A Low Latency Coding Scheme for Compressing Reference Frame in Video Codec

Chun-Lung Lin

Industrial Technology Research Institute (ITRI)
Chung Hsing Rd.,Chutung
Hsingchu,31040, Taiwan
Chunlung@itri.com

Abstract

Designing ultra-low latency coding algorithms has been the key concern for the development of live video system. In the current video coding standards, the capacities of frame memory and bus bandwidth are the major aspects that affect the performance of encoder and decoder. To address this issue, this paper proposes a low-latency reference frame compression algorithm, which utilizes spatial correlation of frame to improve the throughput of frame memory. The proposed approach makes effective use of spatial correlation between pixels through special compression's order, and employs Golomb-Rice coding approach to encode the magnitude of pixel difference optimally according to its occurring frequency.

Key words: Video coding, low latency, reference frame compression, spatial correlation

Introduction

The capacities of frame memory and bus bandwidth are major aspects that significantly affect the performance of hardware for present main standard video compression. Figure 1 shows the schematic of video hardware architecture. The video processor is responsible for processing the video data, such as motion estimation, intra prediction in MPEG-4 and H.264/AVC codec. The overall performance of this multimedia hardware architecture depends on the capacity of external storage and system bus bandwidth as a result of the video processor accesses the reference frames that store in the external video frame memory through the system bus.

A valid approach to improve the system's efficacy is to embed frame memory compression (FMC) hardware unit between the video processor and the system bus as shown in Figure 1. The FMC encoder provides high efficient compression to frame data that processed by the video processor, reduces the data transfer rate, and lowers the usage of the system bus bandwidth and memory. The video processor reads the compressed reference frame data from the external video frame memory and passes them to the FMC decoder to reconstruct the original reference frame data. Even though the transmission data between the system bus and the external memory can be reduced with implementing the FMC algorithm, the latency for the encoding and the decoding process of FMC also delays the data process and transmission time. Meanwhile, the hardware complexity of the video processor increases. If the FMC lowers the latency by using lossy compression, the quality of the reference frame will degrade, and the error propagation of the distortion will significantly affect the video encoding system's performance. In other words, an efficient FMC algorithm includes four properties, low latency, random accessibility, low complexity and near lossless.

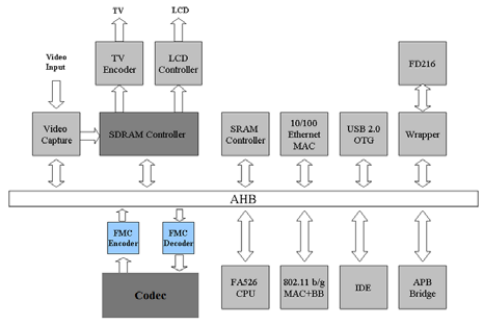

Fig. 1. An example of video processing hardware architecture with the FMC hardware units.

Many researches have been made to achieve these four purposes. Transforming pixels from spatial domain into frequency domain, and using the energy aggregation in frequency domain for efficient compression is a common technique. For example, pixels are transformed into frequency domain by hierarchically conversion, such as Haar transform, and then the encoder quantizes the frequency domain coefficients followed by variable length encoding [3]. Previous techniques also take different consideration like pixel data of greyscale [4], and compression by fixed length or variable length encoding. For compression efficiency and low complexity, another example is to utilize a modified Hadamard transform and Golomb-Rice coding. The compression ratio is fixed at 50% [6] [7]. Frame memory compression techniques for a specific purpose have been proposed. For display devices, the modified Hadamard transform followed by Golomb-Rice coding have been used to achieve memory reduction for display devices [5] [8]. For mobile video applications, the discrete cosine transform and the modified bit plane zonal coding are employed for transformation and compression [10]. However, the transform-based approach, in general, not only needs many additional computational resources but also highly complicates the hardware complexity. In addition, this approach is not suitable for the requirement of low latency.

Another approach uses the spatial domain correlation between pixels for prediction to achieve the four purposes. To achieve the low memory requirement of the reference frames in H.264/AVC, [1] [2] [13] add a decision unit to determine the storage type of each Macroblock (MB) and then compress

them according to their storage type. Furthermore, a pixel-based lossless compression is employed with an address table to achieve random accessibility [11]. To achieve low latency, variable length coding approach is also applied [12]. Utilizing some implementing techniques in the hardware design is also a popular way to reduce the size and access bandwidth for the frame memory. A lossy, 4x4 blocks compression unit is embedded between the video processor and the external storage, and then uses the information from the H.264/AVC intra prediction results with simple quantization like DPCM (Differential Pulse Code Modulation) and the variable length coding for compression [9] [14]. However, compressing these information from the H.264/AVC intra prediction results requires a lot of computational resources in the hardware architecture. The hardware complexity is much higher than the FMC algorithm. In addition, the performance is not adequate when the size of the compression block is small.

This paper proposes a low complexity, low latency and efficient FMC algorithm, which not only highly reduces the hardware requirement between the video processor and the external storage, but also improves the efficiency of accessing the system bus and the external storage. The proposed algorithm can apply to the present video compression standard, such as H.264/AVC and HEVC, to improve the performance of a video compression or decompression hardware system. At the same time, the quality of video is guaranteed, the reference frames can be restored in lossless way and the error propagation of the distortion can be prevented.

The Proposed Method

The FMC compression is a block-based compression where input of the compression unit is a block of the video. Figure 2 shows the flowchart of the proposed algorithm, including quantization, prediction, Golomb-Rice coding, padding, cutting and GOB-based rate control. The bit removed in quantization step will be stored for packing step afterward. The details will be explained later. In Golomb-Rice coding step, a modified Golomb-Rice coding has been applied for better compression efficiency in the algorithm even other variable length coding approaches are also acceptable.

Quantization quantities every pixel and removes the bits from the binary representation of pixel value. The number of the bits to be removed is determined through QP (Quantization Parameter), which is assigned by users. No bits will be removed when QP equals to 0, and 1 bit will be removed when QP equals to 1. Figure 3 shows how the quantization process removes the bits. In the binary representation of the pixel value, the order of removing bits starts from the least significant bit to the most significant. In other word, the least significant bit will be removed at first, and the most significant bit will be the last. The quantization step in the proposed algorithm will retain the removed bits, while others quantization step will drop the bits. In the packing step, if there is space left in the result of the compression, the retained bits can be used for compensation.

A new prediction structure has been proposed here, the prediction uses the boundary and the internal pixel values as shown in Figure 4. Every rectangular box in Figure 4 represents a pixel and the number in the box is the coding order. The prediction starts from the coding order 0 pixel (bottom

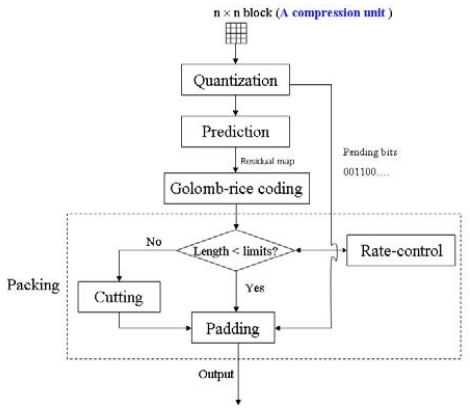

Fig. 2. A flowchart of the proposed FMC algorithm.

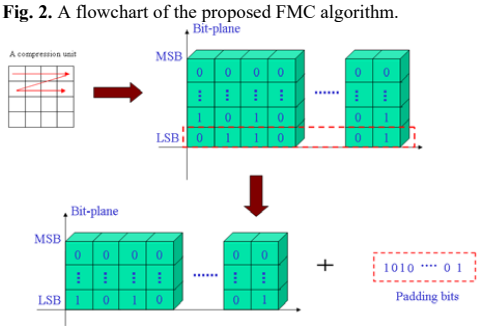

Fig. 3. An example of quantization process.

right corner pixel) which is predicted from the left adjacent pixel (L3) and the upper adjacent pixel (U3). Next, the coding order 1 pixel is predicted from the coding order 0 pixel and the upper adjacent pixel (U3). Then, the coding order 2 pixel can be predicted from the coding order 0 pixel and the coding order 1 pixel. All the predictions are performed according to the order of the prediction structure in Figure 4. The magnitude of the prediction error is residual value, and the residual values of all the pixels constitute a residual map. The residual values are small and focused on few specific values when the correlation between two pixels is higher. Due to this characteristic, the algorithm adopts the concept of Golomb-Rice coding and modifies Golomb-Rice coding for probability distribution to compress the residual values efficiently. If none of the adjacent pixel value is available, the internal encoded pixel value is the only input for the prediction as shown in Figure 5. If only one of the upper adjacent pixel or the left adjacent pixel is available, the prediction works with available adjacent pixel values (upper or left) and the internal encoded pixel value (see Figure 6).

Golomb-Rice coding is a variable length coding, it encodes the most frequent occurrence to the shortest bit stream whereas the less frequent occurrence will use longer bit stream. Therefore, the performance of Golomb-Rice coding depends on the distribution of the input data. The proposed algorithm provides four probability distributions of Golomb-Rice coding according to the statistics and each probability distribution

 ISBN 978-1-5090-4897-7

corresponding to a Golomb-Rice coding table. Each residual value in a compression unit adopts different Golomb-Rice coding table for variable length coding. Each entry of the Golomb-Rice coding table includes at least two columns which are coding value and code word. Association between the coding value and the code word is many to one, in other words, different coding value can associate with the same code word. Numbers of the entries can be reduced by replacing numerous coding values with a represented coding value. Each table includes a special entry, which code word is all 0 or 1.

Distortion is unavoidable from the quantization step for each pixel coding result, and the maximum distortion can go to 1 bit error. Since Golomb-Rice coding result is variable length, the result of the encoded bit streams might be shorter than the maximum allowed bit length. Therefore, the removed bits at the quantization step can be used as padding bits to compensate the distortion if these situations occur. The proposed FMC algorithm applies an important compensation order using the bit importance, bits will be remanded according the order until all pixels compensation done or reach the maximum length allowed.

For the situation Golomb-Rice coding resulting a bit stream exceeds the maximum allowed length for each pixel, the algorithm proposes an overflow cutting. The cutting removes the code words or the residual value's code word that exceed the limited length. Newly useable space after the cutting will be filled up by another suitable code word, which length is shorter than the useable space and the residual value error of the pixel is the lowest. Repeating the padding step until no available suitable code word or all pixels are encoded. If there is a space left when the cutting step is done, padding step will compensate the distortion with the space. At last, the remaining space will fill by all zero.

The Experimental Results

This section evaluates the performance of proposed algorithm, and the previous work [14] has been used for comparison. The two algorithms are evaluated with the settings in Table I, which are the eight common used 1080p video sequences: Pedestrian Area, Blue Sky, Riverbed, Tractor, Toys and Calendar, Rush Hour, Station2 and Sunflower. Each video sequence has different number of frames, from 217 frames to 690 frames.

Comparison of the two algorithms is done under QP=0 and QP=1. The experimental results of the two algorithms are shown in Table II. The best result is all blocks can encode with 100% of QP=0, which means all blocks are compressed lossless and meet the target bit rate. In other word, the ratio of QP=0 stands for the ratio of blocks that can be compressed lossless under the condition of the target bit rate. Higher total ratio in Table II represents more blocks can be encoded with slightly distortion (QP=1) under conforming the target bit rate. The result shows that the proposed FMC algorithm has higher ratio of blocks can be compressed lossless. Compared with the method of [14], the total ratio of the proposed algorithm is also higher, and it can always reach 94%.

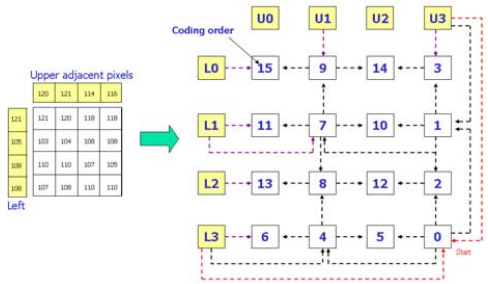

Fig. 4. An example of the proposed prediction structure.

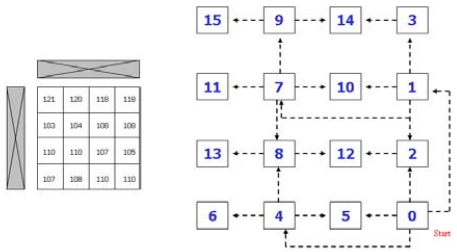

Fig. 5. An example of the proposed prediction structure without any adjacent pixel value.

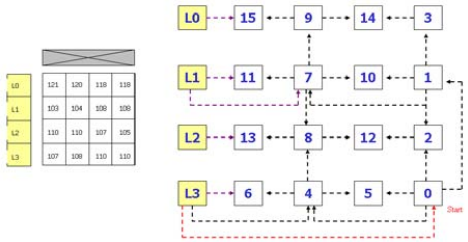

Fig. 6. An example of the proposed prediction structure when

Size of GOB is also an aspect that affects the performance of the algorithm; therefore, different size of GOB is also adopted for the experiments. The result shown in Table III indicates that 16x16 GOB is the best size for the compression efficiency of the proposed algorithm, which is to say the rate control based on GOB concept can positively improve the general compression efficiency.

Conclusion

TABLE I
EXPERIMENTAL SETTINGS

Video sequences	Resolution	Frame number
Pedestrian area	1080p	375
Blue sky	1080p	217
Riverbed	1080p	250
Tractor	1080p	690
Toys and calendar	1080p	250
Rush hour	1080p	500
Station2	1080p	313
Sunflower	1080p	500

TABLE II
THE EXPERIMENTAL RESULTS OF THE PROPOSED METHOD AND [14] UNDER QP=0 AND QP=1 TESTING CONDITIONS.

	The proposed algorithm (%)			**Ref. [14] (%)**		
Video sequences	**QP = 0**	**QP = 1**	**Total**	**QP = 0**	**QP = 1**	**Total**
Pedestrian area	80.5	15.4	95.9	68.3	26.9	95.2
Blue sky	67.6	12.4	80	66.3	21.3	87.6
Riverbed	72.2	23.7	95.9	58.6	29.4	88
Tractor	69.4	23.1	92.5	56.5	32.5	89
Toys and calendar	92.0	7.2	99.2	55.7	33.3	89
Rush hour	92	7.2	99.2	60.8	30.1	90.9
Station2	65.8	27.6	93.4	59	31.6	90.6
Sunflower	87.6	9.8	97.4	61.9	29.3	91.2

This paper proposes a low computation complexity coding scheme for the frame memory compression. The proposed FMC algorithm achieves higher compression efficiency than other previous work. By holding the removed bit at the quantization step for compensation, the distortion can be reduced to the lowest. Residual values resulted from the prediction through the internal and the boundary pixels are encoded by Golomb-Rice coding. Golomb-Rice coding can be replaced by other variable length coding. However, four Golomb-Rice coding tables designed according to the probability distribution in the proposed algorithm are provided to improve the coding efficiency. In the proposed approach, the rate-control of the encoded bit streams is implemented with the concept of GOB, to achieve higher compression efficiency goal. As a result, higher compression efficiency is achieved than with previous work [14]. The proposed algorithm will provide better performance with 16x16 GOB.

TABLE III
THE EXPERIMENTAL RESULTS UNDER VARIOUS GOB SIZES.

	GOB = 4 x 4			**GOB = 16 x 4**			**GOB = 16 x 16**		
Video sequences	**QP=0**	**QP=1**	**Total**	**QP=0**	**QP=1**	**Total**	**QP=0**	**QP=1**	**Total**
Pedestrian area	80.5	15.4	95.9	78.7	18.7	97.5	84.6	13.8	98.5
Blue sky	67.6	12.4	80.0	62.8	16.5	79.3	61.7	21.0	82.7
Riverbed	68.6	28.1	96.7	68.0	30.0	98.1	79.5	20.2	99.6
Tractor	72.2	23.7	95.9	70.3	27.0	97.3	79.4	19.4	98.9
Toys and calendar	69.4	23.1	92.5	68.0	25.6	93.6	70.8	25.6	96.5
Rush hour	92.0	7.2	99.2	92.8	6.9	99.7	97.1	2.8	100
Station2	65.8	27.6	93.4	60.9	34.2	95.1	66.7	30.6	97.4
Sunflower	87.6	9.8	97.4	87.4	11.0	98.4	92.4	7.0	99.5

References

[1]Oscal Tzyh-Chiang Chen et al., "Method for Reducing Buffered-Frame Memory Sizes and Accesses in a Video Codec," Jan. 2005 (US 2006/0171685 A1)

[2] Oscal Tzyh-Chiang Chen et al., "Method for Reducing Buffered-Frame Memory Sizes and Accesses in a Video Codec," Feb. 2005 (TW I277013 (2)).

[3] Faramarz Azadegan et al., "System and Method of Video Frame Memory Reduction of Video Decoders", Jan. 2000 (US 6,693,961 B1).

[4] Jeongnam Youn, "Lossy frame memory compression using intra refresh", Mar. 2008 (US 2009/0245391 A1)

[5] T. L. Bao Yng, B. g. Lee and H. Yoo, "A low complexity and lossless frame memory compression for display devices," *Consumer Electronics, IEEE Transactions on*, vol. 54, no. 3, pp. 1453-1458, August 2008.

[6] T. Y. Lee, "A new frame-recompression algorithm and its hardware design for MPEG-2 video decoders," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 6, pp. 529 - 534, June 2003.

[7] T. Y. Lee, "A new algorithm and its implementation for frame recompression," *Consumer Electronics, IEEE Transactions on*, vol. 47, no. 4, pp. 849-854, Nov 2001.

[8] T. L. B. Yng, B.-G. Lee, H. Yoo, "Low Complexity, Lossless Frame Memory Compression using Modified Hadamard Transform and Adaptive Golomb-Rice Coding," *IADIS International Conference Computer Graphics and Visualization*, pp. 89-96, 2008.

[9] Y. Lee, C.-E. Rhee, H.-J. Lee, "A New Frame Recompression Algorithm Integrated with H.264 Video Compression," *Circuits and Systems, IEEE International Symposium on*, pp.1621-1624, May 2007.

[10] Y. D. Wu, Y. Li, and C. Y. Lee, "A Novel Embedded Bandwidth-Aware Frame Compressor for Mobile Video Applications," *IEEE Intelligent Signal Processing and Communication System*, pp. 1-4, Feb. 2009.

[11] S.-H. Lee, M.-K. Chung, S.-M. Park, C.-M. Kyung, "Lossless frame memory recompression for video codec preserving random accessibility of coding unit," *Consumer Electronics, IEEE Transactions on*, vol. 55, no. 4, pp. 2105-2113, Nov. 2009

[12] S. Lee, N. Eum, M.-K. Chung, and C.-M. Kyung, "Low latency variable length coding scheme for frame memory recompression," *Multimedia and Expo, IEEE International Conference on*, pp.232-237, July 2010.

[13] C.-C. Chen, S.-S. Chen, O.T.C. Chen, C.-F. Chen, and F.-C. Chen, "Effective memory reduction scheme used for reference frames in H.264/AVC video codec," *Circuits and Systems, IEEE International Symposium on*, vol. 2, pp. 1549- 1552, Aug. 2005.

[14] Y. Jin, Y. Lee, and H.-J. Lee, "A New Frame Memory Compression Algorithm with DPCM and VLC in a 4×4 Block," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, no. 1, pp. 1-18, 2009.

 ISBN 978-1-5090-4897-7