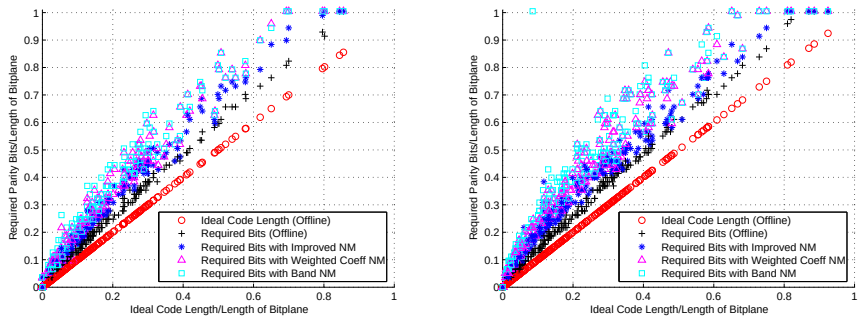
According to the Wyner-Ziv theorem, practical Wyner-Ziv video codec should have similar RD performance with conventional predictive video codec. However, there still exists a large gap between practical Wyner-Ziv video codec and conventional predictive video codec. The performance loss of practical Wyner-Ziv video codec may be introduced by the low quality of side information frame, the inaccurate noise model and the less efficient Slepian-Wolf codec. In order to evaluate the compression efficiency of employed Slepian-Wolf codec, i.e. LDPCA codec [12], Ideal Code Length (ICL) is defined representing the amount of required coding bits by using an ideal arithmetic coding [13] with given soft input. For one bitplane x, it is assumed that estimated soft input $\hat{P}$ fed into LDPCA codec is available at the encoder. The ICL is

calculated as:

$$ICL(x) = \sum_{j=0}^{n} -log\hat{P}(x_j) \tag{5.21}$$

where $x_j \in \{0, 1\}$ and $\hat{P}(x_j)$ represents the probability estimate of x_j, the symbol with index j. The soft input $\hat{P}(x_j)$ is obtained based on a given noise model, the calculation is described in Section 3.3.6.

In order to avoid the influence introduced by different noise models, the required syndrome bits of LDPCA and the ideal code length are compared based on an offline scenario at first. In the offline setting, it is assumed that the the actual difference between original Wyner-Ziv frame and side information frame is known, thus the soft input can be obtained based on the histogram of actual residue. As shown in Fig. 5.10, the gap between Ideal Code Length (offline) and required syndrome bits (offline) indicates that LDPCA codec indeed introduces compression loss. On the other hand, practical required syndrome bits of LDPCA decoder are also measured in order to evaluate different online noise models. As in Fig. 5.10, it can be seen that the amount of required syndrome bits keep approaching to the ideal code length by improving the noise model from band level to improved level [9].



(a) Foreman (b) Soccer

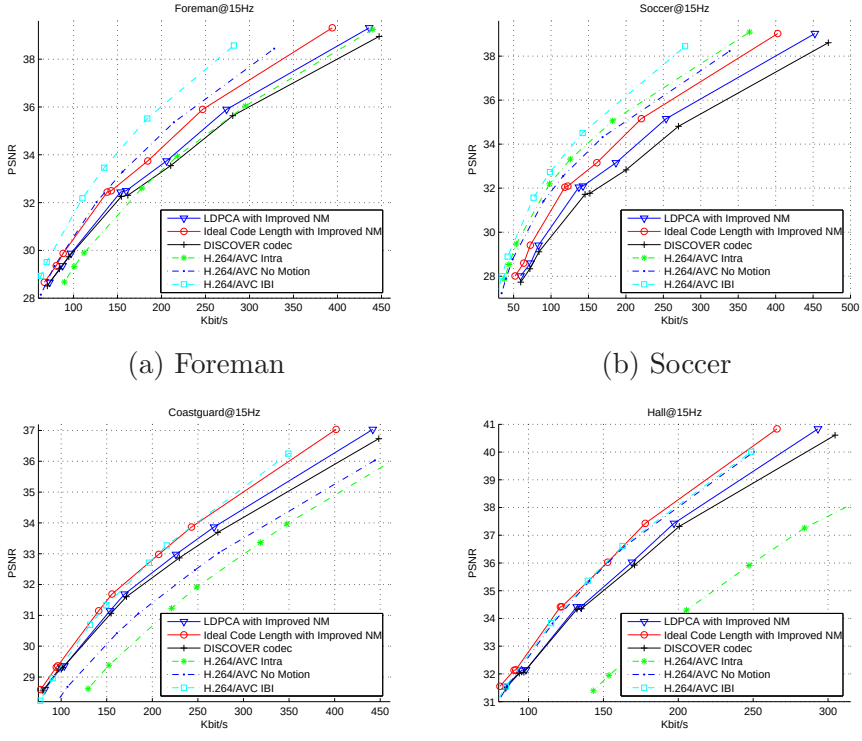
Figure 5.10: Ideal Code Length vs. Required parity bits with different noise models, frame No.2, 4, 6 and 8 of (a) foreman and (b) soccer

With the given improved online noise model [9], RD performances of

Wyner-Ziv video codec with LDPCA coding and assumed ICL coding are compared in Fig. 5.11. Similar results are reported in Fig C.13 in Appendix C if the coefficient level noise model [6] is applied. As shown in Fig. 5.11, it can be seen that there are significant compression loss introduced by LDPCA codec compared with Ideal Code Length. The loss could be caused by different designs on the rate-adaptivity [14], diverse degree distributions of LDPC codes (i.e. regular and irregular degree distributions [12]), finite code length penalty of LDPCA codes and so on. Taking the finite code length penalty as an example, costing bitrate of Wyner-Ziv video codec with LDPCA code length 1584 and 6336 are compared in Table 5.1. For fair comparison, the only difference between LDPCA 1584 and LDPCA 6336 based Wyner-Ziv video codec is that LDPCA 6336 based Wyner-Ziv video codec process four QCIF Wyner-Ziv frames parallel in the test. It can be observed from Table 5.1 that the required bitrate can be slightly reduced by increasing the code length of LDPCA codes from 1584 to 6336.

5.6 Summary

In this chapter, different online noise models within band level and coefficient level are introduced. The impact of different noise models on the coding efficiency of transform domain Wyner-Ziv video coding is evaluated. It shows that RD performance of Wyner-Ziv video codec highly depends on the accuracy of noise model. Therefore, an improved noise model is proposed with the objective to further improve the overall RD performance. The proposed noise model utilizes cross-band correlations to classify coefficients and applies two estimators in different regions, therefore the more accurate Laplacian parameter is obtained. Compared with the best available noise models, the improved noise model can improve coding efficiency up to 0.5 dB for the overall RD performances and 1 dB for the Wyner-Ziv frames. Meanwhile, the existing gap between Wyner-Ziv video codec and conventional video codec H.264/AVC is analyzed. Experimental results indicate that there is some compression loss in Wyner-Ziv video coding introduced by the employed Slepian-Wolf codec, i.e. LDPCA codes, besides the loss caused by low quality side information frame and inaccurate noise model.



(a) Foreman (b) Soccer

(c) Coastguard (d) Hall Monitor

Figure 5.11: RD performance comparison with LDPCA coding and Ideal Code Length

Soccer				
RD Q_i	PSNR (dB)	LDPCA 1584	LDPCA 6336	Δ (Kbits)
8	38.9423	269.34	267.11	-2.23
6	32.9900	123.21	121.71	-1.50
4	31.8949	87.568	86.650	-0.92
2	28.6559	50.222	49.970	-0.25
Foreman				
RD Q_i	PSNR (dB)	LDPCA 1584	LDPCA 6336	Δ (Kbits)
8	39.3753	214.92	214.43	-0.49
6	33.5439	95.802	95.427	-0.38
4	32.2669	63.698	63.069	-0.63
2	29.3838	36.241	35.993	-0.25
Coastguard				
RD Q_i	PSNR (dB)	LDPCA 1584	LDPCA 6336	Δ (Kbits)
8	36.9427	177.84	174.55	-3.29
6	32.5897	65.222	63.334	-1.89
4	31.0476	42.038	40.721	-1.32
2	29.2357	23.090	22.707	-0.38
Hall Monitor				
RD Q_i	PSNR (dB)	LDPCA 1584	LDPCA 6336	Δ (Kbits)
8	40.8666	83.466	79.250	-4.22
6	36.1565	43.590	40.780	-2.81
4	34.5542	27.881	25.920	-1.96
2	32.3211	17.173	16.410	-0.76

Table 5.1: Bitrate comparison of LDPCA codes with length 1584 and 6336

References to Chapter 5

[1] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret. "The discover codec: Architecture, techniques and evaluation", *Picture Coding Symposium*, Nov. 2007.

[2] A. Aaron, S. Rane, E. Setton, and B. Girod. "Transform domain wyner-ziv codec for video", *Proc. SPIE VCIP*, pp. 520–528, Jan 2004.

[3] C. Brites, J. Ascenso, and F. Pereira. "Improving transform domain wyner-ziv video coding performance", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2006.

[4] G. Esmaili and P. Cosman. "Correlation noise classification based on matching success for transform domain wyner-ziv video coding", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2009.

[5] C. Brites, J. Ascenso, and F. Pereira. "Studying temporal correlation noise modeling for pixel based wyner-ziv video coding", *IEEE International Conference on Image Processing*, Oct. 2006.

[6] C. Brites and F. Pereira. "Correlation noise modelling for efficient pixel and transform domain wyner-ziv video coding", *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 18, no. 9, Sept. 2008.

[7] L. Qing, X. He, and R. Lv. "Distributed video coding with dynamic virtual channel mode estimation", *Int'l Symposium on Data, Privacy and E-Commerce*, pp. 170–173, 2007.

[8] S. Kullback. *Information theory and statistics*. John Wiley and Sons, 1959.

[9] X. Huang and S. Forchhammer. "Improved virtual channel noise model for transform domain wyner-ziv video coding", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2009.

[10] DISCOVER Project: www.discoverdvc.org, Dec 2007.

[11] X. Huang and S. Forchhammer. "Improved side information generation for distributed video coding", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 223–228, Oct. 2008.

[12] D. Varodayan, A. Aaron, and B. Girod. "Rate-adaptive distributed source coding using low-density parity-check codes", *EURASIP Signal Process. Journal, Special Section on Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[13] K. Sayood. *Introduction to Data Compression*. Morgan Kaufmann, Second edition, 2000.

[14] J. Ascenso, C. Brites, and F. Pereira. "Design and performance of a novel low-complexity parity-check code for distributed video coding", *IEEE International Conference on Image Processing*, Oct. 2008.

Chapter 6

Wyner-Ziv Decoder with Multiple Side Information

As important factors to influence the coding performance of a Wyner-Ziv video codec, the impact of side information generation and noise model have been discussed in Chapters 4 and 5. However, based on the architecture of transform domain Wyner-Ziv video codec [1], it can be seen that both the generated side information frame and noise model are utilized to estimate the soft input information (conditional bit probabilities) for bitplanes. Soft input information is the essential factor to reduce the number of coding bits of Wyner-Ziv video decoder. The more reliable soft input is, the fewer syndrome bits are required by decoder since the faster convergence will be. Thus, an important way to progress RD performance of Wyner-Ziv video codec is to improve the reliability of soft input information fed into the LDPCA [2] decoder.

A multiple side information based Wyner-Ziv decoder has been proposed in [3], where two different frame interpolation methods are employed to generate the multiple side information. The LDPCA decoder is fed with average value of two soft inputs which are generated based on two different side information estimates and the corresponding noise models. A more reliable soft input is obtained and the RD performance is improved up to 0.3 dB.

Differently, in this chapter, a novel multiple side information based Wyner-Ziv decoder with frame interpolation [4] [5] and extrapolation [6] [7] is proposed. The intuition is that having more different side informa-

tion solutions should allow these to compensate each other's estimation weaknesses depending on the video content, overall leading to a more efficient coding solution. In this context, the extrapolated and the interpolated side information frames can be seen as original frames transmitted through quite different 'channels' and thus each side information frame is seen as an observation with a different amount of 'correlation noise'. With multiple observations, the Wyner-Ziv video decoder can select or combine the available side information estimations to decrease the amount of 'correlation noise' and thus to reduce misleading soft inputs in comparison with the single side information solution. In this way, the novel proposed solution shall reduce the required syndrome bits for each target quality. Therefore, the objective of the proposed Wyner-Ziv decoder is to further progress the RD performance of Wyner-Ziv video coding, also reducing the RD gap regarding conventional video coding such as the H.264/AVC standard, by exploiting not a single but multiple side information.

The rest of this chapter is organized as follows: In Section 6.1, the general structure of the novel Wyner-Ziv decoder with interpolated and extrapolated side information is described. Two main technical novel modules regarding the noise estimation for extrapolation and the soft input combination are described in Section 6.2 and Section 6.3, respectively. Finally, performance results with single and multiple side information are compared in Section 6.4.

6.1 Architecture

The basic idea of the proposed Wyner-Ziv video decoder with multiple side information is to generate better soft-input information by generating first better quality side information, in this case multiple side information through interpolation and extrapolation. The proposed Wyner-Ziv video decoder expects to improve the overall RD performance by also processing extrapolation side information which may be 'better' than interpolation side information for some conditions of the content. The architecture proposed for the novel WZ decoder with multiple side information is presented in Fig. 6.1. The encoder is not changed, the track at the right starting with interpolation (RI and YI) in Fig. 6.1 presents a state-of-art Wyner-Ziv decoder with interpolation. The tech-

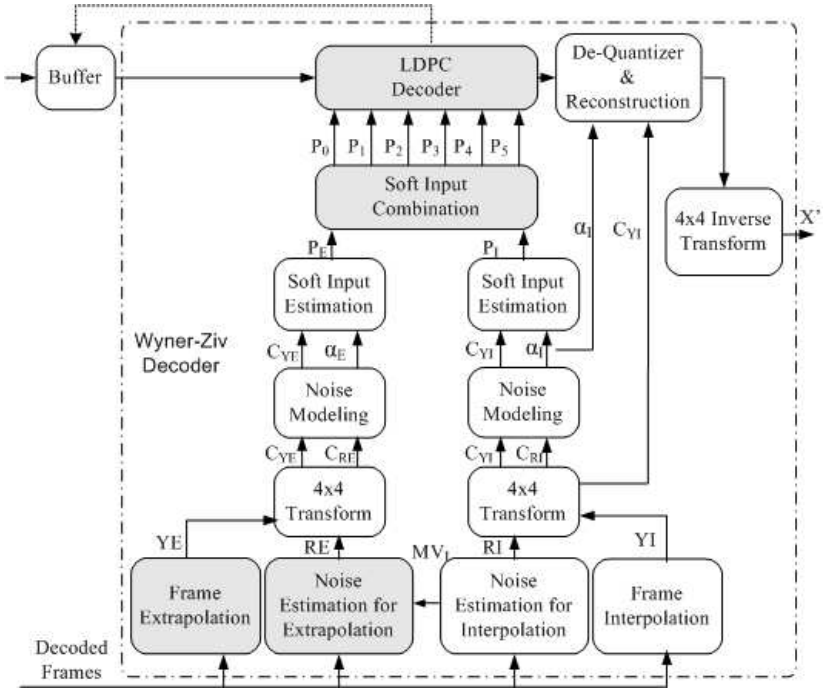


Figure 6.1: Transform domain Wyner-Ziv video decoder with interpolated and extrapolated side information [8]

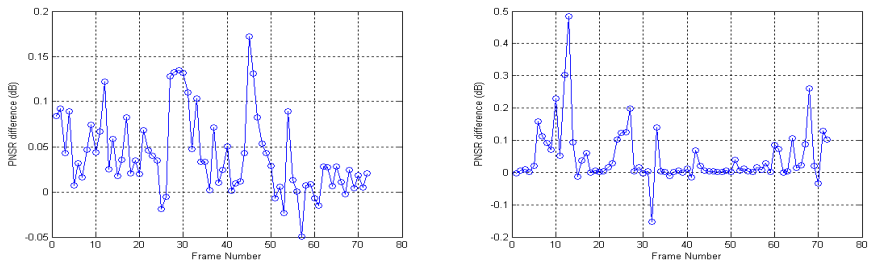
nical novelty of the proposed Wyner-Ziv video decoder includes: i) the noise estimation for extrapolation, ii) the soft inputs combination module, and iii) modified LDPC decoder. The main modules in the novel proposed WZ video decoder are:

• ***Frame Interpolation***: The adopted frame interpolation procedure is introduced in Section 4.2.2 and [5]. Without loss of generality, it generates the side information frame YI_{2i} by using Intra coded frames, X'_{2i-1} and X'_{2i+1} for GOP size 2. It includes forward motion estimation, bi-directional motion estimation, spatial smoothing of Motion Vectors(MV), motion refinement with variable block size and adaptive weighted Overlapped Block Motion Compensation (OBMC).

• ***Noise Estimation for Interpolation***: A motion estimated residue R_{ME} as in Eq. 4.12 (i.e. the difference between X'_{2i-1} and X'_{2i+1} after motion compensation) is taken as the estimated noise residue

RI to express the correlation noise between the Wyner-Ziv frame X_{2i} and the corresponding interpolated frame YI_{2i}.

• ***Frame Extrapolation***: This module creates the extrapolated side information. The procedure is similar to the method introduced in Section 4.1 and [7]. Without loss of generality, the previous coded frames X'_{2i-1} and X'_{2i-2} are used to generate the side information frame YE_{2i} for GOP size 2. It includes motion estimation, spatial smoothing, frame projection, overlapping and filling holes. The difference is that a novel hole filling technique is applied. For the unreferenced/unfilled pixel areas in frame YE_{2i}, both the nearest MVs in the spatial domain and co-located MVs in temporal domain are used to determine the estimated pixels. An average value of these estimates is computed for filling the holes remaining after the frame projection process. As shown in Fig. 6.2, compared with hole filling solution using only co-located MVs in temporal domain, PSNR improvement is achieved by using both of the nearest MVs in the spatial domain and co-located MVs in temporal domain to fill the holes.



(a) Foreman (b) Hall Monitor

Figure 6.2: PSNR improvement by using both of the nearest MVs in the spatial domain and co-located MVs in temporal domain to fill the holes

• ***Noise Estimation for Extrapolation***: The noise residue RE is computed to present the correlation noise between the Wyner-Ziv frame X_{2i} and the corresponding extrapolated side information frame YE_{2i}. Different from noise estimation for interpolation by using a motion estimated residue frame R_{ME}, a combined noise residue is adopted as described in Section 6.2.

• ***Noise Modeling***: After computing the 4×4 integer DCT coefficients C_{YI}, C_{YE}, C_{RI} and C_{RE} for the interpolated and extrapolated side information and the associated residues, the noise distribution between the side information and the corresponding Wyner-Ziv frames is estimated using a Laplacian noise model as described in Section 5.3 or 5.4. Within a given DCT band b_k, the DCT coefficient at coordinates (m, n) is associated to the Laplacian parameter $\alpha_E^{bk}(m, n)$ for extrapolation and $\alpha_I^{bk}(m, n)$ for interpolation. The Laplacian parameter values express the reliability of the side information, i.e. the smaller this value is, the noisier the corresponding coefficient is.

• ***Soft Input Estimation***: With the obtained Laplacian parameters, side information coefficient values and the previous successfully decoded bitplanes, the soft input information (conditional bit probabilities for extrapolation P_E and for interpolation P_I) of each bitplane is estimated as in Section 3.3.6.

• ***Soft Input Combination***: The soft input information to be provided to the LDPCA decoder is generated by combining the soft inputs P_E and P_I in a few predefined modes creating various soft input candidates; see details in Section 6.3.

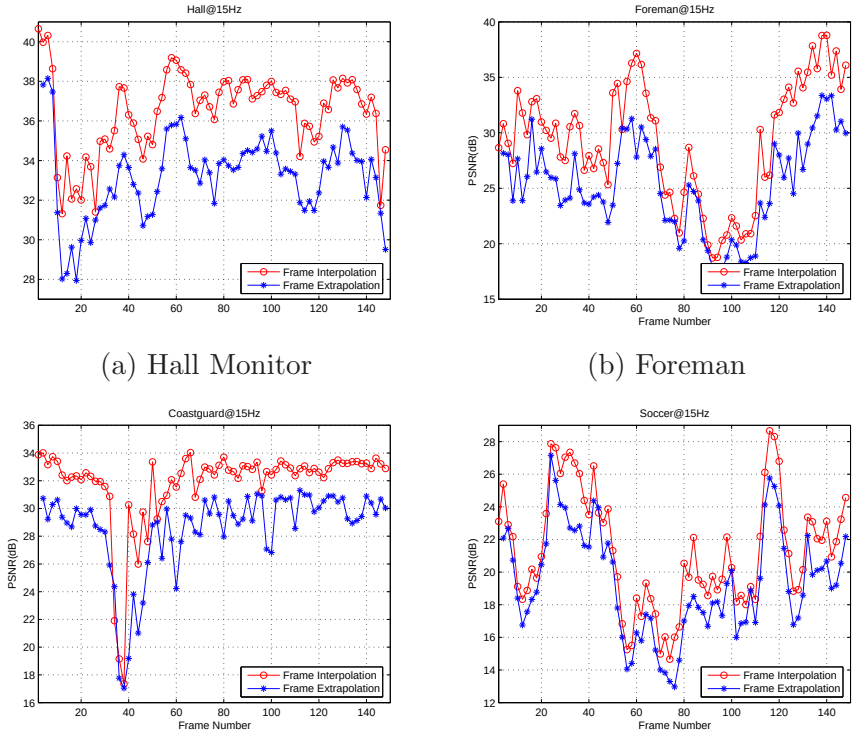
• ***LDPCA Decoder***: All these candidate soft inputs are fed to a modified LDPCA decoder. The soft input which converges (as described in Section 3.3.7) first is chosen by the LDPCA decoder thus minimizing the rate of syndrome bits for a certain target quality.

• ***Reconstruction***: Based on the decoded bins, this module has to recover the coefficient's values also exploiting the available side information. Since the interpolated side information is typically better (see Fig. 6.3), the interpolated side information and its noise modeling parameters are used by the reconstruction module (as described in Section 3.3.8) to recover the decoded Wyner-Ziv frames.

6.2 Noise Estimation for Extrapolation

There are two natural ways to estimate the residue between Wyner-Ziv frames and the corresponding extrapolated side information to represent the correlation noise behavior:

• *Motion Estimated Residue* R_{ME}: Corresponds to the pixel differences between X'_{2i-1} and X'_{2i-2} along the extrapolated MVs.



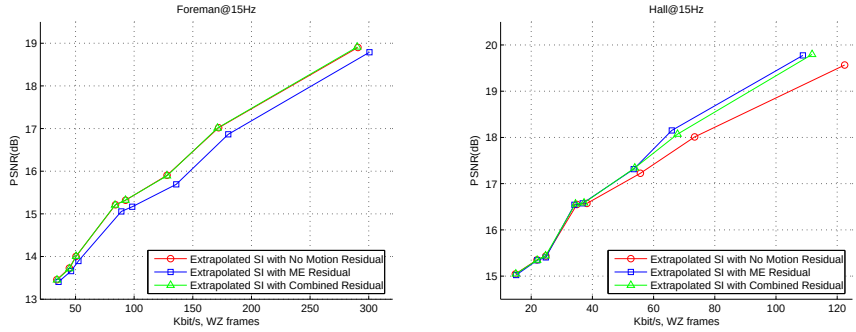
(a) Hall Monitor (b) Foreman

(c) Coastguard (d) Soccer

Figure 6.3: PSNR comparison for the interpolation and extrapolation methods, sequence@15Hz, QCIF, GOP 2, Key frame H.264/AVC Intra coded

• *No Motion Estimated Residue* R_{NO}: Corresponds to the co-located pixel differences between YE_{2i} and X'_{2i-1}. i.e. $R_{NO} = YE_{2i}(x,y) - X'_{2i-1}(x,y)$.

Experiments have shown that, when creating the side information using frame extrapolation, the more commonly used motion estimated residue [9] [10] provides a worse RD performance for high motion sequences while it performs better for low motion sequences in comparison with the no motion estimated residue (see Fig. 6.4). The worse RD performance may be caused by the linear motion assumption adopted for the generation of the unidirectional MVs used for the frame extrapolation process. If these MVs are not fulfilling this assumption, then the



(a) Foreman (b) Hall Monitor

Figure 6.4: RD performances with extrapolated side information using the motion estimated and no motion estimated residues for (a) Foreman and (b) Hall Monitor, QCIF, 15 Hz.

extrapolated block is going to be projected into a wrong position, corresponding to a large real noise residue, while the motion estimated residue R_{ME} will be smaller. Based on this poorly estimated noise residue, the estimated Laplacian parameter will be inaccurate in terms of noise modeling, misleading the LPDC decoder in terms of the soft input P_E. In order to solve this problem, it is necessary to generate a more robust estimate for the noise residue when frame extrapolation is used. In this context, it is proposed here to check the 'accuracy' of the motion vectors obtained by extrapolation MV_E using the motion vectors obtained by frame interpolation MV_I. The intuition is that if the two sets of MVs are similar, then the motion description should be good and thus the motion estimated residue should be used. Following this intuition, a combined noise residue, R_{COM}, is computed by switching between R_{ME} and R_{NO} as:

$$R_{COM}(x,y) = \begin{cases} R_{ME}(x,y), & \text{if } MV_I(m,n)=MV_E(m,n) \\ R_{NO}(x,y), & \text{otherwise} \end{cases} \tag{6.1}$$

where (x, y) are the pixel coordinates and (m, n) are the corresponding block coordinates. The RD performance with single extrapolation side information using the proposed combined noise residue is compared with

the relevant alternatives in Fig. 6.4 for the Foreman and Hall Monitor sequences.

6.3 Soft Input Combination

After the extrapolation soft input P_E and the interpolation soft input P_I are obtained, the soft input combination module has the task of adaptively combining these two soft inputs to generate a set of candidate soft inputs, thus improving the RD performance by reducing the rate of syndrome bits.

Since the values of the Laplacian parameters should express the reliability of the corresponding side information, an unreliability region MAP_{un} is defined as the region of the frame where extrapolation or interpolation indicates areas including discontinuous linear motion. It means there should be little benefit brought by extrapolation outside of the MAP_{un} region within which the motion is relative linear. This MAP_{un} region is determined by:

$$MAP_{un}^{E} = \{(m,n)|\alpha_E^{bk}(m,n) \leq E^*(\alpha_E^{bk}) \cup map_{b_k}^{out_E}\} \quad (6.2)$$

$$MAP_{un}^{I} = \{(m,n)|\alpha_I^{bk}(m,n) \leq E^*(\alpha_I^{bk}) \cup map_{b_k}^{out_I}\} \quad (6.3)$$

$$MAP_{un} = MAP_{un}^{I} \cup MAP_{un}^{E} \quad (6.4)$$

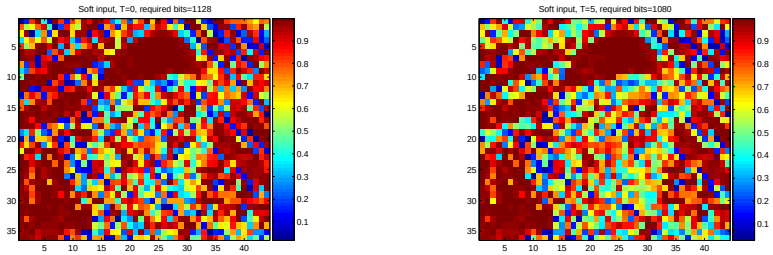
where (m,n) are the block coordinates. $\alpha_E^{bk}(m,n)$ and $\alpha_I^{bk}(m,n)$ are the estimated Laplacian distribution parameters within DCT band b_k for extrapolation and interpolation, respectively. $map_{b_k}^{\bullet}$ represents a classification map used in noise model (introduced in Sections 5.3 and 5.4) to classify Laplacian parameters into inlier region $map_{b_k}^{in\bullet}$ and outlier region $map_{b_k}^{out\bullet}$ (see Eqs. 5.11 or 5.16). It is assumed that the Laplacian parameters in outlier region are unreliable compared to the one in inlier region. $E^*(\alpha^{bk})$ represents the mean value of the Laplacian parameter over the blocks within the inlier region $map_{b_k}^{in}$ of DCT band b_k.

In order to take advantage of the benefits brought by the extrapolation soft input P_E regarding a single interpolation side information solution, a set of candidate soft inputs is generated by combining the extrapolation soft input P_E with the interpolation soft input P_I within the unreliability region MAP_{un}, while only the interpolation soft input

P_I is adopted in the reliable region (there is no expected benefit in also using P_E):

$$P_T(m,n) = \begin{cases} w_T \cdot P_I(m,n) & + \ (1-w_T) \cdot P_E(m,n), \\ & \text{if } (m,n) \in MAP_{un} \\ P_I(m,n), & \text{otherwise} \end{cases} \tag{6.5}$$

where $w_T = \{1-(T/10)|T=0,1,2,3,4,5\}$. All these candidate soft inputs are fed into the LDPCA decoder; the one which first converges will be chosen thus reducing the rate of syndrome bits for the same target quality. By using this set of combined soft inputs, the extrapolation side information track will influence the LDPCA decoding process, reducing the amount of misleading soft inputs provided by the interpolation side information track, following the intuition behind this chapter and reaching the stated objective of improving the overall RD performance based on more and better side information. However, the set of combined soft inputs will increase the complexity of LDPCA decoding up to 6 times.



(a) T=0, required bits=1128 (b) T=5, required bits=1080

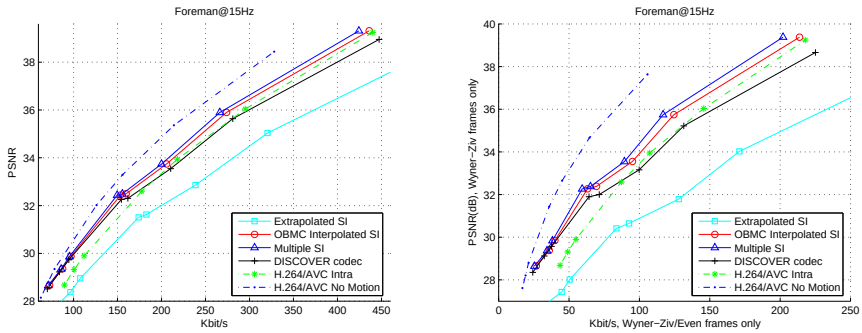
Figure 6.5: Estimated soft input and corresponding required number of syndrome bits of LDPCA [2] for one bitplane (b_k=3, level=5) of Foreman, frame No.4

6.4 Experimental Results

In order to make fair comparisons, the test conditions adopted are the DISCOVER project test conditions [11], which are described in Section 3.4 in detail. The test sequences are *Foreman*, *Soccer*, *Coastguard*

and *Hall Monitor*, at QCIF, 15 frames per second (fps); the GOP size is 2. The key frames are encoded using H.264/AVC Intra and the QPs are chosen so that the average PSNR of the WZ frames is similar to the average PSNR of the key frames. The RD performance is evaluated for the luminance component of both the key frames and Wyner-Ziv frames. The benchmark codecs used are the DISCOVER Wyner-Ziv video codec [11], H.264/AVC Intra codec and H.264/AVC no motion codec. The RD performance of the transform domain Wyner-Ziv video codec with multiple side information is evaluated and compared with the one with single interpolation [5] or extrapolation [7] side information.

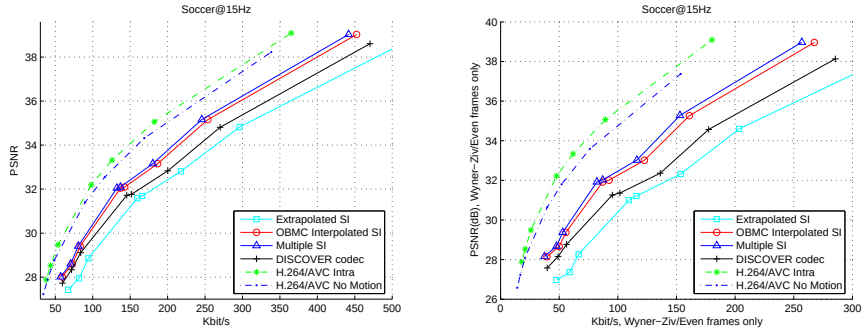
The test results described in Figs. 6.6-6.9 are mainly based on the improved noise model [10] (introduced in Section 5.4). In order to show that the multiple side information structure can work robustly with different noise model, the multiple side information based Wyner-Ziv video decoder with coefficient level noise model [9] are also tested. The results are reported in Figs. C.14-C.17 in Appendix C.



(a) (b)

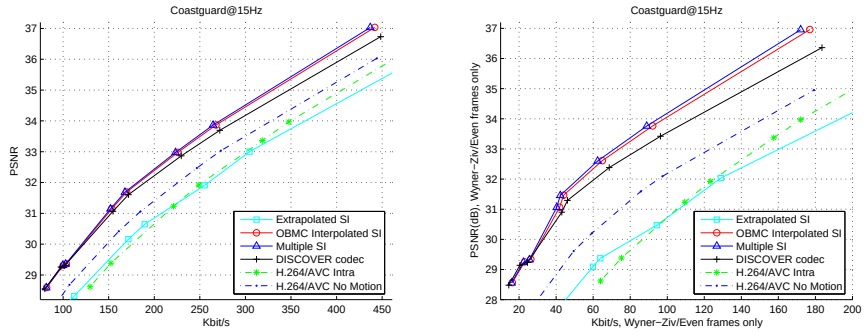
Figure 6.6: GOP2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Foreman@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

According to RD performance results shown in Figs. 6.6-6.9, the RD performance of Wyner-Ziv video coding with multiple side information outperforms the one with single interpolation side information up to 0.4 dB at high bitrates for Wyner-Ziv frames. It indicates that extrapolation side information can contribute to Wyner-Ziv video coding with multiple



(a) (b)

Figure 6.7: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Soccer@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

Figure 6.8: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise mode, sequence Coastguard@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

side information, although the quality of extrapolation side information is worse than interpolation side information. However, since the interpolation side information is quite efficient for low/regular motion sequences, the extrapolation side information brings less RD performance improvements in the context of Wyner-Ziv coding with multiple side information for this type of video content. This means that compared

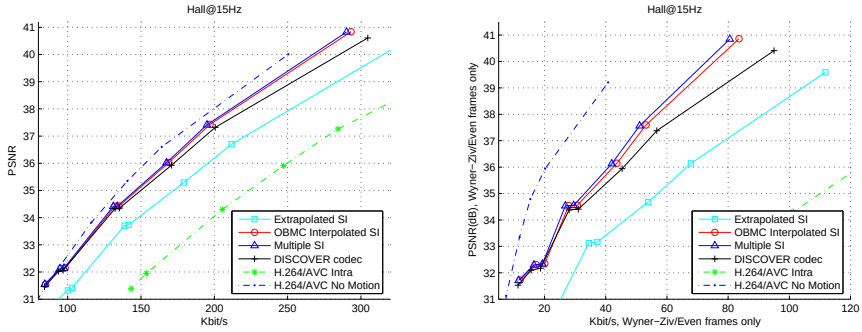


Figure 6.9: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Hall Monitor@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

with low/regular motion sequences like *Hall Monitor* and *Coastguard*, Wyner-Ziv decoding with multiple side information provides larger RD gains for high/irregular motion sequences like *Foreman* and *Soccer*.

Wyner-Ziv video coding with multiple side information gives better RD performance than H.264/AVC Intra coding for *Foreman*, *Coastguard* and *Hall Monitor*; For sequences with more irregular motion like *Soccer*, where the decoder frame estimation process is more difficult, the performance gap between H.264/AVC Intra coding and Wyner-Ziv video coding has been reduced but not yet closed. Compared with H.264/AVC no motion, Wyner-Ziv video coding provides the better coding performance for *Coastguard* only. However, the gaps between H.264/AVC no motion and Wyner-Ziv video coding in *Foreman*, *Soccer* and *Hall Monitor* have been once again reduced. Similarly, the RD performances of multiple side information based Wyner-Ziv codec with larger GOP size are reported in Section C.7 and C.8 for improved noise model and coefficient level noise model, respectively. Compared with the best available single side information based Wyner-Ziv video codec, the RD gains brought by multiple side information are larger than GOP size 2. Visual comparison of the Wyner-Ziv frames coded with multiple side information based codec is illustrated in Section C.9.

6.5 Summary

In this chapter, a novel transform domain Wyner-Ziv video decoder with multiple (interpolation and extrapolation) side information is proposed with the objective to improve the overall RD performance. Although the extrapolated side information frames are significantly worse than the interpolated side information frames, improvement is robustly achieved by generating and combining a set of candidate soft inputs to be fed to the LDPCA decoder, trying to reduce the number of syndrome bits requested by the decoder for a target quality. This process implies adaptively to combine the interpolation and extrapolation derived soft inputs with the aim of using the most reliable side information derived soft input depending on the video content. Compared with state-of-the-art single side information Wyner-Ziv video coding solutions, the proposed transform domain Wyner-Ziv video codec with multiple side information can improve the overall RD performance for the set of test sequences. The RD gains for GOP size 2 can go up to 0.4 dB for the Wyner-Ziv frames with precisely the same H.264/AVC Intra coded key frames. Increasing the GOP size, the RD gains brought by multiple side information could be even larger.

References to Chapter 6

[1] A. Aaron, S. Rane, E. Setton, and B. Girod. "Transform domain wyner-ziv codec for video", *Proc. SPIE VCIP*, pp. 520–528, Jan 2004.

[2] D. Varodayan, A. Aaron, and B. Girod. "Rate-adaptive distributed source coding using low-density parity-check codes", *EURASIP Signal Process. Journal, Special Section on Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[3] D. Kubasov, J. Nayak, and C. Guillemot. "Optimal reconstruction in wyner-ziv video coding with multiple side information", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 183–186, Oct 2007.

[4] J. Ascenso, C. Brites, and F. Pereira. "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding", *5th EURASIP Conf. on Speech and Image Process., Multimedia Commun. and Services*, July 2005.

[5] X. Huang and S. Forchhammer. "Improved side information generation for distributed video coding", *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 223–228, Oct. 2008.

[6] S. Borchet, K. Westerlaken, R. Gunnewiek, and R. Lagendijk. "On extrapolating side information in distrbuted video coding", *Picture Coding Symposium*, Nov. 2007.

[7] L. Natario, C. Brites, J. Ascenso, and F. Pereira. "Extrapolating side information for low-delay pixel-domain distributed video coding", *Int'l Workshop on Very Low Bitrate Video Coding*, Sept. 2005.

[8] X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer. "Distributed video coding with multiple side information", *Picture Coding Symposium (PCS)*, May 2009.

[9] C. Brites and F. Pereira. "Correlation noise modelling for efficient pixel and transform domain wyner-ziv video coding", *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 18, no. 9, Sept. 2008.

[10] X. Huang and S. Forchhammer. "Improved virtual channel noise model for transform domain wyner-ziv video coding", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2009.

[11] DISCOVER Project: www.discoverdvc.org, Dec 2007.

Chapter 7

Conclusion

In this work possible efficient video coding solutions for resource critical applications were investigated and introduced. Some novel techniques for state-of-the-art Distributed Video Coding solution are developed to improve the compression performance.

H.264/AVC Intra and H.264/AVC no motion codecs were presented as natural solutions with low complex encoders derived from the efficient conventional video codec H.264/AVC. The novel multi-frame based post-processing algorithm was presented to further explore the redundancy in temporal domain. It can always improve the quality of different coded sequence of increasing complexity, i.e. from H.264/AVC Intra to H.264/AVC no motion and H.264/AVC Inter coded sequences.

As a new video coding paradigm with low complex encoder, state-of-the-art transform domain Wyner-Ziv video codec was reviewed and implemented. This video coding solution achieves low complexity encoding by removing the motion estimation algorithm from the encoder side but exploiting the redundancy at the decoder. In terms of RD performance, the Wyner-Ziv video codec is getting close to H.264/AVC Intra. For relative low motion sequences with GOP size 2, it may win against H.264/AVC Intra. As important factors to influence the coding performance of Wyner-Ziv video codec, impacts of the quality of side information frame and the accuracy of noise model were evaluated. The adaptive weighted OBMC based frame interpolation algorithm was developed to improve the quality of side information frame. Through adaptive assigning weights to different neighboring blocks, more spatial cor-

relation is taken into account to temporally interpolated frames. Being applied on Wyner-Ziv video coding, the coding efficiency of Wyner-Ziv frame is improved up to 1 dB at higher bitrate of GOP size 2 sequences. Meanwhile, an improved virtual channel noise model was presented to reduce the existing performance gap between Wyner-Ziv video codec and conventional video codec H.264/AVC. It utilizes cross-band correlation to classify the coefficients into different categories. Two different estimators are applied into each category to estimate noise model parameters more accurately. With proposed noise model, coding efficiency of the Wyner-Ziv frames is improved up to 1 dB again at higher bitrate. Finally, a novel multiple side information based Wyner-Ziv video decoder was presented. It adaptively combines multiple side information (interpolation and extrapolation) and generates multiple soft input candidates for the LDPCA decoder. Compared with the best available single side information based Wyner-Ziv video coding solution, multiple side information based Wyner-Ziv video coding further improves the RD performance. The RD gains can be increased up to 0.4 dB for Wyner-Ziv frames of GOP size 2. For coded sequence with larger GOP size, improvements brought by multiple side information solution could be even larger. Experimental results have proved that the proposed algorithms in this work were efficient on improving coding performance of transform domain Wyner-Ziv video codec. These could be valuable contributions for designing future DVC codecs.

So far, for the most common used GOP size 2 setting, Wyner-Ziv video codec already wins against the H.264/AVC Intra for most of the test sequences. For some low/regular motion sequences, Wyner-Ziv video codec can even win against the H.264/AVC no motion codec. However, since the distance between key frames becomes far way for larger GOP sizes, winning against H.264/AVC Intra and no motion codec is getting more difficult. In terms of encoding complexity, Wyner-Ziv video codec is approximately 1/4 of H.264/AVC Intra and 1/8 of H.264/AVC no motion Inter (1/4 of the optimized H.264/AVC no motion Inter). Therefore, it can be concluded that Wyner-Ziv video codec is a promising video coding paradigm for critical encoding resource scenarios.

Besides further improving the quality of side information frame, increasing the accuracy of noise model and enhancing the reliability of soft input information, this work could be extended in one of following

directions:

Removing feedback channel: The existing feedback channel is not applicable for some real time application. In order to avoid using feedback channel, it is a challenge to perform rate control at the encoder side while keeping a low encoding complexity.

Mode decision: Due to low temporal correlation, the current Wyner-Ziv video codec is less efficient on coding the sequences with high motion or larger GOP size. With a mode decision, each block/frame can be adaptively coded with either Intra mode or Wyner-Ziv mode depending on the content of video sequences. Therefore, Wyner-Ziv encoder can avoid coding the frame/block with weak temporal correlation but employ Intra mode instead to improve the coding efficiency.

Slepian-Wolf codec: In Wyner-Ziv video coding structure, the Slepian-Wolf codec plays an important role on correcting the errors in the estimated side information frame. The experimental results indicate that the LDPCA codec introduces some compression loss as the Slepian-Wolf codec. It is a challenge to design an adequate channel code for DVC scenarios.

Appendices

Appendix A

Conference Contributions

A MULTI-FRAME POST-PROCESSING APPROACH TO IMPROVED DECODING OF H.264/AVC VIDEO

Xin Huang, Huiying Li, and Søren Forchhammer

COM Department, Technical University of Denmark, Building 343, Lyngby, DK-2800
Email:{xin,hli,sf}@com.dtu.dk

ABSTRACT

Video compression techniques may yield visually annoying artifacts for limited bitrate coding. In order to improve video quality, a multi-frame based motion compensated filtering algorithm is reported based on combining multiple pictures to form a single super-resolution picture and decimation to the desired format. The algorithm is applied to H.264/AVC decoded sequences and the processing involves a quality estimation based on picture type and local quantization value. Compared with directly decoding, the peak signal to noise ratio (PSNR) of the sequence obtained by the proposed algorithm is improved, and annoying ringing artifacts are effectively suppressed.

Index Terms— Artifacts reduction, motion compensated filtering, H.264/AVC

1. INTRODUCTION

H.264/AVC is the latest video compression standard. Due to its highly efficient performance, it will be used in future video storage and distribution applications. An in-loop de-blocking filter has already been addressed in H.264/AVC, therefore the most annoying artifact is ringing. Many postprocessing methods [1] have been developed based on the MPEG2 and H.263 standards. These methods can remove artifacts but also have a risk of over-smoothing details and sharpness, especially for sequences at medium coding bitrate. H.264/AVC has higher compression efficiency but it also loses many details. In order to remove ringing artifacts, enhance picture resolution, avoid over-smoothing details and preserve the sharpness after decoding, we modify and improve our previous work on MPEG2 [2] for application to H.264/AVC [3] decoded sequences.

The basic idea of the scheme is to apply an adaptive filter along motion trajectories utilizing an estimated quality of the pixel on each trajectory. The process can be divided into quality evaluation, motion compensated upsampling and de-ringing integrated decimation. First, the assumed quality of each pixel in the decoded sequence is estimated based on picture type and quantization step. In the second step, a super-resolution version (quadruple resolution default) of each directly decoded picture is constructed through temporal and spatial upsampling. Finally, a quality based decimation filter is designed to improve video quality and remove ringing artifacts. The motivation for a separate upsampling is an attempt to reduce single frame aliasing and trying to improve sharpness. The aim of this work mainly focuses on artifacts removal and video quality improvement, but by decreasing the decimation degree, higher resolution pictures can be also obtained.

The rest of the paper is organized as follows: In Section 2, a quality metric is designed to estimate each pixel's relative quality in the decoded sequence. A motion compensated upsampling algorithm to construct super-resolution pictures is described in Section 3. The de-ringing integrated decimation filter is described in Section 4. Test results are presented in Section 5.

2. QUALITY METRIC

The coded video sequence is mainly degraded by coarse quantization and inaccurate motion compensation. Macroblocks with different quantization parameter (QP) and prediction types (I, P or B) may have different distortion. Based on different picture types, we define a quality parameter q to reflect the mean squared error (MSE) for pixels in I, P and B pictures. With QP values and picture types available at the decoder, the quality parameter is calculated by $q = \sqrt{12 \times MSE}$, where MSE is determined by picture type and Q_{step} based on curves as shown in Fig. 1. The curves are obtained by measuring the MSE of the luminance components of H.264/AVC decoded sequences. QP determines the quantizer step size, Q_{step}. The results indicate that intra coded pictures (I) provide the best quality, and unidirectional prediction pictures (P) have better quality than bidirectional prediction pictures (B). We only use these training data to describe relative comparisons between different coding modes, it is not an absolute measure. All the settings and testing in later experiments are based on these curves. With this quality parameter, it is feasible to combine pixels with better assumed quality from neighboring pictures into the current picture, and prevent poor quality pixels from degrading better quality pixels.

The MSE caused by the quantization depends on the distribution of transform coefficients. This distribution is hard to estimate accurately due to varying image content. Some studies [4] have proposed to model transform coefficients with the Laplacian distribution, as opposed to the model in [2]. The distortion in pixel domain can be modeled as shown in Fig. 1 in comparison with the measured values.

3. MOTION COMPENSATED UPSAMPLING

Motion Compensated (MC) upsampling tries to form a superresolution picture (default has (V=4) times the resolution vertically and (H=4) times the resolution horizontally) by using the information from current picture and the N_f previous and subsequent pictures. Compared with a directly decoded picture, a MC upsampled higher resolution picture contains more information, which is helpful to remove artifacts and avoid over-smoothing details. MC upsampling starts with sub-pixel accuracy Motion Estimation (ME) to align pixels in the current picture with pixels in reference pictures. Pixels from reference pictures with fractional motion vector are assigned to the corresponding locations in the higher resolution pictures. Pixels from reference pictures with integer motion vectors are combined with decoded pixels in the current picture using a linear filter.

[0]X. Huang, H. Li, and S. Forchhammer "A Multi-frame Based Post-processing Approach to Improve Decoding of H.264/AVC", *Proceedings of the IEEE International Conference on Image Processing 2007*, San Antonio, USA, pp. 381-384, Sept. 2007

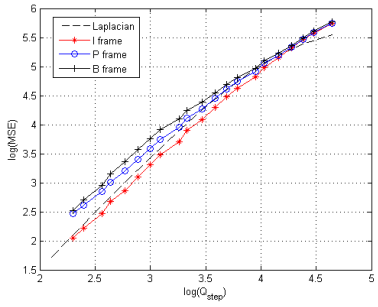


Fig. 1. MSE vs. Q_{step} measured on mobcal(CIF). Rate control is disabled, different QP values are chosen for the different points.

3.1. Motion Compensated Upsampling

In order to obtain reliable and homogeneous motion pixels x_r from reference pictures, a hierarchical block-based ME is utilized. The initial searching block size is set to be 16×16, followed by 4 sub-blocks (8×8). This final block size is our compromise between larger blocks for robustness and smaller blocks for accuracy. The motion vectors are obtained by searching in reference pictures for the best matching 8×8 block. They are denoted by $(m+\Delta m, n+\Delta n)$, where (m, n) is the integer part and $(\Delta m, \Delta n)$ is the fractional part of each motion vector. The fractional part is calculated by refining best matching block to sub-pixel accuracy using interpolation. The interpolated sub-pixels are generated by a six tap filter and then a linear filter as in H.264/AVC [3].

However, block-based motion estimation is not sufficient to guarantee that the best match pixels are in accordance with the true motion. It might introduce errors e.g., at occlusions in the motion compensation process. In order to reduce the risk of errors, we use a rejection criteria to evaluate for each pixel x_r whether it should be placed in the super-resolution picture. As in [2], the evaluation is based on intra-prediction [5]:

$$\hat{x}_{intra} = \begin{cases} min(a,b) & \text{if } c \geq max(a,b) \\ max(a,b) & \text{if } c \leq min(a,b) \\ a+b-c & \text{otherwise} \end{cases} \quad (1)$$

where a, b and c denote the pixel at the left, top and top-left of pixel x_c respectively. We compare the intra-predicted pixels and best match pixels based on sum of absolute difference (SAD). The pixels x_r with larger SAD over an 8×8 block will be rejected.

Let (m_r, n_r) denote the absolute coordinates of the best matching pixel, x_r, with integer motion vectors in a reference picture. Let $(\Delta m, \Delta n)$ denote the relative displacement of interpolated pixels having minimum SAD within an 8×8 block. Its corresponding best match x_r with integer motion vectors is now perceived as an upsampled pixel at position $\big((m_r - m - \Delta m)V, (n_r - n - \Delta n)H\big)$. If more than one reference pixel map to the same position of the current super-resolution picture, the pixel is assigned to be the reference pixel with the best estimated quality (Fig. 1). If these reference pixels have equal quality parameter, the super-resolution pixel is assigned to be their weighted average.

Reference pixels with integer motion vectors ($\Delta m = 0, \Delta n = 0$) may also achieve the minimum SAD. These reference pixels are combined with the directly decoded pixels in the current picture on the same trajectory by using a linear filter. The linear filter is only applied on the condition that the reference pixels have better estimated quality parameters. Let x_c be a pixel in the current decoded picture and x_r a trajectory pixel from a reference picture with integer motion vector. We combine their values to obtain an estimated pixel by:

$$\hat{x} = h_r x_r + h_c x_c \quad (2)$$

To minimize the expected MSE, the coefficients h_r and h_c could be estimated in a training session using original data by solving the Wiener-Hopf equations:

$$\begin{pmatrix} E\{X_r X_r\} & E\{X_r X_c\} \\ E\{X_c X_r\} & E\{X_c X_c\} \end{pmatrix} \begin{pmatrix} h_r \\ h_c \end{pmatrix} = \begin{pmatrix} E\{X X_r\} \\ E\{X X_c\} \end{pmatrix} \quad (3)$$

where X_r and X_c represent stochastic variables of pixel values in reference picture and current picture respectively. X represents a stochastic variable of original pixel values at the same position in original resolution picture. In order to preserve the mean value, coefficients of this filter should be computed under the constraint $h_r + h_c = 1$. Given enough training data, the second-order mean values in (3) could be conditioned on quality of x_r and x_c. To reduce the training h_r and h_c are modeled as in [2]:

$$h_r = 1 - (1 - \alpha)^{(q_c/q_r)^\beta} \quad (4)$$

$$h_c = 1 - h_r \quad (5)$$

This filter is fitted to optimal values of h_r (Fig. 2). The parameter α specifies the *a priori* weight that x_r should carry. The parameter β specifies how much the difference in qualities of x_r and x_c should influence the estimated pixel value. Equation (4) is monotonically increasing in the ratio q_c/q_r from 0 to 1 and it has the property that for $0 \leq \alpha \leq 1$, $\beta \geq 0$, $q_r, q_c \geq 0$ and $0 \leq h_r \leq 1$. Once this filter is applied to the pixels of the current and the reference picture, the estimated pixels in the super-resolution picture are assigned a new quality parameter value:

$$\hat{q} = h_r q_r + h_c q_c \quad (6)$$

3.2. Interpolated Upsampling

After MC upsampling, an unfinished superresolution picture is formed In order to complete the current super-resolution picture with irregular samples, we modify the cubic interpolation process with an irregular sample detection. Cubic spatial interpolation is based on rectangular lattice samples, which can supply true continuity among each segment and produce less jaggy edges. If there are no irregular samples in the nearest 4×4 pixel region, a normal cubic interpolation is implemented. Otherwise, a modified version is used:

$$x_{intp}(m', n') = \textstyle\sum_i \sum_j x_{re}(i,j) K_1 \beta^3(|m' - i|) \beta^3(|n' - j|) + \sum_a \sum_b x_{ir}(a,b) K_2 \beta^3(|m' - a|) \beta^3(|n' - b|) \quad (7)$$

where K_1 and K_2 are normalizing coefficients, $x_{re}(i,j)$ and $x_{ir}(a,b$ represent samples at regular and irregular positions, respectively. $\beta^3(z)$ is a typical cubic convolution kernel [6]:

$$\beta^3(z) = \begin{cases} \frac{3}{2}|z|^3 - \frac{5}{2}|z|^2 + 1 & \text{if } 0 \leq |z| \leq 1 \\ -\frac{1}{2}|z|^3 + \frac{5}{2}|z|^2 - 4|z| + 2 & \text{if } 1 \leq |z| \leq 2 \\ 0 & \text{if } 2 \leq |z| \end{cases} \quad (8)$$

4. DECIMATION

A super-resolution picture for each directly decoded picture is formed after upsampling. In order to reduce ringing artifacts and get the desired picture resolution, we propose a de-ringing integrated downsampling scheme applying a quality based spatial filter. Since ringing artifacts mainly appear in the vicinity of sharp edges, different types of decimation filters are operated in no-edge areas and edge areas, respectively. Canny's method is used for edge detection. In order to reduce the risk of blurring edges in the decimation process, both of the decimation filters are operated in a small 9×9 window.

4.1. No-edge Area Decimation

For the no-edge area, a two-dimensional spatial linear filter combined with adaptive quality weights is applied in the vicinity of each sample position (m_0, n_0) to obtain a lower resolution picture.

$$p_l(m_0', n_0') = \sum_{m,n} g(m,n,m_0,n_0) p_h(m,n) = \sum_{m,n} K g_v(|m-m_0|) g_h(|n-n_0|) w(m,n) p_h(m,n) \quad (9)$$

where $p_l(m_0', n_0')$ represents a downsampled pixel in the lower resolution picture, $p_h(m,n)$ represent the pixels which are adjacent to sample pixel $p_h(m_0, n_0)$ in the super-resolution picture. K is a normalizing factor ($\sum_g = 1$). g_v and g_h are 1-D symmetric filters in the vertical and horizontal direction, respectively. $w(m,n)$ is a weight function for each pixel based on its corresponding quality parameter described below. The 1-D symmetric filters g_v and g_h reflecting the spatial distance are defined by [2]:

$$g_2 = (\ldots, 0, a, 1, a, 0, \ldots) \quad (10)$$

$$g_4 = g_2 * g_2 = (\ldots, a^2, 2a, 1+2a^2, 2a, a^2, \ldots) \quad (11)$$

$$g_v = g_h = g_4 * g_4 \quad (12)$$

Furthermore, the value of a should be adaptive depending on local characteristics (smooth or texture). Therefore, we calculate a standard deviation σ of each downsampling sample $p_h(m_0, n_0)$ within a 9×9 window to obtain an adaptive control value:

$$a = \begin{cases} 1, & \text{if } \sigma \le 10 \\ 0.5, & \text{otherwise} \end{cases} \quad (13)$$

$w(m,n)$ is a weight function reflecting the qualities of different kinds of pixels. It depends on whether $p_h(m,n)$ and $p_h(m_0,n_0)$ are compensated upsampling pixels (p_{cu}) or interpolated upsampling pixels (p_{iu}). If both of them are compensated upsampling pixels, their quality parameters are used to determine the weight of $p_h(m,n)$. If one of them is obtained by interpolation, a constant weight value is assigned [2]:

$$w(m,n) = \begin{cases} \frac{w_0}{\gamma} \gamma^{q(m,n)/q(m_0,n_0)}, & \\ \quad p_h(m,n), p_h(m_0,n_0) \in p_{cu} & \\ 1, \quad p_h(m,n) \in p_{iu}, p_h(m_0,n_0) \in p_{cu} & \\ w_0, \quad p_h(m,n) \in p_{cu}, p_h(m_0,n_0) \in p_{iu} & \end{cases} \quad (14)$$

where the parameter w_0 (set to 6) specifies the *a priori* worth of a compensated upsampling (p_{cu}) pixel compared to an interpolated pixel (p_{iu}). The parameter γ (set to 0.3) is a global parameter reflecting the influence introduced by quality ratio.

4.2. Edge Area Decimation

For the edge areas, de-ringing integrated decimation filters are separately applied on each side of the edge boundary. Only those pixels, which are inside the decimation window and on the same side of the sample pixel $p_h(m_0, n_0)$, are used for this de-ringing filter [7]. Therefore, we define pixel sets $F^{(m_0,n_0)}$ as all the pixels used for the weighted de-ringing filter. The downsampled pixel value $p_l(m_0', n_0')$ is obtained by:

$$p_l(m_0', n_0') = \frac{\sum_{p_h(m,n) \in F^{(m_0,n_0)}} W_{(m,n)}^{m_0,n_0} p_h(m,n)}{\sum_{p_h(m,n) \in F^{(m_0,n_0)}} W_{(m,n)}^{m_0,n_0}} \quad (15)$$

where the weight factor $W_{(m,n)}^{m_0,n_0}$ is the product of local position distance factor $w_d(m,n)$, pixel difference factor $w_l(m,n)$ and quality factor $w(m,n)$. $w_d(m,n)$ and $w_l(m,n)$ are defined as:

$$w_d(m,n) = \begin{cases} \frac{1}{2 \times dis((m,n),(m_0,n_0))}, & \text{if } (m,n) \neq (m_0,n_0) \\ 1, & \text{otherwise} \end{cases} \quad (16)$$

$$w_l(m,n) = e^{-\frac{p_h(m,n) - p_h(m_0,n_0)}{Th}} \quad (17)$$

5. EXPERIMENTAL RESULTS

We used the H.264/AVC reference software JM9.3 [3] for experiments. Several *CIF* sequences (4:2:0) are chosen. They were encoded with different bitrates by enabling rate control. The GOP structure is defined as $(IBBP)_{12}$. In-loop de-blocking filter is on and single encoding reference frame is used. The parameter N_f is set to 5, α and β of the filter (4) are estimated using many frames of different sequences based on Equation (3), (See Fig. 2), the curves yield $\alpha = 0.15$ and $\beta = 0.7$.

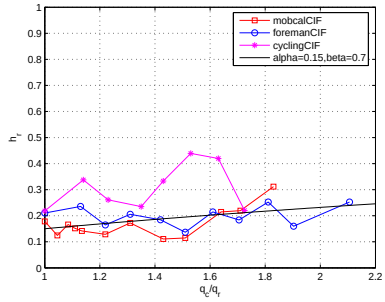
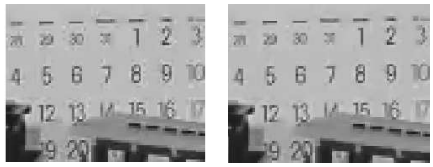


Fig. 2. Filter coefficient h_r as a function of q_c/q_r

Based on these settings, we implemented our algorithm on different directly decoded sequences. Fig. 3 is an example frame with our motion compensated filtering for *mobcal*. The average PSNR performances for the sequences *mobcal* and *foreman* are depicted in Figs. 4 and 5, respectively. From these figures we can clearly see that our algorithm is able to improve the average PSNR performance up to 0.3dB. The more interesting thing is that our algorithm can give improvement for the sequences at medium or relative high bitrate. It can be explained as: the magnitude of the improvements

mainly depends on the relative quality of decoded picture compared to its surrounding pictures. Fig. 6 illustrates the PSNR improvement for each individual picture, it is noted that the algorithm improves all the pictures regardless of their directly decoded quality.



(a) Directly decoded frame (b) With MC filter

(c) Sharpening decoded frame (d) Sharpening MC frame

Fig. 3. Visual comparison, mobcal(CIF) at 498kbit/s, frame 25

6. CONCLUSION

This paper presents a multi-frame approach to improve decoding quality of H.264/AVC sequences. From the experimental results, the average PSNR of the whole sequence is robustly improved especially for sequences at medium or relatively high bitrate. For individual pictures, all the pictures' quality is improved regardless of their directly decoded quality. Visually, ringing artifacts are reduced, sharp details and edge are well preserved.

7. REFERENCES

[1] Y.Nie, H.S.Kong, A.Vetro, and K. Barner, "Fast adaptive fuzzy post-filtering for coding artifacts removal in interlaced video," *Proc. ICASSP*, vol. 2, pp. 993–996, Mar. 2005.

[2] B.Martins and S.Forchhammer, "A unified approach to restoration, deinterlacing and resolution enhancement in decoding mpeg-2 video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 803–811, Sept. 2002.

[3] MPEG AVC/H.264 video reference software JM9.3, *Available: http://iphome.hhi.de/suehring/tml/download/*.

[4] S.Smoot and L.Rowe, "Study of dct coefficient distributions," *Proc. SPIE*, pp. 303–311, Jan. 1996.

[5] JPEG-LS IS 14495-1, "Lossless and near-lossless coding of continuous tone still images," *ISO/IEC International Standard*, 1998.

[6] E.Meijering and M.Unser, "A note on cubic convolution interpolation," *IEEE Trans. Image Process.*, vol. 12, pp. 477–479, Apr. 2003.

[7] H.Li and S.Forchhammer, "Spatial postprocessing filtering for MPEG compressed video," in preparation.

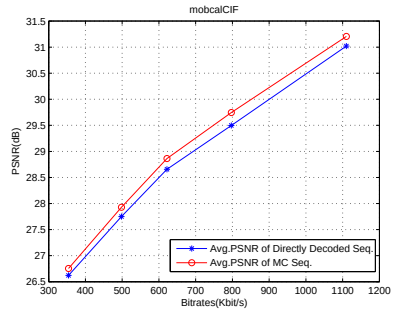


Fig. 4. PSNR performance of MC filter for mobcal(CIF)

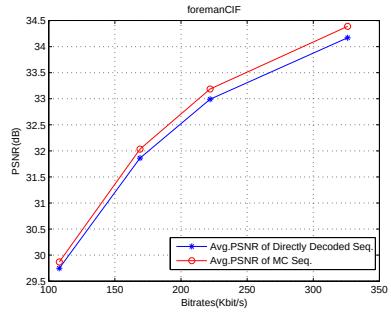


Fig. 5. PSNR performance of MC filter for foreman(CIF)

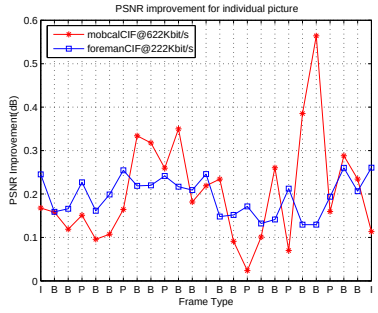


Fig. 6. PSNR improvement for individual pictures

Improved Side Information Generation For Distributed Video Coding

Xin Huang [1] and Søren Forchhammer [2]

DTU Fotonik, Technical University of Denmark
Building 343, Lyngby 2800, Denmark
[1] xin@com.dtu.dk
[2] sf@com.dtu.dk

***Abstract*—As a new coding paradigm, Distributed Video Coding (DVC) deals with lossy source coding using side information to exploit the statistics at the decoder to reduce computational demands at the encoder. The performance of DVC highly depends on the quality of side information. With a better side information generation method, fewer bits will be requested from the encoder and more reliable decoded frames will be obtained. In this paper, a side information generation method is introduced to further improve the Rate-Distortion (RD) performance of transform domain distributed video coding. This algorithm consists of a variable block size based Y, U and V component motion estimation and an adaptive weighted Overlapped Block Motion Compensation (OBMC). The proposal is tested and compared with the results of an executable DVC codec released by DISCOVER group (DIStributed COding for Video sERvices). RD improvements on the set of test sequences are observed.**

I. Introduction

Distributed Video Coding (DVC) has been proposed in] to avoid using complex motion estimation and motion ompensation at the encoder and only explore the video atistics at the decoder side. In many emerging applications, g. wireless video surveillance, wireless PC cameras and obile cameras, due to limited memory and computational ower at the encoder side, DVC might be more suitable an conventional video coding like ISO MPEG-x and ITU- H.26x which have one highly complex encoder and (one) many simpler decoders. DVC is based on two major formation theoretic results: the Slepian-Wolf [2] and Wyner- iv [3] theorems. According to the Slepian-Wolf theorem, it is ossible to achieve the same rate as a joint encoding system / independent encoding but joint decoding of two statistiully dependent signals. The Wyner-Ziv theorem extends the epian-Wolf theorem to a lossy case, which becomes the key eoretical basis of DVC.

A Low-Density Parity-Check (LDPC) based transform doain Wyner-Ziv codec released by DISCOVER [4][5] is one the best DVC codecs available. It improves on the work ised in [6] by introducing an advanced frame interpolation r side information generation [7][8], a finer correlation noise odeling [9][10], and an optimal reconstruction algorithm 1]. However, there are still significant RD performance gaps etween DVC and conventional video coding schemes as .264/AVC. Since the quality of side information frames is a ry natural element influencing the coding efficiency of DVC, there are several different side information generation schemes in the literature including interpolation [7][8] and extrapolation [12][13] based algorithms. In this paper, an interpolation based side information generation scheme is introduced and applied to a transform domain DVC to improve the RD performance. This new scheme improves on the work in [7] and [8], by introducing Y, U and V based motion estimation with variable block size to take advantage of more information and obtain more accurate motion vectors first, combined with an adaptive weighted Overlapped Block Motion Compensation (OBMC) to generate better side information.

The rest of this paper is organized as follows: Section II briefly describes the architecture of the LDPC based transform domain Wyner-Ziv video coding. In Section III, the proposed side information generation scheme is introduced. Test conditions and results are presented in Section IV.

II. Architecture of DVC

The architecture of a state-of-art DVC codec [5][6] is depicted in Fig. 1. A fixed Group of Pictures (GOP=2) is adopted. The video sequence is first split into odd (key) frames and even (Wyner-Ziv) frames. The odd frames are intra coded by using a conventional video coding like H.264/AVC while the even frames are Wyner-Ziv coded.

In the Wyner-Ziv encoder, Wyner-Ziv frames are partitioned into non-overlapped 4x4 blocks and an integer discrete cosine transform (DCT) [14] is applied on each of these. The transform coefficients within a given band $b_k, k \in \{0...15\}$, are grouped together and then quantized with 2^{M_k} levels. DC coefficients and AC coefficients are uniformly scalar quantized and dead zone quantized, respectively. After quantization, the coefficients are binarized, each bitplane is transmitted to a rate-compatible LDPC accumulate encoder [15] starting from the most significant bitplane. For each encoded bitplane, the corresponding accumulated syndrome is stored in a buffer together with an 8-bit Cyclic Redundancy Check (CRC). The amount of bits to be transmitted depends on the requests from the decoder through a feedback channel.

In the Wyner-Ziv decoder, based on two intra coded frames X_{2i-1} and X_{2i+1}, a motion estimation and compensation based frame interpolation algorithm is adopted to create a side information frame Y_{2i} and a motion estimated residual frame R_{ME} (i.e. the difference between X_{2i-1} and X_{2i+1} along

[0]X. Huang and S. Forchhammer "Improved Side Information Generation for Distributed Video Coding", *Proceedings of IEEE International Workshop on Multimedia Signal Processing 2008*, Cairns, Australia, pp. 223-228, Oct. 2008

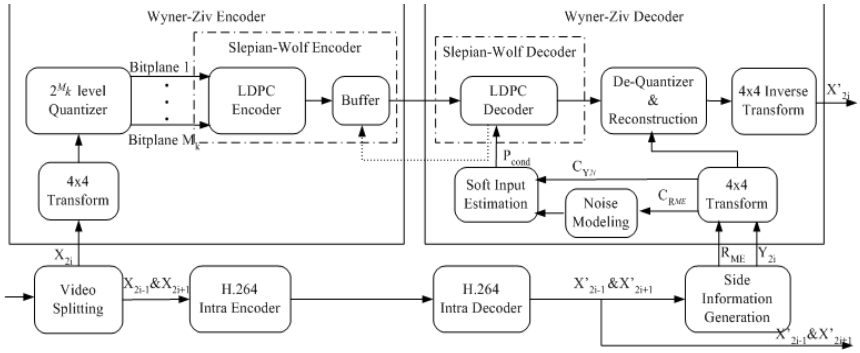


Fig. 1. Diagram of LDPC based transform domain Wyner-Ziv codec architecture

e motion vectors). Y_{2i} and R_{ME} undergo the same 4x4 teger DCT to obtain coefficients $C_{Y_{2i}}$ and $C_{R_{ME}}$. $C_{R_{ME}}$ utilized on-line to roughly model [9] the noise distribution etween corresponding DCT bands of side information frame id Wyner-Ziv frame (i.e. $C_{Y_{2i}}$ and $C_{X_{2i}}$). By using the btained noise distribution, coefficient values of the side formation frame $C_{Y_{2i}}$ and previous successfully decoded tplanes, soft information (conditional bit probabilities P_{cond}) r each bitplane is estimated. With a given soft-input in- rmation P_{cond}, the LDPC decoder starts to process the rresponding bitplanes to correct the bit errors. Convergence tested by computing the Hamming distance between the ceived syndrome and the one obtained by the decoded tplane. If the Hamming distance is different from zero after certain amount of iterations, the LDPC decoder requests ore accumulated syndrome bits from the encoder buffer via e feedback channel. If the Hamming distance is equal to ro, then the 8-bit CRC sum is requested from the buffer to erify successful decoding. A decoded bitplane with correct RC sum is sent to an optimal reconstruction module [11], a tplane with incorrect CRC sum requests more accumulated ndrome bits from encoder buffer to correct the existing bit rors until a low error probability is guaranteed. For more tails refer to [5][15].

III. Side Information Generation

Based on the architecture of the DVC, the output of side formation generation not only influences the soft input esti- ation module but also the reconstruction module. Therefore, e choice of the adopted side information generation scheme n significantly influence the RD performance. Generally, ore accurate side information frames means that fewer bits e requested from the encoder for the same decoding quality. n advanced motion compensated interpolation algorithm [7] reportedly adopted in the executable DVC codec [4][5] re- ased by DISCOVER. It includes forward motion estimation, bi-directional motion estimation, spatial smoothing of motion vectors and bi-directional motion compensation. The work has been improved to extend motion estimation and compensation to sub-pixel accuracy [8].

Although this scheme can generate good side information frames, there are some limitations: First of all, it does not uti- lize all the information available at the decoder side. Secondly, the block size used for motion estimation and compensation might not be an optimal choice. Thirdly, only a simple bi- directional motion compensation is employed. Overcoming these limitations will improve the side information generation and further improve RD performance of DVC. Therefore, an improved side information generation scheme is proposed as shown in Fig. 2. It is divided into two parts: Y, U and V based motion estimation with variable block size is applied to get accurate motion vectors at first. Then an adaptive weighted Overlapped Block Motion Compensation (OBMC) is employed to generate better side information frames.

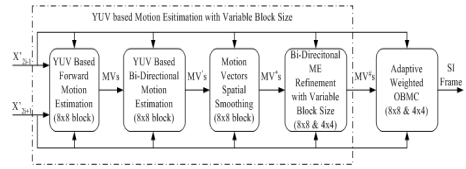


Fig. 2. Improved side information generation scheme

A. YUV based motion estimation with variable block size

In order to take advantage of more information available at the decoder, the chroma components (U and V) in intra decoded key frames are utilized to assist luma component (Y) based motion estimation. Thus the Mean Squared Error (MSE)

ısed motion estimation is determined by:

$$argmin\{\xi_{(m,n)\in block}\{(X^{Y}_{2i-1}(m,n) - X^{Y}_{2i+1}(m+\Delta m, n+\Delta n))^2\} + \lambda \cdot \xi_{(m',n')\in block}\{(X^{UV}_{2i-1}(m',n') - X^{UV}_{2i+1}(m'+\Delta m', n'+\Delta n'))^2\}\} \quad (1)$$

here $X^{Y}_{2i-1}(m,n)$ and $X^{UV}_{2i-1}(m',n')$ are the corresponding ma and chroma values at coordinates (m,n) and (m',n') in ey frame X_{2i-1}, respectively. $(\Delta m, \Delta n)$ and $(\Delta m', \Delta n')$ present the motion vectors. For 4:2:0 video sequences, $m = 2\Delta m', \Delta n = 2\Delta n', m = 2m'$ and $n = 2n'$. λ is parameter to balance the weight between luma and chroma ılues.

Besides YUV based motion estimation, the first three modes in Fig. 2 are similar to [7][8]. With the given two decoded ey frames X_{2i-1} and X_{2i+1}, an 8×8 block based motion ıtimation is applied with full-pixel accuracy first. Since the gid block based motion estimation results in overlapped and ıcovered areas after the frame interpolation, the obtained otion vectors are only seen as candidates. Motion vectors, IV, are selected from the candidates that intercepts the terpolated frame closest to the center of each 8×8 block. In der to obtain more accurate motion vectors $MV^{'}s$, a bidireconal motion estimation scheme [7] with sub-pixel accuracy is plied with a smaller search range. This bidirectional motion ıtimation selects a linear trajectory by using MVs as initial ılues, then the refined $MV^{'}s$ are obtained by a bidirectional mmetric motion search. Afterwards, $MV^{'}s$ are smoothed by ing a weighted vector median filter [7]. A six tap Wiener ter [14] is used to interpolate key frames and consequently otion estimate in sub-pixel accuracy.

Since an 8×8 block based motion estimation is applied in][7][8], it may not perfectly match the true motion especially ound object boundaries. Variable size block based motion ıtimation is more efficient in representing irregular motion. herefore, a bi-directional motion estimation with variable ock size (8×8 and 4×4) is adopted after the motion vector noothing module. Selecting two predefined thresholds τ_{mse} ıd τ_σ, each 8×8 block is evaluated to decide whether to vide it into 4×4 sub-blocks based on:

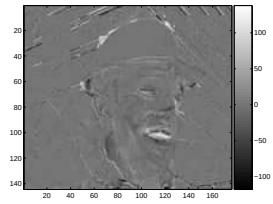
$$MAP_{4\times4} = \begin{cases} True & \text{if } MSE_{8\times8} \geq \tau_{mse} \\ & \text{and } Var(MV) \geq \tau_\sigma \\ False & \text{otherwise} \end{cases} \quad (2)$$

here $MSE_{8\times8}$ is the YUV based MSE value between X_{2i-1} ıd X_{2i+1} over the corresponding 8×8 block, $Var(MV)$ is function to calculate the variance of the relevant motion ctors for the current block in an 3×3 window.

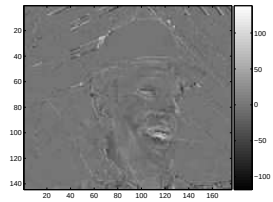
$$Var(MV(m,n)) = \frac{\Sigma^{1}_{i=-1}(MV(m+i,n+i) - \bar{MV})^2}{9} \quad (3)$$

here $\bar{MV}$ is the mean value of MVs. If an 8×8 block tisfies the above conditions, its MV is taken as initial IV of each 4×4 sub-blocks and the relevant $MSE_{4\times4}$ are calculated. A small refinement search range ρ is chosen to find the best matching 4×4 sub-block with minimum $MSE_{4\times4}$.

With variable block size, the smaller blocks are used to describe irregular motion around the edges of objects, the larger blocks are used for homogeneous motion. As shown in Fig. 3, the energy of the motion estimated residual R_{ME} with variable block size is smaller than the one with a fixed 8×8 block. Thus providing an advantage by introducing fewer inaccurate pixels into the side information frame.



(a) With 8×8



(b) With 8×8 and 4×4

Fig. 3. Comparison of motion estimated residuals R_{ME} with fixed block size (8×8) and adaptive block size (8×8 and 4×4)

B. Adaptive Weighted Overlapped Block Motion Compensation

Overlapped Block Motion Compensation (OBMC) is usually applied to reduce blocking artifacts and improve subjective quality in frame rate up-conversion. However, it also has a higher risk of over-blurring the interpolated side information frame compared with the simple bi-directional motion compensation used in [7]. Since the MSE value over each block of the YUV based motion estimation approximately reflects the reliability of its relevant motion vectors, an adaptive OBMC [16] weighted by MSE is employed to reduce the interpolated errors and control the blurring. Let $j \in [0,k]$ denote the index of the neighboring blocks. As shown in Fig. 4, the value of k is varying due to variable block size adopted.

$$Y_{2i}(m_0,n_0) = \frac{\Sigma^{k}_{j=0}\omega_j \hat{Y}_j}{\Sigma^{k}_{j=0}\omega_j} \quad (4)$$

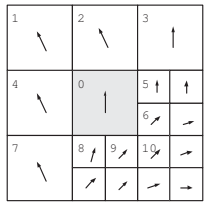
$$\hat{Y}_j = \frac{1}{2} \times (X_{2i-1}(m_0 + \Delta m_j^*, n_0 + \Delta n_j^*) + X_{2i+1}(m_0 + \Delta m_j^\#, n_0 + \Delta n_j^\#)) \quad (5)$$

$$R_{ME}(m_0, n_0) = \frac{\Sigma_{j=0}^{k} \omega_j \hat{R}_j}{\Sigma_{j=0}^{k} \omega_j} \quad (6)$$

$$\hat{R}_j = (X_{2i-1}(m_0 + \Delta m_j^*, n_0 + \Delta n_j^*) - X_{2i+1}(m_0 + \Delta m_j^\#, n_0 + \Delta n_j^\#)) \quad (7)$$

here (m_0, n_0) belongs to current block, $(\Delta m_j^*, \Delta n_j^*)$ and $\Delta m_j^\#, \Delta n_j^\#)$ are backward and forward motion vectors $Block_j$ in X_{2i-1} and X_{2i+1}, respectively. The relation, $\Delta m_j^*, \Delta n_j^*) = -(\Delta m_j^\#, \Delta n_j^\#)$, i.e. linear motion (with OP=2) is assumed. ω_j is the weight of $Block_j$ obtained by ılculating the inverse proportion of the YUV based MSE:

$$\omega_j = (\xi_{(m_j,n_j)\in Block_j}((X_{2i-1}^{YUV}(m_j + \Delta m_j^*, n_j + \Delta n_j^*) - X_{2i+1}^{YUV}(m_j + \Delta m_j^\#, n_j + \Delta n_j^\#))^2))^{-1} \quad (8)$$



g. 4. Utilized neighboring motion vectors and blocks for adaptive weighted 3MC

IV. Experimental Results

First of all, in order to evaluate the proposed Side Informa-ɔn Generation (SIG) scheme, different methods were imple-ented and these are compared in Table I by measuring the ʻerage Peak Signal-to-Noise Ratio (PSNR) of the interpolated ames over the whole sequence: i) bidirectional motion search ısed SIG employed in [13]; ii) advanced SIG employed in]; iii) YUV based SIG; iv) YUV based SIG with Variable lock Size; v) YUV based SIG with fixed block size (8×8) ıd adaptive weighted OBMC; vi) Y based SIG with VBS and laptive weighted OBMC; vii) YUV based SIG with VBS and laptive weighted OBMC.

Then the RD performances of DVC implementation with e proposed SIG method (vii) and the SIG method (ii) based ı [7][8] as in [5] are compared. For testing and comparison, DVC codec was implemented in MATLAB 7 combined ith C. The performance of the basic version was brought -line with the executable DVC codec[4] (see Figs. 6-9) for e comparison. The conditions for the tests are:

- All (149) frames of "Foreman", "Coastguard", "Soccer" and "Hallmonitor" sequences are used. The sequences are in QCIF@15Hz format, and they are available at [4].

TABLE I
THE AVERAGE PSNR RESULTS FOR DIFFERENT METHODS, KEY FRAMES ARE INTRA CODED WITH H.264/AVC WITH FIXED QUANTIZATION PARAMETER (QPS).

Sequence	Foreman	Coastguard	Soccer	Hallmonitor
Key frames	QP=25	QP=26	QP=25	QP=24
i)	27.8192	29.7681	20.6988	35.0267
ii)	28.9047	31.4664	20.8326	36.3338
iii)	28.9843	31.4681	20.8483	36.3339
iv)	28.9999	31.5371	20.8453	36.3735
v)	29.2358	31.7708	21.2874	36.3331
vi)	29.2296	31.8317	21.2961	36.4548
vii)	29.2537	31.8340	21.2967	36.4593

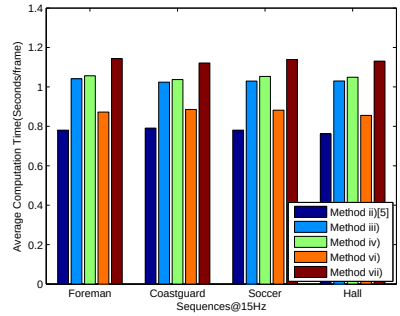


Fig. 5. Complexity comparison for different SIG methods

- As in [4][5], half-pixel accuracy motion estimation is used in the proposed side information generation for fair comparison.
- The most common GOP length (GOP=2) in [5][6] is used. The key frames are encoded by H.264/AVC intra and the QP are chosen so that the average PSNR of Wyner-Ziv frames is similar to the quality of key frames as in [5]. The chosen QPs in Table I are corresponding to the eighth RD point.
- All the RD performance results are evaluated by the average of luminance components (Y) of key frames and Wyner-Ziv frames.
- Parameter λ in YUV based ME is chosen to be 5. The thresholds $\tau_{mse} = \gamma \times Mean(MSE_{8\times8})$ and $\tau_\sigma = 0$ are chosen for variable block size partition, $\gamma = 6$ is chosen experimentally. The refinement search range ρ is defined in ± 1 pixels.

As shown in Table I, each module of the proposed side information scheme generally improves the PSNR step by step. The proposed method (vii) gives the best PSNR result by increasing the complexity of decoder.

The complexity for different SIG methods are evaluated by calculating the average time (on a 3GHz PC) for generating one interpolated frame. As shown in Fig. 5, if the complexity

the decoder should not be significantly increased, the proposed method (vi), which removes YUV based motion estimation from method (vii), gives a good balance between decoder complexity and PSNR performance. We choose method (vii) with the best PNSR results. According to RD results shown in Figs. 6-9, the performance of DVC implementation (with side information method (ii) as in [5][7][8]) is comparable with the results of the DISCOVER executable codec. Compared with the SIG used in [5], the RD performances of DVC for high bitrates are improved up to 0.5dB with the proposed SIG scheme (vii).

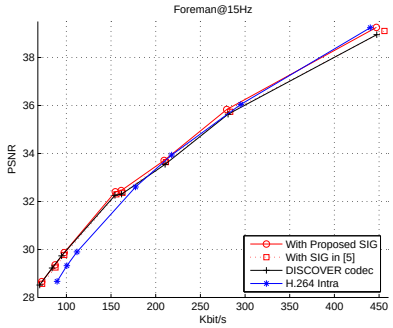


Fig. 6. RD comparison for sequence Foreman

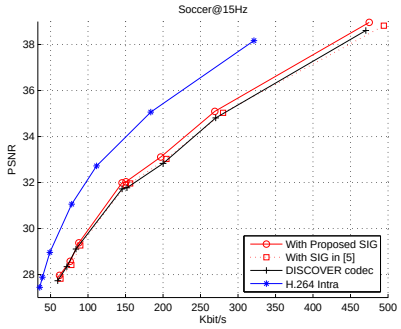


Fig. 7. RD comparison for sequence Soccer

V. Conclusion

In this paper, an improved side information generation scheme is introduced in DVC. It overcomes three limitations of the current scheme utilized in the DISCOVER DVC executable codec. This is obtained by using more information (chroma) at the decoder side, utilizing variable block size for motion estimation and compensation and employing adaptive weighted OBMC. Experimental results show that the proposed scheme can improve coding efficiency of DVC. Compared with the current scheme employed in the DISCOVER executable codec, the proposed scheme improves the RD performance up to 0.5dB at the higher bit-rates.

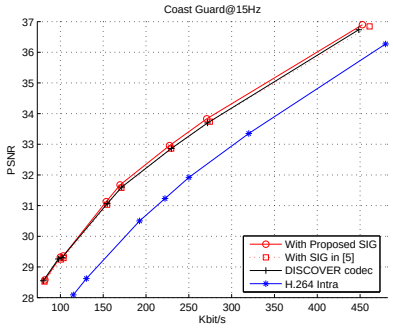


Fig. 8. RD comparison for sequence Coastguard

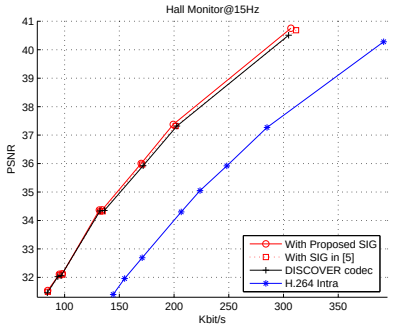


Fig. 9. RD comparison for sequence Hall Monitor

References

[1] A. Aaron, R. Zhang, and B. Girod, "Wyner-ziv coding of motion video," *Proc. Asilomar Conference on Signals and Systems*, Nov. 2002.
[2] J. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. on Inform. Theory*, vol. 19, pp. 471–480, July 1973.
[3] A.D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. on Inform. Theory*, vol. 22, pp. 1–10, Jan. 1976.
[4] Available on: www.discoverdvc.org.
[5] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Knbasov, and M. Ouaret, "The discover codec: architecture, techniques and evaluation," *Picture Coding Symposium (PCS)*, Nov. 2007.

[6] A. Aaron, S. Rane, E. Setton, and B. Girod, "Transform domain wyner-ziv codec for video," *Proc. SPIE VCIP*, Jan. 2004.

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

[8] S. Klomp, Y. Vatis, and J. Ostermann, "Side information interpolation with sub-pel motion compensation for wyner-ziv decoder," *International Conference on Signal Processing and Multimedia Applications*, Aug. 2006.

[9] C. Brites, J. Ascenso, and F. Pereira, "Studing temporal correlation noise modeling for pixel based wyner-ziv video coding," *IEEE Int'l Conf. on Image Proce.*, Oct. 2006.

[0] C. Brites, J. Ascenso, J. Pedro, and F. Pereira, "Evaluating a feedback channel based transform domain wyner-ziv video codec," *Signal Processing, Image Communication 23*, pp. 269–297, 2008.

[1] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal reconstruction in wyner-ziv video coding with multiple side information," *Internationl Workshop on Multimedia Signal Processing*, Oct. 2007.

[2] L. Lu, D. He, and A. Jagmohan, "Side information generation for distributed video coding," *IEEE Int'l Conf. on Image Proce.*, Oct. 2007.

[3] Z. Li, L. Liu, and E. J. Delp, "Rate distortion analysis of motion side estimation in wyner-ziv video coding," *IEEE Trans. on Image processing*, vol. 16, pp. 98–113, Jan. 2007.

[4] I. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley and Sons Ltd., West Sussex, England, 2003.

[5] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive distributed source coding using low-density parity-check codes," *EURASIP Signal Processing Journal, Special Section on Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[6] S. Lee, O. Kwon, and R. Park, "Weighted-adaptive motion-compensated frame rate-up conversion," *IEEE Trans. Consum. Electron.*,, vol. 49, pp. 485–492, Aug. 2003.

IMPROVED VIRTUAL CHANNEL NOISE MODEL FOR TRANSFORM DOMAIN WYNER-ZIV VIDEO CODING

Xin Huang and Søren Forchhammer

DTU Fotonik, Technical University of Denmark, Building 343, Lyngby 2800, Denmark
Email:{xhua, sofo}@fotonik.dtu.dk

ABSTRACT

Distributed Video Coding (DVC) has been proposed as a new video coding paradigm to deal with lossy source coding using side information to exploit the statistics at the decoder to reduce computational demands at the encoder. A virtual channel noise model is utilized at the decoder to estimate the noise distribution between the side information frame and the original frame. This is one of the most important aspects influencing the coding performance of DVC. Noise models with different granularity have been proposed. In this paper, an improved noise model for transform domain Wyner-Ziv video coding is proposed, which utilizes cross-band correlation to estimate the Laplacian parameters more accurately. Experimental results show that the proposed noise model can improve the Rate-Distortion (RD) performance.

Index Terms— DVC, virtual channel, noise model, cross-band correlation

1. INTRODUCTION

Distributed Video Coding (DVC) [1] aims at avoiding complex motion estimation and compensation at the encoder and only explore the video statistics at the decoder side. According to the Slepian-Wolf theorem [2], it is possible to achieve the same rate as a joint encoding system by independent encoding but joint decoding of two statistically dependent signals. The Wyner-Ziv theorem [3] extends the Slepian-Wolf theorem to a lossy case, which becomes the key theoretical basis of DVC. One approach to DVC is to use a feedback channel based transform domain Wyner-Ziv video coding scheme. This was first proposed by the Stanford group in [4], then improved by the DISCOVER group (DIStributed COding for Video sER-vices) [5]. The DISCOVER codec improved coding performance by including a better side information generation scheme [6], an optimal reconstruction [7] and a realistic online noise model [8] at the decoder side. The coding efficiency of DVC is highly dependent on the error correcting capability of the channel code. A more accurate virtual channel noise model between the side information frame and the original frame will lead to improved channel coding performance.

A Laplacian distribution is usually utilized to model the difference of the transformed coefficients between the original frame and the side information in DVC. Accurate estimation of the Laplacian parameter is a complex task in DVC, because the side information frame is not reconstructed at the encoder side and the original frame is not available at the decoder side. Recently, different granularity online models [8][9] have been proposed to estimate the Laplacian distribution, i.e. from band (frame) level to coefficient (pixel) level for transform (pixel) domain Wyner-Ziv video coding. The results indicate that including finer granularity in the noise model improves the Rate-Distortion (RD) performance. In order to further improve the RD performance of transform domain Wyner-Ziv vide coding, an improved noise model with a more accurate estimatic of the Laplacian parameters is proposed. In the proposed model, category map is generated based on previous successfully decode bands, which are utilized to divide transformed coefficients of tl current band into two categories. Different parameter estimators a applied for these two categories to locally calculate the Laplacian p rameters. Finally, each transformed coefficient is assigned a Lapl cian parameter based on its corresponding category and reliability.

The rest of this paper is organized as follows: Section 2 brief describes the architecture of transform domain Wyner-Ziv vide coding. In Section 3, noise models with different granularity a first described. Thereafter the proposed model is introduced. Te conditions and results are presented in Section 4.

2. ARCHITECTURE OF TRANSFORM DOMAIN WYNER-ZIV VIDEO CODING

The architecture of a transform domain Wyner-Ziv video code [4][5] is depicted in Fig. 1. A fixed Group of Pictures (GOP=2) adopted. The video sequence is first split into odd (key) frames ar even (Wyner-Ziv) frames. The odd frames are intra coded by usir a conventional video coding like H.264/AVC while the even fram are Wyner-Ziv coded.

In the encoder, Wyner-Ziv frames are partitioned into no overlapped 4x4 blocks and an integer discrete cosine transfor (DCT) is applied on each of these. The transform coefficients with a given band $b_k, k \in \{0...15\}$, are grouped together and then quai tized [4]. DC coefficients and AC coefficients are uniformly scal quantized and dead zone quantized, respectively. After quantiz tion, the coefficients are binarized, each bitplane is transmitted a rate-compatible LDPC accumulate encoder [10] starting from th most significant bitplane. For each encoded bitplane, the corr sponding accumulated syndrome is stored in a buffer together wi an 8-bit Cyclic Redundancy Check (CRC). CRC is used to aid th decoder detecting the convergence. The amount of bits to be tran mitted depends on the requests from the decoder through a feedbac channel.

In the decoder, an Overlapped Block Motion Compensatic (OBMC) based interpolation algorithm [11] is adopted to create side information frame Y_{2i} and a motion estimated residual fran R_{ME} based on two intra coded frames X_{2i-1} and X_{2i+1}. Y and R_{ME} undergo the same 4x4 integer DCT to obtain coefficien $C_{Y_{2i}}$ and $C_{R_{ME}}$. $C_{R_{ME}}$ is utilized to model the noise distribu tion between corresponding DCT bands of the side information ar Wyner-Ziv frames (i.e. $C_{Y_{2i}}$ and $C_{X_{2i}}$). By using the noise distribu tion obtained, coefficient values of the side information frame C_Y and the previous successfully decoded bitplanes, soft informatic (conditional bit probabilities P_{cond}) for each bitplane is estimate With a given soft-input information P_{cond}, the LDPC decoder star

[0]X. Huang and S. Forchhammer "Improved Virtual Channel Noise Model for Transform Domain Wyner-Ziv Video Coding", *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing 2009*, Taiwan, ROC, April.

to process the corresponding bitplanes to correct the bit errors. Convergence is tested based on the 8-bit CRC and the Hamming distance between the received syndrome and the one obtained by the decoded bitplane: If the Hamming distance is different from zero after a certain amount of iterations, the LDPC decoder requests more accumulated syndrome bits from the encoder buffer via the feedback channel. If the Hamming distance is equal to zero, then the 8-bit CRC sum is requested from the buffer to verify successful decoding. A decoded bitplane with correct CRC sum is sent to a reconstruction module, a bitplane with incorrect CRC sum requests more accumulated syndrome bits from the encoder buffer to correct the existing bit errors until a low error probability is guaranteed.

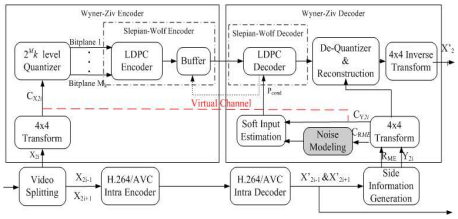


Fig. 1. Diagram of transform domain Wyner-Ziv video codec architecture

3. ONLINE NOISE MODELS

In order to take advantage of side information for decoding, the Wyner-Ziv decoder needs reliable information describing the noise distribution between the original frame and the side information frame R_{XY}. As a realistic solution in [8][9], a motion compensated residual R_{ME} between two key frames X_{2i-1} and X_{2i+1} is used (instead of an unrealistic offline residual R_{XY}) to estimate the Laplacian distribution parameter at the decoder side. Based on the work in [11], OBMC based side information generation is applied, therefore the motion compensated residual R_{ME} is obtained by:

$$R_{ME}(m_0, n_0) = \Sigma_{j=0}^{k} \omega_j \hat{R}_j / \Sigma_{j=0}^{k} \omega_j \quad (1)$$

$$\hat{R}_j = (X_{2i-1}(m_0 + \Delta m_j, n_0 + \Delta n_j) - X_{2i+1}(m_0 - \Delta m_j, n_0 - \Delta n_j)) \quad (2)$$

where (m_0, n_0) is the position within the current block, $(\Delta m_j, \Delta n_j)$ is the motion vector of the neighboring block j ($Block_j$) and k denotes the number of the neighboring blocks. ω_j is the weight of $Block_j$ obtained by:

$$\omega_j = (E_j[(X_{2i-1}(m_j + \Delta m_j, n_j + \Delta n_j) - X_{2i+1}(m_j - \Delta m_j, n_j - \Delta n_j))^2])^{-1} \quad (3)$$

where E_j is the expected value over $(m_j, n_j) \in Block_j$.

Different granularity online noise models for pixel domain and transform domain Wyner-Ziv video coding are discussed in [8][9]. In the following sub-sections, the band level and coefficient level noise models for transform domain Wyner-Ziv video coding are described first, then the proposed noise model is introduced.

3.1. Band Level

With the motion compensated residual R_{ME}, 16 bands of transformed residual coefficients $C^{b_k}_{R_{ME}}, b_k \in \{0...15\}$ are obtained after the 4x4 DCT transform. For a given band b_k, different Laplacian parameters $\alpha^{|\sigma|}_{b_k}$ are used to online model the distribution between transformed coefficients $C^{b_k}_{X_{2i}}$ and $C^{b_k}_{Y_{2i}}$:

$$f(C^{b_k}_{X_{2i}} - C^{b_k}_{Y_{2i}}) \approx \frac{\alpha^{|\sigma|}_{b_k}}{2} e^{-\alpha^{|\sigma|}_{b_k} |C^{b_k}_{R_{ME}}|} \quad (4)$$

$$\alpha^{|\sigma|}_{b_k} = \sqrt{2/\sigma^2_{|b_k|}}, \sigma^2_{|b_k|} = E(|C^{b_k}_{R_{ME}}|^2) - E(|C^{b_k}_{R_{ME}}|)^2 \quad (5)$$

where $\sigma^2_{|b_k|}$ is the variance of the absolute value of the transformed motion compensated residual ($|C^{b_k}_{R_{ME}}|$) within band b_k. The absolute value is chosen for Laplacian parameter estimation, since it is observed that the distribution with parameter $\alpha^{|\sigma|}_{b_k}$ is in general closer to the histogram of the actual residual $C^{b_k}_{R_{XY}} (= C^{b_k}_{X_{2i}} - C^{b_k}_{Y_{2i}})$ compared with the distribution with the parameter $\alpha^{\sigma}_{b_k}$ obtained by residual ($C^{b_k}_{R_{ME}}$) through experiments [8] (See also Fig. 2).

3.2. Coefficient Level

In the band level noise model, the same Laplacian parameter $\alpha^{|\sigma|}_{b_k}$ is utilized for all the coefficients within band b_k. The spatial variation between different blocks is not explored, thus a coefficients level noise model ($c1$) is proposed in [8] to exploit spatial variation.

$$\alpha^{c1}_{b_k}(u,v) = \begin{cases} \alpha^{|\sigma|}_{b_k}, & \text{if } D(u,v)^2 \le \sigma^2_{|b_k|} \\ \sqrt{2/D(u,v)^2}, & \text{if } D(u,v)^2 > \sigma^2_{|b_k|} \end{cases} \quad (6)$$

$$D(u,v) = C^{b_k}_{R_{ME}}(u,v) - E(|C^{b_k}_{R_{ME}}|) \quad (7)$$

where $\alpha^{c1}_{b_k}(u,v)$ represents the estimated Laplacian parameter for the coefficient located at (u,v) within band b_k. $\alpha^{|\sigma|}_{b_k}$ and $\sigma^2_{|b_k|}$ are estimates of the Laplacian parameter and the variance at band level. $E(|C^{b_k}_{R_{ME}}|)$ represents the average absolute value of coefficients in band b_k. $C^{b_k}_{R_{ME}}(u,v)$ is the coefficients value at position (u,v) within band b_k. This coefficient level noise model divides coefficients into two categories by comparing D^2 and the variance $\sigma^2_{|b_k|}$. If D^2 is smaller than the variance, the band level Laplacian parameter $\alpha^{|\sigma|}_{b_k}$ is applied. Otherwise, the coefficient level parameter $\sqrt{2/D(u,v)^2}$ is assigned [8].

3.3. Proposed noise model

A pixel level noise model is proposed in [9] for pixel domain Wyner-Ziv video coding. This work is here extended to a coefficient level noise model ($c2$) for transform domain Wyner-Ziv video coding which weights band level and coefficient level statistics.

$$\alpha^{c2}_{b_k}(u,v) = \frac{\beta \cdot E(|C^{b_k}_{R_{ME}}|) \cdot \alpha^{|\sigma|}_{b_k}}{(\beta - 1) \cdot |C^{b_k}_{R_{ME}}(u,v)| + E(|C^{b_k}_{R_{ME}}|)} \quad (8)$$

where parameter β determines the amplitude of the deviations of $\alpha^{c2}_{b_k}(u,v)$ from $\alpha^{|\sigma|}_{b_k}$. $\beta = 2$ was chosen experimentally [9]. Generally, this noise model assigns Laplacian parameters adaptively based on the absolute magnitude of the transformed motion compensated residual. The larger the absolute transformed residual $|C^{b_k}_{R_{ME}}(u,v)|$ is, the less reliable it is, and therefore a smaller Laplacian parameter $\alpha_{b_k}(u,v)$ is assigned.

As in [8][9], the variance $\sigma^2_{|b_k|}$ is utilized to estimate the Laplacian parameter at band level (Eq. 5) which in turn influences the estimated coefficient level (Eqs. 6 and 8). The maximum likehood estimator can also be used to estimate the Laplacian parameter:

$$\alpha^{|b|}_{b_k} = ((\sum ||C^{b_k}_{R_{ME}}| - E(|C^{b_k}_{R_{ME}}|)|)/N)^{-1} \quad (9)$$

Assuming a Laplacian distribution, these two different estimators (Eqs. 5 and 9) should give the same parameter value. However, as shown in Fig. 2, the experiments indicate that $\alpha^{|b|}_{b_k}$ is generally larger than $\alpha^{|\sigma|}_{b_k}$. The histogram of the actual residual $C^{b_k}_{R_{XY}}$ is more

peaked and has longer tails than the assumed Laplacian distribution. $\alpha_{b_k}^{|b|}$ is closer to the histogram close to zero while the $\alpha_{b_k}^{|\sigma|}$ is closer at the high values. Therefore it is reasonable to classify coefficients into two categories and apply the estimators $\alpha_{b_k}^{|b|}$ (Eq. 5) and $\alpha_{b_k}^{|\sigma|}$ (Eq. 9) for each category, respectively. Further, these estimators will be based on the coefficients within the respective category.

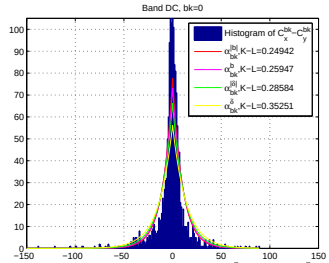


Fig. 2. Histogram of the actual residual $C_{R_{XY}}^0 = C_{X_{2i}}^0 - C_{Y_{2i}}^0$ and the estimated distributions with different estimators (DC coefficients, frame 22 of Foreman). Kullback-Leibler distances (KL) are calculated to compare the distance between the true distribution and modeling distribution.

The coefficient level noise model proposed in [8] classifies coefficients by comparing $D(u,v)^2$ and the variance $\sigma_{|b_k|}^2$ as shown in Eq. 6. However, this calculation is only based on $C_{R_{ME}}^{b_k}$, which may be unreliable in some regions. Only using $C_{R_{ME}}^{b_k}$ (Eq. 6) may lead to inaccurate local parameter calculation. The correlation between classifications of different bands is tested in Fig. 3(a) based on comparing $D(u,v)^2$ and $\sigma_{|b_k|}^2$ of the actual residual $C_{R_{XY}}^{b_k}$.Therefore cross-band correlation can be utilized.

Since the Wyner-Ziv frames can be decoded successively band by band, after successfully decoding one (lower frequency) band b_k, an unfinished decoded frame (Z) can be reconstructed. By calculating the coefficients difference between $C_Z^{b_k}$ and $C_{Y_{2i}}^{b_k}$, an updated residual $C_{R_{ZY}}^{b_k}$ in band b_k is obtained, which is closer to the actual residual $C_{R_{XY}}^{b_k}$ than the motion compensated residual $C_{R_{ME}}^{b_k}$. The $\sigma_{|b_k|}^2$ and $D(u,v)^2$ in Eqs. 5 and 7 are recalculated based on the updated residual $C_{R_{ZY}}^{b_k}$, the classification map of band b_k is obtained as:

$$map_{b_k}^{out} = \{(u,v)|D(u,v)^2 > \sigma_{|b_k|}^2\} \quad (10)$$

$$map_{b_k}^{in} = \{(u,v)|D(u,v)^2 \leq \sigma_{|b_k|}^2\} \quad (11)$$

Due to the existing cross-band correlation, classification map of band b_k can be utilized to estimate the classification of the next (higher frequency) band $b_l, l > k$. The classification estimation follows the decoding order as shown in Fig. 3(b). For instance, after the first band is successfully decoded, the classification map of band 1 (map_1^{out},map_1^{in}) is obtained as described in Eqs. 10 and 11. The classification maps of band 2 and band 3 are simply estimated by copying the map of the neighboring band 1, i.e. $map_3^{out} = map_2^{out} = map_1^{out}$ and $map_3^{in} = map_2^{in} = map_1^{in}$. Similarly, the classification map of band 5 is estimated by using band 2 and band 3 by $map_5^{out} = map_2^{out} \cup map_3^{out}$ and $map_5^{in} = map_2^{in} \cup map_3^{in}$ etc. With the estimated classification, $\alpha_{b_k}^{|b|}$ and $\alpha_{b_k}^{|\sigma|}$ can be calculated within the coefficient sets $map_{b_k}^{in}$ and $map_{b_k}^{out}$, respectively.

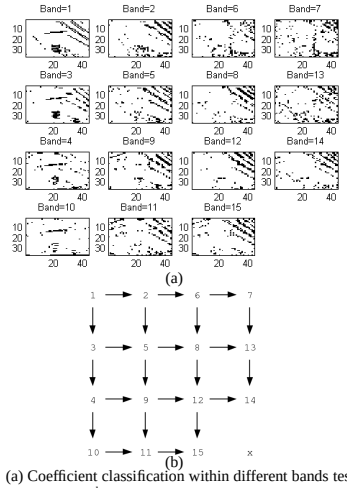


Fig. 3. (a) Coefficient classification within different bands tested on the actual residual $C_{R_{XY}}^{b_k}$ (Frame 22 of Foreman). (b) The classification estimation from lower frequency band to higher frequency band

$$\alpha_{map_{b_k}^{in}}^{|b|} = ((\sum ||C_{R_{ME}}^{map_{b_k}^{in}}| - E(|C_{R_{ME}}^{map_{b_k}^{in}}|))|)/N)^{-1} \quad (12)$$

$$\alpha_{map_{b_k}^{out}}^{|\sigma|} = \sqrt{2/(E(|C_{R_{ME}}^{map_{b_k}^{out}}|^2) - E(|C_{R_{ME}}^{map_{b_k}^{out}}|)^2)} \quad (13)$$

In order to combine the advantages of the two coefficient level noise models described in the subsections 3.2 and 3.3, the Laplacian parameters for lower frequency bands and higher frequency bands are assigned differently. Let $\alpha_{b_k}^{c2}[(u,v)|C_{R_{ME}}^{map_{b_k}^{\bullet}}, \alpha_{b_k}^{|\sigma|}]$ denote the function in Eq. 8. For coefficients $C_{R_{ME}}^{b_k}, b_k \in \{0,1,2\}$,

$$\alpha_{b_k}(\mathbf{u,v}) = \begin{cases} \alpha_{b_k}^{c2}[(u,v)|C_{R_{ME}}^{map_{b_k}^{in}}, \alpha_{map_{b_k}^{in}}^{|b|}] & (u,v) \in map_{b_k}^{in} \\ \alpha_{b_k}^{c2}[(u,v)|C_{R_{ME}}^{map_{b_k}^{out}}, \alpha_{map_{b_k}^{out}}^{|\sigma|}] & (u,v) \in map_{b_k}^{out} \end{cases}$$

For coefficients $C_{R_{ME}}^{b_k}, b_k \in \{3...15\}$,

$$\alpha_{b_k}(\mathbf{u,v}) = \begin{cases} \alpha_{map_{b_k}^{out}}^{|\sigma|} & \text{if } \sqrt{2/D(u,v)^2} \geq \alpha_{map_{b_k}^{out}}^{|\sigma|} \\ & \cup (u,v) \in map_{b_k}^{out} \\ \alpha_{map_{b_k}^{in}}^{|b|} & \text{if } \sqrt{2/D(u,v)^2} \geq \alpha_{map_{b_k}^{in}}^{|b|} \\ & \cup (u,v) \in map_{b_k}^{in} \\ \sqrt{2/D(u,v)^2}, & \text{otherwise} \end{cases} \quad (15$$

4. EXPERIMENTAL RESULTS

The following test conditions are used to obtain the RD performance results: The test sequences (available on [5]) are 149 frames of "Foreman", "Soccer", "Coast-guard" and "Hallmonitor" at 15 frames per second (fps). The most common GOP length of 2 is used. The key frames are encoded by H.264/AVC intra and the QPs

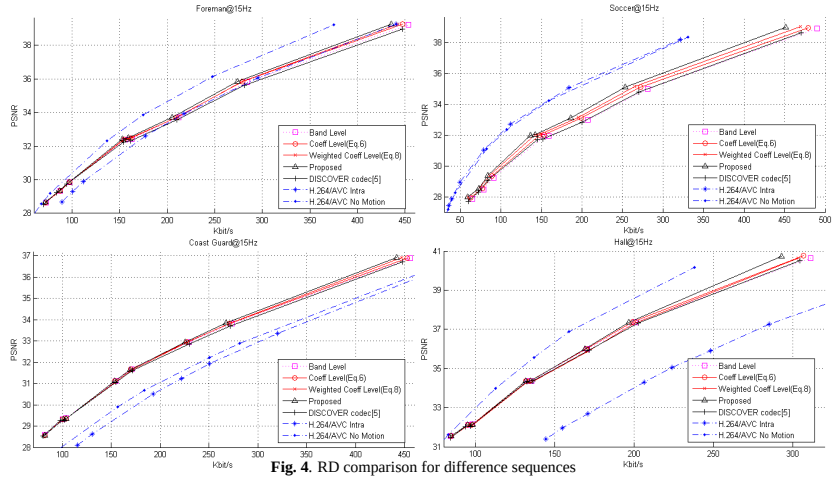


Fig. 4. RD comparison for difference sequences

are chosen so that the average PSNR of Wyner-Ziv frames are similar to the quality of key frames as in [5]. Overlapped Block Motion Compensation (OBMC) based side information generation [11] with half-pixel accuracy is utilized. The RD results are evaluated by the average for the luminance components of key frames and Wyner-Ziv frames. RD performance results of transform domain Wyner-Ziv video coding with different noise models are compared.

The experimental results are depicted in Fig 4. The performance of the DISCOVER executable codec [5]-[8] is depicted for comparison. The performance of H.264/AVC intra coding and H.264/AVC frame difference coding (i.e. No motion estimation with IBI GOP structure) are also included. The band level noise model with side information generation [11] is seen as a baseline. The coefficient level noise models achieve better RD performance than band level noise model. Compared with the coefficient level model [8] (Eq. 6) employed in the DISCOVER codec, the weighted coefficient level model (Eq. 8) gives better RD performance results for sequences "Foreman", "Soccer" and "Coast-guard", but worse RD performance for sequence "Hallmonitor". The proposed noise model achieves better RD performance than all the other noise models. Compared with the coefficient level noise models, the proposed noise model is more robust and it improves the RD performance for high bit-rates up to 0.5 dB.

5. CONCLUSION

In this paper, an improved virtual channel noise model is proposed for transformed domain Wyner-Ziv video coding. It classifies the transformed coefficients into two categories by using the cross-band correlations, applies different estimators to locally calculate the Laplacian parameters and thus adaptively assigns a parameter value for each coefficient. Experimental results show that the proposed noise model can improve the coding efficiency of transformed domain Wyner-Ziv video coding up to 0.5 dB compared with the other noise models.

6. REFERENCES

[1] A. Aaron, R. Zhang, and B. Girod, "Wyner-ziv coding of motion video," *Proc. Asilomar Conf. on Signals and Syst.*, pp. 240–244, 2002.

[2] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, pp. 471–480, July 1973.

[3] A.D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, Jan. 1976.

[4] A. Aaron, S. Rane, E. Setton, and B. Girod, "Transform domain wyner-ziv codec for video," *Proc. SPIE VCIP*, pp. 520–528, Jan. 2004.

[5] Available on: www.discoverdvc.org.

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

[7] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal reconstruction in wyner-ziv video coding with multiple side information," *IEEE Int'l Workshop Multimedia Signal Process.*, pp. 183–186, Oct. 2007.

[8] C. Brites and F. Pereira, "Correlation noise modelling for efficient pixel and transform domain wyner-ziv video coding," *IEEE Trans. on Circuits Syst. Video Technol.*, 2008.

[9] L. Qing, X. He, and R. Lv, "Distributed video coding with dynamic virtual channel mode estimation," *Int'l Symposium on Data, Privacy and E-Commerce*, pp. 170–173, 2007.

[10] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive distributed source coding using low-density parity-check codes," *EURASIP Signal Process. Journal, Special Section on Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[11] X. Huang and S. Forchhammer, "Improved side information generation for distributed video coding," *IEEE Int'l Workshop Multimedia Signal Process.*, pp. 223–228, Oct. 2008.

DISTRIBUTED VIDEO CODING WITH MULTIPLE SIDE INFORMATION

Xin Huang[1], *Catarina Brites*[2], *João Ascenso*[3], *Fernando Pereira*[2], *and Søren Forchhammer*[1]

[1]DTU Fotonik, Technical University of Denmark
[2]Instituto Superior Técnico - Instituto de Telecomunicações, Portugal
[3]Instituto Superior de Engenharia de Lisboa - Instituto de Telecomunicações, Portugal

ABSTRACT

Distributed Video Coding (DVC) is a new video coding paradigm which mainly exploits the source statistics at the decoder based on the availability of some decoder side information. The quality of the side information has a major impact on the DVC Rate-Distortion (RD) performance in the same way the quality of the predictions had a major impact in predictive video coding. In this paper, a DVC solution exploiting multiple side information is proposed; the multiple side information is generated by frame interpolation and frame extrapolation targeting to improve the side information of a single estimation mode. Compared with the best available single side information solutions, the proposed DVC solution with multiple side information robustly improves the RD performance for the set of test sequences.

Index Terms— Distributed Video Coding, multiple side information, soft input.

1. INTRODUCTION

Distributed Video Coding (DVC) [1] proposes to fully or partly exploit the video redundancy at the decoder and not anymore at the encoder as in predictive video coding. According to the Slepian-Wolf theorem [2], it is possible to achieve the same rate by independently encoding but jointly decoding two statistically dependent signals as for typical joint encoding and decoding (with a vanishing error probability). The Wyner-Ziv theorem [3] extends the Slepian-Wolf theorem to the lossy case, becoming the key theoretical basis for Wyner-Ziv (WZ) video coding where some source is lossy coded based on the availability of some correlated source at the decoder from which the so-called side information is derived.

Feedback channel based transform domain Wyner-Ziv video codecs [4] are the most popular approaches to WZ video coding. Since the quality of the side information has a major impact on the final RD performance, there are several side information generation schemes proposed in the literature, notably frame interpolation [5] and frame extrapolation [6] based algorithms. Frame interpolation methods use previous and future decoded frames to generate the side information introducing some delay, while the extrapolation methods only use previously decoded frames. Generally, WZ coding with interpolated side information has better RD performance, notably for small GOP (Group of Pictures) sizes [6]. However, extrapolated side information has benefits for real-time applications due to the lower delay.

Since neither the available interpolation nor the extrapolation solution is perfect in terms of the created side information which is taken as estimation for the frames to WZ encode, the coding efficiency of Wyner-Ziv (WZ) video coding with single side information can be improved. The objective of this paper is to further progress the RD performance of WZ video coding, also reducing the RD gap first development in this area has been proposed in [7], where tw different frame interpolation methods to generate the multiple sic information are used. The channel decoder is fed with the averag of two soft inputs which are generated based on two different sic information estimates and the corresponding noise models. A mo accurate soft input is obtained and the RD performance is improve up to 0.3 dB.

Differently, in this paper, the multiple side information is ge erated by frame interpolation and extrapolation. The intuition he is that having more diverse side information solutions should allo these to compensate each other's estimation weaknesses dependir on the video content, overall leading to a more efficient coding s lution. In this context, the extrapolated and the interpolated side i formation frames can be seen as original frames transmitted throug quite different 'channels' and thus each side information frame seen as an observation with a different amount of 'correlation noise With multiple observations, the WZ video decoder can select combine the available side information estimates to decrease t amount of 'correlation noise' and thus to reduce misleading soft i puts in comparison with the single side information solution. In th way, the novel proposed solution shall reduce the required parity ra for each target quality, improving the RD performance.

The rest of this paper is organized as follows: Section 2 brief describes the state-of-art on transform domain WZ video codir with feedback channel. In Section 3, the novel WZ decoder with i terpolated and extrapolated side information is proposed. Finally, t test conditions and performance results are presented in Section 4.

2. STATE-OF-ART ON TRANSFORM DOMAIN WYNER-ZIV VIDEO CODING

A fixed Group of Pictures (GOP=*N*) is adopted in the state-of-a transform domain WZ video codec with feedback channel [4]. Per odically one frame out of *N* in the video sequence is named as ke frame and intermediate frames are WZ frames. The key frames a intra coded by using a conventional video coding solution with lo complexity such as H.264/AVC intra while the WZ frames are code using a Wyner-Ziv video coding approach.

At the encoder, the WZ frames are partitioned into no overlapped 4×4 blocks and an integer discrete cosine transfor (DCT) is applied to each of them. The transform coefficients a grouped together and then quantized. After quantization, the coefl cients are binarized, and each bitplane is given to a rate compatib Low Density Parity Check (LDPC) accumulate encoder [8] startir from the most significant bitplane. For each encoded bitplane, t corresponding accumulated syndrome is stored in a buffer at t encoder together with an 8-bit Cyclic Redundancy Check (CRC The amount of bits to be transmitted depends on the requests mac by the decoder through a feedback channel (Fig. 1).

[0]X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer "Distributed Video Coding with Multiple Side Information", *Proceedings of IEEE Picture Coding Symposium 2009*, Chicago, USA, May. 2009

[5][6]. Together with an estimated noise residue frame R, Y undergoes the integer DCT to obtain the coefficients C_Y and C_R. C_R is used to model the noise distribution between the corresponding DCT bands of the side information frame and the original WZ frame. Using the noise model [9], the coefficient values of the side information frame C_Y and the previous successfully decoded bitplanes, soft-input P (conditional bit probabilities) for each bitplane is estimated. With this soft-input P, the LDPC decoder starts to process the various bitplanes to correct the bit estimation errors. Convergence is tested by the 8-bit CRC sum and the Hamming distance between the received syndrome and the one obtained from the decoded bitplane: If the Hamming distance is different from zero or the CRC sum is incorrect after a certain amount of iterations, the LDPC decoder requests more accumulated syndrome bits from the encoder buffer via the feedback channel to correct the existing bit errors. If both the Hamming distance and CRC sum are satisfied, convergence is declared, guaranteeing a very low error probability for the decoded bitplane. For more details please refer to [4].

3. WYNER-ZIV DECODER WITH MULTIPLE SIDE INFORMATION

As mentioned before, the choice of the adopted side information generation scheme significantly influences the final coding efficiency. There are several interpolation and extrapolation methods in the literature, all targeting the generation of good quality side information frames [5][6]. The side information frames obtained are going to be used to estimate the soft-input information (conditional bit probabilities) for each bitplane based on a certain noise model [9]. The essential factor to reduce the number of coding bits is the soft-input information which is fed into the LDPC decoder. The more accurate the soft input is, the fewer parity bits are required by the decoder since the faster the convergence will be. Thus, an important way to increase the RD performance is to improve the soft-input information fed into the LDPC decoder.

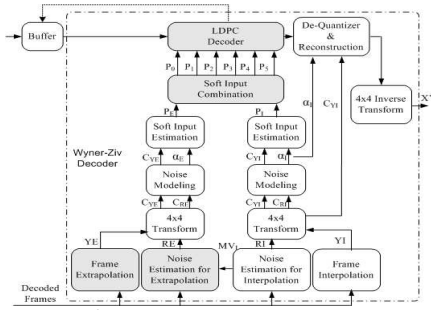


Fig. 1. *Transform domain Wyner-Ziv video decoder with interpolated and extrapolated side information*

The novel proposed WZ video codec with multiple side information follows this approach with the motivation described in Section 2. The encoder is not changed, as the basic idea is to generate better soft-input information by generating first better quality side information, in this case multiple side information through interpolation and extrapolation. While interpolation solutions are the most common in the literature, the WZ video codec proposed in this paper expects to improve the overall RD performance by also processing extrapolation side information which may be 'better' than interpolation side information for some conditions of the content. The architecture proposed for the novel WZ decoder with multiple side information is presented in Fig. 1. The track at the right starting with interpolation (*RI* and *YI*) presents a state-of-art WZ solution with interpolation. The technical novelty of the proposed WZ video decoder includes: i) an improved extrapolation method, ii) the noise estimation for extrapolation, iii) the soft inputs combination module, and iv) modified LDPC decoder.

3.1. WZ Decoder with Multiple Side Information Architecture

The main modules in the novel proposed WZ video decoder are:

• ***Frame Interpolation***: The adopted frame interpolation procedure is the same as in [5]. Without loss of generality, it generates the side information frame YI_{2i} by using intra coded frames, X'_{2i-1} and X'_{2i+1} for GOP size 2. It includes forward motion estimation, bi-directional motion estimation, spatial smoothing of Motion Vectors (MV), motion refinement with variable block size and adaptive weighted Overlapped Block Motion Compensation (OBMC).

• ***Noise Estimation for Interpolation***: A motion estimated residue frame R_{ME} (i.e. the difference between X'_{2i-1} and X'_{2i+1} after motion compensation) is taken as the estimated noise residue *RI* to express the correlation noise between the WZ frame and the corresponding interpolated frame.

• ***Frame Extrapolation***: This module creates the extrapolated side information. The procedure is similar to [6]. Without loss of generality, the previous coded frames X'_{2i-1} and X'_{2i-2} are used to generate the side information frame YE_{2i} for GOP size 2. It includes motion estimation, spatial smoothing, frame projection, overlapping and filling holes. The difference is that a novel hole filling technique is applied. For the unreferenced/unfilled pixel areas in frame YE_{2i}, both the nearest MVs in the spatial domain and co-located MVs in temporal domain are used to determine the estimated pixels; an average of these estimates is computed for filling the holes remaining after the frame projection process.

• ***Noise Estimation for Extrapolation***: The noise residue *RE* is computed to present the correlation noise between the WZ frame and the corresponding extrapolated frame as described in Section 3.2.

• ***Noise Modeling***: After computing the 4×4 integer DCT coefficients C_{YI}, C_{YE}, C_{RI} and C_{RE} for the interpolated and extrapolated side information and the associated residues, the noise distribution between the side information and the corresponding WZ frame is estimated using a Laplacian noise model as described in [9]. Within a given DCT band b_k, the DCT coefficient at coordinates (m, n) is associated to the Laplacian parameter $\alpha_E^{bk}(m, n)$ for extrapolation and $\alpha_I^{bk}(m, n)$ for interpolation. The Laplacian parameter values express the reliability of the side information, i.e. the smaller this value is, the noisier the corresponding coefficient is.

• ***Soft Input Estimation***: With the obtained Laplacian parameters, side information coefficient values and the previous successfully decoded bitplanes, the soft-input information (conditional bit probabilities for extrapolation P_E and for interpolation P_I) of each bitplane are estimated [4].

• ***Soft Input Combination***: The soft input data to be provided to the LDPC decoder is generated by combining the soft inputs P_E and P_I in a few predefined modes creating various soft input candidates; see details in Section 3.3.

• ***LDPC Decoder***: All these candidate soft inputs are fed to a modified LDPC decoder. The soft input which converges (as described in Section 2) first is chosen by the LDPC decoder (Section 3.3) thus minimizing the rate of parity bits for a certain target quality.

• ***Reconstruction***: Based on the decoded bins, this module has to recover the coefficient's values also exploiting the available side information. Since the interpolated side information is typically better (see Fig. 2), the interpolated side information and its noise modeling

parameters are used by the reconstruction module [7] to recover the decoded WZ frames.

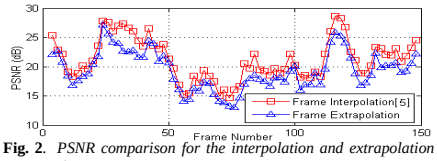


Fig. 2. *PSNR comparison for the interpolation and extrapolation methods for Soccer@15Hz, QCIF, GOP 2.*

3.2. Noise Estimation for Extrapolation

There are two natural ways to estimate the residue between WZ frames and the corresponding extrapolated side information to represent the correlation noise behavior:

• *Motion Estimated Residue* R_{ME}: Corresponds to the pixel differences between X'_{2i-1} and X'_{2i-2} along the extrapolated MVs.

• *No Motion Estimated Residue* R_{NO}: Corresponds to the co-located pixel differences between YE_{2i} and X'_{2i-1}.

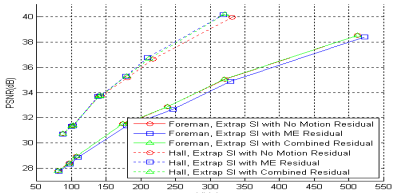


Fig. 3. *RD performance with extrapolated side information using the motion estimated and no motion estimated residues for Foreman and Hall Monitor, QCIF, 15 Hz.*

Experiments have shown that, when creating the side information using frame extrapolation, the more commonly used motion estimated residue [9] will provide a lower RD performance for high motion sequences while it will perform better for low motion sequences in comparison with the no motion estimated residue (see Fig. 3). The lower RD performance may be caused by the linear motion assumption adopted for the generation of the unidirectional MVs used for the frame extrapolation process. If these MVs are not fulfilling this assumption, then the extrapolated block is going to be projected into a wrong position, corresponding to a large real noise residue, while the motion estimated residue R_{ME} will be smaller. Based on this poorly estimated noise residue, the estimated Laplacian parameter will be inaccurate in terms of noise modeling, misleading the LPDC decoder in terms of the soft input P_E. In order to solve this problem, it is necessary to generate a more robust estimate for the noise residue when frame extrapolation is used. In this context, it is proposed here to check the 'accuracy' of the motion vectors obtained by extrapolation MV_E using the motion vectors obtained by frame interpolation MV_I. The intuition is that if the two sets of MVs are similar, then the motion description should be good and thus the motion estimated residue should be used. Following this intuition, a combined noise residue, R_{COM}, is computed by switching between R_{ME} and R_{NO} as:

$$R_{COM}(x,y) = \begin{cases} R_{ME}(x,y), & \text{if } MV_I(m,n)=MV_E(m,n) \\ R_{NO}(x,y), & \text{otherwise} \end{cases} \quad (1)$$

where (x,y) are the pixel coordinates and (m,n) are the corresponding block coordinates. The RD performance with single extrapolation side information using the proposed combined noise residue is compared with the relevant alternatives in Fig. 3 for the

3.3. Soft Input Combination

After the extrapolation soft input P_E and the interpolation soft input P_I are obtained, the soft input combination module has the task of adaptively combining these two soft inputs to generate a set of candidate soft inputs, thus improving the RD performance by reducing the rate of parity bits.

Since the values of the Laplacian parameters should express the reliability of the corresponding side information, an unreliability region map is defined as the region of the frame where extrapolation or interpolation indicates areas including discontinuous linear motion. It means there should be little benefit brought by extrapolation outside of the map region within which the motion is relative linear. This map region is determined by evaluating the Laplacian parameters and their corresponding mean value as:

$$map = \{(m,n) | \alpha_E^{bk}(m,n) < E(\alpha_E^{bk}) \vee \alpha_I^{bk}(m,n) < E(\alpha_I^{bk})\}$$

where $\alpha_E^{bk}(m,n)$ and $\alpha_I^{bk}(m,n)$ are the estimated Laplacian distribution parameters within DCT band b_k for extrapolation and interpolation, respectively. (m,n) are the block coordinates and $E(\alpha^{bk})$ represents the mean value of the Laplacian parameter over all the blocks within DCT band b_k.

In order to take advantage of the benefits brought by the extrapolation soft input P_E regarding a single interpolation side information solution, a set of candidate soft inputs is generated by combining the extrapolation soft input P_E with the interpolation soft input P_I within the unreliability region map, while only the interpolation soft input P_I is adopted in the reliable region (there is no expected benefit in also using P_E):

$$P_T(m,n) = \begin{cases} w_T \cdot P_I(m,n) + (1-w_T) \cdot P_E(m,n), & \text{if } (m,n) \in map \\ P_I(m,n), & \text{otherwise} \end{cases} \quad (3)$$

where $w_T = \{1 - (T/10) | T = 0,1,2,3,4,5\}$. All these candidate soft inputs are fed into the LDPC decoder; the one which first converges will be chosen thus reducing the rate of parity bits for the same target quality. By using this set of combined soft inputs, the extrapolation side information track will influence the LDPC decoding process, reducing the amount of misleading soft inputs provided by the interpolation side information track, following the intuition behind this paper and reaching the stated objective of improving the overall RD performance based on more and better side information. However, the set of combined soft inputs will increase the complexity of LDPC decoding up to $T+1=6$ times.

4. EXPERIMENTAL RESULTS

In order to make fair comparisons, the test conditions adopted in this paper are the DISCOVER project test conditions, commonly used in the DVC literature [4]. The test sequences are *Foreman*, *Soccer*, *Coastguard* and *Hall Monitor*, coded at QCIF, 15 frames per second (fps); the GOP size is 2. The key frames are encoded using H.264/AVC Intra and the QPs are chosen so that the average PSNR of the WZ frames is similar to the average PSNR of the key frames (as in [4]). The RD performance is evaluated for the luminance component of both the key frames and WZ frames. The benchmark codecs used are the DISCOVER WZ video codec [4] and the H.264/AVC Intra codec. For comparison, the performance of some other relevant transform domain WZ video codecs with single (interpolation [5] or extrapolation) and multiple (interpolation and extrapolation) side information is also included.

As shown in Figs. 4-7, the performance of the single interpolation side information WZ video codec is better than the DISCOVER codec due to the OBMC based interpolation side information method [5]. The RD performance with single interpolation side information is better than the one with single extrapolation side information meaning that the additional delay involved really brings additional RD performance. Moreover, based on precisely the same

H.264/AVC intra coded key frames, the multiple side information codec can improve the overall RD performance of single interpolation side information codec up to 0.4 dB at high bitrates for the WZ frames. Since the interpolation side information is quite efficient for low motion sequences, the extrapolation side information brings less RD performance improvements in the context of WZ coding with multiple side information for this type of video content. This means that compared with low motion sequences like *Hall Monitor*, WZ decoding with multiple side information provides larger RD gains for high motion sequences like *Foreman* and *Soccer*. WZ video coding with multiple side information already gives better RD performance than H.264/AVC intra coding for *Foreman*, *Coastguard* and *Hall Monitor*; for sequences with more irregular motion like *Soccer*, where the decoder frame estimation process is more difficult, the performance gap between H.264/AVC intra coding and WZ video coding has been reduced but not yet closed.

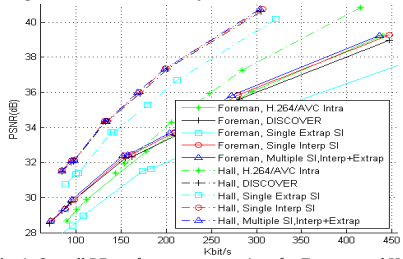


Fig. 4. *Overall RD performance comparison for Foreman and Hall.*

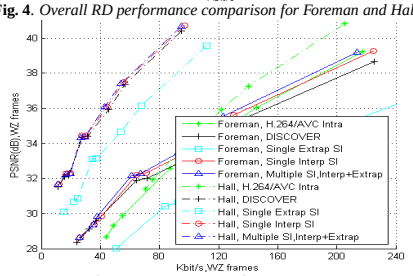


Fig. 5. *RD performance comparison for Foreman and Hall: only WZ frames for precisely the same key frames.*

5. CONCLUSION

A novel transform domain WZ video decoder with multiple (interpolation and extrapolation) side information is proposed in this paper with the objective to improve the overall RD performance. Although the extrapolated side information frames are significantly worse than the interpolated side information frames, improvement is robustly achieved by generating and combining a set of candidate soft inputs to be fed to the LDPC decoder, trying to reduce the number of bits requested by the decoder for a target quality; this process implies adaptively to combine the interpolation and extrapolation derived soft inputs with the aim of using the most reliable side information derived soft input depending on the video content. Compared with state-of-art single side information WZ video coding solutions, the proposed transform domain WZ video codec with multiple side information can improve the overall RD performance for the set of test sequences; the RD gains may go up to 0.4 dB (averaged over the sequence) for the WZ frames with precisely the same H.264/AVC intra coded key frames.

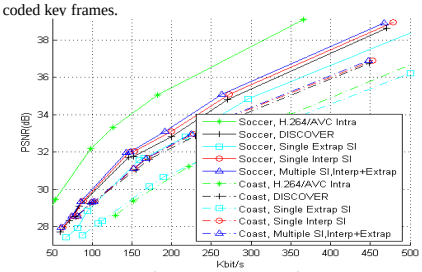


Fig. 6. *Overall RD performance comparison for Soccer and Coast.*

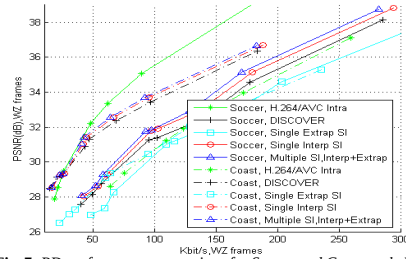


Fig. 7. *RD performance comparison for Soccer and Coast: only WZ frames for precisely the same key frames.*

6. REFERENCES

[1] A. Aaron, S. Rane, E. Setton, and B. Girod, "Transform domain Wyner-Ziv codec for video," *Proc. SPIE VCIP*, San Jose, USA, Jan. 2004.

[2] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, no.4, pp. 471–480, July 1973.

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

[4] DISCOVER Project: www.discoverdvc.org, Dec. 2007.

[5] X. Huang and S. Forchhammer, "Improved side information generation for distributed video coding," *in Proc. IEEE Int'l Workshop Multimedia Signal Process.*, pp. 223–228, Cairns, Australia, Oct. 2008.

[6] L. Natário, C. Brites, J. Ascenso, and F. Pereira, "Extrapolating side information for low-delay pixel-domain distributed video coding," *in Proc. Int'l Workshop on Very Low Bitrate Video Coding*, pp. 16–17, Sardinia, Italy, Sept. 2007.

[7] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal reconstruction in Wyner-Ziv video coding with multiple side information," *in Proc. IEEE Int'l Workshop Multimedia Signal Process.*, pp. 183–186, Chania, Greece, Oct. 2007.

[8] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive distributed source coding using low-density parity-check codes," *EURASIP Signal Process. Journal, Special Section on Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[9] C. Brites and F. Pereira, "Correlation noise modelling for efficient pixel and transform domain Wyner-Ziv video coding," *IEEE Trans. on Circuits. Syst. Video Technol.*, vol. 18, no.9, pp. 1177–1190, Sept. 2008.

Appendix B

Configuration of H.264/AVC

Configuration settings of H.264/AVC Intra coding, H.264/AVC no motion estimation Inter coding and H.264/AVC Inter coding are reported as follows:

Variable	Value	Variable	Value
ProfileIDC	77	LevelIDC	40
IntraPeriod	1	FrameSkip	0
IntraDisableInterOnly	0	Intra4x4ParDisable	0
Intra4x4DiagDisable	0	Intra4x4DirDisable	0
Intra16x16ParDisable	0	Intra16x16PlaneDisable	0
RDPictureDecision	0	RDPictureIntra	0
LoopFilterDisable	0	LoopFilterAlphaC0Offset	0
LoopFilterBetaOffset	0	RestrictSearchRange	2
RDOptimization	1	RandomIntraMBRefresh	0

Table B.1: Configuration setting of H.264/AVC intra coding

[0]The chosen encoding configurations of H.264/AVC motion and no motion Inter coding give similar coding efficiency results compared to the DISCOVER results.

Variable	Value	Variable	Value
ProfileIDC	77	LevelIDC	40
IntraPeriod	1	FrameSkip	1
SearchRange	0	RDOptimization	1
RandomIntraMBRefresh	0	InterSearch16x16	1
InterSearch16x8	0	InterSearch8x16	0
InterSearch8x8	0	InterSearch8x4	0
InterSearch4x8	0	InterSearch4x4	0
NumberBFrames	1	BiPredMotionEstimation	0
RDPictureDecision	0	RDPictureIntra	0
LoopFilterDisable	0	LoopFilterAlphaC0Offset	0
LoopFilterBetaOffset	0	RestrictSearchRange	2

Table B.2: Configuration setting of H.264/AVC no motion inter coding

Variable	Value	Variable	Value
ProfileIDC	77	LevelIDC	40
IntraPeriod	1	FrameSkip	1
SearchRange	16	RDOptimization	1
RandomIntraMBRefresh	0	InterSearch16x16	1
InterSearch16x8	1	InterSearch8x16	1
InterSearch8x8	1	InterSearch8x4	1
InterSearch4x8	1	InterSearch4x4	1
NumberBFrames	1	BiPredMotionEstimation	0
RDPictureDecision	0	RDPictureIntra	0
LoopFilterDisable	0	LoopFilterAlphaC0Offset	0
LoopFilterBetaOffset	0	RestrictSearchRange	2

Table B.3: Configuration setting of H.264/AVC inter coding with GOP *IBI*

Appendix C

Additional Results

C.1 Visual comparison of different side information frames

(a)

(b)

(c)

(d)

Figure C.1: Visual comparison of different side information frames, Foreman frame No. 30 (a) Original frame (b) Extrapolated SI PNSR=23.9389 dB (c) Motion compensation based interpolated SI PSNR=27.4079 dB (d) OBMC based interpolated SI PSNR=27.5019 dB

(a)

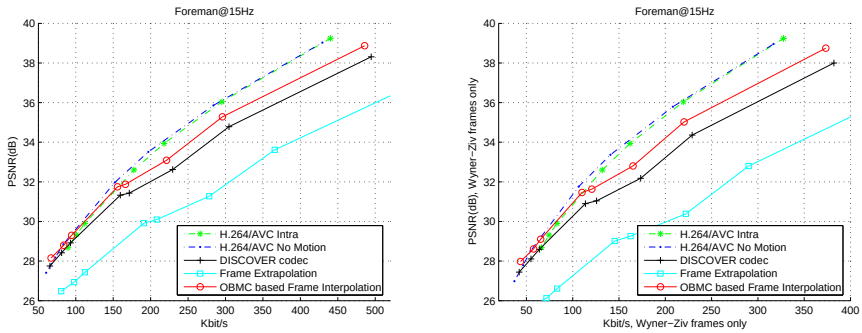
(b)

(c)

(d)

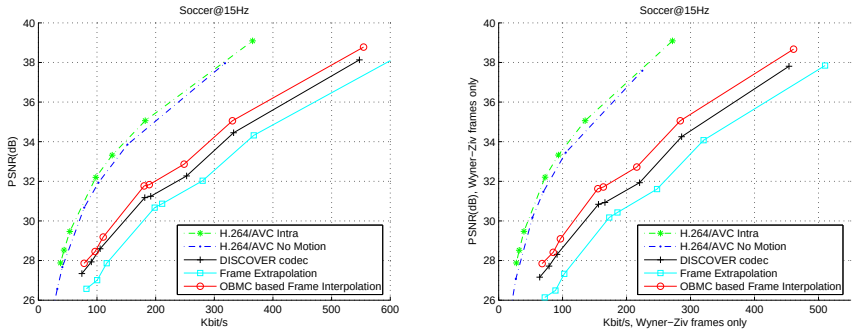
Figure C.2: Visual comparison of different side information frames, Soccer No. 10 (a) Original frame (b) Extrapolated SI PNSR=18.4067 dB (c) Motion compensation based interpolated SI PSNR=18.6658 dB (d) OBMC based interpolated SI PSNR=19.1154 dB

C.2 GOP4 RD performances comparison with different side information generation methods



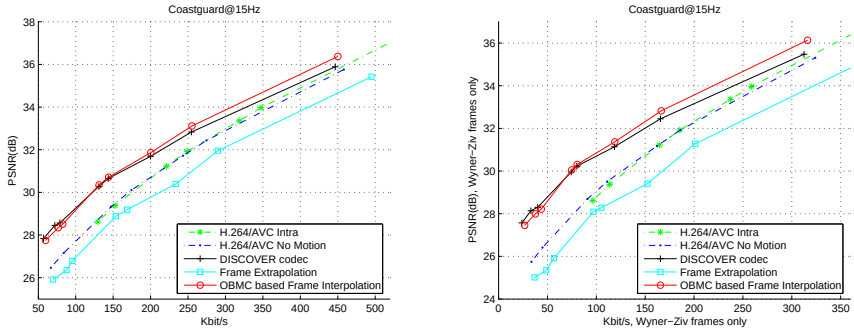
(a) (b)

Figure C.3: GOP4 RD performances comparison for sequence Foreman with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



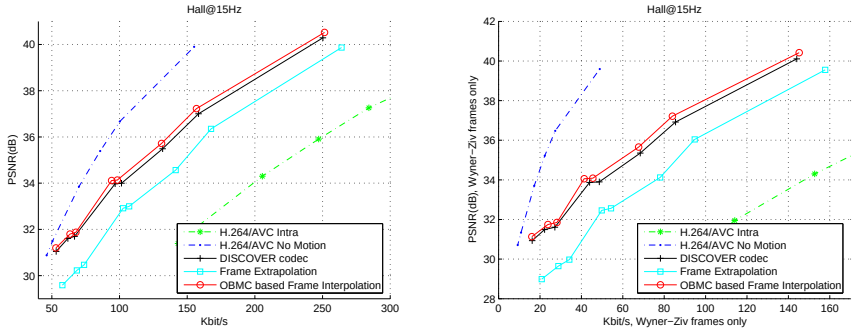
(a) (b)

Figure C.4: GOP4 RD performances comparison for sequence Soccer with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

Figure C.5: GOP4 RD performances comparison for sequence Coastguard with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

Figure C.6: GOP4 RD performances comparison for sequence Hall monitor with different side information generation methods (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

C.3 Band level Laplacian parameters comparison obtained by residue R_{XY} and R_{ME}

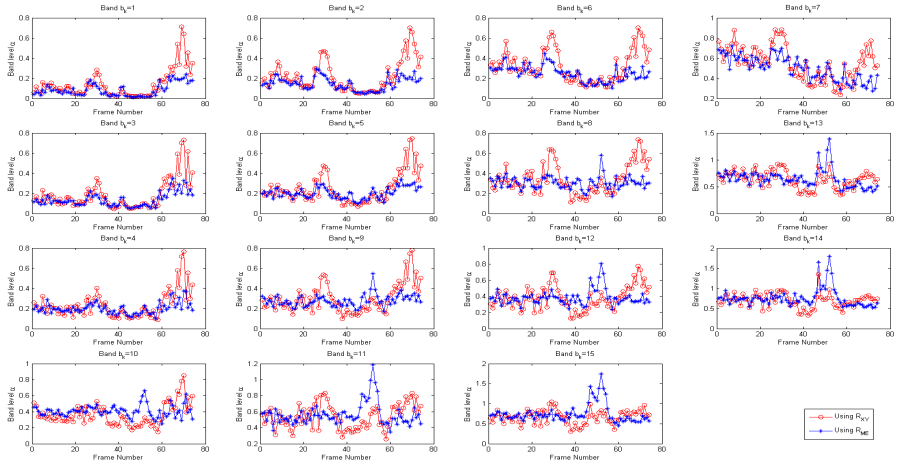


Figure C.7: Band level Laplacian parameters comparison obtained by residue R_{XY} and R_{ME}, sequence Foreman, Q_i=8

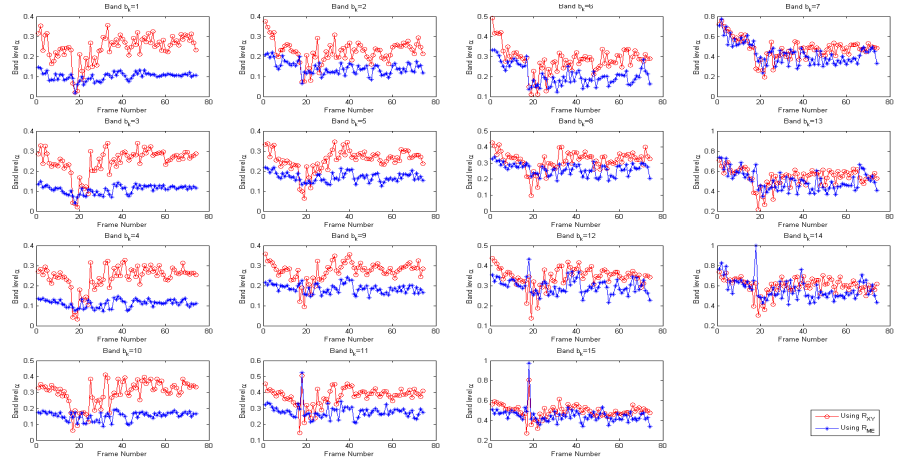
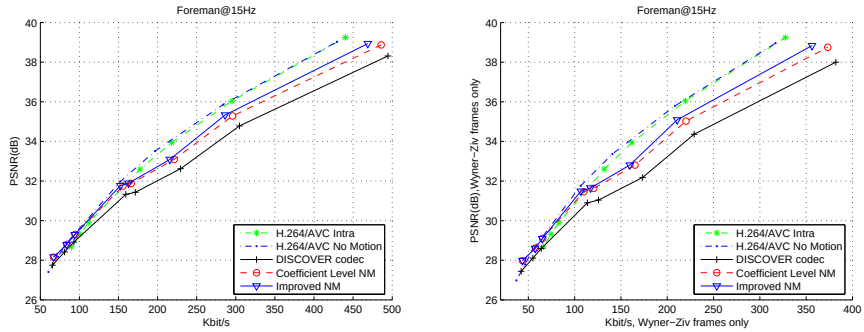


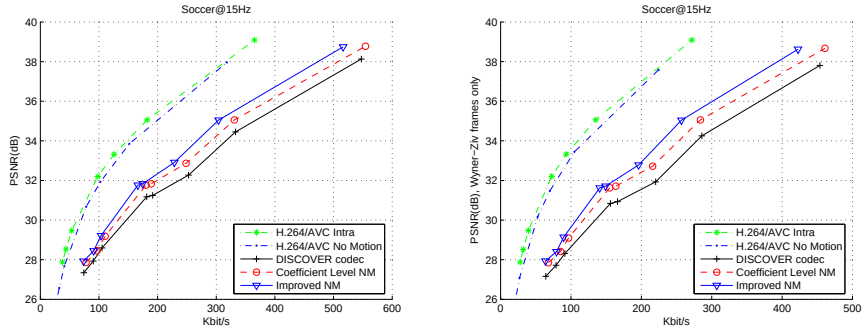
Figure C.8: Band level Laplacian parameters comparison obtained by residue R_{XY} and R_{ME}, sequence Coastguard, Q_i=8

C.4 GOP 4 RD performances comparison with different noise models



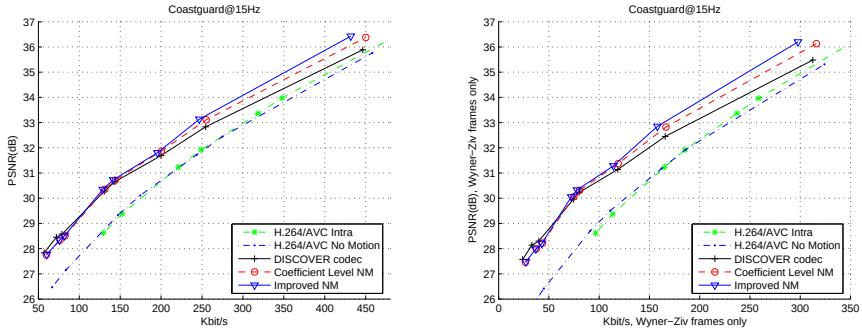
(a) (b)

Figure C.9: GOP4 RD performances comparison for sequence Foreman with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



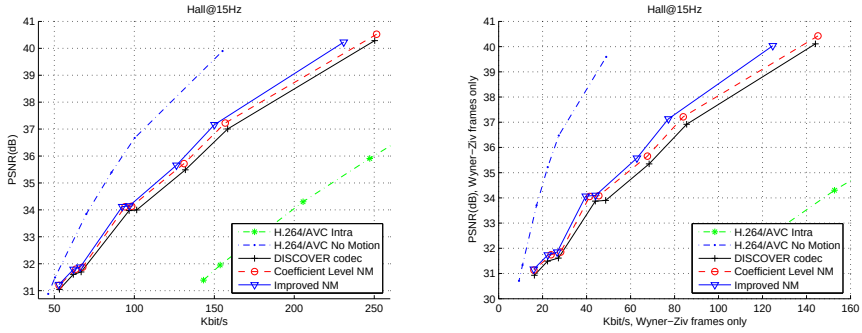
(a) (b)

Figure C.10: GOP4 RD performances comparison for sequence Soccer with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

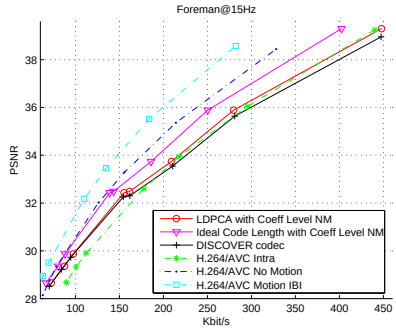
Figure C.11: GOP4 RD performances comparison for sequence Coastguard with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



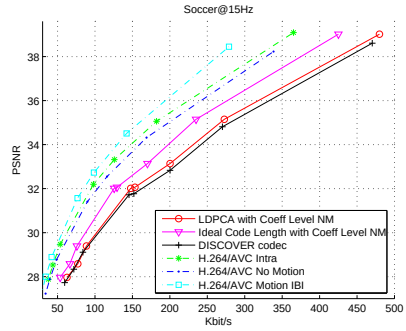
(a) (b)

Figure C.12: GOP4 RD performances comparison for sequence Hall monitor with different noise models (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

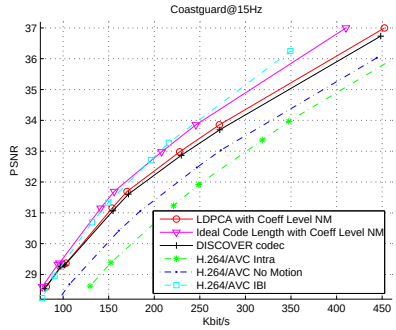
C.5 GOP2 RD performance comparison with LDPCA coding and Ideal Code Length, coefficient level noise model



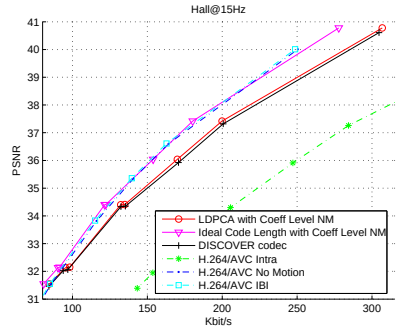
(a) Foreman



(b) Soccer



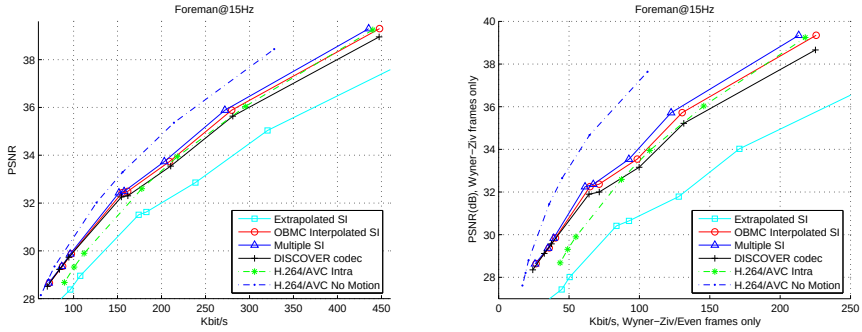
(a) Coastguard



(b) hall

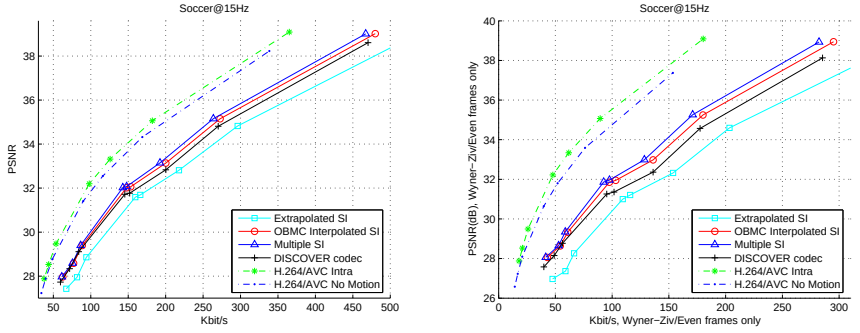
Figure C.13: RD performance comparison with LDPCA coding and Ideal Code Length, Coefficient level noise model

C.6 GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder with coefficient level noise model



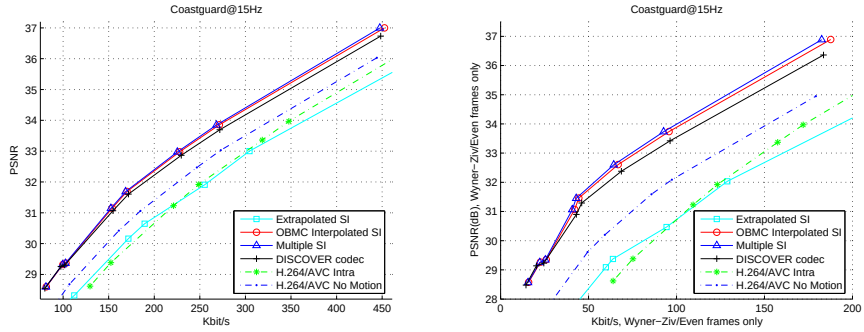
(a) (b)

Figure C.14: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, coefficient level noise model, sequence Foreman@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



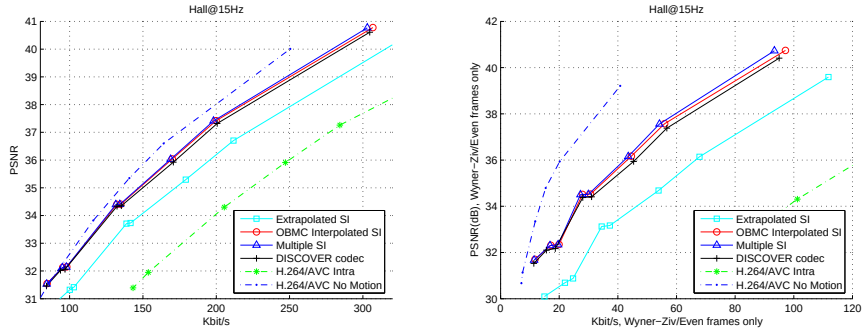
(a) (b)

Figure C.15: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, coefficient level noise model, sequence Soccer@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

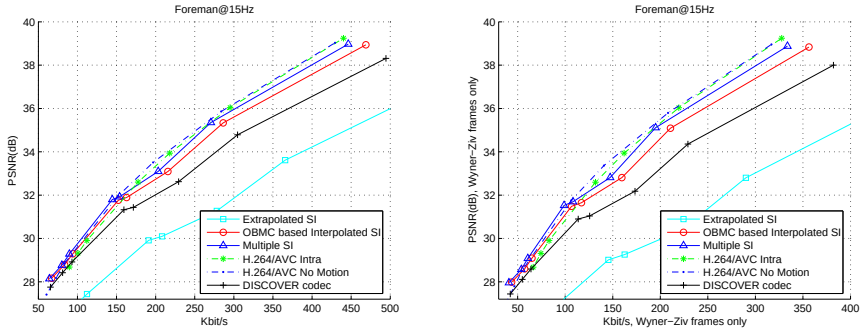
Figure C.16: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, coefficient level noise model, sequence Coastguard@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

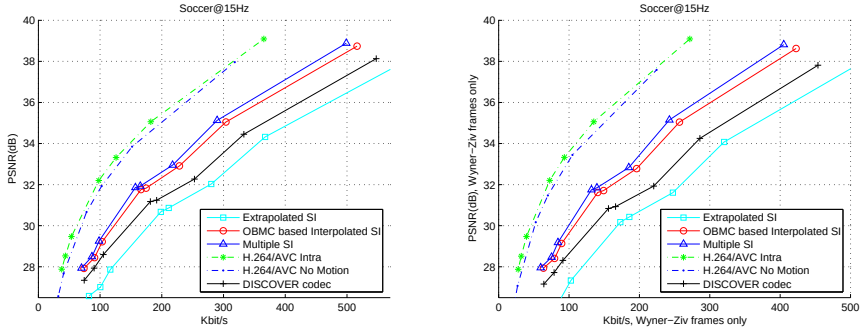
Figure C.17: GOP 2 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, coefficient level noise model, sequence Hall monitor@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

C.7 GOP 4 RD performance evaluation of multiple side information based Wyner-Ziv video coding



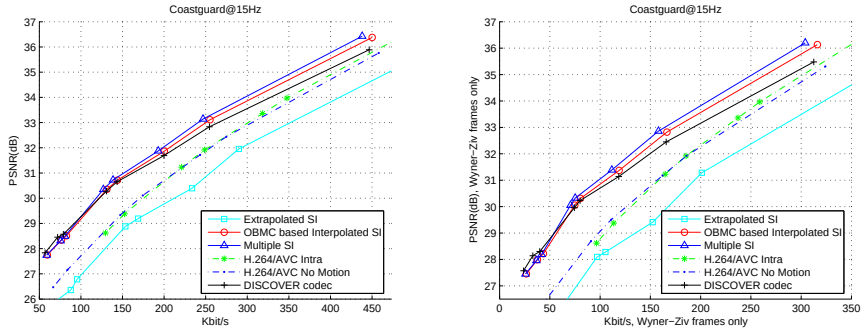
(a) (b)

Figure C.18: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Foreman@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



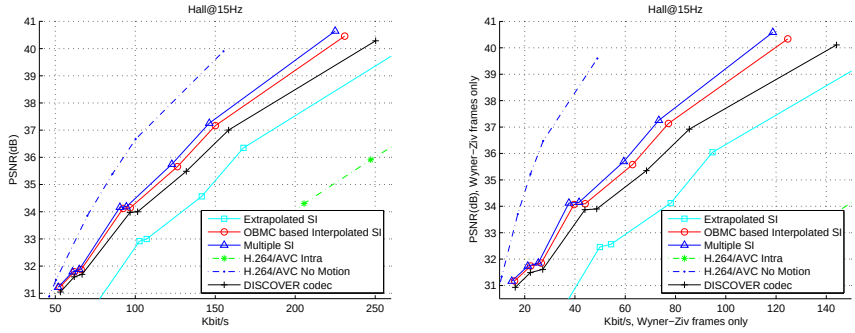
(a) (b)

Figure C.19: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Soccer@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

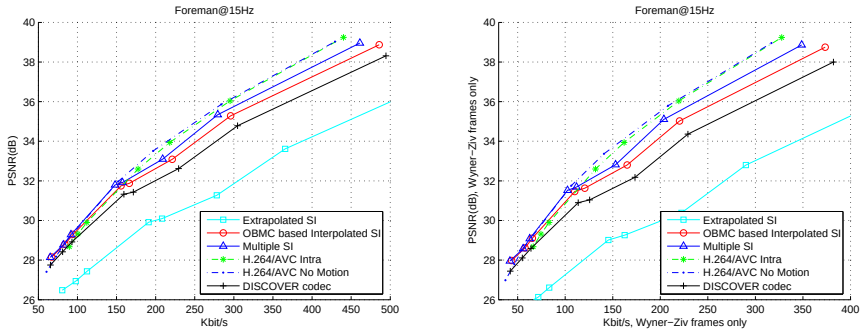
Figure C.20: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Coastguard@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

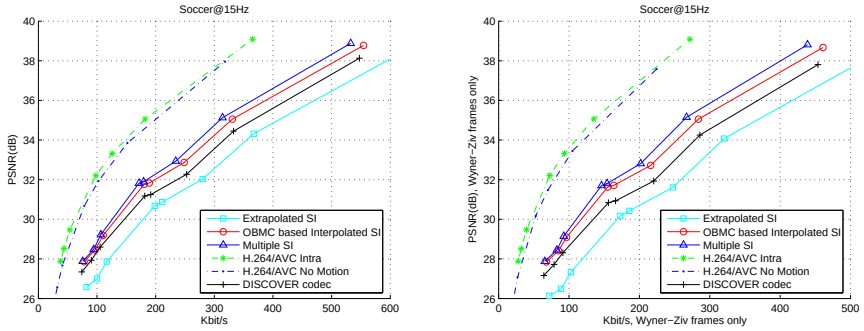
Figure C.21: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Hall monitor@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

C.8 GOP 4 RD performance evaluation of multiple side information based Wyner-Ziv video coding with coefficient level noise model



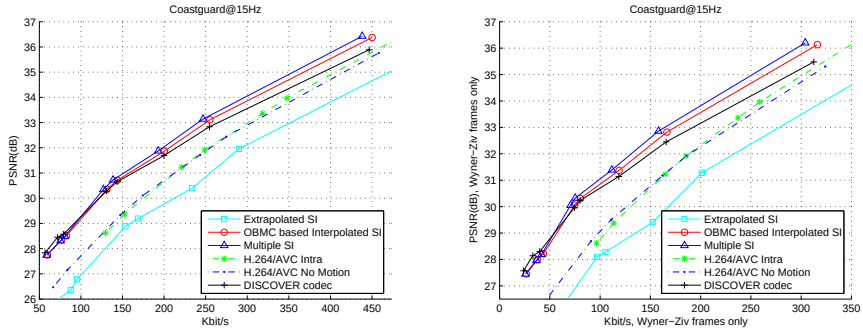
(a) (b)

Figure C.22: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Foreman@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



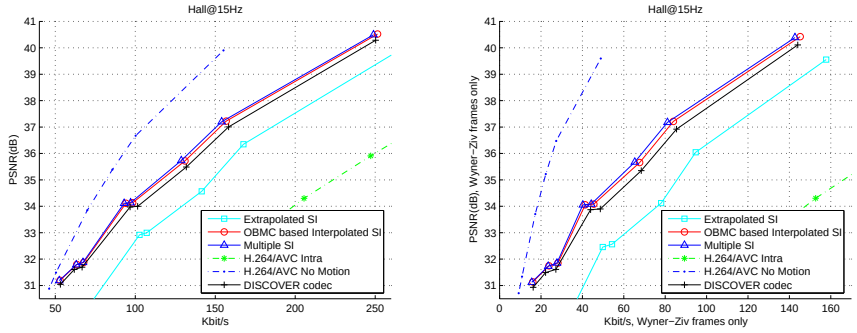
(a) (b)

Figure C.23: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Soccer@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

Figure C.24: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Coastguard@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.



(a) (b)

Figure C.25: GOP 4 RD performance evaluation of multiple SI based Wyner-Ziv video decoder, improved noise model, sequence Hall monitor@15Hz (a), Overall RD performances. (b), Only Wyner-Ziv frames for precisely the same key frames.

C.9 Visual comparison of different Wyner-Ziv codecs

(a) (b)

(c) (d)

Figure C.26: Visual comparison of different Wyner-Ziv codecs, Foreman frame No. 30 (a) Original frame (b) Wyner-Ziv coded frame with OBMC based SI, Bits=35185 PNSR=38.3317 dB (c) Wyner-Ziv coded frame with OBMC based SI and improved noise model, Bits=33431 PSNR=38.3050 dB (d)Wyner-Ziv coded frame with Multiple SI and improved noise model, Bits=29976 PSNR=38.3050 dB

(a)

(b)

(c)

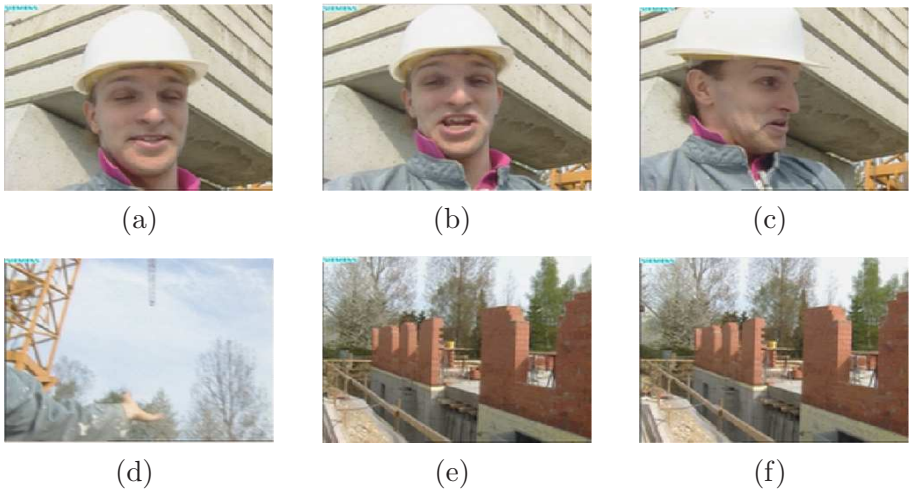
(d)

Figure C.27: Visual comparison of different Wyner-Ziv codecs, Soccer frame No. 10 (a) Original frame (b) Wyner-Ziv coded frame with OBMC based SI, Bits=43222 PNSR=38.0144 dB (c) Wyner-Ziv coded frame with OBMC based SI and improved noise model, Bits=40173 PSNR=37.9835 dB (d)Wyner-Ziv coded frame with Multiple SI and improved noise model, Bits=38014 PSNR=37.9835 dB

Appendix D

Test Material

D.1 Foreman@15Hz

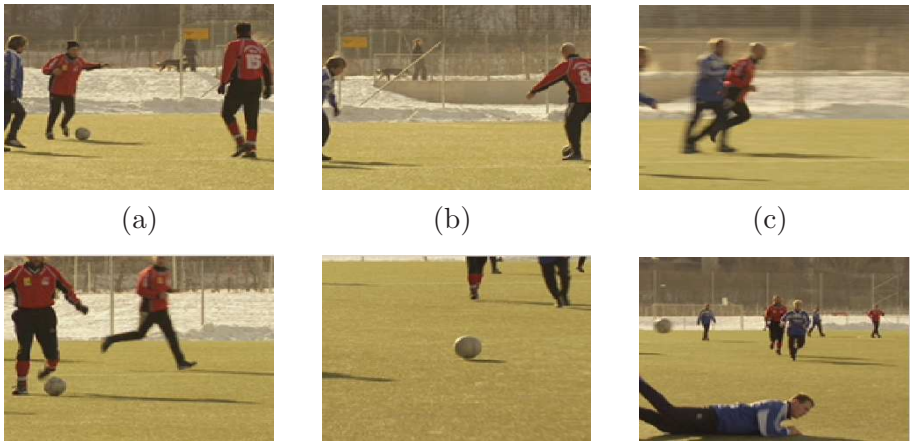


(a) (b) (c)

(d) (e) (f)

Figure D.1: Sequence Foreman@15Hz (a) frame 1 (b) frame 40 (c) frame 80 (d) frame 100 (e) frame 120 (f) frame 149

D.2 Soccer@15Hz

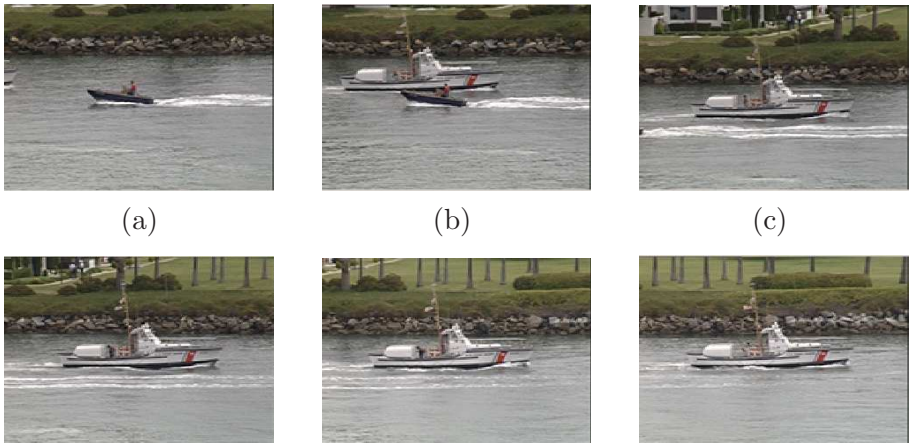


(a) (b) (c)

(d) (e) (f)

Figure D.2: Sequence Soccer@15Hz (a) frame 1 (b) frame 30 (c) frame 60 (d) frame 90 (e) frame 120 (f) frame 149

D.3 Coastguard@15Hz

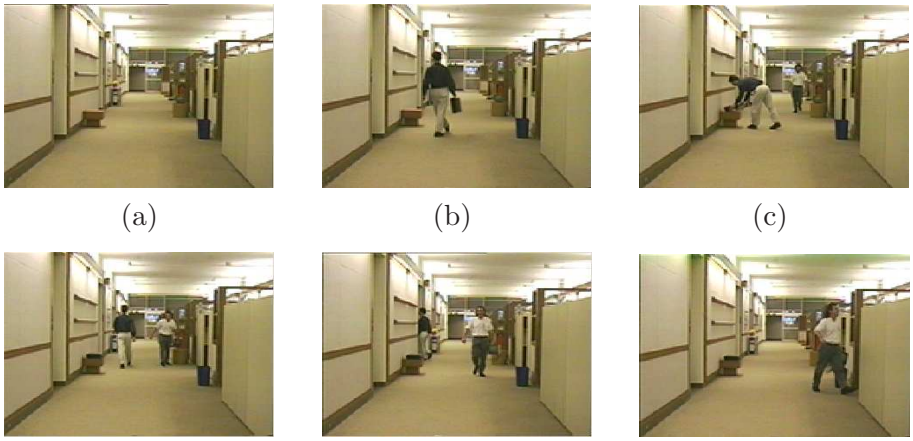


(a) (b) (c)

(d) (e) (f)

Figure D.3: Sequence Coastguard@15Hz (a) frame 1 (b) frame 30 (c) frame 60 (d) frame 90 (e) frame 120 (f) frame 149

D.4 Hall Monitor@15Hz



(a) (b) (c)

(d) (e) (f)

Figure D.4: Sequence Hall Monitor@15Hz (a) frame 1 (b) frame 30 (c) frame 60 (d) frame 90 (e) frame 120 (f) frame 149