An Efficient Mode Selection Method of Macroblock in a GOB of H.263 Encoder

Taewon Hur*, Jeongwoo Kim*, Haesoo Kim**, Juneho Yi*, and Keunyoung Lee*

*Department of Electronics Engineering, Sungkyunkwan University
300, Chunchun-dong, Jangan-gu, Suwon, Kyunggi-do, Korea
E-mail: htw@mickey.skku.ac.kr FAX: +82-331-290-7180

**Department of Control and Instrumentation Engineering, Anyang Technical College
San 39-1, Anyang3-dong, Manan-gu, Anyang, Kyunggi-do, Korea
E-mail: dial8764@aytc.anyang-c.ac.kr FAX: +82-343-42-4400

Abstract

This research features efficient mode selection of macroblock in a GOB (Group of Blocks) of H.263. We propose a novel method based on the Viterbi algorithm that is a minimum-cost search technique. In H.263 standard, mode selection strategies use prediction error or energy of macroblocks and do not consider the amount of bits generated. We consider both prediction error and the amount of bitsgenerated. We apply a Lagrangian cost function to these strategies. In order to determine the Lagrange multiplier (frame-to-frame multiplier and macroblock-to-macroblock multiplier), we have considered a variety of potential approaches. The instantaneous rate of the encoder is controlled by these values.

Experimental results show that PSNR gain of the proposed approach is greater than that of the H.263 test model, TMN5.

1. Introduction

Many block-based video compression techniques employ multi-mode methodologies to achieve efficient coding performance. In most video coding standards such as MPEG1 [1], MPEG2 [2], H.261 [3] and H.263 [4], an input frame is subdivided into unit regions called macroblock and a given macroblock is coded using one of several possible modes. An allowable set of modes is determined by the picture coding type. Users can specify mode selection strategies using bit-stream syntax and decode operations uniquely defined in each video standard. Therefore, we can improve coding efficiency by selecting an appropriate mode. In low bit rate coding, the mode selection has effects on successive coding results for a given target bit rate.

In most video standards, mode selection strategies use prediction error or energy of macroblocks. These methods do not consider the amount of bits generated. The amount of bits frequently exceeds the given bit rate. It is a severe problem for low bit rate coding. Thus, we must consider not only the amount of bits generated but also prediction error with the energy of macroblocks for mode selection. For this purpose, we apply a Lagrangian cost function to these strategies [5].

It is more advantageous to consider mode selection in a GOB rather than in an entire frame. The problem is then finding a combination of modes that minimizes the sum of cost function for a given GOB. We need to compute an optimal path through the trellis which is obtained by a mode selection in a given GOB. The number of paths that need to be searched increases exponentially as the number of modes gets large. Therefore, an efficient searching algorithm is needed. We use the Viterbi algorithm to compute the optimal path through the trellis. The algorithm searches for various combinations of modes until modes are not changed in a GOB. However, this algorithm needs to be iterated several times, resulting in coding delay.

Advanced prediction is an additional capability which enforces overlapped block motion compensation. It permits the use of four motion vectors per macroblock. For experiments, we include the following standards and optional macroblock modes: intra (INTRA), inter with one motion vector (INTER), inter with four motion vectors (INTER4V), and not coded (SKIP).

In this paper, we propose an efficient mode selection method in a GOB. The method searches a mode combination in restricted regions and gives a path through trellis. Though this is not an optimal path, the proposed method is computationally less expensive than other methods by approximating optimal path without any iterations. When the amount of bits generated and distortion depend on adjacent macroblocks, these dependencies are restricted to adjacent macroblocks and paths are searched only in these macroblocks. These dependencies do not propagate to subsequent macroblocks and selected modes are not changed. The search algorithm is executed once in a GOB without any iterations. Using these strategies, an efficient mode selection in a GOB is achieved.

This paper is organized as follows. In Section 2, we formulate a Lagrangian cost function in the available modes in H.263. Section 3 presents a brief overview of the Viterbi algorithm. In Sections 4, we describe the proposed mode selection method and a procedure for updating the lagrange multiplier. Finally, Section 5 reports experimental results.

2. Lagrangian Cost Function

Let us consider an image region which is partitioned into a group of macroblocks (GOB), denoted by $\chi=(X_1,\cdots,X_N)$. Each macroblock in χ can be coded using only one of K possible modes given by the set $\iota=\{I_1,\cdots,I_K\}$. Let $M_i \in \iota$ be the modes to code macroblock X_i. For a given GOB, the modes assigned to the elements in χ are given by the N-tuple, $\mu=(M_1,\cdots,M_N)\in\iota^N$. The problem of finding a combination of modes that minimizes the distortion for a given GOB and bit rate constraint R_C can be formulated as follows:

$$\min_{\mu} D(\chi,\mu) \quad \textit{subject to } R(\chi,\mu)\le R_C \tag{1}$$

Here, $D(\chi,\mu)$ and $R(\chi,\mu)$ represent the total distortion and bit rate, respectively, resulting from the quantization of the GOB χ with a particular mode combination μ. This can also be expressed by a Lagrangian cost function that is composed of distortion measure and the amount of bits generated:

$$\min_{\mu} \sum_{i=1}^{N} J(X_i, \mu) \quad (2)$$

$J(X_i, \mu)$ is the Lagrangian cost function for macroblock X_i, and is given as follows:

$$J(X_i, \mu) = D(X_i, \mu) + \lambda \cdot R(X_i, \mu) \quad (3)$$

In the cases of SKIP or INTRA mode, the Lagrangian cost function associated with each macroblock can be computed irrespective of the operational modes in other macroblocks. For INTER mode, the amount of bits generated R of a particular macroblock is influenced by the immediately preceding macroblock. For INTER4V mode, bit rate and distortion depend not only upon the selected mode of the immediately preceding macroblock but also on the immediately ensuing macroblock. Also, Lagrangian cost functions depend on the modes of preceding and subsequent macroblocks. As a result, we can rewrite (3) at each possible mode as follows:

$$\begin{aligned} SKIP,\ INTRA\ mode &: J(X_i, \mu) = J(X_i, M_i) \\ INTER\ mode &: J(X_i, \mu) = J(X_i, M_{i-1}, M_i) \quad (4) \\ INTER4V\ mode &: J(X_i, \mu) = J(X_i, M_{i-1}, M_i, M_{i+1}) \end{aligned}$$

3. The Viterbi Algorithm

The Viterbi algorithm [6] was originally developed for error control codes and is an example of dynamic programming. Readers refer to Forney [7] for details of this algorithm. The technique was first applied to source coding by Viterbi and Omura [8]. The key idea of the Viterbi algorithm is the following. The minimum distortion path from time 0 to time n is an extension of one of the minimum distortion paths to a node at time $n-1$. In order to find the best possible path of length L, we compute the best path for each state at each time by finding the best extension from the previous states into the current state. We perform this at each time until time L. This idea is known formally as the optimality principle of dynamic programming.

A key fact is that we have a block code with block length L, but we do not have to search a block code of 2^L by computing all 2^L possible distortions. Instead we compute a sequence of K distortions, where K is the number of states, and retain the running distortion for each state along with the accumulated distortions into each state and the accumulated channel symbol sequence. In particular, the time complexity grows with the number of states, not with L.

The algorithm is implemented by keeping track of the following at each level of the trellis: (1) the best path into each node, and (2) cumulative distortions up to that node. The principle of the optimality means that knowing the best path at the current node is equivalent to knowing its best possible predecessor states. When the final nodes are reached, the path map provides the smallest path distortion at the final depth. The Viterbi Algorithm can be described as follows:

Step 0. Given a collection of states $S = \sigma_0, \ldots, \sigma_K$, a starting state s^*, a decoder $\beta(u, s)$ producing a reproduction given channel symbol u in state s, a pre-letter distortion measure d, a source input vector $x_0, \ldots, x_{L-1}$, let $D_j(k)$ denote the distortion for state k at time j. Set $D_0(s^*) = 0$, $D_0(s) = \infty$ for $s \neq s^*$. set $l = 1$.

step 1. For each current state s, find

$$D_l(s) = \min_{\sigma}(D_{l-1}(\sigma) + \min_{u: f(u,\sigma)=s} d(x_l, \beta(u, \sigma)))$$

s' denoted the minimizing value of the previous state and u' the minimizing value of the channel symbol u. The best path into the current state s passes through the previous state s' and is forced by channel symbol u'.

Given the path map $u^{l-1}(s')$ to the best previous state s', an extended path map $u^l(s) = (u^{l-1}(s'), u')$, is formed that gives the best (minimum distortion) path through the trellis from the initial state into the current state s at level l. In other words, for each current state, we find the distortion resulting from extending the K best paths into the previous states into the current state. The best path is picked and the distortion and cumulative path indices are recorded.

Step 2. If $l < L-1$, set $l+1 \rightarrow l$ and go to step 1. If $l = L-1$, pick the final state s_f yielding the minimum of the K values $D_L(s)$. The optimal path map is then $u^L(s_f)$.

4. Proposed Mode Selection Algorithm

In this section, we describe a novel algorithm that effeciently selects modes. As shown in Fig. 1, the selected mode in a GOB is equal to an optimal path which has minimum cost through the trellis. The cost function is given by (4) at each node. We can use the Viterbi algorithm to search for a combination of modes in a GOB. In Fig. 1, a possible mode is SKIP, INTRA, INTER or INTRA4V and eleven nodes are macroblocks in a GOB of QCIF (Quarter Common Intermediate Format) picture format. The number of possible paths in this trellis is 4^{11}. Thus, it is not practical to consider all possible paths because it takes too much time to determine all combinations of modes in a GOB. Therefore, we employ the Viterbi algorithm to impose constraints on possible paths. This is illustrated in Fig 2.

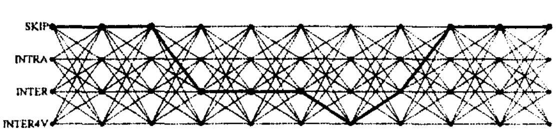


Fig. 1. Path of selected mode in a GOB

In this scheme, the cost function of SKIP or INTRA mode is determined independently of adjacent macroblocks. For INTER mode, the cost function is influenced by the immediately preceding macroblock. In the case of INTER4V mode, the terms representing bit rate and distortion depend not only on the immediately preceding macroblock, but also on the immediately ensuing macroblock. This ensuing macroblock mode is not yet determined. Therefore, we must presume that the next macroblock has one of the possible modes. As a result, when the preceding macroblock mode is SKIP, we can obtain seven Lagrangian cost functions which are denoted by J_{SKIP}, J_{INTRA}, J_{INTER}, $J^{SKIP-SKIP}_{INTER4V}$, $J^{SKIP-INTRA}_{INTER4V}$, $J^{SKIP-INTER}_{INTER4V}$ and $J^{SKIP-INTER4V}_{INTER4V}$. Similarly, we have 7 lagrangian cost function in each case that the preceding macroblock mode is INTRA, INTER and INTER4V. We can select a mode that has minimum cost in i^{th} macroblock. This strategy has an advantage that a mode combination is determined in one pass. The algorithm doesn't iterate until transitional cost doesn't change. It is computationally less expensive than any other method. It may not be an optimal path, however, an efficient mode selection in a GOB is achieved.

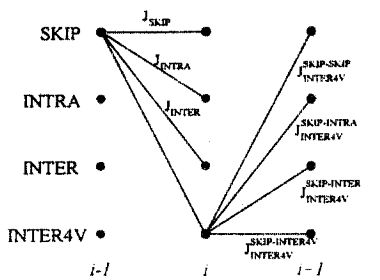


Fig. 2. Constrained search of paths

A final critical consideration with regard to mode selection is the determination of the Lagrange multiplier λ. The frame-to-frame multiplier and the macroblock-to-macroblock multiplier are updated using the equations (5) and (6), respectively. When the amount of bits generated is larger than that of target bits, we use a big frame-to-frame multiplier, λ. In the opposite case, a small λ is used. This Lagrangian multiplier is given as follows:

$$\lambda_{i+1} = \lambda_i \cdot \left(\frac{R_k}{R_c} \right) \qquad (5)$$

where λ_i is for the current frame, and λ_{i+1} for the next frame. R_c is the amount of target bits for the current frame and R_k is the amount of actual bits needed when the current frame is coded.

The macroblock-to-macroblock multiplier λ is determined according to the energy of macroblock. We compute this λ by multiplying 10% of the maximum energy of macroblock with the frame-to-frame multiplier obtained. This Lagrange multiplier is given as follows:

$$\lambda_{ij} = \lambda_i \cdot (1 + SA_j) \quad j \in GOB \qquad (6)$$

λ_{ij} is Lagrange multiplier for cost function of the j^{th} macroblock in the i^{th} frame. SA_j is scaled by ten times as much as the maximum energy and represents the energy of the j^{th} macroblock in a GOB. The scaled version of macroblock activity SA_j is defined as follows:

$$SA_j = \frac{(A_j - A_{avg})}{10 \cdot A_{max}} \qquad (7)$$

where A_j is the energy of the j^{th} macroblock in a GOB, A_{avg} is the average of A_j $(j = 1 \cdots 11)$ and A_{max} is the maximum energy of macroblocks. Thus the activity of macroblock can be stated as follows:

$$A = \sum_{i=1}^{16} \sum_{j=1}^{16} |P(i,j) - MB_{mean}| \qquad (8)$$

$$MB_{mean} = \left(\sum_{i=1}^{16} \sum_{j=1}^{16} P(i,j) \right) / 256 \qquad (9)$$

$P(i,j)$ is a pixel value in a macroblock, and MB_{mean} is the average of $P(i,j)$'s and represents the mean of macroblock.

In the experiments, we use the following number as the initial Lagrange multiplier λ_0:

$$\lambda_0 = 0.6 \cdot \left(\frac{15000}{T_b} \right) \qquad (10)$$

where T_b is the target bit rate. This value is experimentally determined. It is useful to adopt a small initial value when the target bit rate is high. For a smaller λ, the Lagrangian cost function is more influenced by distortion. Therefore, for higher a target bit rate, we should more emphasize the distortion term.
The variation of the Lagrange multiplier λ is depicted in Fig. 3. This result is encoded at 20Kbps for "Suzie" sequence. The figure shows that λ increases as the amount of bits generated large.

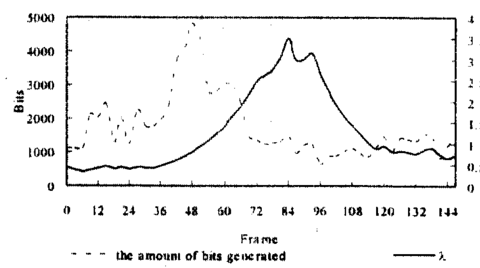


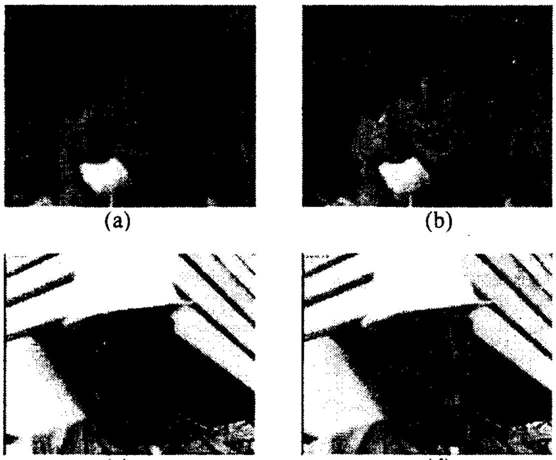
Fig. 3. Variation of λ as the amount of bits generated gets large.

5. Experimental Results

For the experiments, a GOB is defined as a single, horizontal macroblock stripe across a given frame. A QCIF (176×144) image consists of nine macroblock stripes and each stripe contains 11 macroblocks. The performance of the proposed method is compared to that of TMN5, which is the video codec test model for the H.263 standard. "Suzie"(150 frames) and "Foreman"(300 frames) sequences are used for the experiments. Experimental results for the proposed mode selection strategy are provided in Figs. 4-7 for the H.263 video coding standard. Fig 4(a) and (c) shows a reconstructed frame (117^{th}) using H.263 and Fig 4(b) and (d) using the proposed method. PSNR performance is illustrated in Fig. 5 and the average PSNR for a given rate is shown in Fig. 6. These results show that the proposed method is superior to TMN5. Fig. 7 illustrates probability of mode versus a given bit rate. This results shows that the proposed method is more dynamically selected at various bit rates.

6. Conclusion

We have presented an efficient algorithm for selecting operation modes in a block-based video coding system. Our goal is to determine a computationally efficient mode selection for a given GOB. Experimental results show that the proposed mode selection strategy performs better than TMN5 for all test sequences and all bit rates considered. The PSNR gains have improved up to 0.5[dB] for a given bit rate as compared to those of TMN5.



(a) (b) (c) (d)

Fig. 4. Reconstructed frames using (a) (c) H.263 and (b) (d) our method

References

[1] MPEG-1 IS, "Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media up to 1.5Mbit/s," ISO/IEC 11172-1,2,3, 1993.

[2] MPEG-2 DIS, "Information Technology - Coding of Moving Pictures and Associated Audio," ISO/IEC 13818-1,2,3, 1994.

[3] Draft ITU-T Recommendation H.261, "Video Codec for Audiovisual Services at p×64kbits/s." Aug. 1990.

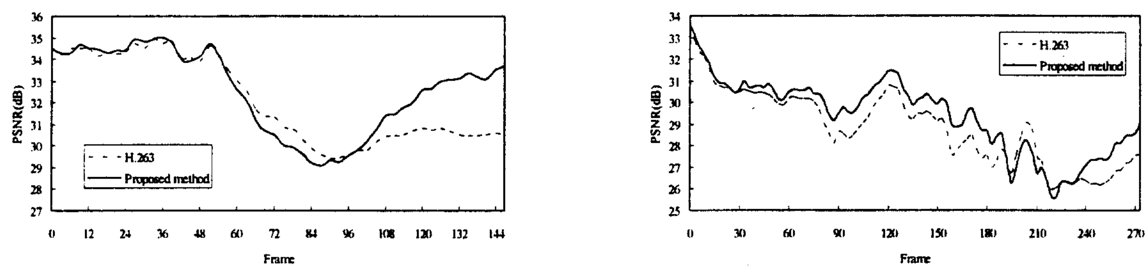
[4] Draft ITU-T Recommendation H.263, "Video Coding for Low Bit Rate Communication," May 1996.

[5] T. Wiegand, M. Lightstion, D. Mukherjee, T. G. Cambell, and S. K. Mitra, "Rate-Distortion Optimized Mode Selection for Very Low Bit Rate Video Coding and the Emerging H.263 Standard," IEEE Transactions on Circuits and Systems for Video Technology, vol 6. no. 2, pp. 182-190, Apr. 1996.

[6] Allen Gersho and Robert M. Gray, "Vector Quantization and Signal Compression," Kluwer Academic Publishers, pp. 566-568.

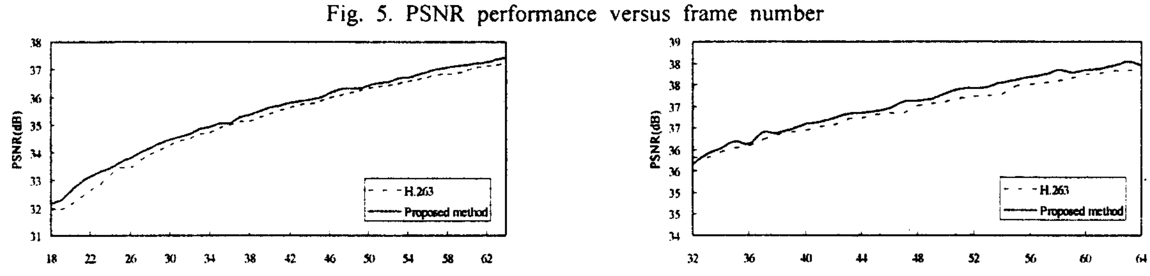
[7] G. D. Forney, "The Viterbi algorithm," Proceedings of IEEE, vol. 61, pp. 268-278, Mar. 1973.

[8] A. J. Viterbi and J. K. Omura, "Trellis encoding of memoryless discrete-time sources with a fidelity criterion," IEEE Transactions on Information Theory, vol. 20, pp. 325-332, May 1974.



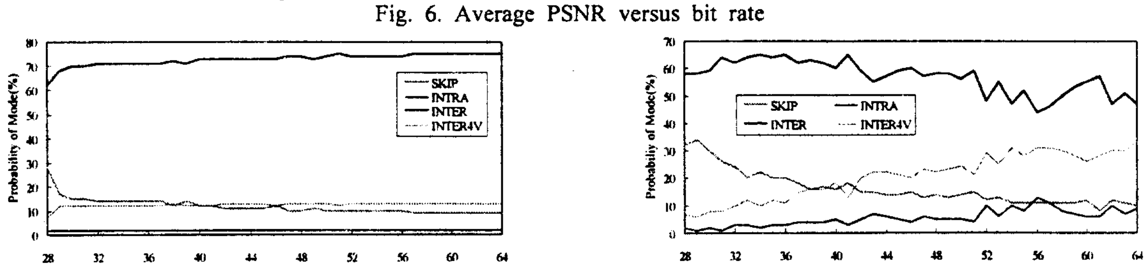
(a) "Suzie" sequence (b) "Foreman" sequence

Fig. 5. PSNR performance versus frame number



(a) "Suzie" sequence (b) "Foreman" sequence

Fig. 6. Average PSNR versus bit rate



(a) H.263 (b) Proposed method

Fig. 7. Probability of a mode versus bit rate for the "Suzie" sequence