A Temporal Dependency Model for Rate-Distortion Optimization in Video Coding

Jingning Han, Paul Wilkins, Yaowu Xu, and James Bankoski
Google LLC
1600 Amphitheatre Parkway, Mountain View, CA 94043, USA
{jingning, paulwilkins, yaowu, jimbankoski}@google.com

***Abstract*—Video codec heavily relies on motion compensated prediction to achieve compression efficiency. The predictive scheme creates temporal dependency across frames, i.e., the quantization distortion in a current block may propagate through motion compensated prediction and affect the coding efficiency of blocks in subsequent frames. The ability to capture such dependency can potentially improve the rate-distortion optimization for coding performance gains. Prior research work builds block-based motion trajectories and uses the correlations between source pixel blocks in the same motion trajectory to estimate the distortion propagation model. This work premises on the realization that the distortion propagation is also largely related to the quantization effect. A novel temporal dependency model that accounts for both block correlation and the quantization effect is proposed. It is experimentally shown to provide considerable compression gains over the existing competitors.**

***Index Terms*—motion compensated prediction, rate-distortion optimization, temporal dependency, video compression**

I. Introduction

Video compression techniques exploit the temporal correlations in video signal, most commonly in the form of motion compensated prediction, to achieve superior coding efficiency. Such predictive coding scheme creates dependency between a current coding block and its reference block. The reconstruction quality of one block can potentially influence the compression efficiency of blocks in the subsequent frames. Intuitively if a reference block predicts a current block well, shifting the bit allocation to improve the reference block reconstruction quality would reduce the overall distortion under the same rate cost. Whereas if the two blocks are less relevant, spending the bits according to their individual needs would make better rate-quality trade off.

A typical rate allocation approach to optimizing the performance of hierarchical coding structure is to use lower quantization parameters (QP) for frames at lower temporal layer, which serve as the reference frames for later higher temporal layer frames [1]. A trellis-based rate allocation optimization is proposed in [2], where each node corresponds to a frame encode at a given QP. The scheme finds the path that minimizes the overall rate-distortion cost as the optimal frame QP combinations for the entire sequence. To achieve optimality for frame level QP selection, the encoding complexity increases significantly.

Recent work [3] [4] exploits inter frame dependency to optimize the rate allocation. A linear model is proposed in [3] to capture the frame level distortion propagation, whose parameters are trained offline. In [4] the inter frame dependency model is updated according to the coding statistics from previously coded frames at the same temporal layer. Both adapt the frame QP according to the derived dependency models. A block based temporal dependency model is proposed in [5] [6]. It conducts forward search over next coding frames to measure the impact of the reconstruction distortion of a current block, based on which the Lagrangian multiplier of each coding tree unit in a current frame will be adjusted [7]. Certain simplifications, including relaxing motion trajectory on-grid alignment constraint (a coding block must be on grid whereas its reference block does not) and assuming all inter-mode coded blocks (ignoring the possible use of intra prediction), are employed to build the model so as to make the whole process complete in a single pass encoding.

A macroblock-tree (MB-tree) scheme [8] is implemented in the x264/5 codec that tracks the temporal dependency through block level motion trajectories. It uses a two-pass encoding approach. The first pass runs with ordinary rate-distortion optimization based mode decision without accounting for the distortion impact on the subsequent frames. The second pass first utilizes the motion vectors and inter/intra mode decisions available from the first pass to build motion trajectories over the source frames (i.e., uncompressed frames). To estimate each block's impact on the subsequent blocks in the same motion trajectory, the MB-tree scheme uses a linear model of the intra- and inter-prediction errors. The correlation between the two blocks is estimated by the difference error divided by the intra-prediction error. The correlations are then recursively propagated through the motion trajectories, which form a temporal dependency model. Based on that the encoder adjusts the rate allocation to improve reconstruction quality of blocks that have higher impact on their subsequent blocks in the motion trajectories. It has been shown that the MB-tree system provides significant compression gains over conventional frame type dependent constant quantization parameter coding scheme.

This work builds on the realization that the MB-tree largely ignores the quantization effect on the temporal dependency by building its model based on the inter- and intra-prediction errors over the source signals, whereas the true distortion propagation depends on the relative energy value between the innovation term and the quantization error. In high resolution

978-9-0827-9703-9/19/$31.00

quantization case, the latter is substantially smaller than the innovation, hence the distortion propagation between frames is minimal. When the quantization error is comparable or exceeds the innovation term (a fairly common situation in medium bit-rate range), the distortion propagation is largely determined by the correlation between blocks in the same motion trajectory. Hence we propose a new temporal dependency model approach that extends the MB-tree scheme to account for the quantization effect on the distortion propagation through the motion trajectory. It is experimentally shown that the proposed scheme provides considerable compression efficiency improvement on top of the MB-tree scheme.

II. The Macroblock-Tree Scheme

We provide a brief description of the MB-tree system initially proposed in [8] for H.264. Similar design principle can be easily applied to later generation codecs like H.265 and VP9. The MB-tree estimates the amount of information each MB contributes to the prediction of future frames, which is used to weight the rate-distortion trade-off for each MB based on its contribution. The scheme works in the reverse frame processing order over the source frames, propagating information from future frames back to the current frame.

For each frame, a propagation step is run for each MB. it operates as follows:

1) Estimate the intra prediction cost in terms of sum of absolute Hadamard transform difference (SATD) noted as $intra\_cost$. It also loads the motion information available from the first-pass encode and estimates the inter prediction cost as $inter\_cost$. Since modern codecs - H.264/5, VP8/9 - all use hybrid inter/intra prediction mode, the $inter\_cost$ value is further upper bounded by $intra\_cost$. A $propagation\_cost$ variable is used to collect all the information flowed back from future processing frames. It is initialized as 0 for all the MBs in the last processing frame in a group of pictures (GOP).
2) The fraction of information from a current MB to be propagated towards its reference block is estimated as
$$progation\_fraction = (1 - inter\_cost/intra\_cost). \quad (1)$$
It reflects how much the motion compensated reference would reduce the prediction error in percentage.
3) The total amount of information the current MB contributes to the GOP is estimated as $intra\_cost + propagation\_cost$. The information that it propagates towards its reference block is captured by
$$\begin{aligned} &propagation\_amount = \\ &\quad (intra\_cost + propagation\_cost) * \\ &\quad propagation\_fraction. \end{aligned}$$
4) Note that the reference block may not necessarily sit on the grid of MBs. The $propagation\_amount$ is dispensed to all the MBs that overlap with the reference block. The corresponding MB in the reference frame accumulates its own $propagation\_cost$ as it receives back propagation:
$$\begin{aligned} &propagation\_cost+ = \\ &\quad (overlap\_area/MB\_area) * \\ &\quad propagation\_amount. \end{aligned}$$
Similar information dispense approach has been used in [9] as well.

In the final encoding stage, the distortion propagation factor of a MB is evaluated as $(1 + propagation\_cost/intra\_cost)$, where the second term captures its impact on later frames in a GOP. The rate allocation is hence adjusted according to the distortion model such that MBs with higher distortion propagation factor get higher rate allocation and vice versa.

III. The Proposed Temporal Dependency Model

This work re-designs the distortion propagation model in MB-tree to account for the quantization effect. Consider the second moment of inter prediction error:
$$\sigma_k^2 = E\{||M_k - \hat{M}_{k-1}||^2\}, \quad (2)$$
where k represents frame index, M_k is the source pixel block and $\hat{M}_{k-1}$ is the reconstructed reference block. Assume the innovation term between M_k and M_{k-1} is largely uncorrelated with the quantization noise at $\hat{M}_{k-1}$ [5], we have
$$\begin{aligned} \sigma_k^2 &= E\{||M_k - M_{k-1} + M_{k-1} - \hat{M}_{k-1}||^2\} \\ &\approx E\{||M_k - M_{k-1}||^2\} + E\{||M_{k-1} - \hat{M}_{k-1}||^2\} \\ &= \sigma_o^2 + D_{k-1}, \end{aligned}$$
where the prediction error σ_k^2 is approximately decomposed into the innovation term σ_0^2 and the quantization distortion in the reference block D_{k-1}.

Under high resolution quantization assumption, it is known that the expected quantization distortion is linear with the input signal energy [10] [11]:
$$D_k = \alpha(\sigma_o^2 + D_{k-1}), \quad (3)$$
where α is decided by the bit-rate and the probability distribution of the input signal. The relationship largely holds in other bit-rate range too. The α value is empirically assumed to be 0.94 in [5] and 1.0 in [8]. Instead this work proposes to directly estimate the effective linear relationship α per block.

When building the temporal dependency model, the encode has the access to the source blocks M_k and M_{k-1}, and their difference
$$R_k = M_k - M_{k-1}. \quad (4)$$
We apply Hadamard transform (this can be replaced with Discrete Cosine Transform for slightly better overall compression performance) to R_k and quantize the transform coefficients to obtain its quantized version:
$$\hat{R}_k = T^{-1}Q(T(R_k)). \quad (5)$$

The Hadamard transform approximates the Discrete Cosine Transform and has simple and fast implementation to reduce the encoder complexity increase. This step allows us to account the transform coding gains for evaluating the quantization noise. The distortion on the innovation term can be estimated by:

$$D_{k0} \approx E\{||R_k - \hat{R}_k||^2\} \tag{6}$$

and the prediction error as:

$$\sigma_o^2 = E\{||R_k||^2\}. \tag{7}$$

In typical encoding settings, a frame usually has larger or similar QP as compared to its reference frames, which implies $D_k \geq D_{k-1}$. Furthermore we assume the quantization noise is bounded by the innovation term, i.e., $\sigma_o^2 \geq D_k$. Hence we assume that $\sigma_o^2 \geq D_{k-1}$ in (3). The quantization effect on the innovation term largely captures the linear relationship in (3):

$$\alpha \approx \frac{D_{k0}}{\sigma_o^2}. \tag{8}$$

Therefore the distortion in the reference block $\hat{M}_{k-1}$ contributes approximately

$$\alpha D_{k-1} = \frac{D_{k0}}{\sigma_o^2} D_{k-1} \tag{9}$$

to block M_k. Accordingly the distortion propagation model in (1) is re-designed as:

$$propagation\_fraction = \frac{D_{k0}}{\sigma_o^2} \cdot (1 - \frac{inter\_cost}{intra\_cost}), \tag{10}$$

where the first term captures the quantization effect and the second term reflects the mutual information between the reference and the current blocks.

Clearly when the quantization noise is significantly smaller than the innovation energy, the inter frame distortion propagation is close to 0, which translates into the fact that there is no need to account for the distortion impact on future frames when conducting the rate-distortion optimization for a current frame coding. When the quantization noise is comparable to the innovation process, the impact of a current block on subsequent blocks in the motion trajectory depends on their correlations. Here we quantify the correlations as the percentage of intra prediction error reduction due to inter prediction.

The proposed temporal dependency model is hence built as such:

1) Gather the $intra\_cost$, $inter\_cost$, and $propagation\_cost$ as discussed in Section II.
2) Apply Hadamard transform (or Discrete Cosine Transform) to the inter prediction residuals, followed by the quantization process. Obtain the prediction error σ_o^2 and quantization error D_{k0} respectively.
3) The fraction of information from a current block to be propagated towards its reference block is estimated as
$$propagation\_fraction = \frac{D_{k0}}{\sigma_o^2} \cdot (1 - \frac{inter\_cost}{intra\_cost}). \tag{11}$$
4) The total amount of information the current block contributes to the GOP is estimated as $intra\_cost + propagation\_cost$. The information that it propagates towards its reference block is captured by
$$propagation\_amount = (intra\_cost + propagation\_cost) * propagation\_fraction.$$
5) The $propagation\_amount$ is dispensed to all the blocks that overlap with the reference block. The corresponding block in the reference frame accumulates its own $propagation\_cost$ as it receives back propagation:
$$propagation\_cost += (overlap\_area/block\_area) * propagation\_amount.$$

The distortion propagation factor of a block is evaluated by $(1 + propagation\_cost/intra\_cost)$. Same rate-distortion optimization trade-off as Section II applies here.

IV. Experimental Results

We implemented both MB-tree and the proposed distortion propagation model in the VP9 codec. The source code can be found at [12]. The baseline framework uses two-pass encoding, where the first pass gathers inter frame statistics to optimize the frame level rate control in the second pass. To validate the efficacy of the proposed approach, we used the distortion propagation model, described in Section II for MB-tree and Section III for temporal dependency model, to adapt the Lagrangian multiplier at 64x64 coding block level.

For every 64x64 block in a frame, we have their distortion propagation factor:

$$dist\_prop[i] = 1 + \frac{propagation\_cost[i]}{intra\_cost[i]}, \tag{12}$$

where i denotes the block index in the frame. We also have the frame level distortion propagation factor:

$$dist\_prop = 1 + \frac{\sum_i propagation\_cost[i]}{\sum_i intra\_cost[i]}. \tag{13}$$

We used (13) to normalize the block distortion propagation factor in (12) and adapted the Lagrangian multiplier at 64x64 block level as:

$$\lambda[i] = \lambda_0 * \frac{dist\_prop}{dist\_prop[i]}, \tag{14}$$

where λ_0 is the multiplier associated with frame level QP. Hence a block with higher relative distortion propagation factor would have a smaller Lagrangian multiplier, which biases the rate-distortion optimization to reduce the reconstruction distortion. Note that there are more complex and advanced algorithms to optimize the rate allocation based on the temporal dependency information. We use the above described simple approach to validate the efficacy of the proposed temporal dependency model referred to as TPL hereafter.

The encoding parameters are set as:

```
./vpxenc input_file.y4m -o output_file.y4m
--target-bitrate=$BIT_RATE
--cpu-used=0 --passes=2
```

where `--cpu-used=0` makes the encoder run at highest complexity mode for best compression efficiency. The test clips include CIF to HD resolutions. The operating bit-rates are set to cover $35dB$ to $45dB$ range for each clip. Their coding performance as compared to the baseline is shown in Table I.

TABLE I
THE COMPRESSION PERFORMANCE OF THE LAGRANGIAN MULTIPLIER OPTIMIZATION USING MB-TREE AND THE PROPOSED TPL RESPECTIVELY AS COMPARED TO BASELINE VP9 ENCODER. THE PERFORMANCE IS EVALUATED IN TERMS OF BD-RATE REDUCTION. A NEGATIVE NUMBER MEANS BETTER COMPRESSION EFFICIENCY.

	MB-tree		TPL	
	PSNR	SSIM	PSNR	SSIM
basketballpass_240p	-1.72%	-3.71%	-1.63%	-4.65%
keiba_240p	-0.75%	-1.22%	-1.00%	-2.00%
football_cif	-0.18%	-0.86%	-0.20%	-1.40%
ice_4cif	-1.79%	-3.09%	-2.08%	-5.42%
RaceHorses_480p	-1.25%	-1.63%	-1.22%	-2.10%
soccer_4cif	-1.27%	-1.81%	-1.44%	-3.05%
harbour_4cif	-1.18%	-1.31%	-1.16%	-1.34%
BalloonFestival_720p	-0.59%	-3.70%	-0.75%	-4.52%
Market3_720p	-0.65%	-2.43%	-0.87%	-3.52%
parkjoy_1080p	-2.38%	-4.19%	-2.68%	-5.58%
factory_1080p	-0.54%	-0.38%	-0.69%	-0.93%
tennis_1080p	-0.82%	-0.59%	-0.90%	-1.39%
pedestrian_1080p	-2.48%	-2.95%	-2.71%	-4.07%
parkscene_1080p	-1.70%	-1.40%	-1.54%	-2.26%
ducks_take_off_1080p	-0.46%	-0.25%	-0.66%	-0.65%
cyclists_720p	-0.11%	0.204%	-0.79%	-2.70%

It is observed that the MB-tree provides fairly consistent compression gains over the baseline, where the frame level QP is optimized according to the first pass encode statistics. The proposed TPL model further outperforms MB-tree when the innovation to quantization noise ratio varies significantly across 64x64 blocks within a frame, e.g., `ice_4cif` and `pedestrian_1080p`. When the innovation process is largely uniform across the frame, e.g., `harbour_4cif`, the quantization effect in (8) uniformly applies to all 64x64 blocks. Its effect will be mostly cancelled out by the normalization step in the above Lagrangian multiplier adaptation scheme. Hence we typically see similar compression performance between MB-tree and TPL.

V. CONCLUSIONS

A novel temporal dependency model is proposed to account for the quantization effect on the distortion propagation through the motion trajectory. Integrated with an adaptive Lagrangian multiplier scheme, the derived model is shown to provide considerable compression performance improvements over the MB-tree. While tested in the VP9 framework, the proposed temporal dependency model is generally applicable to all block based video codec that uses motion compensated prediction.

REFERENCES

[1] Zhao, Tiesong, Zhou Wang, and Chang Wen Chen. "Adaptive quantization parameter cascading in HEVC hierarchical coding." IEEE Transactions on Image Processing 25.7 (2016): 2997-3009.

[2] Liu, Shan, and C-CJ Kuo. "Joint temporal-spatial bit allocation for video coding with dependency." IEEE Transactions on Circuits and Systems for Video Technology 15.1 (2005): 15-26.

[3] Hu, Sudeng, et al. "Rate control optimization for temporal-layer scalable video coding." IEEE Transactions on Circuits and Systems for Video Technology 21.8 (2011): 1152-1162.

[4] He, Jing, et al. "Adaptive quantization parameter selection for h. 265/hevc by employing inter-frame dependency." IEEE Transactions on Circuits and Systems for Video Technology 28.12 (2018): 3424-3436.

[5] Li, Shuai, et al. "Lagrangian multiplier adaptation for rate-distortion optimization with inter-frame dependency." IEEE Transactions on Circuits and Systems for Video Technology 26.1 (2016): 117-129.

[6] Gao, Yanbo, et al. "Temporally dependent rate-distortion optimization for low-delay hierarchical video coding." IEEE Transactions on Image Processing 26.9 (2017): 4457-4470.

[7] Li, Bin, et al. "λ domain rate control algorithm for High Efficiency Video Coding." IEEE transactions on Image Processing 23.9 (2014): 3841-3854.

[8] Garrett-Glaser, Jason. "A novel macroblock-tree algorithm for high-performance optimization of dependent video coding in H. 264/AVC." Tech. Rep. (2009).

[9] Han, Jingning, Vinay Melkote, and Kenneth Rose. "Estimation-theoretic approach to delayed decoding of predictively encoded video sequences." IEEE Transactions on Image Processing 22.3 (2013): 1175-1185.

[10] Sullivan, Gary J., and Thomas Wiegand. "Rate-distortion optimization for video compression." IEEE signal processing magazine 15.6 (1998): 74-90.

[11] Gersho, Allen, and Robert M. Gray. "Vector quantization and signal compression." Vol. 159. Springer Science and Business Media, 2012.

[12] "https://chromium.googlesource.com/webm/libvpx"