Rate Distortion Optimization in H.264

En-hui Yang Xiang Yu
Department of Electrical and Computer Engineering
University of Waterloo, Waterloo, Ontario, N2L 6E3, Canada
Email: ehyang@uwaterloo.ca x23yu@bbcr.uwaterloo.ca

Abstract— **In this paper, we study the best rate distortion performance that an H.264 encoder can possibly achieve. Using soft decision quantization rather than the standard hard decision quantization, we first establish a general framework for jointly designing motion compensation, quantization, and entropy coding in the hybrid coding structure of H.264 to minimize a true rate distortion cost. We then propose three rate distortion optimization algorithms—a graph-based algorithm for optimal soft decision quantization in H.264 baseline profile encoding given motion compensation and quantization step sizes, an iterative algorithm for optimal residual coding in H.264 baseline profile encoding given motion compensation, and an iterative overall algorithm for optimal H.264 baseline profile encoding—with them embedded in the indicated order. The graph-based algorithm for optimal soft decision quantization is the core; given motion compensation and quantization step sizes, it is guaranteed to perform optimal soft decision quantization to certain degree. The proposed iterative overall algorithm has been implemented based on the reference encoder JM82 of H.264. Comparative studies show that it achieves a significant performance gain, which can be as high as 25% rate reduction at the same PSNR when compared to the reference encoder.**

I. Introduction

H.264, the newest hybrid video compression standard [2], has proved its superiority in coding efficiency over its precedents, e.g., it shows a more than 50% rate reduction over the popular MPEG-2. However, as the enormous volume of video data constantly demands for better and better compression, it is desirable to study how rate distortion (RD) methods can be used to further enhance the compression performance in the H.264 standard-compliant coding environment.

RD methods can be classified into two categories. The first category computes the theoretical RD function based on a given statistic model for video data [1]. These methods usually encounter a problem of model mismatch. The second category uses an operational RD function, which is computed based on the data to be compressed. In general, there are two main problems. First, in most operational methods, the formulated optimization problem is restricted and the RD cost is optimized only over motion compensation and quantization step sizes. Second, there is no simple way to solve the restricted optimization problem if the actual RD cost is used because of hard decision quantization. Hence, an approximate RD cost is often used in the restricted optimization problem in many operational methods. For example, the optimization of motion compensation in [5] is based on the prediction error instead of the actual distortion, which is the quantization error.

In this paper, we shall use the actual RD cost and take a different approach to design an operational RD method. Using soft decision quantization (SDQ) instead of hard decision quantization, we discover that the quantized residual itself is a free parameter that can be optimized in order to improve compression performance. Then, we formulate a general framework for jointly designing motion compensation, quantization, and entropy coding in the H.264 hybrid coding structure. Surprisingly, this generality not only improves the compression performance in term of the RD tradeoff, but also makes the optimization problem tractable at least algorithmically. Indeed, with respect to the baseline profile of H.264, we propose three RD optimization algorithms—a graph-based algorithm for optimal soft decision quantization given motion compensation and quantization step sizes, an iterative algorithm for optimal residual coding given motion compensation, and an iterative overall algorithm for optimal H.264 baseline profile encoding—with them embedded in the indicated order. The algorithm for optimal SDQ is the core. The SDQ design is based on a graph structure developed specifically for the context adaptive variable length coding (CAVLC) method in the baseline profile of H.264. Given motion compensation and quantization step sizes, the graph-based algorithm is guaranteed to perform optimal soft decision quantization to certain degree. The proposed iterative overall algorithm has been implemented based on the reference encoder JM82 of H.264. Comparative studies show that it achieves a significant performance gain over other baseline-based methods reported in the literature, which can be as high as 25% rate reduction at the same PSNR when compared to the reference encoder.

The proposed rate distortion optimization algorithms for H.264 video coding are inspired by a fixed-slope universal lossy data compression scheme considered in [4], which was first initiated in [6]. Other related works on practical SDQ include without limitation SDQ in JPEG image coding and H.263+ video coding (see [7], [8], [11] and references therein). In [7], partial SDQ called rate-distortion optimal thresholding was considered. Recently, Yang and Wang [8] successfully developed an algorithm for optimal SDQ in JPEG compatible image coding.Without considering optimization over motion compensation and quantization step sizes, Wen et. al [11] proposed a trellis-based algorithm for optimal SDQ in H.263+ video coding, which, however, is not applicable to H.264 due to the inherent difference in the entropy coding stages of H.264 and H.263+.

This paper is organized as follows. In Section II, we develop a framework for jointly designing the hybrid coding structure in H.264. Section III is then dedicated to the core algorithm of SDQ based on CAVLC. Simulation results and conclusion are presented in Section IV and V, respectively.

II. The syntax-constrained optimization framework for H.264 video compression

Besides the well-studied variabilities in H.264, e.g., motion vectors, prediction modes, and quantization step sizes, we discovered in [9] a somehow hidden parameter, i.e., the quantized coefficient, which can be optimized in order to improve compression performance, leading to an SDQ design. Using SDQ instead of conventional hard-decision quantization a joint design framework of motion compensation, quantization and entropy coding in the hybrid structure is formulated as follows,

$$\min_{\boldsymbol{m},\boldsymbol{V},\boldsymbol{q},\boldsymbol{U}} d(\boldsymbol{X},\hat{\boldsymbol{X}}) + \lambda \cdot (r(\boldsymbol{m}) + r(\boldsymbol{V}) + r(\boldsymbol{q}) + r(\boldsymbol{U})), \quad (1)$$

where $\boldsymbol{X}$ is a given frame, $\hat{\boldsymbol{X}}=\boldsymbol{P}(\boldsymbol{m},\boldsymbol{V})+\hat{\boldsymbol{Z}}(\boldsymbol{q},\boldsymbol{U})$ is the reconstruction with $\boldsymbol{P}$ denoting the prediction, $\hat{\boldsymbol{Z}}=\{\hat{z}: \hat{z}=\mathrm{T}^{-1}(q\boldsymbol{u})\}$, T^{-1} denotes the inverse DCT transform in H.264, $d(\cdot)$ is a distortion measure, $r(\cdot)$ is the rate function for CAVLC, λ is a positive constant corresponding to the slope of a point on the RD curve, $\boldsymbol{m},\boldsymbol{V},\boldsymbol{q},\boldsymbol{U}$ are the prediction modes, motion vectors, quantization step sizes, and quantized transform coefficients of the frame, respectively, and $(q,\boldsymbol{u})$ correspond to a block.

To make the problem tractable, an iterative solution is proposed based on three algorithms—one for optimal soft decision quantization given motion compensation and quantization step sizes, one for optimal residual coding given motion compensation, and an overall iterative algorithm for jointly designing motion compensation, quantization and entropy coding—with them embedded in the indicated order. Specifically, the SDQ algorithm for given motion compensation and quantization step sizes is formulated as,

$$\min_{\boldsymbol{U}} d(\boldsymbol{X}-\boldsymbol{P},\hat{\boldsymbol{Z}}(\boldsymbol{q},\boldsymbol{U})) + \lambda \cdot r(\boldsymbol{U}). \quad (2)$$

The overall iterative algorithm is obtained by alternately optimizing residual coding and motion compensation as follows.

1. *Optimal residual coding.* For given motion prediction $\boldsymbol{P}$, residual coding is optimized by
$$\min_{\boldsymbol{q},\boldsymbol{U}} d(\boldsymbol{X}-\boldsymbol{P},\hat{\boldsymbol{Z}}(\boldsymbol{q},\boldsymbol{U})) + \lambda \cdot (r(\boldsymbol{q}) + r(\boldsymbol{U})). \quad (3)$$
2. *Motion compensation optimization.* For given residual reconstruction $\hat{\boldsymbol{Z}}$, motion compensation is optimized by
$$\min_{\boldsymbol{m},\boldsymbol{V}} d(\boldsymbol{X}-\hat{\boldsymbol{Z}},\boldsymbol{P}(\boldsymbol{m},\boldsymbol{V})) + \lambda \cdot (r(\boldsymbol{m}) + r(\boldsymbol{V})). \quad (4)$$

As discussed in [3], the core of the above iterative solution is the SDQ algorithm, based on which the optimal residual coding algorithm is developed. In the following, we propose the SDQ algorithm based on the CAVLC entropy coding method in H.264, while the residual coding algorithm for given SDQ outputs and the motion compensation algorithm are simple and not included here (See [3] for more details.) At this point, it is not clear whether or not the above iterative joint optimization algorithm will converge to the global optimal solution of (1). However, the iterative joint optimization algorithm does converge in the sense that the actual rate distortion cost is decreasing at each iteration step.

III. Soft Decision Quantization Algorithm Design

This section presents the core SDQ algorithm for (2) based on the CAVLC entropy coding method in H.264. Clearly, for given motion prediction and $\boldsymbol{q}$, the distortion term in (2) is block-wise additive. Note that $\boldsymbol{U} = \{\boldsymbol{u}_1, \cdots, \boldsymbol{u}_{16K}\}$. In H.264, encoding of each block $\boldsymbol{u}_k$ depends not only on $\boldsymbol{u}_k$ itself, but also on its two neighboring blocks. However, such dependency is very weak, and the number of bits needed to encode $\boldsymbol{u}_k$ largely depends on $\boldsymbol{u}_k$ itself. Therefore, in the given optimization problem, we will decouple such weak dependency. In doing so, the optimization of the entire frame can be solved in a block by block manner with each block being 4×4. By omitting the subscript, the optimization problem given in (2) now reduces to,

$$\boldsymbol{u} = \arg\min_{\boldsymbol{u}} \ d(\boldsymbol{x}-\boldsymbol{p},\mathrm{T}^{-1}(\boldsymbol{u}\cdot q)) + \lambda \cdot r(\boldsymbol{u}) \quad (5)$$

where $r(\boldsymbol{u})$ is the number of bits needed for CAVLC to encode $\boldsymbol{u}$ given that its two neighboring blocks have been optimized.

We now address the computation issue for the distortion term in (5) as it contains the inverse DCT transform. Consider that DCT is a unitary transform, which maintains the Euclidean distance. We use the Euclidean distance for $d(\cdot)$. Then, the problem of (5) becomes

$$\boldsymbol{u} = \arg\min_{\boldsymbol{u}} \quad ||\boldsymbol{c}-\boldsymbol{u}\cdot q||^2 + \lambda \cdot r(\boldsymbol{u}), \quad (6)$$

where $\boldsymbol{c} = \mathrm{T}(\boldsymbol{x}-\boldsymbol{p})$ is computed before SDQ. Besides the computational simplicity, the computation of distortion in the DCT domain facilitates a dynamic programming solution of the SDQ problem because the distortion is now computed in an element-wise additive manner.

A. Review of CAVLC

The CAVLC entropy coding method is briefly summarized as follows (see [2] for details, including notations mentioned but otherwise undefined in this paper),

1. Initialization. An input sequence is scanned in the reverse order to form the run-length codewords, as well as to initialize parameters such as $\mathit{TotalCoeffs}$, $\mathit{T1s}$, and $\mathit{TotalZeros}$.
2. Encoding trailing levels with value ± 1. It is named the trailing ones coding rule.
3. Encoding other levels. 7 variable length coding tables, named as Vlc(i) with $0 \leq i \leq 6$, are used to encode levels one by one. The table selection criteria are summarized in the following pseudo codes.

```
//Choose a table for the first level
if(TotalCoeffs>10&&T1s<3)use Vlc(1);
else                     use Vlc(0);
// Update the table selection
vlc_inc[7]={0,3,6,12,24,48,inf}   ;
if(level>vlc_inc[i])      i ++    ;
if(level>3 && FirstLevel) i = 2   ;
```

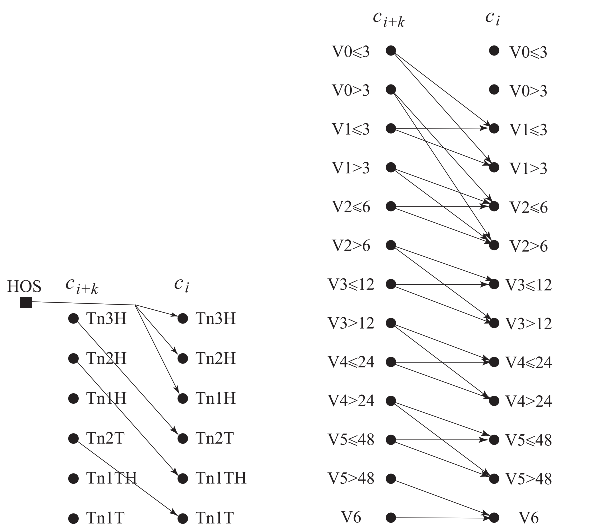

Fig. 1. States and transitions. Left panel: defined according to the trailing one coding rule of CAVLC. Right Panel: defined according to the *level* coding process.

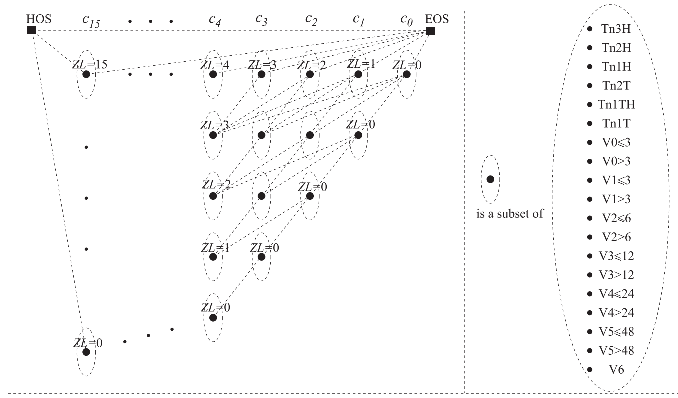

Fig. 2. The graph structure for soft decision quantization based on CAVLC. There are 16 columns according to 16 coefficient. A column consists of multiple state groups, according to different $ZerosLeft$. The left panel shows the connections between these groups. Each group initially contains a set of states defined on the right panel, while eventually only those states that receive connections from HOS will remain. Note that ZL in the figure stands for $ZerosLeft$ in the text.

4. Encoding runs. A parameter, $ZerosLeft$, defined as the number of zeros between the current $level$ and the scan end is used to select one table out of 7 to encode the current run.

B. Graph Design for Soft Decision Quantization

A graph structure is developed to solve the minimization problem in (6). Specifically, a graph is constructed to represent the vector space of the quantization outputs, with each transition standing for a $(run, level)$ pair and each path in the graph giving a unique sequence of quantization output.

Figure 1 shows the states and transitions defined according to the trailing ones coding rule and the $level$ coding process in CAVLC. The state definition implies a restriction to the state output. For example, the output for the state V$i > T_i$ must be greater than T_i. Consider the dynamic range of $[1, 255]$ for a $level$ in H.264. The output range for V$i \leq T_i$ is $[1, T_i]$, while the output for V$i > T_i$ will be any integer in $[T_i + 1, 255]$. For V6, the output range will be the full range of $[1, 255]$.

Consider the runs coding process of CAVLC. As shown in Figure 2, a state group is defined for each different $ZerosLeft$ as a set of all states defined according to the $level$ coding process and the trailing one rule. For coefficient $c(i)$, there are $(i + 1)$ groups, corresponding to $ZerosLeft = 0, 1, \cdots, i$. Connections between groups are shown in the left panel of Figure 2.

Based on the state definition for trailing ones and the state group formation, we are able to follow the $level$ coding table initialization rule in the graph design. Specifically, for a path starting at $c(i)$ with $ZerosLeft$, we know that $TotalCoeffs = i+1-ZerosLeft$. Thus, the rule is applied to build connections from trailing one states to the initial $level$ coding states.

Finally, we expand the main structure in the left panel of figure 2 into a full graph. Specifically, there are 16 columns, each of them corresponding to one coefficient. Each column contains several groups of states. There are $i + 1$ groups for the column of $c(i)$. The graph structure starts from HOS, with connections going to Tn1H in all groups, to Tn2H in groups with $i+1-ZerosLeft > 1$, to Tn3H in groups with $i+1-ZerosLeft > 2$, to V0$\leq$3 and V0>3 in groups with $i+1-ZerosLeft \leq 10$, and to V1$\leq$3 and V1>3 in groups with $i+1-ZerosLeft > 10$. Then, as discussed in the above, connections between state groups are established according to the rules shown in Figure 1 and the $level$ coding table initialization rule. Eventually, while each group initially contains 19 states as shown in the right panel of Figure 2, only those states that receive connections from HOS will remain valid. The graph ends at a dummy state EOS. There is also a connection from HOS directly to EOS, corresponding to a case where the whole sequence is quantized to 0.

Theorem: Given a 4×4 residual block, applying Viterbi algorithm for a search in the proposed graph gives the optimal solution to the soft decision quantization problem of (6).

Proof: We first show that for a given input sequence $\boldsymbol{c} = (c_{15}, \cdots, c_0)$ any possible sequence of quantization outputs accords to a path in the proposed graph by introducing parallel transitions between two connected states in the graph. Then, we define a metric for each transition in the graph so that for any path the accumulated metric equals to the RD cost of (6). Consequently, Viterbi algorithm can be used to search for a path in the graph to minimize the RD cost, and the obtained path gives the optimal quantization outputs for solving (6).

We now define parallel transitions between two connected states in the proposed graph. As discussed in the above, the output of a state based on a $level$ coding table will be any integer within a given range. Consider a connection from a state s_1 at c_i to a state s_2 at c_j. Denote the output range of s_2 as $[u_{\text{low}}, u_{\text{high}}]$. There will be $(u_{\text{high}} - u_{\text{low}} + 1)$ parallel transitions from s_1 to s_2, with each according to a unique quantization output within the given range. Consider that for each coding table two complementary states are defined to cover the whole dynamic range of a *level* (or in case of Vlc(6) the state V6 covers the whole range). It is not hard to see that the proposed graph represents the entire 16-dimensional vector space for the quantization outputs.

To assign a metric to each transition, we study three types of transitions, a transition starting from HOS, a transition ending at EOS, and a transition from a state s_1 at c_i to another state s_2 at c_j. Specifically, the RD cost for a transition from HOS to a state s_1 at c_i is

$$g_{\text{head}}(c_i, s_1) = \sum_{k=i+1}^{15} c_k^2 + \lambda \cdot r(\mathit{ZerosLeft},\, \mathit{T1s},\, \mathit{TotalCoeffs}) + (c_i - u_i \cdot q)^2 + \lambda \cdot r_{s_1}(u_i), \quad (7)$$

where the first term is the distortion for quantizing coefficients from c_{15} to c_{i+1} to zero as the encoding starts with c_i, the second term gives the rate cost for coding the three parameters, the last two terms accord to the RD cost for quantizing c_i to u_i, and q is the quantization step size.

For a normal transition from state s_1 at c_i to state s_2 at c_j, $(15 \geq i > j \geq 0)$, the metric is defined as

$$g_{\text{n}} = \sum_{k=j+1}^{i-1} c_k^2 + \lambda \cdot r_{s_1}(i-j-1) + (c_j - u_j \cdot q)^2 + \lambda \cdot r_{s_2}(u_j), \quad (8)$$

where the first term computes the distortion for quantizing coefficients in the between to zero, the second term is the rate cost for coding the run with $r_{s_1}(i-j-1)$ given by the run coding table at state s_1, the last two terms are the RD cost for quantizing c_j to u_j with $r_{s_2}(u_j)$ determined by the level coding table at state s_2.

Finally, for a transition from a state at the column of c_j to EOS, the RD cost is

$$g_{\text{end}}(c_j) = \sum_{k=0}^{j-1} c_k^2, \quad (9)$$

which accords to the distortion for quantizing the remaining coefficients from c_{j-1} to c_0 to zero.

By examining details of CAVLC, it is not hard to see that the accumulated metric along any path leads to the same value as evaluating the RD cost in (6) for the corresponding output sequence. Thus, Viterbi algorithm is applicable to find the path with the minimize RD cost, and the obtain path gives the quantization output sequence to solve (6).

In practice, the number of parallel transitions from a state s_1 to a state s_2 can be much less than the $(u_{\text{high}} - u_{\text{low}} + 1)$ to reduce the complexity because the distortion is a quadratic function of the quantization output. Simulation results also show that it is sufficient to compute as few as 4 parallel transitions. Thus the complexity is reduced to a fairly low level.

IV. Experimental Results

The proposed joint optimization method is implemented based on the H.264 reference software Jm82[12]. Simulations have been conducted over a range of typical video sequences. Figure 3 illustrates the RD performance of four methods for coding various a video sequence. The video quality is measured by PSNR, which is defined as

$$\mathit{PSNR} = 10 \log_{10} \frac{255^2}{\mathit{MSE}},$$

where MSE is the mean square error. Compared to the method in [5] with the baseline profile, the proposed method significantly reduces the coding rate while maintaining the same quality. Compared to the method in [5] with the main profile CABAC, the proposed method results in a codec that has slightly better coding performance but enjoys a much faster decoding process.

The right panel of figure 3 shows the relative rate savings of the three optimization methods over the H.264 codec without any RD optimization. Given two methods A and B the rate saving of A relative to B is defined as [5],

$$S(\text{PSNR}) = 100 \cdot \frac{R_{\text{A}}(\text{PSNR}) - R_{\text{B}}(\text{PSNR})}{R_{\text{A}}(\text{PSNR})} \%,$$

where $R_{\text{A}}(\text{PSNR})$ and $R_{\text{B}}(\text{PSNR})$ are the rate with given PSNR for methods A and B, respectively. It is shown that with the same coding setting the proposed RD optimization method achieves $20 \sim 25\%$ rate gain over the codec without RD optimization, while the baseline-based method in [5] has a gain of $10 \sim 15\%$.

Hybrid video compression generally implies high correlation among frames due to the application of inter-frame prediction. The method proposed in this paper optimizes the coding performance for each individual frame. Figure 4 shows the simulation results of the relative rate savings, which is obtained by averaging over various numbers of coding frames. Clearly, the relative rate savings decreases as N increases due to the error accumulation. However, compare the results for the proposed method in the left panel and the result for the method in [5] in the right panel. The proposed method show a constant gain, indicating a positive effect of the proposed optimization method on the RD performance for coding the whole sequence.

V. Conclusion

In this paper, we have proposed a framework for jointly designing motion compensation, quantization and entropy coding in the hybrid coding structure of H.264. The core algorithm, i.e., a graph-based SDQ algorithm, was proven to achieve the optimal soft decision quantization for a block with given motion compensation and the quantization step size in the sense of minimizing the actual RD cost. The proposed method achieves a significant compression gain, as up to 25% rate reduction at the same PSNR when compared to the reference codec.

In general, the SDQ-based joint optimization framework is applicable to any coding method with a hybrid structure. Although this paper was focused on its application to H.264, it can be applied to other hybrid coding standards by developing algorithms, particularly the SDQ algorithm, accordingly.

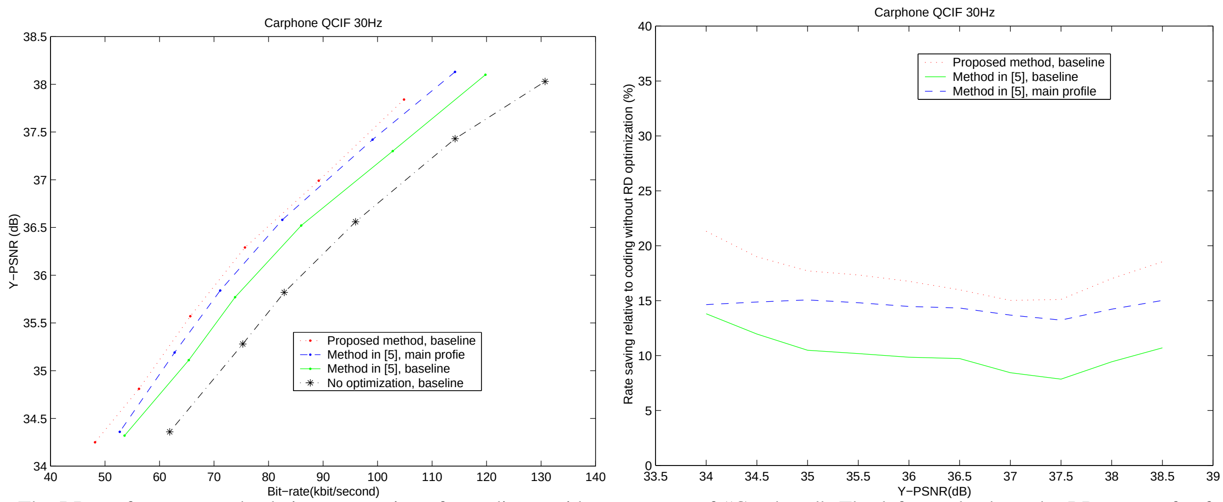

Fig. 3. The RD performance and relative rate savings for coding a video sequence of "Carphone". The left panels show the RD curves for four coding methods. In the right panels, the relative rate savings over one common method, i.e., the method without RD optimization, are presented for the three RD optimization methods.

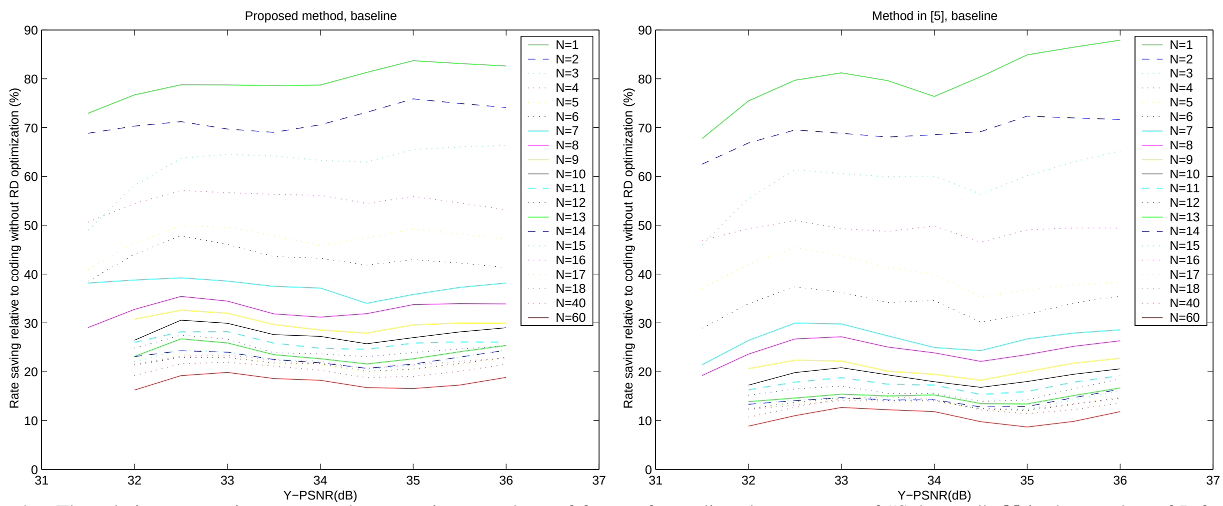

Fig. 4. The relative rate savings averaged over various numbers of frames for coding the sequence of "Salesman". N is the number of P-frames.

Acknowledgment

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grants RGPIN203035-98 and RGPIN203035-02 and under Collaborative Research and Development Grant, by the Premier's Research Excellence Award, by the Canada Foundation for Innovation, by the Ontario Distinguished Researcher Award, and by the Canada Research Chairs Program.

References

[1] T. Berger, *Rate Distortion Theory-A Mathematical Basis for Data Compression*, Englewood Cliffs, NJ: Prentice-Hall, 1971.

[2] T. Wiegand, G. J. Sullivan and A. Luthra, "Draft ITU-T Rec. H.264/ISO/IEC 14496-10 AVC," JVT of ISO/IEC MPEG and ITU-T VCEG, Doc. JVT-G050r1, 2003.

[3] E.-h. Yang, X. Yu, "Rate Distortion Optimization for H.264 Video Coding: A General Framework and Algorithms", submitted to IEEE Transactions on Image Processing.

[4] E.-h. Yang, Z. Zhang, and T. Berger, "Fixed-slope Universal Lossy Data Compression," *IEEE Transactions on Information Theory*, Vol. 43, No. 5, pp.1465-1476, September 1997.

[5] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-Constrained Coder Control and Comparison of Video Coding Standards", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 7, pp.688-703, July 2003.

[6] E.-h. Yang, S.-y. Shen, "Distortion program-size complexity with respect to a fidelity criterion and rate distortion function," *IEEE Transactions on Information Theory*, Vol. IT-39, pp. 288–292, 1993.

[7] M. Crouse and K. Ramchandran, "Joint Thresholding and Quantizer Selection for Transform Image Coding: Entropy Constrained Analysis and Applications to Baseline JPEG," *IEEE Trans. Image Processing*, vol. 6, pp. 285-297, Feb. 1997.

[8] E.-h. Yang and L. Wang, "Joint Optimization of Run-length Coding, Huffman Coding and Quantization table with Complete baseline JPEG Decoder Compatibility," U.S. patent application 2004.

[9] E.-h. Yang, X. Yu, "Optimal Soft Decision Quantization Design for H.264", *Proc. of the 9th Canadian Workshop on Information Theory (CWIT'2005)*, pp.223-226, Montreal, Quebec, June 2005.

[10] K. Ramchandran, A. Ortega, M. Vetterli, "Bit Allocation for Dependent Quantization with Applications to Multiresolution and MPEG Video Coders", *IEEE Transactions on Image Processing*, Vol. 3, Issue 5, pp.533-545, Sept. 1994.

[11] J. Wen, M. Luttrell, and J. Villasenor, "Trellis-based R-D Optimal Quantization in H.263+", *IEEE Transaction on Image Processing*, Vol. 9, No. 8, pp.1431-1434, August 2000.

[12] HHI, H.264 reference software, http://bs.hhi.de/ suehring/tml/