INTER FRAME CODING WITH TEMPLATE MATCHING AVERAGING

Yoshinori Suzuki [1], Choong Seng Boon[1] and Thiow Keng Tan [2]

[1] NTT DoCoMo, Inc., [2] M-Sphere Consulting Pte. Ltd.

ABSTRACT

A template matching prediction based on a group of reconstructed pixels surrounding a target block enables prediction of pixels in the target block without motion information. The predictor of a target block is produced by minimizing the matching error of the template. Due to the freedom possessed by the template, the residuals of a target block may become large in flat regions. Our previous paper proposed to predictively encode the decimated version of a target block in flat regions to suppress the prediction errors. In this paper, the performance of template matching prediction is further improved. Multiple candidates are created by template matching at decoder. An average of the multiple candidates then forms the final predictor, which can reduce coding noise residing in the reference frames. Simulation results show that the proposed scheme improves coding efficiency of H.264 up to 7.9%.

Index Terms— Video coding, Prediction methods, Motion compensation

1. INTRODUCTION

Inter prediction, reducing temporal redundancy between a target frame and reference frame, is a key component of video coding techniques. Two groups of motion estimation methods are employed in inter frame prediction [1]. The first group of techniques uses forward motion estimation in which motion vectors are needed to be transmitted. This technique, based on block matching mechanism, is extensively adopted in video codec standards such as H.264 [2] and MPEG-4 [3]. In these standards, partitioning of a target block for inter prediction is applied to improve the prediction performance. The well-known rate-distortion optimization [4][5] based on Lagrangian optimization technique is usually used to choose the best partition block size for each target block. The second group of techniques uses a recursive (backward) motion estimation in which motion vectors are recursively estimated based on the transmitted pixels such as shown in [6][7]. In this technique, no motion vector needs to be transmitted to a decoder. However, since it assumes that image intensity is constant along the motion trajectory, the estimation problem tends to become ill-posed.

In a different way from these two groups of techniques, we have proposed Template Matching Prediction (TMP) method [8][9][10] for inter prediction, which has been applied to intra-frame coding [11]. TMP can predict pixels in a target block without transmission of motion vector, as similar to the recursive motion estimation. At the same time, it is expected that the prediction performance of TMP is comparable to that of the block matching when the correlation between a target block and its template is high. TMP determines the predicted signals of a target block at the encoder, as similar to the forward motion estimation. Additionally, in [10], the decimated version of the target block is predictively encoded to enhance the performance of TMP in flat regions. Because of a high similarity between a target block and its decimated version in flat regions, coding bits for residuals of the target block can be reduced with little loss of image quality.

The averaging of more than three candidate samples can make prediction signals of a target block smoother. This technique is applied to inter-frame coding [12][13] and intra-frame coding [14]. In [12] and [13], the advantage of increasing the number of candidate samples for averaging, which are produced by block matching, has been demonstrated. In [14] for intra frames, TMP is enhanced by multiple candidates averaging to improve and refine the predictor creation method by template matching. This paper extends our previous work done in [10] by introducing the multiple candidates averaging as similar to [14].

The rest of this paper is organized as follows. The coding scheme with TMP and decimation of a target block [10] is described in section 2. The extension of it is proposed in section 3 and its simulation results are shown in section 4, followed by conclusion in section 5.

2. TEMPLATE MATCHING PREDICTION (TMP)

This section briefly describes our previous work [10]. TMP exploits the correlation between the pixels in a target block and the reconstructed pixels surrounding the target block, on the top and to the left, which is called the template. The predictor of a target block is produced by minimizing the matching error of the template to a search region in a previously reconstructed frame.

As shown in Figure 1, using the target block on a current frame F_t , indicated by B, and a template formed by a group of pixels straddling on top and to the left of the target block, indicated by T_B, the best-matched template T_p is searched within a reconstructed frame F_{t-1} by minimizing the sum of absolute difference (SAD) between T_B and any template on

F_{t-1}. The block P adjacent to the best-matched template T_p, hereafter shown as TMP block, is assigned as the predictor of the target block B. According to this process, the TMP block P is estimated without any motion vector information since the decoder can produce the same predictor as the encoder by using the reconstructed pixels in T_B.

When the template contains features such as complex texture, an appropriate predictor can be produced using the correlation between a target block and its template. However, in flat regions, the residuals of the target block may become large as the template may be matched to an arbitrary region where the adjacent predicted block does not match well with the target block.

To suppress the increase of prediction errors in these regions, a reduced resolution measure has been adopted in the coding scheme with TMP. Concretely, two types of TMP, i.e., TMP-fEature (TMP-E) and TMP-fLat (TMP-L) are prepared. In TMP-E, residuals of a target bock are predictively encoded as conventional coding method. In TMP-L, a target block and a TMP block are decimated and residuals between these decimated blocks are encoded as shown in Figure 2. By exploiting the similarity between a target block and its decimated version, coding bits for residuals in flat regions can be reduced with a little loss of image quality. At the decoder, the reconstructed decimated block is upsampled [15] to reconstruct the target block.

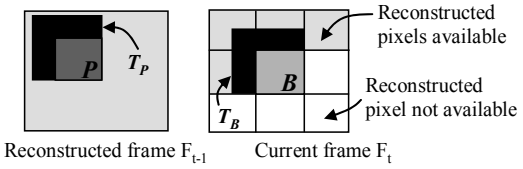

Figure 1 Template matching prediction.

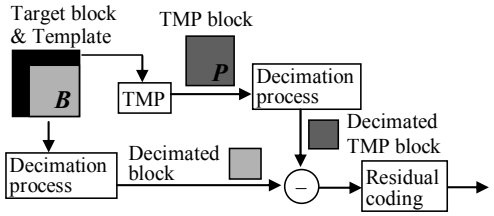

Figure 2 Reduced resolution coding for TMP-L

3. TEMPLATE MATCHING AVERAGING (TMA)

3.1. Expectation by multiple candidates averaging

We can expect that the averaging of multiple candidates makes prediction signals of a target block smoother if these candidates are similar to each other assuming that the additive noise is white noise.

Let the target block, represented by the vector $\mathbf{x}$, be estimated by the candidate blocks $\mathbf{x}_i$, i being the index of the candidate blocks which are similar each other. Assuming that each $\mathbf{x}_i$ are independent and identically distributed (i.i.d.) and have the same mean, $\boldsymbol{\mu}$, and standard deviation, $\boldsymbol{\sigma}$, where

$$\boldsymbol{\mu} = E[\mathbf{x}_i] \tag{1}$$

$$\boldsymbol{\sigma}^2 = E[\mathbf{x}_i^2] - \boldsymbol{\mu}^2 \tag{2}$$

then the central limit theorem states that the average of $\mathbf{x}_i$, denoted as

$$\mathbf{x}_{avg} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \tag{3}$$

tends towards a normal distribution with a mean, $\boldsymbol{\mu}_{avg}$, and standard deviation, $\boldsymbol{\sigma}_{avg}$ given by

$$\boldsymbol{\mu}_{avg} = E[\mathbf{x}_{avg}] = \boldsymbol{\mu} \tag{4}$$

$$\boldsymbol{\sigma}^2_{avg} = E[\mathbf{x}_{avg}^2] - \boldsymbol{\mu}_{avg}^2 = \frac{1}{N}\boldsymbol{\sigma}^2 \tag{5}$$

Equation (5) shows that averaging of N candidate blocks results in a better predictor than any of the individual candidate blocks. $\boldsymbol{\sigma}^2_{avg}$ is N times smaller than $\boldsymbol{\sigma}^2$, thus statistically a smaller prediction error can be expected. This translates into improvements in coding efficiency as a smaller prediction error requires less transform coefficients to be coded and transmitted to the decoder.

3.2 Proposed predictor

To further improve the prediction performance of TMP described in 2.1, the multiple candidates averaging is integrated into TMP. The conventional schemes with the multiple candidates averaging need to send motion vectors to create their candidates. Our proposal does not need to encode any motion vectors, since the multiple candidates are searched at decoder using TMP.

As shown in Figure 3, multiple candidate blocks P_1 –P_N, for creating the final predictor of the target block B by averaging, are detected within a reconstructed frame F_{t-1}. Using the template T_B, the first N of matched templates T_{p1} - T_{pN} with the lowest SAD value between T_B and any template on a reconstructed frame F_{t-1} is searched by template matching. The blocks P_1 –P_N adjacent to the N of matched templates T_{p1} - T_{pN} are assigned as multiple candidate predictors of the target block B.

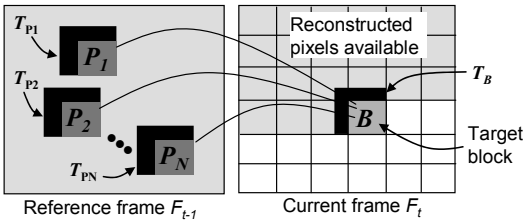

Figure 3 Template Matching Averaging.

To keep the correlation among candidates high, a prediction block P_i corresponding to a template T_{Pi} with SAD greater than (threshold Th) + (SAD of T_{P1}) is removed from the candidates. Since N=1 is selected when the difference between T_{P1} and T_{P2} is large, TMP is included in this method. The final sample predictor P_B is formed as follows:

$$P_B(x,y) = \frac{1}{N}\sum_{n=1}^{N} P_n(x,y), (x,y) \in B \cdot \quad (6)$$

We call this proposed method as Template Matching Averaging (TMA).

3.3. Block-based encoding and decoding algorithm

When introducing TMA into the inter frame coding of H.264, for every target block, the best prediction candidate is selected among forward motion compensated prediction (MCP), intra prediction, TMA-E and TMA-L in the criteria of the rate-distortion optimization [5]. In addition to the prediction modes of H.264, we added one macroblock (MB) mode, consisting of four TMA blocks, to the list of MB modes, and one sub-MB mode, consisting of one TMA block, to the list of sub-MB modes. If a MB mode or a sub-MB mode is 'TMA', a TMA mode, which consists of TMA-E and TMA-L, is additionally encoded. The meanings of '-E' and '-L' are same as described in section 2.

The block diagram of the proposed decoder is shown in Figure 4. The entropy decoding process produces prediction modes, motion vectors, and residual signals of the target blocks. If the prediction mode is 'MCP', the prediction signal is retrieved from the reference frame stored in the frame memory based on the decoded motion vector. If the prediction mode is 'TMA', a TMA mode is additionally decoded in the entropy decoding process. If the TMA mode is 'TMA-E', the prediction signal of TMA is added to the reconstructed residual signal. If the TMA mode is 'TMA-L', the entropy decoding process produces residual signals of the decimated block. Next, the decimated version of prediction signal of TMA is added to the residual signal to generate the reconstructed decimated block. After that, the reconstructed decimated block is upsampled to generate the reconstructed block. Note that intra mode is not shown in Figure 4.

TMA requires the decoder to perform the search. This may be a significant complexity for some decoder architectures. However, the search is only needed to perform if the encoder indicates that TMA mode was used for the block. Therefore the increase in complexity is correspondingly rewarded by the better bitrate performance.

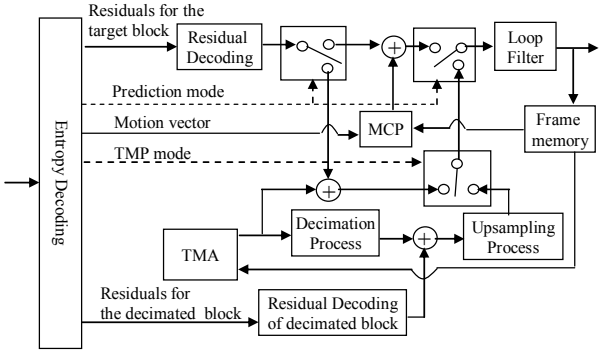

Figure 4 Diagram of proposed decoder

4. SIMULATION AND RESULTS

To confirm the improvement attained by our proposal, the proposed coding scheme described in subsection 3.3 is incorporated into the reference software of H.264 (JM8.6). The simulation condition is shown in Table 1. Using the assessment method for comparison of coding efficiency [16], the performance of the proposed coding scheme is compared to H.264. The block size of TMA-L is set to 8×8, since the block size of residual coding is 4×4.

Table 1 Simulation conditions.

Test sequences	Carphone, Crowd, F1, Foreman, Coastguard, Football (CIF, 15fps, around 10 seconds)
Prediction structure	IPPP…. (Luminance only)
Max No. of candidates	N=16
Block size for TMA-E	6 × 6 with the bottom-right 4 × 4 cut out
Block size for TMA-L	12 × 12 with the bottom-right 8 × 8 cut out
Entropy coding	Arithmetic coding (CABAC)
No. of reference frames	1 frame
RD optimization	On

Figure 5 shows the results of bitrate saving attained by 'proposal' as compared to 'H.264'. The plus value means that there is coding gain. The results show the proposed method reduces the total coding bitrate of 'H.264' up to 7.9%. Figure 6 shows the rate-distortion performance of 'Foreman' as an example, and the results indicate that the proposed method outperforms 'H.264'.

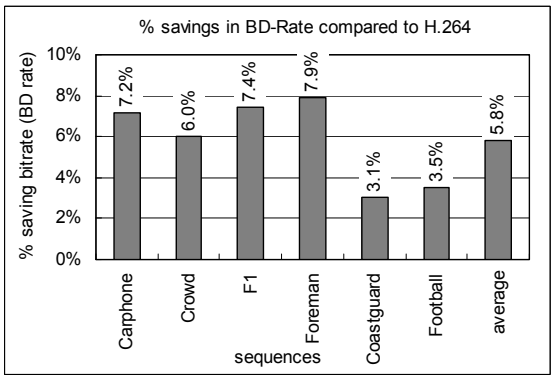

Figure 5 Bitrate saving

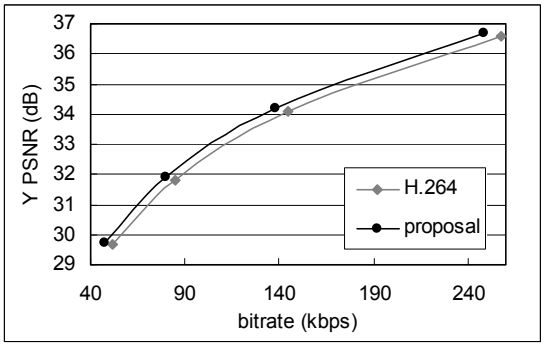

Figure 6 Example of coding performance (Foreman)

· Maximum number of predictor for averaging

In the experiments above, N=16 was used as the maximum number of multiple candidates. This value was selected according to the results of preliminary experiments shown in Figure 7 which indicates the saving bitrate attained by TMA-E of 8×8 block size as compared to H.264 when the value of N was changed. The 'Average' means the averaged BD rate of 6 sequences listed in Table 1. The results show that the performance peaks at N=16 on average. It is considered that the assumption described in subsection 3.1 (the candidate sub blocks $\mathbf{x}_i$ are i.i.d) is no longer kept when N becomes larger. Though the appropriate value of N may differ from one block to another according to the results of some sequences, we set to N=16 in this paper.

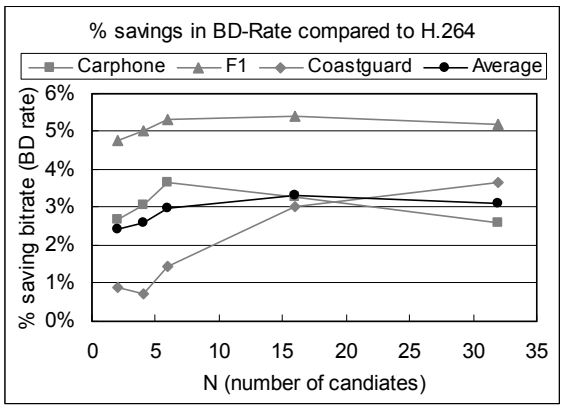

Figure 7 BD-Rate results for N averaging

· Block size for TMA-E

For the block size of TMA-E, 4×4 was used. Figure 8 shows the saving bitrate attained by TMA-E with N=16 as compared to H.264 for different block sizes. The results show the smaller block size is better. It is considered that the smaller block for averaging could be agreeable to the assumption of subsection 3.1.

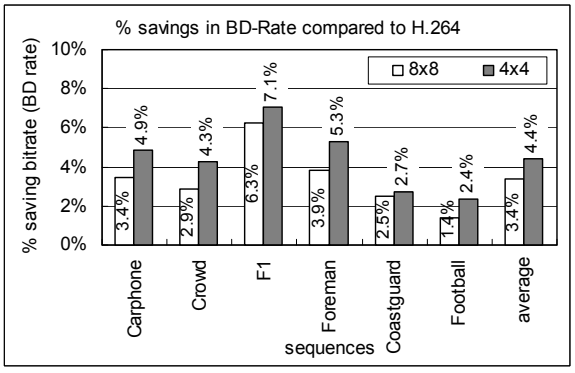

Figure 8 BD-Rate results for prediction block size

5. CONCLUSIONS

This paper proposed the incorporation of multiple candidates averaging into template matching prediction for improving the coding performance of our previous coding scheme. The noise component in the predictor is suppressed by averaging of multiple candidates.

Simulation results show that the proposed method reduces the total coding bit of H.264 up to 7.9 %. The appropriate number of blocks, N=16, and block size, 4×4, are considered from the viewpoint of the expectation of multiple candidates averaging.

6. REFERENCES

[1] H. Li, A. Lundmark and R. Forchheimer, "Image Sequence Coding at Very Low Bitrates: A Review", *IEEE Trans. Image Processing, vol. 3, No. 5,* September 1994, 589-609.

[2] "ITU-T Rec. H.264 | ISO/IEC 14496-10, Advance Video Coding for generic audiovisual services", May 2003.

[3] "Text of ISO/IEC 14496-2: 2004 (Third Edition)", MPEG-4 Part 2: Visual, June 2004.

[4] G.J. Sullivan and T. Wiegand, "Rate-Distortion Optimization for Video Compression", *IEEE Signal Processing Magazine*, November 1998, 74-90.

[5] T. Wiegand et. al., "Rate-Constrained Coder Control and Comparison of Video Coding Standards", *IEEE Trans.Circuits and Systems for Video Technology, vol. 13, No. 7,* July 2003, 688-703.

[6] A.N. Netravali and J.D. Robbins, "Motion-Compensated Television Coding: Part I", *The Bell Syst. Techn. Jorn.*, vol. BSTJ-58, no. 3, March 1979, 399-412.

[7] S.N. Efstratiadis and A.K. Katsaggelos, "A Model-Based Pel-Recursive Motion Estimation Algorithm", *Proc. ICIP 1990*, October 1990.

[8] K. Sugimoto et. al., "Inter Frame Coding with Template Matching Spatio-Temporal Prediction", *Proc. ICIP 2004*, Singapore, October 24-27, 2004.

[9] M. Kobayashi et. al., "Reduction of Information with motion prediction using template matching", *Proceedings of the 20th Picture Coding Symposium of Japan*, Nov 9-11, 2005, pages 17-18.

[10] Y. Suzuki et. al, "Block-based Reduced Resolution Inter Frame Coding with Template Matching Prediction ", *Proc. ICIP 2006*, Atlanta, GA, USA, October 8-11 2006.

[11] T.K. Tan et. al, "Intra Prediction by Template Matching ", *Proc. ICIP 2006*, Atlanta, GA, USA, October 8-11 2006.

[12] M. Flierl et. al, "Rate-Constrained Multihypothesis Prediction for Motion-Compensated Video Compression", *IEEE Trans. CSVT, vol. 12, No. 11*, November 2002.

[13] Y. Suzuki et. al., "Reduction of Coding Bits for Residuals by Smoothing of Prediction Signal in Block-based Inter Frame Coding", *Proc. the 21st Picture Coding Symposium of Japan*, November 8-10 2006.

[14] T.K. Tan et. al, "Intra Prediction by Averaged Template Matching Predictors", *Proc. CCNC 2007*, Las Vegas, NV, USA, January 11-13 2007.

[15] "Draft Text of Recommendation H.263 Version 2 ("H.263+") for Decision", *ITU-T SG16 Q.15/16*, February 1998.

[16] G.Bjontegaard, Calculation of average PSNR differences between RD-curves, *VCEG-M33*, April 2001.