Regional Context Model and Dynamic Huffman Binarization for Adaptive Entropy Coding of Multimedia

Jiangming Wu*, Wenrui Dai* and Hongkai Xiong*

*Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.
E-mail: robinwu1984@139.com, daiwenrui@sjtu.edu.cn, xionghongkai@sjtu.edu.cn.

***Abstract*—Classical context modeling and binarization algorithms on multimedia do not fully exploit their spatial correlations under the sequential assumption. This paper proposes a novel entropy coding scheme incorporating regional context modeling (RCM) and dynamic Huffman binarization (DHB) for multimedia. RCM evaluates the context order with the line distance in Cartesian coordinate system, so that the 2-D spatial correlations can be directly exploited. RCM predicts each symbol with the weighted estimation based on the line distance and appearing frequency of its context, which coincides with the principle for context-based prediction. On the other hand, DHB is proposed to adaptively assign the length of bins based on the estimated probability of symbols for coding. For optimal prediction and coding, frequent symbols are assigned with short bins to constrain the propagation of code redundancy. Applied into multimedia coding, extensive experimental results show that the proposed scheme achieves better coding performance than the benchmark coders.**

I. Introduction

With the increasing demands for multimedia dissemination and broadcasting, the efficient coding of multimedia, e.g. videos and images, is required. The common track of multimedia coding splits the problem into two parts [1]: modeling the statistics of the binarized source [2] and generate the code assignment of the source based on such statistics [3]. Therefore, binarization and context modeling play a significant role in improving the coding efficiency.

Classical context modeling algorithms [4], [5] construct their contexts in a sequential manner, which is typically the suffix of predicted subsequences. Among them, the state-of-the-arts are context-tree weighting (CTW [6], [7]) and prediction by partial matching (PPM [8]–[10]). CTW estimated the weighted mixture over all finite-order models to asymptotically predict with the optimal model. While PPM adaptively switched within a set of finite-order models to fit the statistics of the source. In summary, they incorporate variable-order Markov models [11] to adaptively represent the finite-order statistics. However, their performance for multimedia coding is degraded by the sequential and uni-directional modeling.

The work was supported in part by the NSFC, under grants U1201255, 61271218, and 61228101, and the High Technology Research and Development Program of China, under grants 2012AA011702 and 2011AA01A107.

As an alternative, non-sequential [12]–[15] and multi-dimensional extensions [16], [17] are adopted. [12]–[14] combined the predicted symbols with arbitrary positions to obtain the contexts for predicting. While [15] enumerated models with distinctive characterization and make an efficient estimation by mixing all these models. As a bi-dimensional extension, [16] extended CTW to predict based on contexts in two directions, but it could not be generalized to context trees with three or more dimensions. CABAC [17] proposed a specific binarization and context modeling scheme for video frames, which is prevailing in the state-of-the-art video coding scheme, e.g. H.264/AVC [18]. However, it can only construct contexts with the neighboring samples and its static binarization scheme generates and propagate additional code redundancy.

In this paper, we propose a novel regional context modeling (RCM) and dynamic Huffman binarization (DHB) scheme for adaptive entropy coding of multimedia. The contribution of this paper is twofold. Firstly, regional context modeling is proposed to exploit the spatial correlations in video frames. RCM evaluates the context order with the line distance in Cartesian coordinate system and predicts with the weighted estimation based on the line distance. These weights depend on the line distance of contexts and their appearing frequencies, which coincides with the principle for context-based prediction. The model redundancy for RCM is proven to solely depend on the line distance of its contexts, but is independent of the data size.

On the other hand, dynamic Huffman algorithm is adopted for binarization. Due to the pre-determined precision for CABAC, code redundancy is derived and propagated through the static binarization scheme. To reduce the code redundancy, DHB is proposed to adaptively assign the length of bins based on the estimated probability of symbols for coding. Consequently, frequent symbols are assigned with short bins to constrain the propagation of code redundancy.

The remainder of this paper is organised as follows. Section II proposes regional context model, including its formulation, weighted prediction, and derivation of model redundancy. Binarization with dynamic Huffman algorithm is adopted in Section III. For validation, an extensive range of experimental results for multimedia are provided in Section IV. Finally, Section V draws the conclusion.

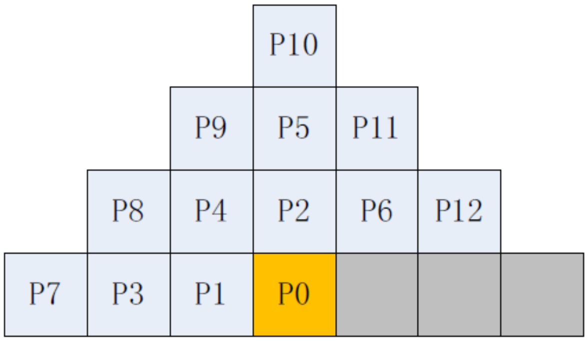

Fig. 1. Regional contexts with maximal line distance 3, where the line distances of P1-P2, P3-P6 and P7-P12 are 1, 2, and 3, respectively.

II. Regional Context Modeling

As we all know, spatial correlations come down with the growth of distances from contexts to current symbol. In classical context modeling, separable modeling or sequentialization of the morphology in local regions destruct the 2-D spatial correlations in video frames. To directly exploit these correlations, regional context modeling is proposed to generalize the order of sequential contexts to their line distances.

Definition 1. *For arbitrary symbols s_1 and s_2 with coordinates (x_1, y_1) and (x_2, y_2) in 2-D Cartesian coordinate system, their line distance $l(s_1, s_2)$ is defined as the cumulation of the distance on each dimension.*

$$l(s_1, s_2) = l_x + l_y = |x_1 - x_2| + |y_1 - y_2| \tag{1}$$

In Definition 1, symbol generally represents "bin" or "pixel" in compression. Therefore, line distances of symbols consisting in contexts are obtained by fixing s_2 with the current symbol in Eq. (1). For simplicity, we fix the current symbol at the origin of 2-D Cartesian coordinate system and select its contexts from its neighboring causal region. As shown in Fig. 1, P0 is the current symbol for prediction and P1-P12 are contexts taken in the causal neighborhood of P0. Line distances of contexts are determined with their distance to the current symbols in horizontal and vertical directions.

RCM predicts each symbol with the weighted estimation based on various contexts c_i, where the weights w_i depends on the line distance of all the symbols comprising c_i. Without loss of generality, we take it as the set of line distance for context c_i.

$$P_{RCM}(s) = \sum_i w_i(\{l_i\})\, Pr(s|c_i) \tag{2}$$

To balance the estimation led by contexts with various line distances and various frequencies of emerging combinations of symbols under the same line distance, all possible contexts are categorized into sub-contexts based on their line distances. For each kind of sub-contexts, normalized factor are assigned for the weighted estimation.

Definition 2. *Given the maximal line distance D (from the neighboring causal symbol to the current one), the normalized factor Z_i for the sub-context with the line distance i is defined as*

$$Z_i(d_i) = \begin{cases} \frac{(D+2-i)^{d_i}}{(D+3-i)^{2i}} & 1 < i < D \\ \frac{2^{d_D}}{3^{2D-1}} & i = D \end{cases} \tag{3}$$

where d_i is the count of sub-context with line distance i.

Eq. (3) implies that the normalized factor Z_i for each sub-context is affected by its line distance i and count d_i. It varies inversely with the line distance i, but increases with the growth of d_i. This observation coincides with the fact that those contexts composed of symbols appearing close to the current symbol tend to be significant in prediction.

In RCM, the line distance of contexts can range from 1 to D. Recalling Definition 1, given arbitrary line distance $i<D$, its corresponding count d_i can be in the interval $[0, 2i]$. While $d_D \in [0, 2D-1]$ for the maximal line distance D, as the last symbol with line distance D must be selected in the contexts. Taking Fig. 1 for example, P12 must be selected in the contexts when $i = 3$. Consequently, Proposition 1 proves that the normalized factor Z_i given by Definition 3 satisfies for contexts with arbitrary $i \le D$.

Proposition 1. *Given arbitrary $D>0$, $Z_i(d_i)$ given by Definition 3 is the normalized factor for RCM.*

$$\sum_{d_i} Z_i(d_i) = 1 \tag{4}$$

Proof: Firstly, considering the case $1 < i < D$, there are $C_{2i}^{d_i}$ ways to obtain line distance d_i from the maximum value $2i$. Thus, we can obtain

$$\sum_{d_i} C_{2i}^{d_i} Z_i(d_i) = \sum_{d_i=0}^{2i} C_{2i}^{d_i} \frac{(D+2-i)^{d_i}}{(D+3-i)^{2i}} = \frac{(D+2-i+1)^{2i}}{(D+3-i)^{2i}} = 1. \tag{5}$$

When $i = D$, d_D takes values in the interval $[0, 2D-1]$. Consequently, there are $C_{2D-1}^{d_D}$ ways to construct contexts with line distance d_D. It satisfies

$$\sum_{d_D} C_{2D-1}^{d_D} Z_D(d_D) = \sum_{d_D=0}^{2D-1} C_{2D-1}^{d_D} \frac{2^{d_D}}{3^{2D-1}} = 1. \tag{6}$$

As a result, $Z_i(d_i)$ normalizes the weights for arbitrary $i \le D$, which means that it is the normalized factor for RCM. ■

The normalized factors serves as weights for contexts composed of sub-contexts with various line distances and counts of symbols. The weighted probability is estimated for predicting each sub-context. Fig. 2 shows the normalized factor for each context with maximal line distance 2.

The introduction of normalized factors is due to the fact that contexts with same line distance can be composed of different symbols. Therefore, an additional model redundancy is derived to specify these contexts in prediction. Denote $(d_1, \ldots, d_D)$ the counts of symbols with line distance from 1 to D. Comparing

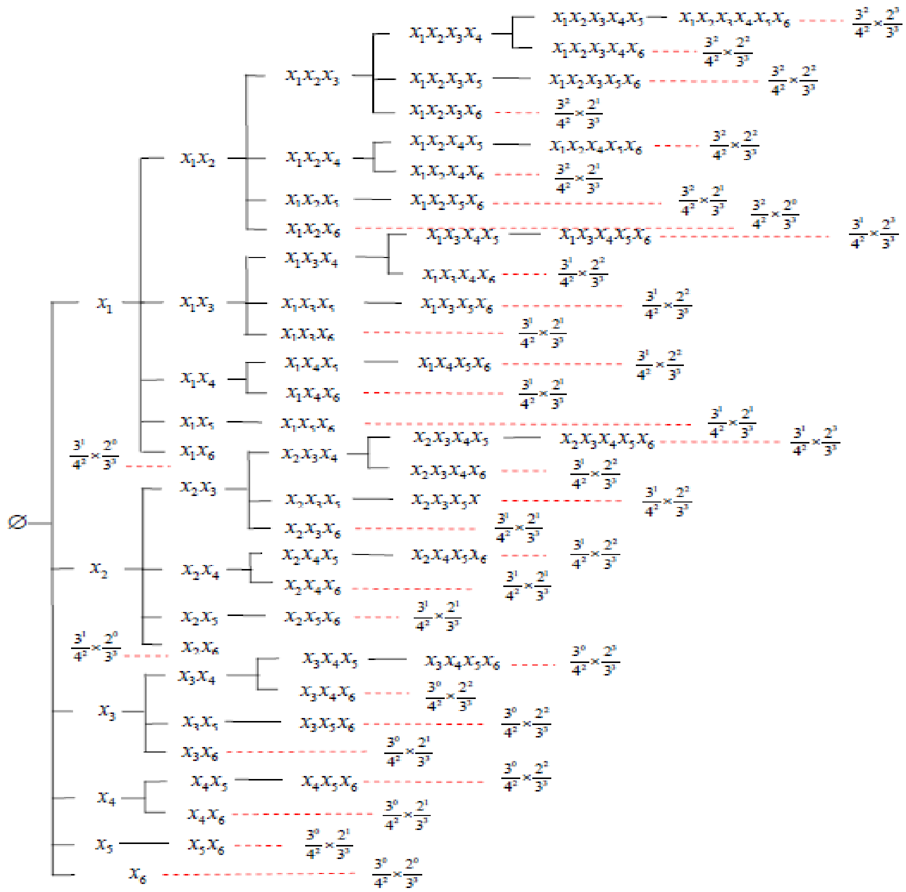

Fig. 2. The context tree with the maximal line distance is 2 and the normalized factor

weighted estimation from RCM with the one from CTW, the model redundancy ρ_{RCM} for RCM is derived.

Proposition 2. *In comparison to CTW, the additional model redundancy for RCM depends solely on its maximal line distance D. For arbitrary regional context models with counts* $(d_1, \ldots, d_D)$*, its additional model redundancy is*

$$\rho_{RCM} = -\log k_{(d_1,\ldots,d_D)}, \tag{7}$$

where $k(d_1, \ldots, d_D)$ *is the cumulative normalized factor for the actual model with counts* $(d_1, \ldots, d_D)$*.*

$$k_{(d_1 d_2 \ldots d_D)} = [\prod_{i=1}^{D-1} Z_i(d_i)] \cdot Z_D(d_D) \tag{8}$$

Proof: The additional model redundancy led by RCM is formulated as

$$\rho_{RCM}(\mathbf{x}) = \log \frac{1}{P_{RCM}(\mathbf{x})} - \log \frac{1}{P_{CTW}(\mathbf{x})}, \tag{9}$$

where $\mathbf{x}$ *is the source for prediction and* $P_{CTW}(\cdot)$ *is the prediction from CTW. Since CTW is the sequential context modeling, it is a specific case for RCM.* ρ_{RCM} *is derived as*

$$\begin{aligned}\rho_{RCM}(\mathbf{x}) &= \log \frac{P_{CTW}(\mathbf{x})}{P_{RCM}(\mathbf{x})} = \log \frac{P_{CTW}(\mathbf{x})}{\sum_i k_{(d_1 d_2 \ldots d_D)} P_i(\mathbf{x})} \\ &= \rho_{CTW} - \log k_{(d_1 d_2 \ldots d_D)} \end{aligned} \tag{10}$$

In Eq. (10)*, the derivation of* ρ_{CTW} *can be referred to [6], and* $k(d_1, \ldots, d_D)$ *is the cumulative normalized factor for the actual model with counts* $(d_1, \ldots, d_D)$*. Thus, the additional model redundancy is associated with the cumulative normalized factor for the actual model, which depends solely on D.* ■

Proposition 2 shows that the additional model redundancy for RCM depends on its line distance, but it is not the double exponential function [16] of maximal order and number of dimensions. This fact implies that the empirical entropy for coding vanishes with the growth of source size. Remarkably, RCM can be naturally extended to multi-dimensional cases by considering the components with their line distances in all the dimensions.

Proposition 3. *Given the maximal line distance D, there exist* $[\prod_{i=1}^{D-1}[\sum_{d_i=0}^{2i} C_{2i}^{d_i}]] \times [\sum_{d_D=0}^{2D-1} C_{2D-1}^{d_D}]$ *contexts with* *normalized factor* $[\prod_{i=1}^{D-1} Z_i(d_i)] \times Z_D(d_D)$.

$$\left[\prod_{i=1}^{D-1}[\sum_{d_i=0}^{2i}(C_{2i}^{d_i} Z_i(d_i))]\right] \cdot \sum_{d_D=0}^{2D-1}[C_{2D-1}^{d_D} Z_i(d_D)\Bigg] = 1 \quad (11)$$

Proof: For contexts with counts for line distance $(d_1, \cdots, d_D)$*, its corresponding normalized factors are* $Z_1(d_1), \cdots, Z_D(d_D)$*. Consequently, for* $1 \leq i < D$*, it holds*

$$\sum_{d_i=0}^{2i} C_{2i}^{d_i} Z_i(d_i) = 1. \quad (12)$$

While it holds for $i = D$

$$\sum_{d_D=0}^{2D-1} C_{2D-1} d_D Z_D(d_D) = 1. \quad (13)$$

As a conclusion, Proposition 3 is drawn by combining Eq. (12) *and* (13)*.* ■

Given the maximal line distance D, the computational complexity of RCM is $[\prod_{i=1}^{D-1}[\sum_{d_i=0}^{2i} C_{2i}^{d_i}]] \times [\sum_{d_D=0}^{2D-1} C_{2D-1}^{d_D}]$ times the one of CTW in theory.

III. Dynamic Huffman Binarization

CABAC [17] empirically predefines a specific binarization along with a context modeling scheme for video frames, which is prevailing in the state-of-the-art video coding scheme, e.g. H.264/AVC [18]. However, additional code redundancy is led by the static binarization in CABAC, especially when the probabilities of symbols are close to 0 or 1. For each most probable symbol(MPS) x, the actual interval of its estimated probability P_x is restricted to [0.01875,0.98125]. Thus, overflow tends to occur after 128 successive times of the MPS, as the minimum step for state range change is 2 and the threshold is 256. Consequently, the code redundancy ρ_c led by CABAC is derived as

$$\begin{aligned} \rho_c(MPS|P_{MPS}{>}0.98125) &\approx \tfrac{2}{256} - \log_2 \tfrac{1}{P_x} \\ &= \tfrac{1}{128} + \log_2 P_x. \end{aligned} \quad (14)$$

In the static binarization, CABAC gives each symbol several methods to generate bins, including Unary binarization, Golomb binarization, Exponential-Golomb binarization, and etc. The code redundancy rises when the number of bins in MPS increases. Therefore, Huffman code would give the lower redundancy for binarization by assigning proper MPS probability.

The proposed DHB scheme adopts dynamic Huffman algorithm to adaptively conduct binarization for each syntax element (SE). For various SEs, DHB consider the number of bins they have and estimates their probabilities. Constructing Huffman tree for these symbols, the MPS probability is estimated for the corresponding bins in the tree. Algorithm 1 and 2 describes the dynamic Huffman algorithm for binarization in the CABAC encoding and decodeing procedure, repectively. The proposed DHB scheme considers the probability distribution for the cases of double, triple, and quadruple symbols. For each bin, various numbers of bins are assigned and a Huffman tree is accordingly constructed to estimate their probabilities. The probabilities for bins are estimated and adjusted to reduce the code redundancy, as CABAC cannot deal with those bins with probabilities outside interval $[0.01875, 0.98125]$.

Algorithm 1 Dynamic Huffman algorithm for binarization in the CABAC encoding procedure

1: Initialize Bin for each SE.
2: Construct Huffman tree for current SE by considering its initial distribution.
3: For two-symbol distribution $\{x_1, x_2 | P(x_1) = P_1, P(x_2) = P_2, P_1 \leq P_2\}$, one bin is required. Its MPS probability is $P_{MPS} = \frac{P_2}{P_1+P_2}$.
4: For three-symbol distribution $\{x_1, x_2, x_3 | P(x_1) = P_1, P(x_2) = P_2, P(x_3) = P_3, P_1 \leq P_2 \leq P_3\}$, two bins are required. A three-tuple Huffman tree is constructed to estimate their MPS probability.
5: For four-symbol distribution, combine the two least frequent symbols and repeat step 4.
6: Seek the node corresponding to current symbol in binary tree.
7: Encode the bins of the node by CABAC.
8: Update the access count of the obtained node and update current tree.
9: Goto step 2 to encode next symbol.

Algorithm 2 Dynamic Huffman algorithm for binarization in the CABAC decoding procedure

1: Initialize Bin for each SE.
2: Construct Huffman tree for current SE by considering its initial distribution.
3: For two-symbol distribution $\{x_1, x_2 | P(x_1) = P_1, P(x_2) = P_2, P_1 \leq P_2\}$, one bin is required. Its MPS probability is $P_{MPS} = \frac{P_2}{P_1+P_2}$.
4: For three-symbol distribution $\{x_1, x_2, x_3 | P(x_1) = P_1, P(x_2) = P_2, P(x_3) = P_3, P_1 \leq P_2 \leq P_3\}$, two bins are required. A three-tuple Huffman tree is constructed to estimate their MPS probability.
5: For four-symbol distribution, combine the two least frequent symbols and repeat step 4.
6: Decode the bins of the node by CABAC.
7: Seek the node corresponding to current bins in binary tree.
8: Update the access count of the obtained node and update current tree.
9: Goto step 2 to decode next symbol.

If the size of the symbol set is n, the computational complexity of constructing Huffman tree for DHB is $O(n^2)$ and the average computational complexity of searching node in Huffman tree for DHB is $O(\log_2 n)$.

IV. Experimental Results

To validate the efficacy of the proposed entropy coding scheme incorporating regional context modeling and dynamic

TABLE I

IMPROVEMENT OF COMPRESSION PERFORMANCE (%) OF DHB IN COMPARISON TO CABAC FOR JAVA CLASS FILES. THE COMPRESSION RATIO IS OBTAINED BY (ORIGINAL_BYTES − COMPRESSED_BYTES)/ORIGINAL_BYTES × 100%.

Object	Utf8 length	Integer	Float	Long	Double	Class
ΔCR	0.075	9.822	12.772	2.920	6.688	0.702
Object	String	FieldRef	MethodRef	InterfaceRef	NameAndType	ConstantPool
ΔCR	0.047	0.676	0.048	0.840	0.046	27.780

TABLE II

IMPROVEMENT OF COMPRESSION PERFORMANCE (%) OF DHB IN COMPARISON TO CABAC FOR DNA SEQUENCES.

Sequence	HuRef				
Chromosome	chr1	chr2	chr3	chr4	Average
ΔCR	5.68	6.95	6.74	6.01	6.35
Sequence	FraVesHawaii 1.0				
Chromosome	chrLG1	chrLG2	chrLG3	chrLG4	Average
ΔCR	7.37	6.84	7.03	6.81	7.01

Huffman binarization, we employ it on a wide range of multimedia, e.g. video sequences, Java class files, and DNA sequences, respectively. To be concrete, DHB is compared with CABAC over Java class files and DNA sequences. While RCM is validated by comparing with the classical context modeling methods over video sequences. Finally the entropy coding scheme incorporating RCM and DHB is applied in the video coding for evaluation with CABAC. In RCM, the maximal line distance is 3. In practice, both the encoder and decoder operate on a PC with a 2.8GHz Intel Core i5 processor and complied with VC++ 10.0 with same configuration ("DEBUG" mode).

A. Class File Compression

Java class file contains a Java bytecode to be executed on the Java Virtual Machine [19], which is the prevailing platform in multimedia dissemination. DHB conducts binarization for class files according to the data contained in the constant pool. Table I shows the compression performance gain of DHB over CABAC for 11 kinds of data in the constant pool. DHB is more effective in immutable type data, e.g. Float and Integer.

B. DNA Sequence Compression

Storage and delivery of DNA sequences are widely concerned by both academic and industrial society [20]. DNA sequences are composed of repeated patterns of nucleotide symbols with the exception of insertion, deletion, and substitution. Obviously, they are not characterized with sequential statistics. In this paper, DHB is compared with CABAC over eight chromosomes: first four chromosomes of *fragaria vesca* sequence FraVesHawaii 1.0 and *homo sapiens* sequence HuRef[1], respectively. Table.II shows that DHB can improve the compression performance for DNA sequence by 6.35% and 7.01% in average, when compared with CABAC.

C. Video Sequence Compression

To evaluate RCM, we compare it with the classical context modeling methods in lossless video coding. For RCM, video frames are predicted based on weighted estimation and their prediction errors are encoded with range coder. Table III shows the compression performance of RCM, LPAQ [15], PPMZ2, and 7Zip, respectively. Among them, PPMZ2 and 7zip are the improved version of PPM algorithm with variable order Markov models. While LPAQ is the optimal lossless predictor that enumerates models with distinctive characterization and make an efficient estimation by mixing all these models. Table III implies that RCM predicts more accurately than the classical context modeling methods. The margin of gain over LPAQ is up to 4%.

Finally, we validate the proposed entropy coding scheme based on DHB and RCM over six video sequences with the YUV 4:2:0, two resolutions including CIF (352×288) and 576p (704×576). The proposed scheme is compared with CABAC, where the codec of SEs such as mb_type, $intra\_predmode$, $unary\_binarization$, $ExpGolomb$ has been modified to improve the coding performance. However, the proposed scheme is unsuitable for SE like ExpGolomb, as coding performance is degraded by excessive nodes in the Huffman tree.

Table IV shows that the proposed scheme improves the coding performance by 3% to 8% for frame-based coding. Since the reference software for CABAC resets bin state at the begin of the slice encoding, we also applied the proposed scheme into slice-based mode for H.264/AVC. There is also a gain up to 0.53% for the slice-based mode. In Table IV, R_{all} stands for the compression ratio for frame-based mode and R_{slice} for slice-based mode, respectively. At the meantime, the computational complexity is slightly increased in comparison to CABAC, which mainly lies on modeling for extensive symbols and adaptive binarization. Denote T_{enc} and T_{dec} the encoding and decoding time. Table IV suggests that the proposed scheme increases the encoding and decoding complexity by 2.1% and 3.02% in average, respectively.

V. CONCLUSION

This paper proposes a novel entropy coding scheme incorporating regional context model and dynamic Huffman

[1] DNA sequences can be downloaded from NIH website ftp://ftp.ncbi.nih.gov/genomes.

TABLE III
COMPRESSION PERFORMANCE (%) OF RCM IN COMPARISON WITH CLASSICAL CONTEXT MODELING METHODS FOR VIDEO SEQUENCES.

Sequence	Size	RCM	LPAQ	PPMZ2	7Zip
Akiyo	176×144	82.1	78.0	79.4	82.8
Foreman		53.0	50.5	48.3	45.8
Salesman		58.1	49.4	54.8	63.6
Foreman	352×288	43.9	43.0	40.0	36.9

TABLE IV
IMPROVEMENT OF COMPRESSION PERFORMANCE (%) AND INCREMENT OF COMPUTATIONAL COMPLEXITY IN COMPARISON TO CABAC FOR VIDEO SEQUENCES.

Sequence	Size	R_{all}	R_{slice}	T_{enc}	T_{dec}
City	704×576	6.38	0.08	3.52	5.07
Crew		3.00	0.53	3.83	2.05
Harbour		5.46	0.36	0.91	0.28
Soccer		5.13	0.20	1.40	0.34
Football	352×288	8.09	0.25	2.26	5.50
Foreman		3.90	0.02	0.68	4.85
Average		5.33	0.24	2.10	3.02

binarization. RCM derives the normalized factor according to line distance and count of contexts for the weighted estimation for prediction. The excess model redundancy depends solely on the maximal line distance, which vanishes with the growth of source size. DHB assigns bins with estimatd MPS to syntax elements in an adaptive manner, which is demonstrated to reduce the code redundancy led by binarization. Extensive experimental results on multimedia shows that the proposed scheme based on RCM and DHB improves the coding performance with a slight increment of computational complexity, when compared with the benchmark coders.

REFERENCES

[1] J. Rissanen and G. G. Langdon Jr, "Universal modeling and coding," *IEEE Trans. Inf. Theory*, vol. 27, no. 1, pp. 12-23, Jan. 1981.

[2] C. Bloom, "Solving the Problems of Context Modeling," *Informally Published Report*, California Institute of Technology, 1998. [Online]. Available: http://www.cbloom.com/papers

[3] A. Moffat, R. Neal, and I. Witten, "Arithmetic coding revisited," *ACM Trans. Inf. Syst.*, vol. 16, no. 3, pp. 256C294, 1998.

[4] J. Rissanen, "A universal data compression system," *IEEE Trans. Inf. Theory*, vol. 29, no. 5, pp. 656-664, Sep. 1983.

[5] M. J. Weinberger, J. Rissanen, and M. Freder, "A universal finite memory source," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 634-652, May 1995.

[6] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, "The context-tree weighting method: Basic properties," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 653-664, May 1995.

[7] A. Garivier, "Redundancy of the context-tree weighting method on renewal and Markov renewal processes", *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5579-5586, Dec. 2006.

[8] J. G. Cleary and I. H. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Trans. Commun.*, vol. 32, no. 4, pp.396-402, Apr. 1984.

[9] A. Moffat, "Implementing the PPM data compression scheme," *IEEE Trans. Commun.*, vol. 38, no. 11, pp. 1917-1921, Nov. 1990.

[10] D. Shkarin, "PPM: One step to practicality," in *Proc. Data Compression Conf.*, Snowbird, UT, USA, Mar. 2002, pp. 202-211.

[11] R. El-Yaniv, R. Begleiter, and G. Yona, "On prediction using variable order Markov models," *J. Artif. Intell. Res.*, vol. 22, pp. 385-421, Dec. 2004.

[12] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, "Context weighting for general finite-context sources," *IEEE Trans. Inf. Theory*, vol. 42, no. 5, pp. 1514-1520, Sep. 1996.

[13] X. Wu, Y. Sun, B.-L. Yeo, *et al.*, "Compression using multiple Markov chain modeling," U.S. Patent 7 319 417, Nov. 18, 2005.

[14] W. Dai, H. Xiong, and L. Song, "On non-sequential context modeling with application to executable data compression," in *Proc. Data Compression Conf.*, Snowbird, UT, USA, Mar. 2008, pp. 172-181.

[15] M. V. Mahoney, "Adaptive weighting of context models for lossless data compression," 2005. [Online]. Available: http://www.cs.fit.edu/mmahoney/compression

[16] E. Ordentlich, M. J. Weinberger, and C. Chang. "On multi-directional context sets," *IEEE Trans. Inf. Theory*. vol. 57, no. 10, pp. 6827-6836, Oct. 2011.

[17] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 620-636, Jul. 2003.

[18] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560-576, Jul. 2003.

[19] B.Venners. *Inside the Java Virtual Machine*, Second Edition. McGraw-Hill Inc., 1999.

[20] S. Kahn, "On the future of genomic data," *Science (Washington)*, vol. 331, no. 6018, pp. 728-729, Feb. 2011.