An Enhanced Rate Control Based on Mode Decision and Early Motion Estimation for H.264/AVC

Siavash Es'haghi, Hassan Farsi

Abstract**—** The H.264/AVC video coding standard delivers a significantly better performance compared to previous standards, supporting higher quality video over lower bit rate channels. Rate control plays an important role in real-time video communication applications using H.264/AVC. An important step in many existing rate control algorithms is to determine the target bits for each P frame. This paper aims in improving video distortion by allocating more bits to frames with higher complexity and fewer bits to low complexity frames. In this work, the distribution of Macro Block (MB) modes in a frame is considered as a measure of its complexity. Also, an early motion estimation approach is introduced and used for complexity estimation. The bit budget is then allocated to frames according to their complexity and buffer status. Simulation results show that the proposed method effectively improves the PSNR average and meets the target bit rate more closely. In addition the proposed technique is less complex than other existing frame layer bit allocation schemes that are based on frame complexity.

Keywords**—**H.264, Rate Control, Bit Allocation, Frame Complexity, Mode Decision, Motion Estimation

I. INTRODUCTION

DIGITAL video compression techniques have played an important role in the world of telecommunication and multimedia systems where bandwidth is still a valuable commodity. Hence, video coding techniques are of prime importance for reducing the amount of information needed for a picture sequence without losing much of its quality, judged by the human viewers.

International study groups, VCEG (Video Coding Experts Group) of ITU-T (International Telecommunication Union - Telecommunication sector) and MPEG (Moving Picture Experts Group) of ISO/IEC, have researched the video coding techniques for various applications of moving pictures since the early 1990s. Since then, ITU-T developed H.261 as the first video coding standard for videoconferencing application. MPEG-1 video coding standard was accomplished for storage in compact disk and MPEG-2 [3] (ITU-T adopted it as H.262) standard for digital TV and HDTV as extension of MPEG-1[7]. Also, for covering the very wide range of applications such as shaped regions of video objects as well as rectangular pictures, MPEG-4 part 2 [6] standard was developed. This includes also natural and synthetic video / audio combinations with interactivity built in. On the other hand, ITU-T developed H.263 [5] in order to improve the compression performance of H.261, and the base coding model of H.263 was adopted as the core of some parts in MPEG-4 part 2. MPEG 1, 2 and 4 also cover audio coding.

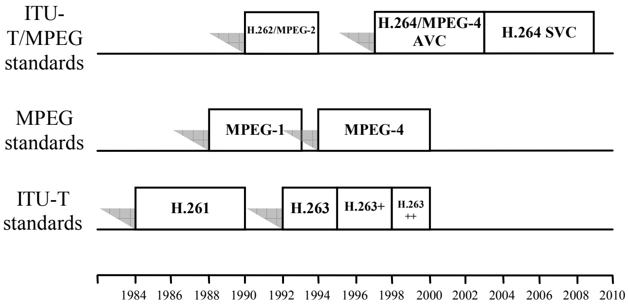


Fig. 1 Video Coding Standards Development History

In order to provide better compression of video compared to previous standards, H.264 / MPEG-4 part 10 [1], [2] (H.264/AVC) video coding standard was recently developed by the JVT (Joint Video Team) consisting of experts from VCEG and MPEG. H.264/AVC fulfills significant coding efficiency, simple syntax specifications, and seamless integration of video coding into all current protocols and multiplex architectures. Thus H.264/AVC can support various applications like video broadcasting, video streaming, video conferencing over fixed and wireless networks and over different transport protocols.

The Scalable Video Coding extension (SVC) [8] of the H.264/AVC is the latest amendment for this successful specification, Fig. 1. SVC allows partial transmission and decoding of a bit stream. The resulting decoded video has lower temporal or spatial resolution or reduced fidelity while retaining a reconstruction quality that is close to that achieved using the existing single-layer H.264/AVC design with the same quantity of data as in the partial bit stream. SVC provides network-friendly scalability at a bit stream level with a moderate increase in decoder complexity relative to single

Manuscript received February, 2010: Revised version received May 22, 2010.

S. Es'haghi is a lecturer in Islamic Azad University, Birjand, Iran. (Phone: +98-561-4342261, Fax: +98-561-4342260, Email:esshaghi@gmail.com).

H. Farsi is an assistant professor in Electronics and communications group, University of Birjand, Birjand Iran. (Email: hfarsi@birjand.ac.ir).

layer H.264/AVC. Furthermore, it provides the functionality of lossless rewriting of fidelity-scalable SVC bit streams to single-layer H.264/AVC bit streams. The SVC extension of H.264/AVC is suitable for video conferencing as well as for mobile to high-definition broadcast and professional editing applications.

A standard defines a coded representation, syntax, which describes the video in a compressed form. In other words, a standard only specifies the output of the encoder, i.e., the input of the decoder, instead of the codec itself. The scope of the standardization is illustrated in Fig. 2.

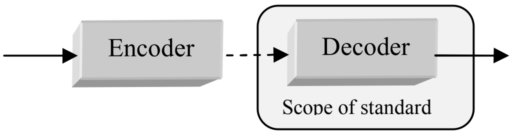


Fig. 2 Scope of video coding standardization

H.264/AVC video coding standard as previous standards takes advantage of block-based hybrid coding scheme which refers to the combination of motion-compensated prediction and transform coding, as illustrated in Fig. 3. A picture is partitioned into fixed-size macroblocks that each covers a rectangular picture area of 16x16 samples of the luma component. This partitioning into macroblocks has been adopted into all previous video coding standards since H.261 [4]. Macroblocks are the basic building blocks of the standard for which the decoding process is specified.

Motion estimation (ME) of a macroblock involves finding a region in a reference frame that closely matches the current macroblock. The selected best matching region in the reference frame is subtracted from the current macroblock to produce a residual macroblock (motion compensation) that is encoded and transmitted together with a motion vector describing the position of the best matching region (relative to the current macroblock position). The H.264 encoder performs the ME of the variable macroblock sizes and then selects the best mode among all of the possible macroblock modes (mode decision).

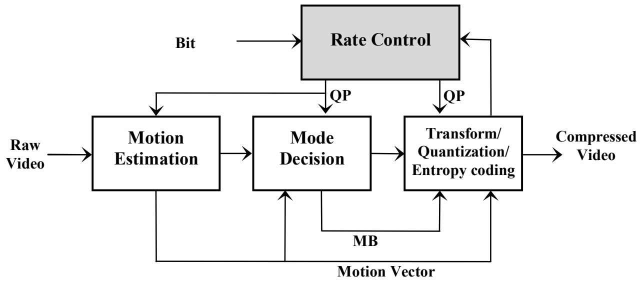


Fig. 3 block-based hybrid coding scheme of H.264

Rate control is a key component of an efficient encoder which regulates varying bit rate characteristics of a coded video bit stream in order to produce high quality decoded frame at a given target bit rate. In general, rate control first allocates a target number of bits to each video frame and then determines the Quantization parameter (QP) to meet the target as close as possible. The solution is heavily dependent upon rate-distortion (R-D) models. Like VM-18, H.264 rate control scheme [9] is based on a quadratic rate-quantization (R-Q) model [11]. Rate control is not a part of the H.264 standard, but the standard group has issued a non-normative guidance to aid in implementation.

II. REVIEW OF H.264/AVC

The intent of the H.264/AVC project was to create a standard capable of providing good video quality at substantially lower bit rates than previous standards (e.g. half or less the bit rate of MPEG-2, H.263, or MPEG-4 Part 2), without increasing the complexity of design so much that it would be impractical or excessively expensive to implement.

A. Applications

H.264/AVC standard is designed to provide a technical solution appropriate for a broad range of applications [1], at least including:

- Broadcast over cable, satellite, cable modem, DSL, terrestrial.
- Interactive or serial storage on optical and magnetic devices, DVD, etc.
- Conversational services over ISDN, Ethernet, LAN, DSL, wireless and mobile networks, modems.
- Video-on-demand or multimedia streaming services over cable modem, DSL, ISDN, LAN, wireless networks.
- Multimedia messaging services over DSL, ISDN.

B. Coding Scheme and New Features

The H.264/AVC design covers a Video Coding Layer (VCL), which efficiently represents the video content, and a Network Abstraction Layer (NAL), which formats the VCL representation of the video and provides header information in a manner appropriate for conveyance by particular transport layers or storage media.

H.264/AVC video coding layer has the same basic functional elements as previous standards (MPEG-1, MPEG-2, MPEG-4 part 2, H.261, H.263) [7], i.e., transform for reduction of spatial correlation, quantization for bit rate control, motion compensated prediction for reduction of temporal correlation, entropy encoding for reduction of statistical correlation, as depicted in Fig. 4. However, in order to fulfill better coding performance, the important changes in H.264 occur in the details of each functional element by including intra-picture prediction, a new 4x4 integer transform, multiple reference pictures, variable block sizes and a quarter precision for motion compensation, a de-blocking filter, and improved entropy coding.

There is no single coding element in the VCL that provides the majority of the dramatic improvement in compression efficiency, in relation to prior video coding standards. Rather, it is the plurality of smaller improvements that add up to the significant gain.

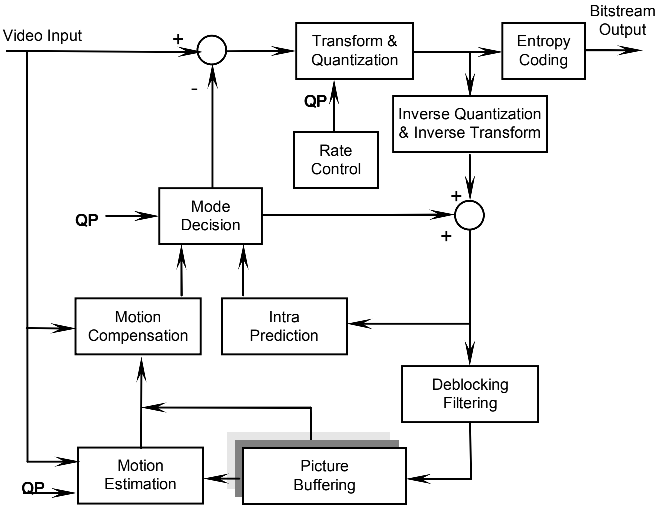


Fig. 4 H.264 Encoder

Improved coding efficiency comes at the expense of added complexity to the coder/decoder. H.264 utilizes some methods to reduce the implementation complexity. Multiplier-free integer transform is introduced. Multiplication operation for the exact transform is combined with the multiplication of quantization.

The noisy channel conditions like the wireless networks obstruct the perfect reception of coded video bit stream in the decoder. Incorrect decoding by the lost data degrades the subjective picture quality and propagates to the subsequent blocks or pictures. So, H.264 utilizes some methods to exploit error resilience to network noise. The parameter setting, flexible macroblock ordering, switched slice, redundant slice methods are added to the data partitioning, used in previous standards.

For the particular applications, H.264 defines the Profiles and Levels specifying restrictions on bit streams like some of the previous video standards. A *profile* defines a set of coding tools or algorithms that can be used in generating a conforming bit-stream, whereas a *level* places constraints on certain key parameters of the bitstream.

C. Hypothetical Reference Decoder (HRD)

One of the key benefits provided by a standard is the assurance that all the decoders compliant with the standard will be able to decode a compliant compressed video. To achieve that it is not sufficient to just provide a description of the coding algorithm. It is also important in a real time system to specify how bits are fed to a decoder and how the decoded pictures are removed from a decoder. Specifying input and output buffer models and developing an implementation independent model of a receiver achieves this. That receiver model is also called Hypothetical Reference Decoder (HRD). An encoder is not allowed to create a bit stream that cannot be decoded by the HRD. Hence, if in any receiver implementation the designer mimics the behavior of HRD, it is guaranteed to be able to decode all the compliant bit streams.

In H.264/AVC HRD specifies operation of two buffers: (i) Coded Picture Buffer (CPB) and (ii) Decoded Picture Buffer (DPB). CPB models the arrival and removal time of the coded bits. The HRD design is similar in spirit to what MPEG-2 had, but is more flexible in support of sending the video at a variety of bit rates without excessive delay. As unlike MPEG-2, in H.264/AVC, multiple frames can be used for reference, the reference frames can be located either in past or future arbitrarily in display order, the HRD also specifies a model of the decoded picture buffer management to ensure that excessive memory capacity is not needed in a decoder to store the pictures used as references.

III. RATE CONTROL

An encoder employs rate control as a way to regulate varying bit rate characteristics of the coded bit stream in order to produce high quality decoded frame at a given target bit rate. Rate control is thus a necessary part of an encoder, and has been widely studied in standards, like MPEG 2, MPEG 4, H.263, and so on [11], [13], [14], [15], [16]. Block-based hybrid video encoding schemes are inherently loss processes. They achieve compression not only by removing truly redundant information from the bit stream, but also by making small quality compromises in ways that are intended to be minimally perceptible. In particular, the quantization parameter (QP) regulates how much spatial detail is saved. When QP is very small, almost all that detail is retained. As QP is increased, some of that detail is aggregated so that the bit rate drops – but at the price of some increase in distortion and some loss of quality.

A rate control algorithm dynamically adjusts encoder parameters such as quantization parameter (QP) for the current frame according to the specified bit-rate and the statistics of the current frame, like the mean absolute difference (MAD) and the header bits of each macroblock, to achieve a target bit rate, Fig. 5. The rate control module in H.264 is more complex than the previous standards as well, since the statistics of the current frame is not available for the rate control of H.264 [9], [18]. Note that the quantization parameters are involved in both rate control and Rate Distortion Optimization (RDO) of H.264 while it is only involved in rate control of MPEG 2, MPEG 4 and H.263.

Rate control is not a part of the H.264 standard, but the standard group has issued non-normative guidance [17], [18]. A linear model is suggested to predict the current frame MAD as follows:

$$MAD\ (j) = a_1 \times MAD\ (j-1) + a_2 \qquad (1)$$

where MAD(j) is the predicted MAD for jth frame and MAD(j-1) is the actual MAD of (j-1)th frame, and a1 and a2 are two coefficients. The initial value of a1 and a2 are set to 1 and 0, respectively. They are updated by a linear regression method similar to that for the quadratic R-D model parameters estimation in MPEG-4 rate control after coding each basic unit.

Furthermore, a fluid flow traffic model is presented to compute the occupancy of virtual buffer $B_c(n_{i,j})$, before coding the jth frame in the ith GOP which generates b($n_{i,j}$) bits. Then we have

$$B_c(n_{i,j+1}) = B_c(n_{i,j}) + b(n_{i,j}) - \frac{u}{F_r} \qquad (2)$$

$$B_c(n_{1,1}) = 0$$

where u is the available channel bandwidth which can be either a variable bitrate (VBR) or a constant bitrate (CBR) case and F_r is the predefined frame rate.

The rate control algorithm which is discussed here is composed of two layers: group of pictures (GOP) layer rate control and frame layer rate control. Total number of remaining bits for all non-coded frames is computed in GOP layer. The frame layer rate control scheme consists of two stages: pre-encoding and post-encoding stages. The objective of pre-encoding stage is to compute quantization parameter for each frame. In the post-encoding stage, model parameters, such as those in (1) and the coefficients in the quadratic R-Q model, are updated.

A. GOP Layer Rate Control

In this layer, we need to compute the total number of remaining bits T_r for all non-coded frames in each GOP and to determine the starting quantization parameter of each GOP.

Without loss of generality, we assume that the GOP structure is or IPP…P with a length of N_{gop} frames. In the beginning of the ith GOP, the total number of bits allocated for the ith GOP is computed as follows:

$$T_r(n_{i,0}) = \frac{u}{F_r} \times N_{gop} - B_c(n_{i-1,N_{gop}}) \qquad (3)$$

Total number of remaining bits for all non-coded frames is updated frame by frame as follows:

$$T_r(n_{i,j}) = T_r(n_{i,j-1}) - b(n_{i,j-1}) \qquad (4)$$

The starting quantization parameter of the first GOP is a predefined quantization parameter and it is set by considering of available bits for each pixel of frame. For the other GOPs the starting QP is computed based on the average QP of the P frames in the previous GOP and is limited to have a difference of two with starting QP of the previous GOP.

B. Frame Layer Rate Control

The frame layer rate control scheme consists of two stages: pre-encoding and post-encoding.

1) Pre-encoding Stage

The objective of this stage is to compute quantization parameter for each frame. First, target bit for each frame is determined and then QP is computed according the quadratic model.

A target buffer level for the jth (j>2) frame in the ith GOP is predefined as

$$Tbl(n_{i,j+1}) = Tbl(n_{i,j}) - \frac{Tbl(n_{i,2})}{N_p - 1} \qquad (5)$$

Target buffer level for the first P in each GOP, $Tbl(n_{i,2})$, is the actual buffer occupancy after coding the first P.

Using linear tracking theory, the target bits allocated for the jth frame in the ith GOP is determined based on the target buffer level, the frame rate, the available channel bandwidth and the actual buffer occupancy as follows:

$$T_{buf}(n_{i,j}) = \frac{u}{F_r} + \gamma \times (Tbl(n_{i,j}) - B_c(n_{i,j})) \qquad (6)$$

where γ is a constant and its typical value is 0.5.

Furthermore, target bit for the frame based on remaining bits for the total number of N_r of non-coded frames in the GOP is computed as follow:

$$T = \frac{T_r(n_{i,j})}{N_r} \qquad (7)$$

Target bit for the frame is a weighted combination as

$$T(n_{i,j}) = \beta \times T + (1-\beta) \times T_{buf}(n_{i,j}) \qquad (8)$$

where β is a weighting factor and its value in our work is 0.7 to emphasize on the role of frame complexity in bit allocation.

Also a lower bound and an upper bound for the target bits of each frame are determined by considering the hypothetical reference decoder (HRD).

The drawback of this bit allocation scheme is that scene content of each frame is not considered and bits are distributed equally between frames.

After estimating the number of header bits for the frame, the number of bits for texture is given by

$$T_{texture}(n_{i,j}) = T(n_{i,j}) - T_{header}(n_{i,j}) \qquad (9)$$

Finally, QP is computed by using quadratic model and the predicted MAD as follows

$$T_{texture}(n_{i,j}) = c_1 \frac{MAD(j)}{Q_{step}} + c_2 \frac{MAD(j)}{Q_{step}^{\ 2}} \quad (10)$$

QP is computed based on Qstep. To maintain the smoothness of visual quality among successive frames, the computed QP for the jth frame is limited to change within a range [9]. The final quantization parameter is further bounded by 51 and 0. The quantization parameter is then used to perform RDO for each MB in the current frame. The above discussion is shown in Fig. 5.

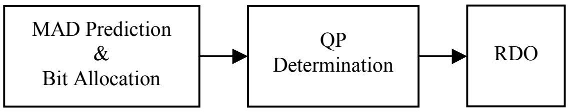


Fig. 5 Rate Control concept in H.264

2) Post-encoding Stage

After encoding a frame, the parameters of linear prediction model (1), as well as coefficients of quadratic model (10) are updated. A linear regression method similar to [11] and [16] is used to update these parameters. Meanwhile, the actual bits generated are added to the current buffer occupancy.

To ensure that the updated buffer occupancy is not too high, the frame skipping parameter N is set to zero and increased until the following buffer condition is satisfied:

$$B_c(n_{i,j+N}) < B_s \times 0.8 \quad (11)$$

where Bs is the buffer capacity.

In addition of weakness of bit allocation scheme in this algorithm, in some special cases such as scene changes the predicted MAD is inaccurate as well.

IV. MODE DECISION AND MOTION ESTIMATION

Motion compensated prediction is a powerful tool to reduce temporal redundancies between frames and is thus used extensively in video coding standards (i.e., H.261, H.263, MPEG-1 and MPEG-2) as a prediction technique for temporal differential pulse code modulation (DPCM) coding. The concept of motion compensation is based on the estimation of motion between video frames. Both prediction error and motion vectors are transmitted to the receiver. For a block in an inter frame, the rate-constrained motion estimation is first done to find the optimal motion vector by minimizing:

$$J = D + \lambda_{motion} \times R \quad (12)$$

D is the sum of absolute differences (*SAD*) between the original and predicted values. *R* represents the bits needed for signaling motion vector and reference frame index and λ_{motion} is Lagrange parameter and is a function of QP.

One of the main improvements of the present standard is the improved prediction process both for inter and intra. The accuracy of motion compensation is in units of one quarter of the distance between luma samples. In case the motion vector points to an integer-sample position, the prediction signal consists of the corresponding samples of the reference picture; otherwise the corresponding sample is obtained using interpolation to generate non-integer positions. The prediction values at half-sample positions are obtained by applying a one-dimensional 6-tap FIR filter horizontally and vertically. Prediction values at quarter-sample positions are generated by averaging samples at integer- and half-sample positions.

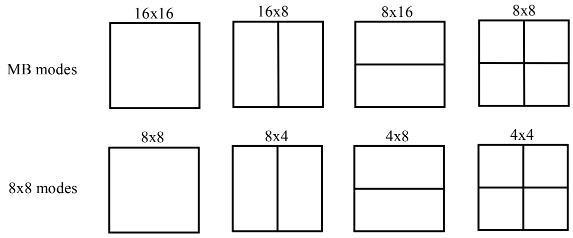


Fig. 6 Various block sizes in one macroblock in H.264/AVC

In addition to the intra macroblock coding types, various predictive or motion-compensated coding types are specified as P macroblock types. Each P macroblock type corresponds to a specific partition of the macroblock into the block shapes used for motion-compensated prediction. Partitions with luma block sizes of 16x16, 16x8, 8x16, and 8x8 samples are supported by the syntax. In case partitions with 8x8 samples are chosen, one additional syntax element for each 8x8 partition is transmitted. This syntax element specifies whether the corresponding 8x8 partition is further partitioned into partitions of 8x4, 4x8, or 4x4 luma samples. To select the best mode, RDO is employed such that for each MB, all the MB modes such as Inter 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4, Intra and SKIP modes are tried and the one that leads to the least rate-distortion (RD) cost is selected. This is to achieve the best trade-off of the rate and distortion performance. RD cost is calculated using Lagrangian function defined in [9] as:

$$RD\cos t = Distortion + \lambda_{\mathrm{mod}e} \times Rate \quad (13)$$

The best mode selection is highly dependent on $\lambda_{\mathrm{mod}e}$ value. With smaller $\lambda_{\mathrm{mod}e}$ value smaller macroblock size selection is more probable.

V. REVIEW OF EXISTING IMPROVED METHODS FOR RATE CONTROL AND BIT ALLOCATION

Rate control is a key component of an efficient encoder and consists of two steps: Bit Allocation and QP determination based on a quadratic rate-distortion model. The accuracy of the model is essential to achieve the target bits. Several approaches are proposed in literature to improve the rate-distortion model, as in [28]. Bit allocation plays an important role in rate control and is responsible for distributing bits among frames to produce high and constant quality decoded sequence. Current studies for frame-level bit allocation can roughly be classified into two categories: one is predominantly based on buffer status, as in [15] and [16], and the other uses a scene content complexity-based approach, as in [13] and [14]. In [15], the bit budget for each frame throughout the whole video sequence is nearly constant. In [16], the initial target bit number for each new P-frame is scaled based on the current buffer level to maintain buffer occupancy of about 50% after encoding each frame. Obviously, such bit allocation schemes are unable to achieve optimal performance because they do not match the non-stationary characteristics of video signals. The above discussed H.264 rate control [9], [18] falls into this category. In (7), the number of remaining bits T is calculated without considering frame content, and the bits are equally allocated to all non-coded frames. However, the remaining bits should be distributed to all non-coded frames according to their statistics and complexities.

For target bit allocation, it is important to distribute an adequate bit budget for each allocation unit with an identical perceptual quality. A feasible bit allocation strategy will allocate the available bits according to the content activities of each allocation unit [14]. Several models are proposed to estimate the target number of bits allocated to a video frame or an MB [13], [14], [19], [20], [21]. The quantization parameter can he employed to control the bit rate according to the content activities of a video sequence and the planned buffer fullness. For example, the quantization scale for an MB in MPEG TM5 [14] is decided by the normalized MB activity properly calculated from the minimum variance of the luminance frame-based or field-based sub block of 8x8 pixels in an MB as well as the planned buffer fullness. Jordi and Lei [21] presented a rate control method for operating typical DCT video codecs for video communications. One type of rate control approaches uses an explicit bit model to predict the number of compressed bits when a certain quantization scale is used, as the control schemes developed in [11]. To achieve an optimal bit allocation, some approaches allocate bits among video frames by a multi-pass encoding strategy [23], but they are more appropriate for off-line encoding due to the time consuming multi-pass procedures. In [13], an optimized method is proposed to assign target bits to each frame according to frame complexity, which is measured by frame energy. Much work estimate current frame complexity based on previous frames statistics as [24] and little work use current frame statistics as [10], [23].

In [26], Motion complexity is measured by the bits that allocated to the encoded frames. Therefore, the target bits allocated to the current frame is estimated by the previous frame bit consumption. However, there is a difference between the estimated bits and finally allocated bits and the estimation should be precise. Finally the complexity is computed as the ratio of the predicted bits to actual allocated bits to the previously encoded frames. A rate control algorithm which include both frame and macroblock level strategies is proposed in [27]. In frame layer, the frame MAD is used as frame complexity and quadratic rate-distortion model is used to compute QP. However in macroblock level and in the case of small MAD, it is proposed that the model is not accurate and QP computation is adjusted. In [24], the predicted MAD of current frame is used as a measure to represent the complexity of frame's motion. But it is difficult to obtain a good estimate of MAD for H.264 frame prior to the actual coding, due to the complex rate-distortion optimization (RDO) procedure, as described in [18]. At scene changes, the predicted MAD is far from the actual value and this method is completely failed.

Rate control in [28,29,30] proposed a rate-quantization model, which is designed from the rate distortion function of Laplacian distribution. In order to control fluctuation of picture quality within each image frame, macroblocks are classified into three different groups according to their characteristics. The quantization parameter for each macroblock is adjusted by the type of the previous macroblock. The Sobel mask is used to measure the complexity of each macroblock. A PSNR-based frame complexity estimation is presented in [23] to improve H.264/AVC rate control. MAD ratio is defined as the ratio of the predicted MAD of current frame to the average MAD of all previously encoded P frames in the GOP. The PSNR of current frame is calculated based on the previous reconstructed frame and the current original frame as in the case of dropping the current frame. PSNR drop of the current frame is the difference between the PSNR of the current frame and previous frame PSNR. Then the ratio of the PSNR drop to the average of PSNR drops of all the previous frames presents the frame complexity. Although this approach uses current frame statistics for representing the frame complexity, but is not accurate due to not performing motion estimation. Furthermore, it has complex computational process.

VI. OUR PROPOSED APPROACH

The complexity of motion contents refers to the moving picture contents of two consecutive frames in a video sequence. If the objects in a frame are static, or a frame can be best matched to the previous frame with very low difference, then such a frame is a low motion frame. If there is a scene change between two consecutive frames, or a moving object is unable to find a best match in the previous frame, then such a frame is a high motion frame. Generally bit allocation obeys the following rule: more bits are allocated to high motion frames, while fewer bits are allocated to low motion frames. Furthermore, it has complex computational process.

A. Frame complexity estimation based on mode decision

If we encode all P frames in a sequence with same encoding parameters and the decoded frames have constant PSNR then the frames bits will be in accordance with the complexity of frame's motion contents and the allocated bits can be a good measure to represent the complexity of frame's motion. Fig. 1 shows the bit allocation of the sequence "Foreman" divided by PSNR of each frame in the case of disabling rate control. This curve is used as a reference for representing the complexity of motion contents.

There are several possible macro-block sizes in H.264/AVC such as 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, and 4x4. The H.264 performs RDO to select the best macroblock mode among all of block modes. Skip and larger macro-block sizes are more possible for low motion frames while smaller macro-block sizes and intra modes are more probable in high motion frames.

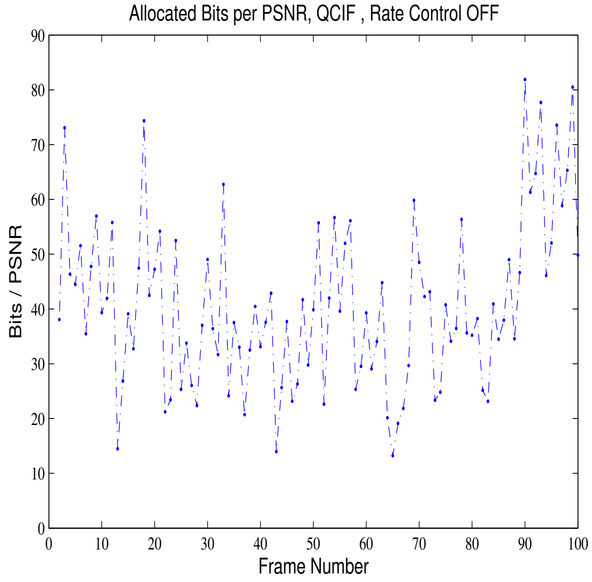


Fig. 1 Bits allocated to 'Claire' sequence divided by each frame PSNR in the case of disabling rate control

Therefore, we can estimate frame complexity by considering different modes in the frame as

$$C_{frame} = \frac{\sum w_i \times N_{\mathrm{mode}(i)}}{w_{skip} \times N_{skip}} \tag{13}$$

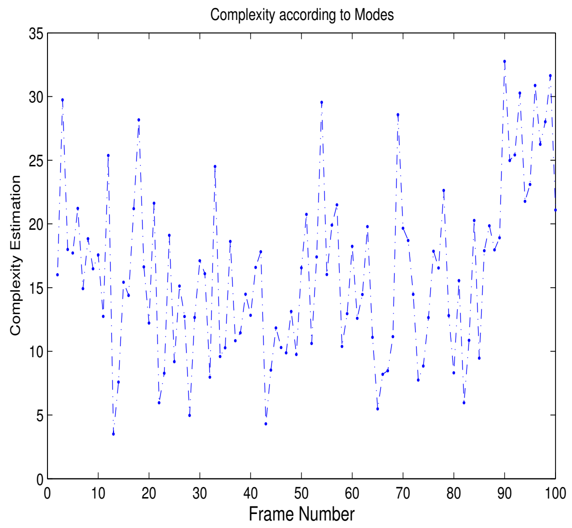


Fig. 2 Complexity estimation by counting different modes of each frame

where $N_{\mathrm{mode}(i)}$ (i=1,2,…,7) and N_{skip} are the percents of macroblocks modes of Inter 16x16, 16x8 plus 8x16, 8x8, 8x4 plus 4x8, 4x4, Intra16x16, Intra4x4 and skip macroblocks and submacroblocks modes, respectively and w_i (i=1,2,…,7) and w_{skip} are corresponding weighting factors. Intra macroblocks and smaller macroblock sizes represent more complexity and we assign larger weighting factors to them. So we choose these weighting factors as $w_1<w_2<\ldots<w_7$. The above frame complexity is depicted in Figure 2 which is highly correlated with Figure 1. We conclude that the proposed frame complexity can be used instead of allocated bits to represent the motion complexity.

However, when we enable rate control in H.264, QP varies to regulate the bit rate according to channel bandwidth constraints. The best mode selection is highly dependent on λ_{mode} value which is a function of QP. For instance, with smaller QP and as a result smaller λ_{mode} value, smaller macroblock size selection is more probable. Therefore, comparing modes percents in two frames coded with different λ_{mode} values is not a fair and accurate estimation of their complexity any more. For example a P8x8 coded macroblock with QP1 is more complex than a P8x8 coded macroblock with QP2 if QP1>QP2. Therefore we modify the above complexity estimation by considering the QP as

$$\begin{aligned} & if\,|QPc-QP_{ref}| \le 2 \\ & \quad C_{frame}=(C_{frame})\times(1+0.2(QPc-QP_{ref})) \\ & if\,|QPc-QP_{ref}| \succ 2 \\ & \quad C_{frame}=(C_{frame})\times(1+0.2(QPc-QP_{ref}))*0.7 \end{aligned} \tag{14}$$

where QPc and QP_{ref} are current frame quantization parameter and a reference quantization parameter, respectively, and C_{frame} is the frame complexity estimation defined is ()6. In our work, the value of QP_{ref} is initial QP of the sequence plus 2. The above frame complexity is depicted in Figure 3 and compared with frame complexity estimation without considering QP in the case of enabling rate control.

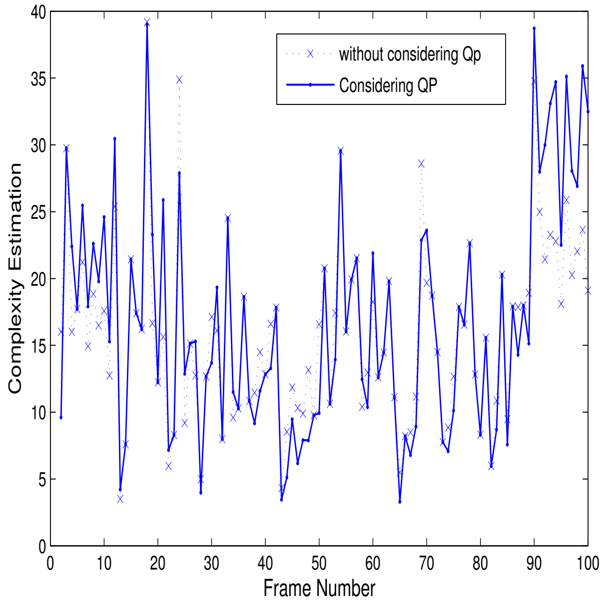


Fig. 3 Modifying frame complexity by considering QP of each frame and without considering the QP

According to above discussions the bit allocation is performed for each frame based on its complexity comparing with reference complexity which is defined as

$$C_{ref} = \alpha \times C_{avg} + (1-\alpha) \times C_{prev} \tag{15}$$

where C_{avg} is the average complexity of all P frames before last coded P frame, C_{prev} is the frame complexity of P frame before last coded P frame and α is a constant value of 0.5 which is chosen experimentally. We used this weighted sum to remove unreasonable results. In order to the bits for each frame based on its complexity, we compare each frame complexity with the complexity reference and allocate the bits to the next frame as follow

$$\begin{aligned} & if\ C_{frame} > C_{ref} \\ & \qquad T_1 = (1+\beta_1) \times \frac{T_r(n_{i,j})}{N_r} \\ & if\ C_{frame} < C_{ref} \\ & \qquad T_1 = (1-\beta_2) \times \frac{T_r(n_{i,j})}{N_r} \end{aligned} \tag{16}$$

where β_1 and β_2 are constant values of 0.2 and 0.22. These values are chosen experimentally.

B. Early 16x16 motion estimation to estimate the complexity

To estimate the current frame motion complexity, first we can find the best 16x16 block matched in the references for all MBs in the current frame. It can be performed in two ways: find only the best matched block size of 16x16 without considering motion estimation cost, or by computing and minimizing the motion cost with a coarse QP value equal to of previous frame QP. However, the later will be affected by QP. But our experiments show that the effect of QP is negligible in low bit rate cases and almost has no effect in high bit rate encoding or low motion sequences. In our proposed algorithm the second approach is adopted. Our detailed approach is discussed here.

After performing motion estimation for all MBs in the frame, total SAD of 16x16 block motion compensated frame is computed. Based on this value of SAD of current frame, SAD_{curr}, and previous frame, SAD_{prev}, and the average value of previous frames, SAD_{avg}, we can compare the current frame complexity with previous frames. First we define the comparison value, SAD_{ref}, for current frame as

$$SAD_{ref} = \alpha \times SAD_{prev} + (1-\alpha) \times SAD_{avg} \tag{17}$$

where α is a constant value of 0.5.

Comparing SAD_{curr} and SAD_{ref} represents current frame complexity and bits are allocated to the frame accordingly by adjusting (7) as

$$T_2 = \frac{T_r(n_{i,j})}{N_r} \times (1 + \frac{SAD_{curr}}{SAD_{ref}} \times \gamma) \tag{18}$$

where γ is a constant value to normalize the frame complexity. In this case, remaining bits are distributed among remaining frames based on their complexity.
The final target bits is computed as follows:

$$T = K \times T_1 + (1-K) \times T_2 \tag{19}$$

Here, the remaining bits are distributed to all non-coded frames according to their statistics and complexities. Then we use this value of T in (3)3 to compute target bits for the frame.

C. More Other Modifications

Furthermore, the predicted MAD for current frame is adjusted with a weighted combination based on the SAD value. In special case of scene change, value of SAD_{curr} divided by SAD_{ref} exceeds a threshold. When scene change is detected, some other improvements are applied as well. First, the predicted MAD is altered as

$$MAD_{curr} = MAD_{pred} \times (1 + \frac{SAD_{curr}}{SAD_{ref}} \times \gamma) \qquad (20)$$

Second, all model parameters such as those in linear model and quadratic model are reset to initial values. Updating methods for these parameters including sliding windows are reset as well.

Finally, after above computation, QP is computed and RDO is performed with all valid modes except the intra 16x16. Our proposed algorithm is shown in Fig. 2.

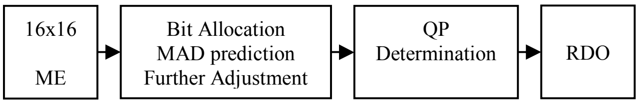


Fig. 4 Proposed Algorithm

VII. EXPERIMENTAL RESULTS

In our experiments, the JM9.6 test platform is adopted. The JVT rate control software is selected as a reference of our comparison. The proposed algorithm is implemented in JVT scheme, and the test sequences are in QCIF 4:2:0 format.

TABLE I. Average PSNR

SEQUENCE	Reference Software	Proposed
Suzie-Trevor	34.33	36.78
Forman	34.87	35.19
Salesman	35.42	35.53

Primary results show that our proposed algorithm improves the average PSNR, as shown in Table I. Also the standard deviation of the PSNR is significantly improved by using the proposed method. The PSNR of each frame in “Foreman” and “Suzie-Trevor” encoded at 56kbps are plotted in Fig. 3 and Fig. 4, respectively. The results of other test videos are listed in table II. As illustrated the new technique has achieved an average PSNR gain of 0.43 dB and reduced PSNR deviation as well in average of 21.25%. This is while using our proposed algorithm, adds a little computational complexity. Table II also shows that our scheme meets target bit rates more closely than JM9.6.

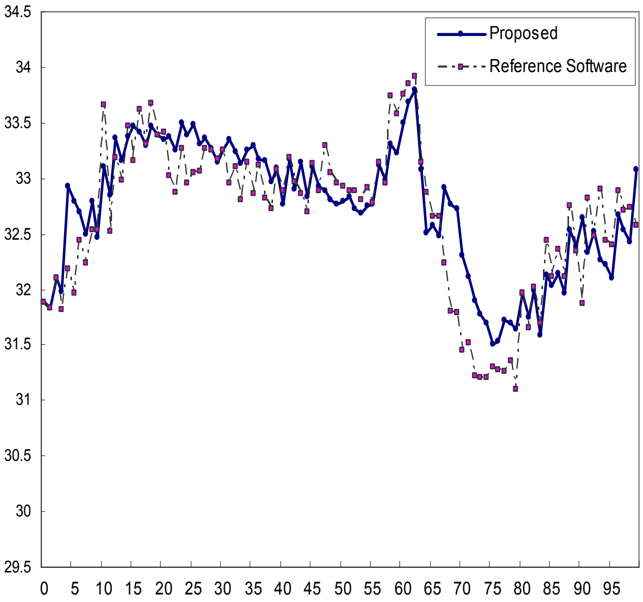


Fig. 5 PSNR of each frame for sequence “Foreman“.

VIII. CONCLUSION

In this paper we have proposed a new method for frame layer bit allocation based on frame complexity so that it can maintain a video stream with a smoother PSNR variation which is highly desirable in real-time video coding and transmission. The proposed approach uses the distribution of MB modes in a frame as a measure of its complexity. Also the early motion estimation approach is introduced and is used in complexity estimation. The current frame’s complexity is then estimated based on previously encoded frames and bit budget is allocated to this frame based on its complexity. Our experimental results show that our proposed approach outperforms the JM9.6 rate control algorithm by improving the PSNR in average of 0.43 dB and PSNR deviation reduction of 21.25%. Furthermore, the proposed improvements can be

TABLE II. Simulation Results of Proposed Algorithm and JM9.6 reference software

Sequence	Average PSNR		PSNR Std. Deviation		Bit Rate (kbps)	
	JM9.6	Proposed	JM9.6	Proposed	JM9.6	Proposed
Foreman (100kbps)	34.87	35.68	1.29	1.08	101.32	100.26
Foreman (56kbps)	32.34	32.85	1.09	0.78	57.36	56.12
Carphone (100kbps)	36.82	37.41	1.16	1.01	100.24	100.33
Carphone (56kbps)	34.34	34.79	0.76	0.64	56.50	56.48
Claire (100kbps)	45.48	45.83	1.02	0.73	99.80	99.71
Claire (56kbps)	42.53	42.67	1.56	1.16	56.18	55.92
Salesman (100kbps)	39.63	40.11	2.14	1.68	101.7	100.49
Salesman (56kbps)	35.42	35.56	1.58	1.23	56.44	56.31

extended to MB level rate control by allocating bits to basic units of a frame according to basic units' complexity.

Fig. 6 PSNR of each frame for sequence "Suzie-Trevor".

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard*," IEEE Trans. Circuits Syst. Video Technol,* vol. 13, no. 7, pp. 560-576, 2003.

[2] Joint Video Team of ITU-T and ISO/IEC JTC 1, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)," Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, March 2003.

[3] ITU-T and ISO/IEC JTC 1, "Generic coding of moving pictures and associated audio information – Part 2: Video," ITU-T Recommendation H.262 – ISO/IEC 13818-2 (MPEG-2), Nov. 1994.

[4] ITU-T, Video Codec for Audiovisual Services at px64 kbit/s, ITU-T Recommendation H.261, Version 1: Nov. 1990; Version 2: Mar. 1993.

[5] ITU-T, "Video coding for low bit rate communication," ITUT Recommendation H.263; version 1, Nov. 1995; version 2, Jan. 1998; version 3, Nov. 2000.

[6] ISO/IEC JTC1, "Coding of audio-visual objects – Part 2: Visual," ISO/IEC 14496-2 (MPEG-4 visual version 1), April 1999; Amendment 1 (version 2), February, 2000; Amendment 4 (streaming profile), January, 2001.

[7] M. Ghanbari, "Standard Codecs*:* Image Compression to Advanced Video Coding." London, IEE, 2003.

[8] H. Schwarz, D. Marpe, T. Wiegand,"Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Trans. Circuits Syst. Video Technol,* VOL. 17, NO. 9, PP. 1103-1120, 2007.

[9] G. Sullivan, T. Wiegand and K. P. Lim, "Joint Model Reference Encoding Methods and Decoding Concealment Methods; Section 2.6: Rate Control" JVT-I049, San Diego, September 2003.

[10] S. Eshaghi, O. Fatemi, M. R. Hashemi, "A Frame Layer Bit Allocation for H.264 Based on Mode Decision," *IEEE Int. Conf. Signal Processing and Communications,* 2007.

[11] H. J. Lee and T. H. Chiang and Y.-Q. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 878-894, 2000.

[12] J. Ribas-Corbera, P. A. Chou, and S. Regunathan: "A Generalized Hypothetical Reference Decoder for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol*, Vol. 13, No. 7, pp. 674- 687, July 2003.

[13] J. Ribas-Corbera and S.-M. Lei, "A frame-layer bit allocation for H.263+," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, Oct. 2000.

[14] *ISO/IEC JTCI/SC29/WG11, Test Model 5*, 1993.

[15] *ITU-T/SG15, Video Codec Test Model, TMN8, Portland*, June 1997

[16] *MPEG-4 Video Verification Model V18.0, Coding of Moving Pictures and Audio N3908, ISO/IEC, JTC1/SC29/WG11*, Jan. 2001.

[17] Z. Li, F. Pan, K. Pang, "Adaptive Basic Unit Layer Rate Control for JVT," JVT-G012, Pattaya, Thailand, March 2003.

[18] Zhengguo Li, Wen Gao, Feng Pan, "Adaptive Rate Control with HRD Consideration," JVT-H014, Geneva, May 2003.

[19] T. Chiang and Y. *Q.* Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol,* vol. 7, no. 1, pp. 246-250, Feb. 1997.

[20] B. Tao, B. W. Dickinson, and H. A. Peterson, "Adaptive model-driven bit allocation for MPEG video coding," *IEEE Trans. on Circuits ond Systems for Video Technologv,* vol. 10,no. 1, pp. 147.157, Feb. 2000.

[21] R. C. Jordi and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol,* vol. 9, no. 1, pp. 172-185, Feb. 1999.

[22] J. Cabrera, A. Ortega, and I. 1. Ronda, "Stochastic ratecontrol of video coders for wireless channels," *IEEE Trans. Circuits Syst. Video Technol,* vol. 12, no. 6, pp. 496-510, June 2002.

[23] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Trans. Image Processing*, vol. 3, pp. 533–545, Sept. 1994.

[24] M. Jiang, X. Yi, and N. Ling, "On Enhancing H.264 Rate Control by PSNR-Based Frame Complexity Estimation ," *IEEE Int. Conf. on Consumer Electronics*, PP. 231-232, Las Vegas, January 2005.

[25] M. Jiang, X. Yi, and N. Ling, "Improved frame-layer rate control for H.264 using MAD ratio," *IEEE Int. Sym. on Circuits Syst.* Vol. III, PP. 813-816, Vancouver, Canada, May 2004.

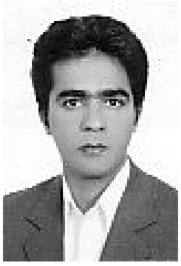
[26] H. Yu, F. Pan, and Z. Lin "A New Bit Estimation Scheme for H.264 Rate Control," *Int. Symp. Consumer Electronics*, 2004.

[27] *J. Xu, Y. He* "A NOVEL RATE CONTROL FOR H.264," *Int. symp. Circuits Syst.* 2004.

[28] S. Kim, Y. S. Ho "Rate Control Algorithm for H.264lAVC Video Coding Standard Based on Rate-Quantization Model," *IEEE Int. Conf.* Multimedia Expo. 2004.

[29] M. Arfan Jaffar, Bilal Ahmed," Color video segmentation using fuzzy C-Mean clustering with spatial information," in proceeding of NAUN international journal of Circuits, Systems and Signal Processing, Issue 5, Vol. 5, May 2009, pp.123-134.

[30] M. G. Sarwer, Q. M. Wu, " An efficient search range decision algorithm for motion estimation in H. 264/AVC," in proceeding of NAUN international journal of Circuits, Systems and Signal Processing, Issue 3, Vol. 3, May 2009, pp. 173-160.



Siavash Es'haghi received the B.Sc. degree from Iran University of Science and Technology, Tehran, Iran, and M.Sc. degree from Malek-Ashtar University of Technology, Tehran, Iran in 2002 and 2006, respectively. From 2005 to 2006, he joined in Multimedia Processing Laboratory in School of Electrical & Computer Engineering, University of Tehran.

In 2007, he was employed in Islamic Azad University, Branch of Birjand, Iran, as a lecturer in the department of computer and Electronics engineering. His interests include video, image and digital signal processing and implementation.

Hassan Farsi received the B.Sc. and M.Sc. degrees from Sharif University of Technology, Tehran, Iran, in 1992 and 1995, respectively.

From 1992 to 1993, he worked in research communications centre, Tehran, Iran on waveform speech coders.

In 1995, he was employed in University of Birjand, Birjand, Iran as a lecturer in the department of Electronics and communications engineering. Since 2000, he started his Ph.D in the Centre of Communications Systems Research (CCSR), University of Surrey, Guildford, UK, in field of speech quality improvement provided by low bit rate speech coders and received the Ph.D degree in 2004. His research has also led to improving speech quality coded by 2.4 kb/s standard MELP coder. Besides speech processing, he is also interested in image and video processing and he is doing his research in this area as well.