Distributed Video Coding with Majority Rule on Multiple Sets of Side Information

Jung Ah Park, Doug Young Suh

College of Electronics and Information, Kyung Hee University, Korea
kiwilang@khu.ac.kr

Abstract

This paper proposes a method for retrieving the original information X out of multiple sets of side information (SI) such as Y₁, Y₂, Y₃ etc. These sets are generated using key frames that are encoded by a conventional video codec. By using the method of majority rule, the most reliable bits from the decoded Yᵢ’s are selected for the composition of Y, the conditional entropy H(X|Yᵢ) of which becomes much lower than any individual conditional entropy H(X|Yᵢ). The method is demonstrated using 3 typical image sequences with different characteristics.

Keywords: distributed video coding (DVC), multiple side information, majority rule, turbo code

1. Introduction

In recent years, distributed video coding (DVC) has become an increasingly popular topic of research. As proved by Pereira et al., DVC is suitable for emerging applications such as wireless video cameras and wireless low-power surveillance networks, disposable video cameras, certain medical applications, and sensor networks [1]. A new paradigm is needed for video compression for these applications in which most of the computational burden has been shifted from the encoder to the decoder. Such an approach was introduced by Slepian and Wolf [2] for lossless encoding and by Wyner and Ziv [3] for lossy cases. Wyner-Ziv video coding used the correlation between the original frame and a predicted frame at the decoder. This predicted frame, called side information (SI), uses the available information.

They used a single Y (SI) and they have tried to generate Y as close to X (the original information) as possible. Some generated an SI from multiple reference frames [4]. Adikari et al. [5][6] extended this approach to the decoding scenario, with an SI generated from multiple reference frames. This scenario was considered earlier in [7]. Adikari et al. [5][6] could improve the compression ratio by directly combining two SI streams.

Based on the estimated error probability, the Turbo decoder decides which SI stream is used for decoding a given block. However, decoder-side estimation is not accurate and the block-wise approach results in discontinuities in the reconstructed image. The turbo decoder requests more parity bits from the encoder until it receives enough parity bits to decode the original systematic bits at a predetermined error probability (10⁻³).

Another method of using multiple side information is found in [8]. This paper presents an optimal reconstruction approach, which exploits the Laplacian correlation model between the source and the side information to minimize the mean squared error (MSE) of the reconstructed sample. The decoder needs feedback from the encoder, such as the estimated bit error rate after Turbo decoding.

This paper proposes a new decoding method in which multiple sets of SI are used and a novel technique for combining them. Based on majority rule, the best segments are selected from bitstreams of multiple sets of SI.

The remainder of the paper is organized as follows. In Section 2, we briefly summarize the Slepian-Wolf theorem. Section 3 presents how to use it in deciding what are the most feasible bits of the decoded SI. Section 4 contains a detailed analysis of the proposed method’s performance. Section 5 concludes the paper with a suggestion for future research.

2. The foundation of DVC

According to the Slepian-Wolf theorem [2], the amount of information required for decoding the original information X is H(X|Y) given the SI Y in the decoder. The higher the correlation between X and Y, the less H(X|Y). In DVC, Y is modeled on the output of a BSC (Binary Symmetric Channel), while X is the input. Conditional entropy H(X|Y) is a function of bit error rate (BER) p as follows,

\[ H(X|Y) = p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p}, \quad (1) \]

By receiving parity bits with a slightly higher parity rate R than H(X|Y), as shown in Fig. 1, the decoder can retrieve X. Since Y is modeled as X corrupted by bit errors, X can be retrieved by an error-correction algorithm such as the turbo code. Instead of sending X in a WZ encoder, the encoder sends only a subset of the parity bits of X [9]. Turbo codes are mostly used as channel codes. SI Y is generated using image interpolation between key frames which are encoded by a conventional video codec.
Since X is unknown in the decoder, the interpolation is unreliable. Using various interpolation techniques, various kinds of side information can be generated. Accuracy of interpolation is dependent on interpolation techniques and the characteristics of images.

3. Multiple sets of SI

In this section, we describe a decoding scenario with more than one set of SI used to decode X. The uncertainty of X and conditional entropy was reduced greatly by using more sets of SI. This idea resembles the idea that a packet is repeatedly sent and that bit errors are corrected based on majority rule.

\[
H(X|Y_1) \geq H(X|Y_1Y_2) \geq H(X|Y_1Y_2Y_3) \geq \cdots \tag{2}
\]

The decoder using multiple sets of SI is shown in Fig. 2. Following is the process:

1. Parity bitstreams of X were generated by the Turbo encoder in the WZ encoder. They were punctured to make transmission rate R and were delivered to the decoder.
2. At the WZ decoder, multiple sets of SI Y_i were created using various prediction techniques. They were used as corrupted X in the Turbo decoder.
3. Using the received parity bits, each SI bitstream Y_i was error-corrected to result in pre-decoded X_i.
4. By applying majority rule at every bit of X_i’s at the same bit location, Y_o was resulted. In this process, if the numbers 0 and 1 are the same, a rule is needed in order to make a decision.
5. Y_o was again decoded with the received parity bitstreams and became X'.

Using this method, X can be retrieved at the expense of consuming more computational power at a lower transmission rate R. Even though R>H(X|Y_i) for every Y_i \(\in\ Y\), where Y = \{Y_1, Y_2, ..., Y_n\}, one can compose Y_o, which satisfies R>H(X|Y_o).

4. DVC with multiple sets of SI

In order to apply multiple SI sets in DVC, the decoder initially corrects errors in Y_i’s one by one using transmitted parity bits.

The method of generating the final X’ is depicted in Fig. 3. In order to reduce complexity, only the most significant coefficients were decoded, such as DC and AC coefficients near DC. The other AC coefficients were copied from the most reliable Y_i in every 8 x 8 block. By using the MSE between the decoded coefficients of X' and those of X_i, the most reliable Y_i in every block was selected. DC values were dominant.

5. Results and discussion

The proposed approach was tested with several test video sequences such as ‘Foreman,’ ‘Football,’ and ‘Mobile,’ which are all different in spatial and temporal characteristics. In the experiment, all coefficients of 8x8 blocks in the WZ frame are rearranged as bit-planes. For example, the seven MSB’s of the DC coefficients are rearranged as 7 bit-planes. A bit-plane is 396 bits-long because of one bit per each of 396 8 x 8 blocks of QCIF image.
Two parity bitstreams, one for original bitstream X and the other for the interleaved original bitstream, were calculated for every 7 x 396 bit long original bitstream using the turbo encoder. The parity bitstreams were punctured at every M bit so that R = 2 x 7 x 396/M bits were sent to the decoder. The puncturing index M was determined to ensure R was larger than H(X|Y) so that X could be decoded out of Y. As shown in Fig. 1, M was recommended to be 11 at a bit error rate of about 2.5%, at which H(X|Y) = 0.167 < R = 0.182.

As shown in Table 1, the residual BER’s of X’i with a puncturing index M of 15 was still high since the parity rate R was not large enough. However, it was larger than H(X|Y) when M was 11. By selecting the most reliable bits, the residual BER of Yx became quite low and the residual BER of X’ was close to zero. Thus, our proposed method improved error the correction capability by selecting the most reliable information.

The experiment at a BER of 3% was performed, increasing R from 10% to 40% of H(X). If only Y1 was used, 34% of H(X) should have been transmitted for perfect decoding, as shown in Fig. 4. Since M=11, the ideal amount of R in Fig. 1 at a BER of 3% was more than 16.7%. If three sets of SI, such as Y1, Y2, and Y3 were to be used, a 1.5% error would be corrected by a majority rule, and the bit rate R would be equal to 9% of H(X). This would result in perfect decoding, as shown in Fig. 4, compared to the ideal amount of 18% (= 2/11) at a BER of 3%, as shown in Fig. 1. If four sets of SI, such as Y1, Y2, Y3, and Y4 were to be used, the starting BER would become 0.6% and the required rate for successful decoding would be 18%, as in Fig. 4. The results for three and four sets of SI were not much different. If four sets of SI were to be used and 1 or 0 was chosen randomly, performance would not improve. Therefore, in order to apply the majority rule, it is effective to use an odd number of sets of SI. Rate saving gains of the proposed method were about 47% (34%=>18%, (34-18)/34) and 65% (34%=>12%, (34-12)/34) for three and five sets of SI compared to a single set of SI.

In order to investigate the performance of our proposed method, we used randomly-generated multiple sets of SI from 30 dB to 31 dB. The first 50 frames of sequences (Foreman, Football, Mobile) were used in the experiment. Their average BER was about 3% and the average transmission rate R was 2/11 of H(X) in all sequences. Figure 5 shows the PSNRs of Yx and the final X’ of the luminance component for the multiple sets of SI. Table 2 shows the average PSNRs of Fig. 5; it also shows that multiple SI sets improved compression efficiency compared to a single set of SI in all three sequences by 5.45 dB, 1.78 dB, and 6.8 dB, respectively. The ‘Football’ sequence performed the worst because it contained more unpredictable motion than the ‘Foreman’ and ‘Mobile’ sequences. The ‘Football’ SI was differed greatly from the original X than did the SI of the other two sequences.

6. Conclusion

This paper proposed a method for improving DVC performance using multiple sets of SI. It revealed there is a trade-off between the coding efficiency and complexity of the decoder. Even though generation of more sets of SI increased computational complexity in the decoder, it reduced the uncertainty of the original information X and, as a result, the transmission bitrate R.

Multiple sets of SI could be generated based on various kinds of image interpolation techniques. In conventional video codecs, multiple reference frames are used for motion estimation and the closest block in the multiple frames is selected for each block of the image frame to be encoded.

Similarly, instead of calculating the SAD (sum of absolute differences), we selected bits to be decoded out of multiple bitstreams by using a majority rule. The bits selected using a majority rule were composed to be the ultimate SI Yx, for which the conditional entropy H(X|Yx) was much less than any H(X|Yi), where Yi is a single set of SI.

Experimental simulations showed significant improvement in the PSNR (more than 1dB) compared to the conventional DVC using a single set of SI. If the resulting quality were set to be the same, the proposed method would reduce the transmission rate by 65%. In the future, we will focus on developing better rules for the selection.
Fig. 5. Performance comparison of the proposed technique

Table 2: Performance comparison of the proposed technique

<table>
<thead>
<tr>
<th>average</th>
<th>Ys</th>
<th>Ym</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC</td>
<td>30.05</td>
<td>30.84</td>
<td></td>
</tr>
<tr>
<td>DC–AC1</td>
<td>29.99</td>
<td>31.06</td>
<td></td>
</tr>
<tr>
<td>DC–AC2</td>
<td>29.11</td>
<td>31.21</td>
<td></td>
</tr>
<tr>
<td>DC–AC3</td>
<td>29.13</td>
<td>31.54</td>
<td></td>
</tr>
<tr>
<td>DC–AC4</td>
<td>29.12</td>
<td>31.81</td>
<td></td>
</tr>
</tbody>
</table>

36.95 (+5.45)

<table>
<thead>
<tr>
<th>average</th>
<th>Xs</th>
<th>Ys</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC</td>
<td>30.20</td>
<td>30.10</td>
<td></td>
</tr>
<tr>
<td>DC–AC1</td>
<td>30.21</td>
<td>31.13</td>
<td></td>
</tr>
<tr>
<td>DC–AC2</td>
<td>30.22</td>
<td>31.22</td>
<td></td>
</tr>
<tr>
<td>DC–AC3</td>
<td>30.23</td>
<td>31.27</td>
<td></td>
</tr>
<tr>
<td>DC–AC4</td>
<td>30.22</td>
<td>31.28</td>
<td></td>
</tr>
</tbody>
</table>

32.28 (+1.78)

<table>
<thead>
<tr>
<th>average</th>
<th>Xs</th>
<th>Ys</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC</td>
<td>29.75</td>
<td>30.50</td>
<td></td>
</tr>
<tr>
<td>DC–AC1</td>
<td>29.71</td>
<td>30.63</td>
<td></td>
</tr>
<tr>
<td>DC–AC2</td>
<td>29.67</td>
<td>30.79</td>
<td></td>
</tr>
<tr>
<td>DC–AC3</td>
<td>29.66</td>
<td>30.95</td>
<td></td>
</tr>
<tr>
<td>DC–AC4</td>
<td>29.65</td>
<td>31.06</td>
<td></td>
</tr>
</tbody>
</table>

37.24 (+6.6)

Acknowledgements
This work was supported by the IT R&D program of MKE/IITA. [2009-F-032-01, Development of Ultra Low Complexity Video Coding Technique for Next-generation Mobile Video Service]

References