

# Hardware Architecture Design of Hybrid Distributed Video Coding with Frame Level Coding Mode Selection

Chieh-Chuan Chiu\*, Hsin-Fang Wu\*, Shao-Yi Chien\*, Chia-Han Lee†

V. Srinivasa Somayazulu‡, and Yen-Kuang Chen‡

\* Graduate Institute of Electronics Engineering and Department of Electrical Engineering  
National Taiwan University, Taipei, Taiwan

† Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan

‡ Intel Corporation, USA

Intel-NTU Connected Context Computing Center, Taipei, Taiwan

**Abstract**—Distributed video coding (DVC), a new video coding paradigm based on Slepian-Wolf and Wyner-Ziv theories, is a promising solution for implementing low-power and low-cost distributed wireless video sensors since most of the computation load is moved from the encoder to the decoder. In this paper, the hardware architecture design of an efficient distributed video coding system, hybrid DVC with frame-level coding mode selection, is proposed. With the fully block-pipelined architecture, coding mode pre-decision, and specially-designed LDPC code engine, the proposed hardware is an efficient solution for distributed video sensors with high rate-distortion performance.

## I. INTRODUCTION

Recently, developing low-power, low-cost distributed wireless video sensors for applications in sensor and machine-to-machine (M2M) networks becomes an increasing interest. In these scenarios, a large amount of sensor nodes collect video data which will be sent to the powerful aggregation node for further processing or analysis. The sensors are expected to have limited computational capability and are powered by batteries. Although the state-of-the-art H.264 video codec can achieve high coding efficiency, it is not suitable for the scenario due to its high encoding complexity.

Distributed video coding (DVC), a novel video coding scheme based on the Slepian-Wolf and Wyner-Ziv distributed source coding theories [1], is more suitable for the sensor and M2M networks since the heavy coding complexity is shifted from the encoder to the decoder. However, the coding efficiency of DVC is inferior to that of the H.264 video codec. Several research groups have tried to boost the performance of DVC systems and have proposed algorithms to improve the quality of side information [2][3] and the accuracy of correlation noise modeling [4][5]. However, DVC is still unable to compete against H.264 No Motion, which also has low encoding complexity, especially for low motion sequences.

In order to improve the performance of the DVC codec, we borrow the ideas of residual coding, skip mode, and entropy coding from traditional video codecs, and propose a hybrid DVC architecture. A frame-level coding mode selection (CMS) mechanism is also proposed to choose the optimal mode for

each frame. For the hardware design, bandwidth consumption and frame buffer can be reduced by fully block-pipelined architecture, coding mode pre-decision, and specially-designed low-density parity-check (LDPC) code engine proposed in this paper.

The rest of this paper is organized as follows. In Section II, we will give a brief introduction to our proposed hybrid DVC architecture. In Section III, we will propose a frame-level coding mode selection scheme. The details of the proposed hardware design is then presented in Section IV. Finally, we will show some experimental results in Section V and conclude this paper in Section VI.

## II. HYBRID DISTRIBUTED VIDEO CODING

In DVC, video frames are splitted into key frames (conventional intra codec) and Wyner-Ziv (WZ) frames (intra-encoded). Wyner-Ziv frames are transformed and quantized before encoded bitplane by bitplane. The Wyner-Ziv encoder generates parity bitstreams for each bitplane and only transmits some of the parity bits to the decoder. The DVC decoder takes the side information (generated using key frames) as the erroneous source and uses the received parity bits to reconstruct the Wyner-Ziv frames.

We combine three coding techniques used by the conventional video codecs, *residual coding*, *skip mode*, and *entropy coding*, into the DVC system. Details are presented in the following subsections.

### A. Residual Coding

Residual coding can greatly reduce the redundancy between frames to achieve high compression ratio for low motion sequences, while the encoding complexity is still low [6][7]. In our proposed DVC architecture, the residue between the previous key frame and the Wyner-ziv frame is coded.

The correlation noise modeling of the residual coding remains unchanged, thus we can use the same method in DVC for noise modeling [5].

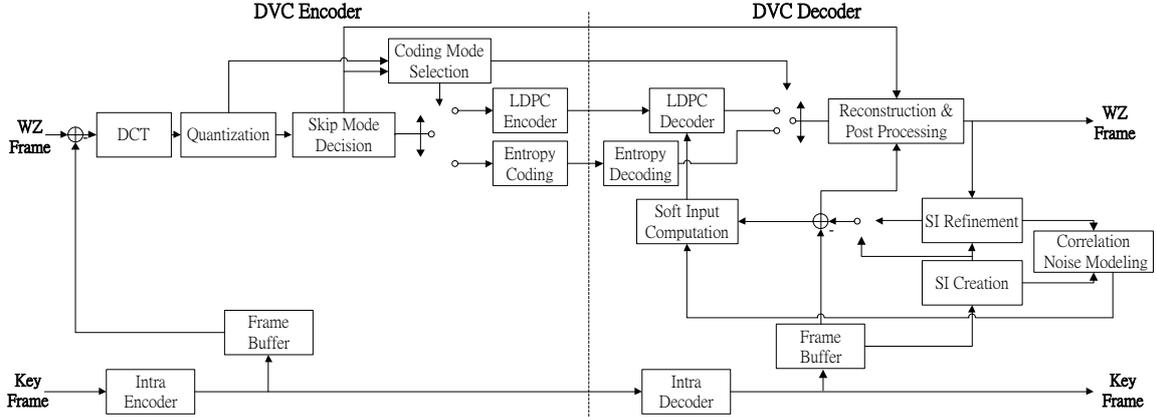


Fig. 1. The hybrid DVC framework.

The modified quantization method is designed to prevent small quantization step size. We set a minimum value of quantization step size  $Q_{i,\min}$  for different bands of different QP values. The modified quantization step size of the  $i$ th band  $Q_i$  is set as

$$Q_i = \max\left(\frac{2|\max(b_i)|}{2^M - 1}, Q_{i,\min}\right), \quad (1)$$

where  $\max(b_i)$  is the maximum value of band  $i$ ,  $M$  is the number of bitplanes, and  $|\cdot|$  denotes the absolute value.

### B. Skip Mode

The skip mode is widely used in the traditional video codec. In this work, we proposed a  $4 \times 4$  block-based skip mode to determine which blocks should be skipped. The residual block would be skipped if the distortion of the residual block is lower than the predefined threshold. The distortion function is specified as

$$Dist(n) = \sum_{u=0}^3 \sum_{v=0}^3 q_n^2(u, v), \quad (2)$$

where  $q_n(u, v)$  is the  $(u, v)$ -th coefficient of the  $n$ th transformed and quantized block of the residual frame. The skip mask is set to 1 if the distortion of block  $n$  is below the threshold  $\tau$ . The skip mask is then encoded by the traditional run length coding with the well-trained Huffman table.

### C. Entropy Coding

Although residual coding and the skip mode boost the performance of the DVC system, DVC is still unable to compete against H.264 No Motion. The reason is that the DVC architecture requires the encoder to code all bitplanes no matter how good the side information is. By using entropy coding, the encoder does not need to code excess data, thus outperforming DVC when the residues are small.

## III. FRAME LEVEL CODING MODE SELECTION

In our proposed DVC architecture, encoder can dynamically select different coding modes. The frame-level coding mode selection mechanism at the encoder is proposed to improve the

coding efficiency. We adopt CAVLC in our codec and there are four kinds of coding modes available:

- **Channel Coding:** channel coding is used to code all bands.
- **Hybrid Mode 1:** channel coding is used to code the lower three frequency bands, and the others are coded by CAVLC.
- **Hybrid Mode 2:** channel coding is used to code the lower six frequency bands, and the others are coded by CAVLC.
- **Entropy Coding:** CAVLC is used to code all bands.

With the help of these modes, we can handle different kinds of sequences. For low motion frames, the encoder tends to choose entropy coding; for high motion frames, channel coding is used. Mode decision then becomes a critical issue and a frame-level coding mode selection criterion is proposed for automatic mode selection.

The coding mode selection works as follows. First the encoder calculates the average amplitude of each band,  $b_{\text{avg}}$ . Then we rearrange  $b_{\text{avg}}$  in the zigzag scan order and cluster them into three groups. The first group consists of the lowest three frequency bands, the second group consists of the next three low frequency bands, and the other bands are grouped into the third group. The coding mode is selected according to the energy of each group and the threshold of each band.

## IV. HARDWARE ARCHITECTURE DESIGN

In this section, the hardware architecture design for the proposed DVC encoder is presented. With some algorithm modification, it can be mapped to a fully block-pipelined architecture, which can achieve high throughput, low on-chip memory, and low off-chip memory bandwidth requirements.

### A. Residual Coding and Integer Transform

The input data of this module are the block data of the key frame and the Wyner-Ziv frame. The input SRAM here is designed as a ping-pong buffer (one for reading a new block row and the other is accessed by this module), which means there are two 4-line buffers for each key frame and Wyner-Ziv frame. The residual coding module would read data from the input SRAM first (four pixel per read), and then

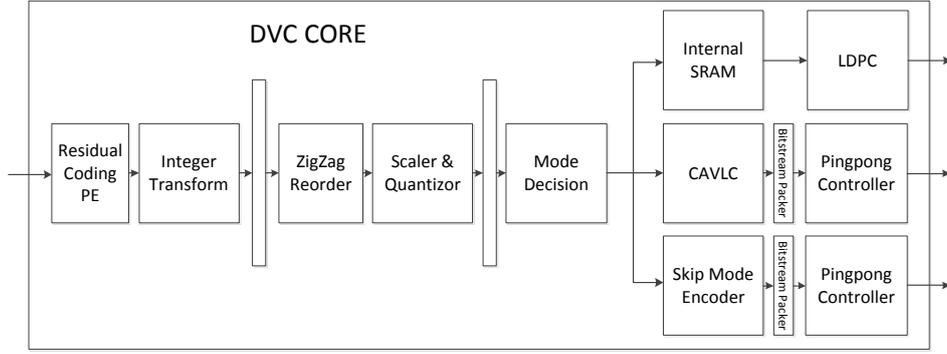


Fig. 2. The hardware architecture design of proposed encoder.

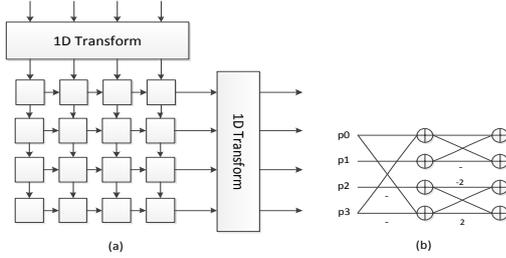


Fig. 3. The block diagram of integer transform.

the corresponding residues for a  $4 \times 4$  block are calculated and output to the next module. The hardware design of integer transform is shown in Fig. 3, where the residues are processed by two one-dimensional transform modules, which are designed with a butterfly structure. 16 register buffers are involved to transpose the data.

### B. Scaler and Quantization

In the original scheme, the transformed coefficients of the whole frame are stored into the external memory and the quantization step sizes are then calculated accordingly. To remove this data dependency which leads to a frame-level data access, we predefine the quantization step size for each DCT band with different  $Q_i$  to avoid halting the encoding process until quantization step size is ready, and hence the bandwidth is reduced. Because both entropy coding and channel coding operate in zigzag order, the input transformed coefficients are first buffered and reordered. In this stage, the scaling operation and quantization for each coefficient are also performed. Those operations are changed to shift and multiplication operations, and no divider is required in this module. Consequently, the hardware cost of this stage becomes low.

### C. Skip Mode and Frame Level Coding Mode Decision

The skip mode decision and frame level coding mode decision are merged in this module. Similarly, in order to remove the frame-level data dependency, a coding mode pre-decision technique is proposed. That is, we use the skip mode and coding mode of the previous frame to encode the current Wyner-Ziv frame. Therefore, all stages can be pipelined without any stall, and the bandwidth consumption is reduced. However, there is a tradeoff that the overall coding

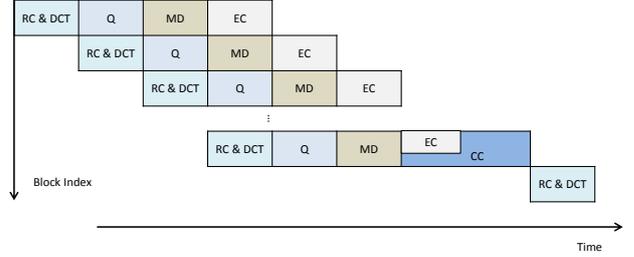


Fig. 4. The schedule of DVC encoder.

efficiency may drop due to the inaccuracy of the coding mode selection. The related analysis will be show in Section V.

### D. Channel Coding

The channel coding consumes lots of on-chip memory for hardware implementation because the channel coding needs to interleave the original data for generating the parity bits. In addition, the code length is often large. In our original design, a block-length of 6336 bits is employed to encode the bit-planes. That is, the channel coding process halts until all 6336 bits are ready, which results in a large memory requirement. To save both the bandwidth and memory size, the code length of channel coding should be shorten without dramatic performance degradation. After conducting extensive experiments, we modified the code length from 6336 to 1584 bits. Hence, the memory size is reduced by 75%. Furthermore, no data access to the external memory is required, which saves the total required bandwidth.

### E. Scheduling

With the above-mentioned techniques to reduce the hardware cost of the proposed encoder, all frame buffers are removed. As shown in Fig. 4, a block-based pipelined scheduling is used to encode each block. The channel coding starts generating parity bits right after the 58 bit-planes (1584 bits) are ready. The whole encoding process continues until all blocks are encoded.

## V. EXPERIMENTAL RESULTS

In this section, several experimental results are shown to evaluate the rate-distortion performance of the proposed hardware architecture of our hybrid DVC codec. The test conditions are set as follows:

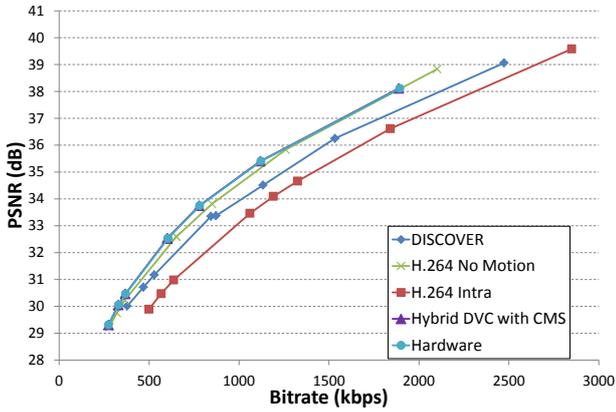


Fig. 5. R-D performance comparison for the Foreman sequence.

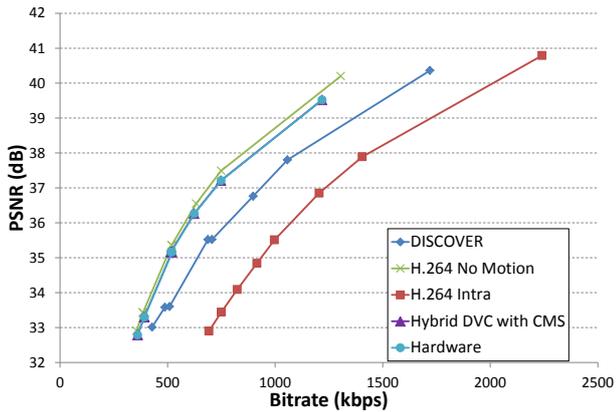


Fig. 6. R-D performance comparison for the Hall sequence.

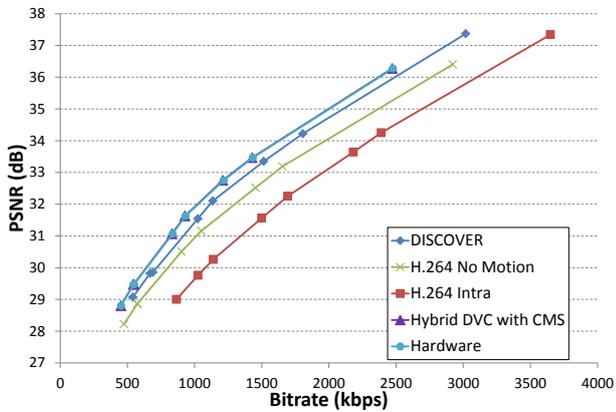


Fig. 7. R-D performance comparison for the Coastguard sequence.

- Test sequences: Foreman, Coastguard, and Hall Monitor. Each sequence has different characteristics.
- Resolution and frame rate: CIF ( $352 \times 288$ ) at 30 Hz.
- Rate-distortion points: Here we use  $Q$ -tables provided by DISCOVER [8] to quantize each DCT band.
- Benchmark codecs: H.264/AVC Intra and No Motion.
- GOP size: 2.

The R-D performance evaluation results in Fig. 5, Fig. 6, and Fig. 7 show that the hybrid DVC codec achieves higher compression ratio than the DISCOVER codec [8]—up to 2 dB

gain in PSNR is achieved for the Hall sequence and 1 dB gain for the Foreman sequence. Although the R-D performance of our DVC system is still worse than H.264 No Motion for the Foreman sequence, the hybrid DVC with the coding mode selection is as good as H.264 No Motion for the Hall sequence and is better for the Coastguard sequence. From the experimental results, the bitrate for low motion sequences is further reduced with the help of the frame-level coding mode selection. Moreover, with above-mentioned modifications, the R-D performance of hardware implementation is still equal to that of the original hybrid DVC codec.

## VI. CONCLUSION

In this work, we propose a hybrid distributed video coding (DVC) structure which significantly improves the coding efficiency. The proposed hybrid DVC system incorporates several coding techniques used in the conventional video codecs: residue coding, skip mode, and entropy coding. With the help of the frame-level coding mode selection, our codec achieves high coding efficiency and matches the performance of H.264 No Motion for low motion sequences. For hardware architecture design, we propose a simplified version of DVC encoder to reduce the total size of on-chip SRAM and required bandwidth. For quantization part, a predefined quantization level is employed instead of online quantization step updating for each frame. For the frame-level coding mode calculation, a pre-decision scheme is proposed repeating the mode of the previous frames. The original frame level pipeline structure then becomes the block pipeline structure, which can greatly increase the throughput of the DVC system.

## ACKNOWLEDGMENT

This work was supported by National Science Council, National Taiwan University and Intel Corporation under Grants NSC 100-2911-I-002-001 and 101R7501.

## REFERENCES

- [1] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.
- [2] R. Martins, C. Brites, J. Ascenso, and F. Pereira, "Refining side information for improved transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 9, pp. 1327–1341, Sept. 2009.
- [3] J. Ascenso, C. Brites, and F. Pereira, "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding," in *Proc. 5th EURASIP Conf. on Speech and Image Processing, Multimedia Communications and Services*, 2007.
- [4] G.R. Esmaili and P.C. Cosman, "Wyner-Ziv video coding with classified correlation noise estimation and key frame coding mode selection," *IEEE Transactions on Image Processing*, vol. 20, no. 9, pp. 2463–2474, Sept. 2011.
- [5] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 9, pp. 1177–1190, Sept. 2008.
- [6] A. Aaron, D. Varodayan, and B. Girod, "Wyner-Ziv residual coding of video," in *Proceedings Picture Coding Symposium*, 2006.
- [7] M. B. Badem, H. K. Arachchi, S. T. Worrall, and A. M. Kondoz, "Transform domain residual coding technique for distributed video coding," *Picture Coding Sym*, 2007.
- [8] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Proc. Picture Coding Symposium (PCS'07)*, Nov. 2007.