MULTIPLE LDPC DECODING USING BITPLANE CORRELATION FOR TRANSFORM DOMAIN WYNER-ZIV VIDEO CODING

Huynh Van Luong, Xin Huang, and Søren Forchhammer

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

ABSTRACT

Distributed video coding (DVC) is an emerging video coding paradigm for systems which fully or partly exploit the source statistics at the decoder to reduce the computational burden at the encoder. This paper considers a Low Density Parity Check (LDPC) based Transform Domain Wyner-Ziv (TDWZ) video codec. To improve the LDPC coding performance in the context of TDWZ, this paper proposes a Wyner-Ziv video codec using bitplane correlation through multiple parallel LDPC decoding. The proposed scheme utilizes inter bitplane correlation to enhance the bitplane decoding performance. Experimental results show that the proposed scheme reduces the bit rate up to 3.9% and improves the rate-distortion (RD) performance of TDWZ.

Index Terms— Wyner-Ziv video coding, multiple decoders, bitplane correlation

1. INTRODUCTION

Distributed Video Coding [1][2] proposes to fully or partly exploit the video redundancy at the decoder, rather than at the encoder as in predictive video coding. 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, becoming the theoretical basis for DVC when source data are lossy coded and decoded based on a correlated source at the decoder providing the so-called side information.

Transform Domain Wyner-Ziv (TDWZ) video coding is a popular approach to DVC. This approach was first proposed in [5], and thereafter improved by e.g. advanced side information generation schemes [6]-[9], finer noise models [7][10] and refinement schemes [11][12]. Despite the advances in practical TDWZ video coding, the RD performance of TDWZ video coding still remains to reach the performance of conventional video coding, such as H.264/AVC. The coding efficiency of error correcting codes, an LDPC Accumulate (LDPCA) codec [13] in this paper, plays a key role in TDWZ in terms of overall RD performance. To improve the RD performance, a Wyner-Ziv codec with multiple LDPCA decoders is proposed in this paper. The proposed scheme is inspired by the work in [14] using joint bitplane LDPC decoding. Different from [14], the proposed Wyner-Ziv codec utilizes multiple LDPCA decoders in parallel and takes inter bitplane correlation into account during decoding, thereby improving the overall RD performance of the TDWZ codec. The modifications involve the buffer part and the decoder, while the Wyner-Ziv encoder is not changed.

The rest of the paper is organized as follows. Section 2 presents the state-of-the-art TDWZ video codec adopted in this paper. Section 3 describes the proposed Wyner-Ziv codec with multiple LDPCA decoders. Section 4 analyzes the performance of our approach and compares with other existing methods.

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

The architecture of a state-of-the-art TDWZ video codec is depicted in Fig. 1. It basically follows the same architecture as the one developed by the DISCOVER project [6]. However, a better side information generation scheme [8] and an improved noise model [10] are adopted to achieve a better RD performance.

At the encoder, 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 with low complexity such as H.264/AVC Intra, while the WZ frames in between are coded with a Wyner-Ziv approach. WZ frames are transformed using a 4x4 block size and the transformed coefficients within the same frequency band are grouped together and then quantized. DC coefficients and AC coefficients are uniformly scalar quantized and dead-zone quantized, respectively. Thereafter quantized coefficients are decomposed into bitplanes, each bitplane is fed to a rate-compatible LDPCA encoder [13] starting from the most significant bitplane (MSB) to least significant bitplane (LSB). 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 made by the decoder through a feedback channel as shown in Fig. 1.

At the decoder, a side information frame is interpolated and the corresponding noise residue is generated by using previously decoded frames. Given the available side information, soft-input information (conditional bit probabilities Pr) within each bitplane is estimated using a noise model. Thereafter the LDPCA decoder starts to decode the various bitplanes, ordered from MSB to LSB, to correct the bit errors. For each bitplane, convergence is tested by the 8-bit CRC sum and the Hamming distance between the received syndrome and the decoded bitplanes [6]. After all the bitplanes are successfully decoded, the Wyner-Ziv frame can be decoded through combined de-quantization and reconstruction followed by an inverse transform.
For the LDPCA decoding, a Belief-Propagation (BP) algorithm is used to retrieve each transmitted bitplane. The BP algorithm is a soft-decoding approach, which is passing a Log-Likelihood Ratio (LLR) of $Pr$ back and forth between source nodes and the syndrome nodes. Let $X=(b_{m-1}, ..., b_{1}, b_{0})$ denote a quantized DCT coefficient of a Wyner-Ziv frame, where $b_{m-1}$ is an MSB bit and $b_{0}$ is an LSB bit and $Y$ denotes a quantized DCT coefficient of the side information. The LLR of a bit $b_{i}$ ($0 \leq i \leq m-1$) of the $i^{th}$ significant bitplane is described as:

$$L(b_{i}) = \log \left( \frac{Pr(b_{i}=0|Y,b_{m-1},...b_{1})}{Pr(b_{i}=1|Y,b_{m-1},...b_{1})} \right) \quad (1)$$

where $b_{m-1},...,b_{i+1}$ represent bits from previous successfully decoded bits of the transformed coefficient. The LDPCA decoder utilizes information from previous successfully decoded bitplanes for decoding future bitplanes. The BP algorithm performs an approximation of the Maximum-Likelihood decoding to determine an estimate of the transmitted bits.

### 3. WYNER-ZIV CODEC WITH MULTIPLE LDPCA DECODERS

In the TDWZ codec described in Section 2, the LDPCA decoder utilizes side information, modeled noise correlation and the information from previous decoded bitplanes to decode future bitplanes. However, the inter bitplane correlation is not fully explored during decoding, although a refinement scheme is explored during decoding, although a refinement scheme is proposed in [12] to utilize the bitplane correlation to update soft-inputs. The inter bitplane correlation is not fully explored during decoding, although a refinement scheme is explored during decoding, although a refinement scheme is proposed in [12] to utilize the bitplane correlation to update soft-information. Let $\beta_{k}=Pr(b_{k}=0)$ denote a probability of bitplane $k$. Moreover, the decoding order of our approach does not consider the significance of bitplanes. The LLR described in formula (1) only uses the bits from previous successfully decoded bitplanes and decodes from MSB to LSB. Here the LLR expression is generalized for a bit $b_{i}$ of bitplane $i$ as:

$$L(b_{i}) = \log \left( \frac{Pr(b_{i}=0|Y,\beta_{m-1},...\beta_{2},\beta_{1},\beta_{0})}{Pr(b_{i}=1|Y,\beta_{m-1},...\beta_{2},\beta_{1},\beta_{0})} \right) \quad (2)$$

where $\beta_{k}$ are soft-input values for the same coefficient as $b_{k}$.

To understand the method, we should take into account both bitplane (bit) and coefficient (symbol) levels to get soft side information updated via one BP algorithm used for LDPCA decoding which is propagated to bit level and thereafter symbol level. Similar to [14], the key idea is to use the BP mechanism during the decoding of a frame and to convert the LLR back and forth between symbol level and bit level. Distinctly, in the proposed method, the soft-input is only updated after the multiple LDPCA decoders of one coefficient band are completely processed (using a certain number of iterations) at bit level based on the given syndrome bits. Let $Pr^{(t)}(b_{i}=0)$ denote the probability of bit $b_{i}$ at the iteration $t-1$ at bit level. The LLR of bit $b_{i}$ is updated at iteration $t$ as an approximation of (2):

$$L^{(t)}(b_{i}) = \log \left( \frac{\sum_{X}Pr(X|Y,b_{i}=0) \prod_{i \neq t} Pr^{(t)}(b_{i})}{\sum_{X}Pr(X|Y,b_{i}=1) \prod_{i \neq t} Pr^{(t)}(b_{i})} \right) \quad (3)$$

where $X=(b_{m-1},...,b_{1},b_{0})$ and $S$ indicates the set of values \{0,1,2,...,$2^{m-1}$\} for the coefficient X which is coded by $m$ bitplanes (for DC and the magnitude of AC coefficients). $Pr(X|Y,b_{i})$ is calculated at symbol level by using the estimated noise distribution between the side information frame and the original Wyner-Ziv frame via a noise model as shown in Fig. 2 and selecting $X$ with $b_{i}=0$ and $b_{i}=1$ in the numerator and denominator in (3), respectively.

The LLRs at iteration $t$ noted by $L^{(t)}(b_{i})$ are in turn input to multiple LDPCA decoders. After one LDPCA is processed, $L^{(t)}(b_{i})$
is temporarily achieved as output. The updated \( P_{t}(b_{i}) \) values are obtained based on LLR definition:

\[
L^{(t)}_{i}(b_{i}) = \log \left( \frac{P_{t}(b_{i} = 0)}{1 - P_{t}(b_{i} = 0)} \right)
\]

(4)

i.e. for the next iteration, we have:

\[
P_{t+1}(b_{i} = 0) = \frac{1}{2} \left( 1 + \tanh \left( \frac{L^{(t)}_{i}(b_{i})}{2} \right) \right)
\]

(5)

This \( P_{t}^{(i)}(b_{i}) \) is used as a new probability of bit \( b_{i} \) to compute new LLRs, \( L^{(t+1)}(b_{i}) \), for the next iteration of multiple LDPCA decoding based on (3).

Since all LDPCA decoders are running in parallel, once a bitplane is successfully decoded, instantaneously, the re-initialization procedure is performed. The new soft-inputs for the rest of the bitplanes are assigned conditional on the successfully decoded bitplane. The LDPCA decoder with the successfully decoded bitplane will no longer request syndromes from the buffer. Assume \( b_{i} \) is successfully decoded with value 0, then \( P_{t}^{(i)}(b_{i}=0)=1 \) and the iteration count is reset as \( t=0 \). In the remaining unfinished bitplanes are re-initialized by \( P_{t}^{(i)}(b_{i}=0)=1/2 \). The LDPCA decoders are iteratively updated up to a maximum numbers of iterations (\( T_{\text{max}} \)) with the given syndrome bits. If they are not successful after this number of iterations, the LDPCA decoders request more syndrome bits from the buffer via the feed-forward channel. Then a new process is started until all the bitplanes of the DCT coefficient are successfully decoded. Let \( N_{\text{max}} \) denote a maximum numbers of syndromes.

Overall, the multiple LDPCA decoding is handled as follows:

1. **Initiate parameters.** Iteration count \( t=0 \); Number of syndrome bits \( n=0 \); For all bits \( b_{i} \), \( P_{0}^{(i)}(b_{i}=0)=1/2 \).

2. **Increase and check conditions.**
   a. **Syndrome bit condition:** Increase \( n=n+1 \). If \( n \geq N_{\text{max}} \) then end, go to Step 2.b.
   b. **Iteration count condition:** Increase \( t=t+1 \). If \( t > T_{\text{max}} \) go to Step 3, else return to 2.a.

3. **Compute the LLRs.** At bit level, formula (3) is computed to get \( L^{(t)}_{i}(b_{i}) \), by multiplying the soft side information, \( P_{t}(X|Y,b_{i}) \) of symbol level, and the probabilities, \( P_{t}^{(i)}(b_{i}) \), of bitplane level \((k,\xi)\).

4. **Check if any LDPCA is successfully decoded?**
   a. **No:** Compute probabilities of bitplanes. \( L^{(t)}_{i}(b_{i}) \) are forwarded to multiple LDPCA decoders where \( L^{(t)}_{i}(b_{i}) \) are received from LDPCA outputs. New probabilities of bitplanes, \( P_{t}^{(i)}(b_{i}) \), are obtained by (5).
   b. **Yes:** Re-initialize the process. Assume LDPCA \( b_{i} \) is successfully decoded with value \( b_{i}=0 \), assign \( P_{t}^{(i)}(b_{i}=0)=1 \).

5. **Check all LDPCA decoders.** The process is ended if all bitplanes are successfully decoded, otherwise, go to step 2.b.

The above procedure is repeated for all bands of the DCT coefficients for which Wyner-Ziv bits are transmitted. Restarting the decoding of single LDPCA does increase complexity of the decoding.

### 4. PERFORMANCE EVALUATION

In this section, the RD performance of the proposed approach is presented and compared with the state-of-the-art TDWZ video codec described in Section 2 as well as relevant benchmarks. The test sequences are 149 frames of Foreman, Hall Monitor, Soccer, and Coast-guard with 15Hz frame rate and QCIF format. GOP (group of pictures) size is 2, where the first frame is coded as a key frame using H.246/AVC Intra and other frame is coded using Wyner-Ziv coding. Eight RD points \((Q)\) are considered corresponding to eight 4x4 quantization matrices [6]. The values within these matrices determine the number of bitplanes associated to the DCT coefficient bands, therefore, the number of LDPCA decoding instances is known. The proposed model uses \( m \) (number of bitplanes of a given band) regular LDPC accumulate decoders [13] with a length of 1584 bits for each. At these settings, exactly 1584 transform coefficients per given band of a frame can be decoded at a time by \( m \) LDPCA each decoding one bitplane.

Table 1 shows rate and PSNR values of the proposed TDWZ codec with multiple LDPCA decoders (WZMD) as well as the savings in total rate, \( \Delta R \) (in %), and WZ rate, \( \Delta R_{\text{WZ}} \) (in %), compared with the state-of-the-art TDWZ codec [10]. The WZMD achieves a reduction of bit-rate for WZ frames up to 1.8% for Foreman; 2.59% for Hall Monitor; 2.26% for Soccer; 1.82% for Coast-guard. In terms of the overall bit-rate, it saves up to 0.82% for Foreman sequence; 0.59% for Hall Monitor; 1.46% for Soccer; 0.52% for Coast-guard. It can be noted that the same PSNR values were obtained for both WZMD and TDWZ [10].

In some cases, the required number of syndromes consumed for the LSB is (close to) \( N_{\text{max}} \) even though there is still some correlation. This is due to a (relative) loss in the LDPCA decoder, which may be reduced by first coding the LSB independently and thereafter apply WZMD to the remaining bitplanes having decoded the LSB. This is called WZMD(LSB). As a result, the coding efficiency in terms of bit-rate is improved, Table 2 depicts the bit rate savings for WZMD and WZMD(LSB) compared with TDWZ [10]. The results shows that WZ rate savings up to 3.9% for Foreman and 3.77% for Soccer.

**Table 1.** Total rate and WZ rate savings (in %) for WZMD based TDWZ compared with TDWZ [10]

<table>
<thead>
<tr>
<th>Qj</th>
<th>Foreman</th>
<th>Hall</th>
<th>Soccer</th>
<th>Coast-guard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate</td>
<td>PSNR</td>
<td>$\Delta R$</td>
<td>$\Delta R_{\text{WZ}}$</td>
<td>Rate</td>
</tr>
<tr>
<td>[kbps]</td>
<td>[dB]</td>
<td>[%]</td>
<td>[%]</td>
<td>[kbps]</td>
</tr>
<tr>
<td>1</td>
<td>72.34</td>
<td>28.67</td>
<td>0.49</td>
<td>1.32</td>
</tr>
<tr>
<td>2</td>
<td>86.98</td>
<td>29.36</td>
<td>0.62</td>
<td>1.51</td>
</tr>
<tr>
<td>3</td>
<td>96.60</td>
<td>29.87</td>
<td>0.53</td>
<td>1.30</td>
</tr>
<tr>
<td>4</td>
<td>152.38</td>
<td>32.44</td>
<td>0.68</td>
<td>1.66</td>
</tr>
<tr>
<td>5</td>
<td>158.49</td>
<td>32.50</td>
<td>0.78</td>
<td>1.80</td>
</tr>
<tr>
<td>6</td>
<td>204.06</td>
<td>33.74</td>
<td>0.82</td>
<td>1.78</td>
</tr>
<tr>
<td>7</td>
<td>271.93</td>
<td>35.90</td>
<td>0.73</td>
<td>1.61</td>
</tr>
<tr>
<td>8</td>
<td>433.19</td>
<td>39.31</td>
<td>0.66</td>
<td>1.35</td>
</tr>
</tbody>
</table>
Table 2. Bit rate savings (in %) of WZMD and WZMD (LSB)

<table>
<thead>
<tr>
<th>Qj</th>
<th>Foreman</th>
<th>Soccer</th>
<th>WZMD</th>
<th>WZMD(LSB)</th>
<th>WZMD</th>
<th>WZMD(LSB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ΔR [%]</td>
<td>ΔRWz [%]</td>
<td>ΔR [%]</td>
<td>ΔRWz [%]</td>
<td>ΔR [%]</td>
<td>ΔRWz [%]</td>
</tr>
<tr>
<td>1</td>
<td>0,49</td>
<td>1,32</td>
<td>1,44</td>
<td>3,90</td>
<td>1,26</td>
<td>1,88</td>
</tr>
<tr>
<td>2</td>
<td>0,62</td>
<td>1,51</td>
<td>1,48</td>
<td>3,60</td>
<td>1,11</td>
<td>1,62</td>
</tr>
<tr>
<td>3</td>
<td>0,53</td>
<td>1,30</td>
<td>0,99</td>
<td>2,41</td>
<td>1,38</td>
<td>2,06</td>
</tr>
<tr>
<td>4</td>
<td>0,68</td>
<td>1,66</td>
<td>0,68</td>
<td>1,66</td>
<td>1,19</td>
<td>1,88</td>
</tr>
<tr>
<td>5</td>
<td>0,78</td>
<td>1,80</td>
<td>0,78</td>
<td>1,80</td>
<td>1,46</td>
<td>2,26</td>
</tr>
<tr>
<td>6</td>
<td>0,82</td>
<td>1,78</td>
<td>1,05</td>
<td>2,26</td>
<td>1,35</td>
<td>2,06</td>
</tr>
<tr>
<td>7</td>
<td>0,73</td>
<td>1,61</td>
<td>0,86</td>
<td>1,89</td>
<td>1,14</td>
<td>1,80</td>
</tr>
<tr>
<td>8</td>
<td>0,66</td>
<td>1,35</td>
<td>0,79</td>
<td>1,62</td>
<td>0,73</td>
<td>1,24</td>
</tr>
</tbody>
</table>

The experimental results in Fig. 3 demonstrate that the proposed approach significantly improves RD performance compared with the DISCOVER codec, with PSNR gains up to about 0.7 dB for Foreman and 0.9 dB for Soccer. The performance of H.264/AVC (Intra) and the H.264/AVC (No Motion) codecs are also included. The WZMD is more efficient than H.264/AVC (Intra) for Foreman. H.264/AVC (No Motion) codec is more efficient than the TDWZ codecs for both sequences since it exploits co-located frame differences at the encoder.

5. CONCLUSION

This paper proposes a Wyner-Ziv video codec using multiple parallel LDPC decoding to utilize inter bitplane correlation. The technique takes bitplane correlation into account by iteratively refining the soft-input for each bitplane during decoding. Experimental results show that the proposed multiple LDPC decoding can improve the coding efficiency of TDWZ in terms of WZ rate savings up to 3.9% compared with the existing TDWZ [10] and provide better RD performance than DISCOVER codec.

6. REFERENCES