Filtered Intra Template Matching Prediction For Future Video Coding

Ramin G. Youvalari
Nokia Technologies
Tampere, Finland

Done Bugdayci Sansli
Nokia Technologies
Tampere, Finland

Pekka Astola
Nokia Technologies
Tampere, Finland

Jani Lainema
Nokia Technologies
Tampere, Finland

***Abstract*—Intra template matching (intraTMP) is a block copy method for image coding which provides measurable gains for screen content. However, the trade-off between the coding efficiency and complexity of this method for camera-captured content is less favorable compared to the screen content case. This paper proposes an enhanced intra template matching method which adapts the contents of a copied block to the local characteristics of the current block using an adaptive filter, to improve the performance of the tool for camera-captured content. Parameters of the adaptive filter are derived from the local neighborhood of the current block in both encoder and the decoder sides. Experiments illustrate that the proposed method provides 0.12% to 0.16% average luma coding gain over Enhanced Compression Model (ECM-7.0) exploration software under the common test conditions of the exploration study. Moreover, the proposed method improves the coding efficiency of the intraTMP tool in ECM-7.0 by around 50%.**

***Index Terms*—intra prediction, image compression, template matching, intra block copy, video compression, adaptive filtering**

I. Introduction

Versatile Video Coding (VVC), published in 2020, is the latest international video coding standard jointly developed by the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). In addition to providing similar video quality at a reduced bitrate objectively and subjectively over its predecessor High Efficiency Video Coding (HEVC), VVC supports efficient coding of various content types such as HDR/WCG, screen content and 360 degree videos starting from the first version of the standard [1].

Soon after the finalization of VVC, the Joint Video Experts Team (JVET) of VCEG and MPEG, has initiated an exploration activity to advance image and video compression tools beyond VVC. Promising coding tools proposed in the exploration activity are incorporated into the exploration software known as the Enhanced Compression Model (ECM) [2], which is built on top of VVC's reference software VTM [3]. As of October 2022, ECM has reached 8.22%, 16.45% and 17.55% coding gains in Y, Cb and Cr components, respectively, over VTM-11.0 in all intra configuration [4].

The newly introduced intra prediction tools are the decoder side intra mode derivation (DIMD), fusion for template based intra prediction (TIMD), intra template matching (intraTMP), convolutional cross-component model (CCCM) and gradient linear model (GLM). There are also extensions of the existing tools including but not limited to spatial geometric partitioning mode (SGPM), extended and template based multiple reference line (MRL), combined intra inter prediction (CIIP) with TIMD and TM-merge, and secondary MPM. The interested reader is encouraged to refer to [5] for a more thorough summary of the tools included in this exploration.

Intra template matching [6] is a well known intra prediction tool whose coding efficiency and complexity can be scaled based on the selected search range, search granularity, i.e. block vector resolution, and template size. ECM-7.0 implementation reportedly provides 0.26% compression gain for the luma component [7] [8]. Section II provides a detailed description of this tool. CCCM is a newly introduced tool [9], which involves deriving a 7-tap filter from the reconstructed template areas of luma and chroma components of the same block, in order to obtain a chroma prediction from the current reconstructed luma component. This highly effective tool brings 1.43% luma and 3.50% chroma gain respectively in All intra configuration.

This paper proposes an adaptive linear filter similar to CCCM filter, in order to improve the prediction performance of the intraTMP tool. The proposed filter utilizes a 6-tap linear filter model in order to adapt the texture characteristics of the reference area to the current block. Experiments conducted using ECM-7.0 software, demonstrate that the proposed filtering method enhances the coding performance of the intraTMP tool by around 50%.

The paper continues in section II with a general description of intra block copy methods and intraTMP, followed by details of the method in ECM-7.0. Section III provides the details of the proposed method and its implementation. Experimental results and related discussion is found in Section IV and section V concludes the paper.

II. Template Matching based Intra Prediction

The process of generating predictions in intra template matching tool is similar to the method used in intra block copy (IBC) [10], where a previously coded and reconstructed block from the same image is copied and used as the predictor for the current block. However, intraTMP mode differs from the traditional IBC methods in the way the block copy vector is obtained. In intraTMP method, the decoder conducts a search to derive the block vector, while in IBC, the encoder explicitly signals the block vector in the bitstream.

ISBN: 978-9-4645-9360-0

Both the encoder and decoder in intra template matching method use the same search mechanism that derives the same block vector in order to prevent error drift. The block matching does not use the current block as the decoder does not have the original for it, but instead matching is done for a predefined template area around the block which consists of already reconstructed samples.

The goal of the search is to find the block that has the minimum difference between the template regions around the current and reference blocks. The best matching block is selected by comparing the block similarity which can be measured using various similarity metrics such as sum of absolute differences (SAD) or sum of squared errors (SSE). Figure 1 illustrates the intra TMP operation.

The IBC method was adopted in the screen content extension of the High Efficiency Video Coding (HEVC) standard [11] and the first version of the VVC standard as a screen content coding tool [10]. Although block copy methods perform very well for screen content, which frequently contains exact repetitive patterns throughout the picture, these methods have less impact on coding efficiency for natural content, especially when considering the run time increase and memory access complexity of such tools.

In camera-captured content, exact repetitive patterns are less common and it is challenging to find good matches in the image even for objects and backgrounds with repetitive textures. This is because of various factors such as noise, lighting changes, reconstruction errors, texture complexity, object and camera motion. To address these issues, an adaptive filter derived from the local neighborhood samples can be applied to provide an alternative for the codec, in case one or more of the described issues are present for the current block.

The intraTMP method in ECM-7.0 enables the tool for camera-captured natural content and is chosen as a trade-off between coding gain and computational complexity. As in any other search based method, complexity of the search depends on the search area, granularity of the search step and the similarity measure. The search area is restricted to the three neighbouring coding tree units (CTUs) in addition to the CTU of the current block, which are the left, top left and top CTUs. The initial search step is two pixels and final refinement around the best block so far, is done in one pixel resolution. The intraTMP method is only applied to luma blocks in ECM. The similarity metric used is the sum of absolute differences (SAD), and the template used for determining the block vector is the L-shaped template with four pixels width and height, as illustrated in Figure 1, [5].

III. Proposed Method

It is proposed to adapt the reference block of an intraTMP method to the local signal characteristics by deriving a filter over the template regions of reference and current blocks and applying to the prediction block. It can be assumed that the template area around a block is highly correlated with the block itself. Then, an adaptive filter which captures the differences between the template regions of the reference and current block will help to adapt the prediction block from reference area to the current block.

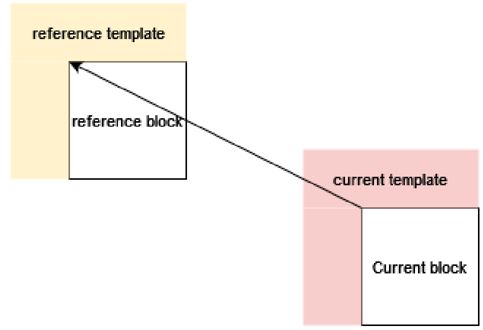

Fig. 1. Illustration of intraTMP where the block vector is decided via template cost based search

A. Adaptive Filter Generation

Given an input signal $x[n]$, a corresponding reference signal $s[n]$ and a Wiener filter $w[n]$ of order N with coefficients $\{c_0, ..., c_{N-1}\}$, filter coefficients are derived by minimizing the mean square error $e[n]$ between the filtered signal $y[n]$ and the reference signal.

$$y[n] = \sum_{i=0}^{N-1} c_n x[n-i] \quad (1)$$

$$e[n] = s[n] - y[n] \quad (2)$$

$$c_i = argmin E[e^2[n]] \quad (3)$$

For the proposed method, the input signal $x[n]$ consists of the reconstructed samples within the template of the reference block whereas the reference signal $s[n]$ consists of the reconstructed samples within the template of the current block.

The adaptive filter proposed in this study, is a six-tap linear filter composed of five spatial taps and a bias term. Spatial taps are picked from a cross-shaped pattern where the center tap is the current pixel and the following four spatial taps are the neighbouring pixels to the north (N), south (S), west (W) and east (E) of the current pixel respectively. The bias term is a fixed term which is equal to the middle value of the operating range, for example for 10-bit content the bias value is equal to 512.

The proposed filter shape is illustrated in Figure 2. The prediction samples that are directly copied from the reference block are denoted as $p(x, y)$ whereas the filtered prediction samples are denoted as $p'(x, y)$:

$$p'(x, y) = \sum_{n=0}^{5} c_n f_n \quad (4)$$

where:

$$\begin{aligned} f_0 &= p(x, y), \\ f_1 &= p(x, y-1), \\ f_2 &= p(x, y+1), \\ f_3 &= p(x-1, y), \\ f_4 &= p(x+1, y), \\ f_5 &= (1 << (bitdepth - 1)) \end{aligned} \tag{5}$$

ECM-7.0 software includes a solver designed to address linear systems of equations utilizing integer arithmetic and based on the well-known concept of LDL decomposition. This method was initially introduced through the CCCM tool [9]. In our proposed adaptive filter, we leveraged this solver for calculating the coefficients of the filter. The template area that is used in coefficient generation is the block width and height extended by four lines of samples. Figure 3 illustrates the filter parameter derivation and filter application phases of the proposed method. On the left, the reference block and its corresponding flipped L shaped template area (4 lines above and 4 lines to the left) are shown. The yellow region on the left figure shows the reference template area and 1 line padding extensions, from which the filter parameters are derived. On the right, the yellow region illustrates the reference block and its 1 line padding extensions, over which the filter is applied to generate the filtered intra template matching prediction.

This method is signaled as an extension of intraTMP method. A context coded flag is signaled in the bitstream to indicate whether the intraTMP block is coded with regular or filtered intraTMP.

B. Mode selection algorithm

The straightforward application of the proposed method for an encoder decoder pair which already utilizes an intraTMP mode, is to use result of the intraTMP search to derive and apply a filter on this particular block. Then, the encoder can test both methods and select either the filtered or unfiltered version of the intraTMP block, which can be signaled to the decoder through the use of a single bit.

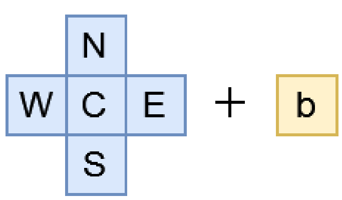

Fig. 2. Adaptive filter with five spatial taps and a bias term

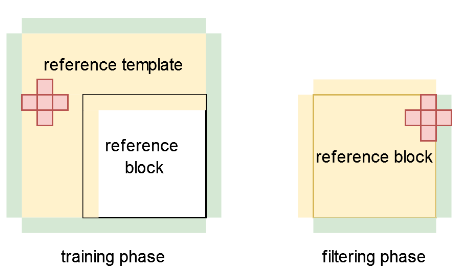

Fig. 3. Filter parameter derivation (left) and filter application (right) phases of the proposed method

Because the filtering operation changes the block statistics significantly, the optimal implementation would be to integrate the filter generation and application to the search process, where the filter is derived and applied for each location in the search route. This would divert the search of filtered intraTMP from that of regular intraTMP as the cost of each new location would be different for these two cases. This implementation would require to have two separate searches and the complexity brought by the repetitive derivation of the filter coefficients at each point on the route and filtering these blocks, would be substantial.

An alternative approach that could potentially increase the coding gain by checking more candidates with a reasonable increase in computational complexity would be to create a list of N candidate blocks that are obtained during the regular intraTMP search and derive filters for all or a subset of these blocks based on a selection criteria. In this paper, SAD cost based list is sorted in ascending order and filter parameters are derived and applied for a predetermined number of candidates. Then the final reference block is selected based on lowest filtered SAD cost. The sum of absolute transform differences (SATD) cost of both unfiltered and filtered intraTMP candidates are calculated. ECM7.0 encoder sorts the SATD cost of many different tools and tests a subset of these tool in full rate-distortion (RD) check. Based on their SATD costs, zero, one or two intraTMP based methods may be tested in full RD before the encoder makes it final decision.

IV. Experimental Results

The proposed algorithm is implemented over Enhanced Compression Model (ECM-7.0) software [2] [5] of JVET. The experiments are done using JVET common test conditions [12] with All Intra (AI) configuration. Common test conditions include a range of videos divided into separate classes based on their resolution and content type.

The performance is measured by Bjøntegaard delta bitrate metric [13] where negative numbers indicate improved performance over the anchor method by measuring the difference between the RD curves generated over four points of fixed quantization parameter (i.e., 22, 27, 32, 37) encodings.

Different tests were conducted in which various number of candidates in the list, as described before, is considered. For this, we considered number of candidates $N \in 1, 2, 4, 8, 16$ to investigate their impact for filtering intraTMP on coding efficiency.

Table I shows the impact of the proposed method on the coding efficiency in all components when using only the best candidate from the intraTMP search for the filtering process. As shown in the table, the proposed method improves the BD-Rate performance on average by 0.12% in luma component compared to ECM-7.0. Moreover, the BD-Rate impact in Cb and Cr components are 0.13% and 0.15%, respectively.

Table II illustrates the performance of the proposed method in AI configuration using different number of candidates for filter derivation and selection process. The results in the table indicate the BD-Rate numbers, which solely represent the performance of the approach for the luma component. This is due to the fact that the intraTMP technique utilized in ECM-7.0 is exclusively applied to the luma component.

The proposed method shows superior performance in terms of BD-Rate improvements over ECM-7.0 in all tested list sizes. Furthermore, the results indicate that the proposed filtering applied to the best candidate from the intraTMP search process leads to an average BD-Rate reduction of 0.12%. The performance also improves with an increase in the number of candidates in the list, as more candidate blocks are considered when generating the best filter coefficients for the current block. The proposed method performs better in sequences suitable for intraTMP coding tools, such as Class E sequences containing repeating texture throughout the frame. adapting the reference area texture to the local texture of the current block.

Although this tool is more appropriate for camera-captured content than screen content, the experiments indicate that it can also improve the screen content coding by reducing the coding artifacts in the reference area, resulting in better block prediction. As shown in the table, the proposed method provides BD-Rate gains of more than 0.30% in Class F which include mostly screen content sequences.

TABLE I
DETAILED BD-RATE (%) RESULTS OF THE PROPOSED METHOD WHEN USING 1 CANDIDATE

Class	BD-Rate (%) results over ECM-7.0		
	Y	Cb	Cr
Class A1	-0.01 %	-0.08 %	-0.04 %
Class A2	-0.17 %	-0.22 %	-0.18 %
Class B	-0.15 %	-0.10 %	-0.23 %
Class C	-0.05 %	0.00 %	-0.02 %
Class E	-0.23 %	-0.32 %	-0.25 %
Overall	**-0.12 %**	**-0.13 %**	**-0.15 %**
Class D	-0.01 %	-0.24 %	-0.02 %
Class F	-0.32 %	-0.24 %	-0.28 %
Enc. Time	100 %		
Dec. Time	100 %		

TABLE II
SUMMARY OF BD-RATE (%) RESULTS OF THE PROPOSED METHOD

Class	BD-Rate (%) results over ECM-7.0				
	N = 1	***N = 2***	***N = 4***	***N = 8***	***N = 16***
Class A1	-0.01%	-0.01%	-0.01%	-0.01%	0.01%
Class A2	-0.17%	-0.19%	-0.19%	-0.21%	-0.22%
Class B	-0.15%	-0.16%	-0.18%	-0.18%	-0.20%
Class C	-0.05%	-0.04%	-0.05%	-0.06%	-0.06%
Class E	-0.23%	-0.25%	-0.28%	-0.32%	-0.35%
Average Y	**-0.12%**	**-0.13%**	**-0.14%**	**-0.15%**	**-0.16%**
Class D	-0.01%	-0.01%	-0.02%	-0.03%	-0.03%
Class F	-0.32%	-0.29%	-0.28%	-0.35%	-0.36%
Average CbCr	-0.14%	-0.14%	-0.17%	-0.22%	-0.21%
Average YCbCr	-0.13%	-0.13%	-0.15%	-0.17%	-0.17%
Enc. Time	100%	100%	101%	104%	106%
Dec. Time	100%	100%	100%	102%	103%

V. CONCLUSION

This paper proposed an adaptive 6-tap linear filter to enhance the prediction performance of the intra template matching tool. The proposed filter aims to adapt the reference area texture to the characteristics of the texture in local area. This is realized by calculating the filter coefficients for each block coded with this mode, based on the reconstructed samples in the template areas of reference and current blocks. The results presented in this paper indicate that the proposed algorithm outperforms ECM-7.0 in all tested list sizes and achieves a high BD-Rate improvement. Furthermore, the coding efficiency of the intraTMP method in ECM-7.0 is increased by 50% due to the attained coding gain of the proposed method. In this study, the proposed method is tested in combination with intraTMP, however, the extension of the tool to other block copy methods such as IBC is expected to provide similar or even higher benefits, which is left as future work.

REFERENCES

[1] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the Versatile Video Coding (VVC) Standard and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021.

[2] "ECM-7.0 reference software." [Online]. Available: https://vcgit.hhi.fraunhofer.de/ecm/ECM.git

[3] "Versatile Video Coding (VVC) reference software." [Online]. Available: https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM

[4] V. Seregin, J. Chen, F. Le Leannec, and K. Zhang, "JVET AHG report: ECM software development (AHG6), JVET-AB0006," *Joint Video Experts Team*, October 2022.

[5] M. Coban, F. Le Léannec, R.-L. Liao, K. Naser, J. Ström, and L. Zhang, "Algorithm description of enhanced compression model 7 (ECM 7), JVET-AB2025," *Joint Video Experts Team*, October 2022.

[6] T. K. Tan, C. S. Boon, and Y. Suzuki, "Intra prediction by template matching," in *2006 International Conference on Image Processing*, 2006, pp. 1693–1696.

[7] K. Naser, T. Poirier, F. Galpin, and A. Robert, "IntraTMP adaptation for camera-captured content, JVET-AA0043," *Joint Video Experts Team*, July 2022.

[8] ——, "EE2-1.14: IntraTMP adaptation for camera-captured content, JVET-AB0130," *Joint Video Experts Team*, October 2022.

[9] P. Astola, J. Lainema, R. G. Youvalari, A. Aminlou, and K. Panusupone, "AHG12: Convolutional cross-component model (CCCM) for intra prediction, JVET-Z0064," *Joint Video Experts Team*, April 2022.

[10] T. Nguyen, X. Xu, F. Henry, R.-L. Liao, M. G. Sarwer, M. Karczewicz, Y.-H. Chao, J. Xu, S. Liu, D. Marpe *et al.*, "Overview of the screen content support in vvc: Applications, coding tools, and performance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3801–3817, 2021.

[11] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649 – 1668, 2012.

[12] M. Karczewicz and Y. Ye, "Common test conditions and evaluation procedures for enhanced compression tool testing, JVET-Y2017," *Joint Video Experts Team*, January 2022.

[13] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," *ITU SG16 Doc. VCEG-M33*, 2001.