IEEE Transactions on Multimedia (2013). 15(5), 1094-1109.
DOI: http://dx.doi.org/10.1109/TMM.2013.2241414

Mode Decision-based Algorithm for Complexity Control in H.264/AVC

Amaya Jiménez-Moreno*, Eduardo Martínez-Enríquez, *Student Member, IEEE,* and Fernando Díaz-de-María, *Member, IEEE*

***Abstract*—The latest H.264/AVC video coding standard achieves high compression rates in exchange for high computational complexity. Nowadays, however, many application scenarios require the encoder to meet some complexity constraints.**

This paper proposes a novel complexity control method that relies on a hypothesis testing that can handle time-variant content and target complexities. Specifically, it is based on a binary hypothesis testing that decides, on a macroblock basis, whether to use a low- or a high-complexity coding model. Gaussian statistics are assumed so that the probability density functions involved in the hypothesis testing can be easily adapted. The decision threshold is also adapted according to the deviation between the actual and the target complexities.

The proposed method is implemented on the H.264/AVC reference software JM10.2 and compared with a state-of-the-art method. Our experimental results prove that the proposed method achieves a better trade-off between complexity control and coding efficiency. Furthermore, it leads to a lower deviation from the target complexity.

***Index Terms*—Complexity control, H.264/AVC, hypothesis testing, mode decision.**

I. Introduction

Nowadays, in a world of multimedia portable devices, signal processing systems must be designed to run on a variety of platforms, each one endowed with specific computational and memory resources. Therefore, the conception of algorithms capable of adapting their computational complexity (obviously in exchange for performance, memory, delay, etc.) to those supported by specific devices becomes an important challenge that will be of interest in years to come.

Video coding is one of the numerous signal processing systems that, in some scenarios, are required to be complexity-adaptive. Although many research efforts have been devoted to reduce the complexity of video compression algorithms [1]–[13], only a few works have been devoted to actually *control the complexity* [14]–[24]. In this paper, the problem of complexity control is tackled in the framework of H.264/AVC, the latest video coding standard of the Joint Video Team (JVT).

It is well- known that H.264/AVC achieves a significantly higher coding efficiency than previous video coding standards, such as MPEG-2/H.262, MPEG-4 part 2, and H.263. As a result of this higher efficiency, H.264 is the most suitable coding standard for a wide range of applications demanding high quality and low bit rates. To achieve this coding efficiency, H.264/AVC makes use of a variety of techniques, such as quarter-pixel-accuracy motion estimation (ME), multiple reference frames, various block sizes, in-loop deblocking filter, 4×4 DCT transform, and context-based adaptive binary arithmetic coding (CABAC). Given a macroblock (MB), the encoder has to choose among a variety of potential coding options in an optimum manner. For this purpose, H.264/AVC uses a rate-distortion optimization method (RDO).

Complexity control algorithms aim to provide the best possible rate-distortion (R-D) performance while satisfying a specific complexity constraint. In other words, the goal is no longer to just reduce the complexity of an H.264/AVC implementation, but also to keep it around a certain target complexity.

This work aims to design an algorithm capable of keeping its complexity around a certain externally- provided target value with minimum losses in terms of coding efficiency, even when the target complexity is very low. The proposed approach, which relies on tools that have proven to be effective in complexity reduction, has been devised to satisfy the following specifications: low miss-adjustment error with respect to the target complexity, capability to adapt to a time-variant complexity target and to the video content, and capability to operate on a large dynamic range of target complexities and to work with any image resolution.

The rest of the paper is organized as follows. Section II gives a brief review of the most relevant contributions to the complexity control problem in H.264/AVC. Section III explains in detail the proposed method. Section IV describes the experiments conducted to prove the strengths of the method, and shows and discusses the results. Finally, section V summarizes our conclusions.

II. Related Work

A. Background: RDO in H.264/AVC

Since most of the algorithms that deal with complexity control in H.264/AVC work on the RDO process, which involves the ME and the mode decision (MD) subsystems, a brief summary of this process is in order to provide an appropriate background.

As mentioned in the introduction, the H.264/AVC encoder selects the best coding option for each MB by means of an RDO process. This optimization process significantly contributes to the coding efficiency, but at the expense of a notable increment of the encoder complexity. The RDO process entails

This work has been partially supported by the National Grant TEC2011-26807 of the Spanish Ministry of Science and Innovation.

Amaya Jiménez-Moreno, Eduardo Martínez-Enríquez and Fernando Díaz-de-María are with the Department of Signal Theory and Communications, Carlos III University, Leganés (Madrid), Spain (e-mail: ajimenez@tsc.uc3m.es, emenriquez@tsc.uc3m.es, fdiaz@tsc.uc3m.es).

assessing every coding option for each MB to find the one that minimizes a distortion measure subject to a rate restriction [25]. This problem can be solved by using a Lagrangian optimization, which turns the original constrained optimization problem into an unconstrained one [26].

The typical H.264 encoder implementations sequentially perform two RDO stages. First, the encoder carries out the ME to find the best reference frame (Ref) and motion vector (MV) for any possible block size. Second, the encoder chooses the optimal mode (partition size). The H.264/AVC standard allows for several MB (16×16, 16×8, 8×16, and 8×8 pixels) and sub-MB (8×4, 4×8, and 4×4 pixels) partitions. Moreover, two additional modes, the so-called Direct and SKIP, which are a particular case of the 16×16 MB partition in B and P slices, respectively, are also considered. This whole set is composed of modes known as Inter modes.

The RDO-based ME is solved by means of a Lagrangian optimization, which aims to minimize the following cost function:

$$J_{motion} = SAD(MV, Ref) + \lambda_{motion} R_{motion}(MV, Ref), \quad (1)$$

where SAD denotes the sum of absolute differences between original and predicted blocks (given MV and Ref) and is used as a distortion measure, λ_{motion} is a Lagrange multiplier, and R_{motion} is an approximation to the number of bits needed to encode the motion information.

The MD problem, the solution of which allows the encoder to choose the optimal mode, that is, the optimal partition size k, is solved in the same manner. In this case, the cost function to be minimized is as follows:

$$\begin{aligned} J_{mode,k} = \quad & SSD(\{MV\}_k, \{Ref\}_k, k) + \\ & \lambda_{mode} R(\{MV\}_k, \{Ref\}_k, k), \end{aligned} \quad (2)$$

where the distortion measure is now SSD, the sum of square differences between the original and the reconstructed blocks; λ_{mode} is again a Lagrange multiplier, and R is the number of bits required to encode the headers, MVs, Ref indexes, and residual transform coefficients.

Additionally, an alternative set of modes known as Intra modes is available in the encoder. In this case, the prediction is formed from already encoded pixels of the current slice. As in the Inter case, there are also several block sizes to chose from: 16×16, 8×8, and 4×4 pixels.

The RDO process is responsible for choosing the best possible mode, in R-D terms, among all the Intra and Inter modes.

B. Complexity Control in H.264/AVC

A huge research effort has been devoted to the complexity reduction problem in H.264/AVC since its publication as a standard in 2003. In particular, both the ME and MD processes have received a lot of attention: [1]–[6] are contributions to reduced-complexity ME and [7]–[13] to fast MD, just to name a few examples. Nevertheless, the results of the complexity reduction methods depend heavily on the video content, and therefore these techniques are not capable of guaranteeing that the complexity is kept around a given target.

Focusing now on the complexity control problem, the most common approach involves adding a complexity term to the cost functions that are minimized in the RDO process. In [14], an estimation of the high frequency content of a block and a target complexity are included in a novel cost function so that the ME process relies on it to decide which partitions are taken into account for each MB. In [15], modified versions of both J_{motion} and J_{mode} cost functions are proposed by adding a complexity term that is based on the computation time and the number of instructions required. Moreover, the modes are rearranged according to a texture analysis, so that, given an available complexity for an MB, the encoding process picks modes according to the resulting arrangement, and stops whenever the accumulated complexity exceeds the target complexity; once a subset of modes has been selected in this manner, the modified cost functions are used to decide on the best representation for the MB. It is also worth mentioning that this method requires a costly off-line estimation of the Lagrange multipliers involved in the cost functions. In [16], an algorithm that relies on encoding-time statistics to reach a given complexity target is proposed. In particular, the algorithm estimates the encoding complexity from a buffer occupancy measurement and manages this complexity by means of a Lagrangian rate-distortion-complexity cost. Additionally, the encoder drops frames when the complexity target cannot be met. In [17] a complexity scalable video encoder that is capable of adapting *on-the-fly* to the available computational resources is presented. Specifically, this algorithm works at both frame and MB levels. At the frame level, the algorithm decides the maximum number of SAD calculations according to the complexity budget. At the MB level, the complexity budget is allocated among the MBs in proportion to the distortion of the co-located MBs in previous frames. In [18], an algorithm capable of finding an appropriate encoder configuration is proposed. Given a working bit rate, it finds optimal operating points taking into account distortion and complexity. The authors propose two fast approaches that do not require an exhaustive evaluation of encoder configurations. An extension of this work is presented in [19] following the same principles. In [20], an allocation of computational resources based on a virtual buffer is proposed. Additionally, to guarantee that the used resources do not exceed the estimated ones, two complexity control schemes are defined, one on the ME and the other on the MD. For the ME, a search path and a termination point are defined according to R-D considerations and the allocated complexity. For the MD, a search order and a termination point are defined according to the most frequent modes in neighboring MBs and the allocated complexity. In [21], the MBs in a frame are encoded using only Intra and SKIP modes. Then, the encoding of the MBs producing the highest costs is further refined using additional modes. The number of mode decisions is controlled by means of a parameter that allows this method to be scaled for different complexity targets.

In [22] the Bayesian decision theory is used for complexity control. In particular, a threshold to comply with an average target complexity level is determined using a probability model where the corresponding cumulative density functions are

estimated based on motion measurements and the quantization parameter (QP) value. To this purpose, an off-line precomputed relationship among these parameters is required. This method is limited to SKIP/non-SKIP decisions.

The works described so far were tested on QCIF and CIF resolutions, since complexity control was considered attached to low-power devices, which were not able to work with higher resolutions. Nowadays, however, the fast growth in computational power has made even hand-held devices capable of working with higher resolutions. The works by Queiroz et al. ([23], [24]) tackle the complexity problem for higher resolutions. In [23] complexity is controlled by allowing only for a subset of modes in the MD process. Specifically, the most likely modes are sorted, and only those that do not exceed a pre-established complexity limit are evaluated. In [24] the values of distortion, rate, and complexity achieved by a set of specific encoder configurations are collected by means of an off-line training process. These values are tabulated and a desired level of complexity is reached by applying the corresponding encoder configuration. The weakness of this off-line training process is the difficulty of adapting the model to time varying conditions in both complexity requirements and video content.

The proposed algorithm, as a few of the previously mentioned ([16], [17]), relies on a parameter estimation process that is carried out *on-the-fly*, avoiding both the generalization problems inherent to an off-line estimation and the computational cost associated with the training process. In this manner, the algorithm can easily adapt to changes in both target complexity and video content. As a result, the proposed method is simple and capable of efficiently operating on different video contents and resolutions and on changing complexity targets, exhibiting quite remarkable convergence properties. Furthermore, these high levels of simplicity and flexibility are achieved in exchange for acceptable losses in coding efficiency.

The next section explains the proposed method in detail.

III. Proposed method

A. Motivation and Overview

The proposed algorithm is based on the application of a hypothesis testing whose decision threshold is automatically set to reach the desired coding complexity level. This approach has been adopted for two reasons: 1) it allows for defining a cost policy adapted to the specific problem at hand, thus providing a valuable degree of flexibility; and 2) as shown in our previous work regarding the fast MD problem [27], this approach has proved its ability to act effectively on the complexity while maintaining a high coding efficiency level.

In particular, the proposed algorithm relies on a binary hypothesis testing. For every MB, a decision between low- or high-complexity coding is made. On the one hand, when low-complexity is selected, the MB can be encoded as SKIP, Inter 16×16, or Intra 16×16. On the other, when high-complexity is selected, the MB can be encoded as any of the available Inter or Intra modes. The following argument supports the definition of these two complexity levels. For the algorithm

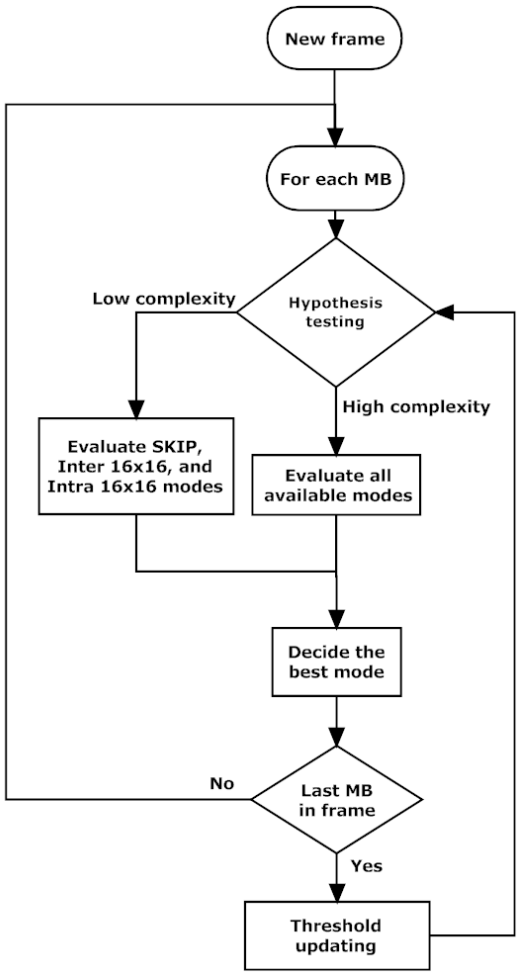


Fig. 1. Flowchart of the proposed algorithm.

to meet tough complexity constraints, the amount of modes in the low-complexity level must be kept as low as possible. Therefore, it would have been desirable for this hypothesis to involve only the SKIP mode, which does not require ME; however, considering only the SKIP mode would have led to significant losses in coding efficiency. Consequently, to avoid these efficiency losses and still keep the complexity at reasonable low levels, the Inter 16×16 mode had to be included. Furthermore, the Intra 16×16 mode had to be included as well to achieve a satisfactory performance in those cases where the ME process does not work properly, i.e., when the penalty in coding efficiency for not allowing Intra modes is high.

Once all MBs in a frame are encoded, the complexity control algorithm must check the achieved complexity and compute the deviation from the target. Then, the complexity control algorithm adjusts the decision threshold of the hypothesis testing according to this deviation, so that this new threshold is used for the next frame to be encoded. The flowchart in Fig. 1 summarizes the whole process.

Mathematically, the formulation of the hypothesis testing derives from the Bayesian decision theory. Given two possible hypotheses H_0 and H_1, and two corresponding decisions D_0 and D_1, the likelihood ratio test (LRT) is defined as follows:

$$\frac{\Pr(x|H_1)}{\Pr(x|H_0)} \gtrless_{D_0}^{D_1} \frac{(C_{10}-C_{00})}{(C_{01}-C_{11})}\frac{\Pr(H_0)}{\Pr(H_1)}, \quad (3)$$

where C_{ji} are the costs of deciding j when the correct hypothesis is i, $\Pr(x|H_i)$ are the likelihoods of obtaining the observation x given the hypothesis H_i and $\Pr(H_i)$ are the *a priori* probability of each hypothesis.

The following subsections explain in detail the main building blocks of the proposed method. Subsection III-B describes the feature selection process, i.e., the selection of the feature x to be used in the LRT expression (3). Subsection III-C presents the specific LRT formulation used. Finally, subsection III-D describes the algorithm that provides the proper threshold to meet the target complexity.

B. Feature Selection

As previously mentioned, the LRT (3) is computed according to an observation x. In particular, the hypothesis test is built on the probability density functions (pdfs) of this observation conditioned to each considered hypothesis ($\Pr(x|H_i)$), with $i = \{0, 1\}$. Consequently, the selection of this input feature x becomes crucial to the success of the proposed method. For this reason, a comprehensive feature selection process is conducted to choose the most appropriate x for describing our decision domain, i.e., the observation x that produces the most separable pdfs $\Pr(x|H_0)$ and $\Pr(x|H_1)$. As stated before, hypothesis H_0 entails a low-complexity encoding model (SKIP, Inter 16×16, or Intra 16×16), while H_1 entails a high-complexity encoding model (any available mode).

Different features have been used in the literature to make an early mode decision. The J_{mode} cost has been proved to be one of the most informative features for this purpose [12] (for a comprehensive statistical analysis of these J_{mode} costs, the reader is referred to [28]). Now, we need to study if J_{mode} costs are also suitable to the complexity control problem. In particular, we seek the most appropriate J_{mode} cost to make an early detection of the MBs that should be encoded as SKIP, Inter 16×16, or Intra 16×16, without causing significant efficiency coding losses. For this purpose we compute the probability of the cost J_k, the J_{mode} associated with the k mode, when hypothesis H_i, with $i = \{0, 1\}$, is true:

$$\Pr(J_k|H_i). \quad (4)$$

In our case, since the modes SKIP, Inter 16×16, and Intra 16×16 are assessed for all the MBs and their corresponding J_{mode} costs are available, we consider the next set of possible costs J_k as candidates for input feature x to our hypothesis testing: J_{SKIP}, $J_{Inter16\times16}$, $J_{Intra16\times16}$, and $J_{min(SKIP,Inter16)}$, where $min(SKIP, Inter16)$ is the minimum cost between the SKIP and the Inter 16×16 modes.

To select the most appropriate cost to be the input feature, we rely on two different tests, the Bhattacharyya distance and the mutual information (MI). The Bhattacharyya distance measures the distance between two pdfs and, for the Gaussian case, is defined as follows:

TABLE I
D_{bhat} AND MI COMPUTED FOR EACH J_k CONSIDERED FOR "RUSH HOUR" (HD) AT QP 24.

	$J_{min(SKIP,Inter16)}$	J_{SKIP}	$J_{Inter16\times16}$	$J_{Intra16\times16}$
D_{bhat}	0.44	0.04	0.03	0.01
MI	0.20	0.19	0.17	0.10

TABLE II
D_{bhat} AND MI COMPUTED FOR EACH J_k CONSIDERED FOR "FOREMAN" (CIF) AT QP 32.

	$J_{min(SKIP,Inter16)}$	J_{SKIP}	$J_{Inter16\times16}$	$J_{Intra16\times16}$
D_{bhat}	0.21	0.10	0.02	0.01
MI	0.14	0.11	0.11	0.09

$$D_{bhat} = \frac{1}{8}(\mu_2-\mu_1)^T\left[\frac{\sigma_1^2+\sigma_2^2}{2}\right]^{-1}(\mu_2-\mu_1) + \frac{1}{2}ln\frac{\frac{\sigma_1^2+\sigma_2^2}{2}}{\sqrt{|\sigma_1^2\sigma_2^2|}}, \quad (5)$$

where μ_1 and μ_2 are the means and σ_1^2 and σ_2^2 are the variances of the two involved pdfs. In our case, we have to compute the distance between $\Pr(J_k|H_0)$ and $\Pr(J_k|H_1)$ for every J_k considered and choose as optimal the J_k that maximizes the distance. In other words, the larger the difference between the distributions, the better J_k is as an input feature for the hypothesis testing.

Likewise, the MI is a statistical tool that measures the shared information between two variables z and y, quantifying how much the knowledge of one of these variables reduces the uncertainty about the other:

$$MI(z; y) = H(z) - H(z|y), \quad (6)$$

where $H(\cdot)$ denotes entropy. In our case z denotes our decision, i.e., if an MB is encoded at either low or high complexity, and y denotes the J_k cost. Therefore, $H(z|J_k)$ represents the entropy of the decision when the J_k cost is known, and $MI(z; J_k)$ the mutual information between the optimal decision and the J_k cost. In this case, the higher the MI, the lower the uncertainty about the decision, and the better J_k is as an input feature for the hypothesis testing. In our experiments, we used the estimator described in [29] to compute the MI.

To select the most suitable feature, we relied on a set of 10 video sequences of different resolutions (4 CIF, 4 QCIF, and 2 HD), and we considered a variety of quality levels (QP = 24, 28, 32, 36, and 40). We computed both the Bhattacharyya distance and the MI in all the cases. According to the Bhattacharyya distance, the results achieved are remarkably consistent and in favor of $min(SKIP, Inter16)$. When the MI is considered, the results are not so consistent, but again $min(SKIP, Inter16)$ turns out to be the most voted. Tables I, II, and III illustrate these results for three selected examples: "Rush Hour" (HD) at QP 24, "Foreman" (CIF) at QP 32, and "Carphone" (QCIF) at QP 36.

As can be observed in Tables I, II and III, the J_k cost associated with $min(SKIP, Inter16)$ is the most suitable for our design, since both the MI and the Bhattacharyya distance are maximum. Therefore, this cost, hereafter $J_{SKIP,16}$, will be used as an input feature in our hypothesis testing.

Figure 2 depicts the resulting pdfs for the same examples. The left part of the figure shows $\Pr(J_{SKIP,16}|H_0)$, in blue,

TABLE III
D_{bhat} AND MI COMPUTED FOR EACH J_k CONSIDERED FOR "CARPHONE" (QCIF) AT QP 36.

	$J_{min(SKIP,Inter16)}$	J_{SKIP}	$J_{Inter16\times16}$	$J_{Intra16\times16}$
D_{bhat}	0.49	0.09	0.02	0.002
MI	0.18	0.10	0.15	0.08

and $\Pr(J_{SKIP,16}|H_1)$, in red, for the sequence "Rush Hour" (HD) at QP 24; the central part shows the same pdfs for "Foreman" (CIF) at QP 32; and the right part shows them for "Carphone" (QCIF) at QP 36. As can be observed, the separability of the distributions is enough to make reliable decisions.

Furthermore, the $J_{SKIP,16}$ cost is a content-dependent feature. Consequently, the pdfs considered must be estimated *on-the-fly* to properly follow the changing properties of these distributions. This content-adaptive property is the main advantage of this proposal. On the other hand, the potential disadvantage would be the computational cost associated with the estimation of the pdfs. This issue is addressed by assuming Gaussian distributions, so that only their means and standard deviations have to be estimated. As shown in Fig. 2, the Gaussianity assumption seems quite reasonable.

The next section explains in detail the hypothesis testing approach.

C. A Content-Adaptive Hypothesis Testing

Once the hypotheses H_0 and H_1 are defined, the input feature $x = J_{SKIP,16}$ is selected, and the resulting conditional pdfs $\Pr(J_{SKIP,16}|H_0)$ and $\Pr(J_{SKIP,16}|H_1)$ are modeled as Gaussian distributions, the LRT defined in (3) can be rewritten accordingly:

$$\frac{\exp(\frac{-(J_{SKIP,16}-\hat{\mu}_1)^2}{2\hat{\sigma}_1^2})}{\exp(\frac{-(J_{SKIP,16}-\hat{\mu}_0)^2}{2\hat{\sigma}_0^2})}\frac{\hat{\sigma}_0^2}{\hat{\sigma}_1^2} \gtrless_{D_0}^{D_1} \frac{\hat{P}(H_0)}{\hat{P}(H_1)}\frac{C_{10}}{C_{01}}, \quad (7)$$

where $\hat{\mu}_0$ and $\hat{\mu}_1$ are the estimated means of the class conditional pdfs ($\Pr(J_{SKIP,16}|H_0)$ and $\Pr(J_{SKIP,16}|H_1)$), respectively; $\hat{\sigma}_0$ and $\hat{\sigma}_1$ are the estimated standard deviations of the same distributions; $\hat{P}(H_0)$ and $\hat{P}(H_1)$ are the estimated *a priori* probabilities of the hypothesis; and the cost associated with correct decisions (C_{00} and C_{11}) are considered to be zero. The parameters of the pdfs, $\hat{\mu}_0$, $\hat{\mu}_1$, $\hat{\sigma}_0$, and $\hat{\sigma}_1$, as well as the *a priori* probabilities $\hat{P}(H_0)$ and $\hat{P}(H_1)$, are estimated *on-the-fly* as described later, so that the decision process is adapted to the specific video content.

An exponentially averaged estimation, in which distant samples are less significant than current samples, is used to estimate the values of the means and standard deviations. Specifically, the updating equations are the following:

$$\hat{\mu}_i(n) = \alpha\hat{\mu}_i(n-1) + (1-\alpha)J_{SKIP,16}(n), i = \{0,1\} \quad (8)$$

$$\hat{\sigma}_i^2(n) = \beta\hat{\sigma}_i^2(n-1) + (1-\beta)(J_{SKIP,16}(n) - \hat{\mu}_i(n))^2, i = \{0,1\}, \quad (9)$$

where n denotes a index associated with the times that the H_i hypothesis is selected; $\hat{\mu}_i(n-1)$ and $\hat{\sigma}_i^2(n-1)$ are the estimated mean and variance, respectively, at the instant $(n-1)$; $\hat{\mu}_i(n)$ and $\hat{\sigma}_i^2(n)$ are the estimated mean and variance, respectively, at the instant n; $J_{SKIP,16}(n)$ is the cost for the involved MB at the instant n; and α and β are the parameters defining the forgetting factors of the exponentially averaged estimation process. Both α and β are experimentally set to 0.95.

Following a similar procedure, the *a priori* probabilities $\hat{P}(H_0)$ and $\hat{P}(H_1)$ are also estimated *on-the-fly*. In this case, the estimated maximum values are limited in order to avoid *winner-takes-all*.

Finally, it is worth mentioning that the hypothesis test does not begin its operation until a reasonable estimation of all of these parameters is reached.

D. A Content-Adaptive Decision Threshold

The most usual expression for the hypothesis test is obtained by taking logarithms in (7):

$$-\frac{(J_{SKIP,16}-\hat{\mu}_1)^2}{2\hat{\sigma}_1^2} + \frac{(J_{SKIP,16}-\hat{\mu}_0)^2}{2\hat{\sigma}_0^2} + ln\frac{\hat{\sigma}_0^2}{\hat{\sigma}_1^2} \gtrless_{D_0}^{D_1} ln(\frac{\hat{P}(H_0)}{\hat{P}(H_1)}) + ln(\frac{C_{10}}{C_{01}}). \quad (10)$$

Furthermore, to simplify the notation in the previous equation, hereafter we will denote the left and right sides of this equation as follows:

$$\theta \gtrless_{D_0}^{D_1} \eta + \epsilon, \quad (11)$$

where the classical expression is slightly modified to distinguish two components in the right part of the inequality. Specifically, η refers to the logarithm of the *a priori* probability ratio, and ϵ refers to the logarithm of the cost ratio.

To control the complexity, we propose to act on ϵ (cost ratio) in (11). By acting on ϵ, we are varying the threshold according to which the hypothesis testing decides whether an MB is encoded using the low-complexity mode (only the SKIP, Inter 16×16, and Intra 16×16 modes are evaluated) or the high-complexity mode (all the available modes are evaluated). The larger the ϵ, the higher the number of low-complexity encoded MBs.

It should be noticed that by acting on ϵ we are actually modifying the relative importance of C_{01} and C_{10}. When low complexity is required, the cost of deciding the high complexity hypothesis when the other was the correct one is large. In such a case, C_{10} takes a high value and, consequently, ϵ also takes a high value. In contrast, when a high value of complexity is acceptable, the complexity control algorithm should focus on coding efficiency. In this case, deciding low complexity when high complexity was the correct decision becomes more relevant; C_{01} takes a high value, and ϵ a low value. In summary, high values of C_{10} promote complexity saving, while high values of C_{01} benefit coding efficiency.

The goal of the complexity control is to act on ϵ to achieve a certain target complexity TC. This TC is expressed as a percentage of the full complexity, i.e., $TC = 100$ means that the target complexity is that of the full mode evaluation, or $TC = 20$ means that the target complexity is 20% of the full mode evaluation. This TC value could be obtained according to one or several parameters, as in the current battery level in a mobile device, the buffer occupancy in rate-controlled

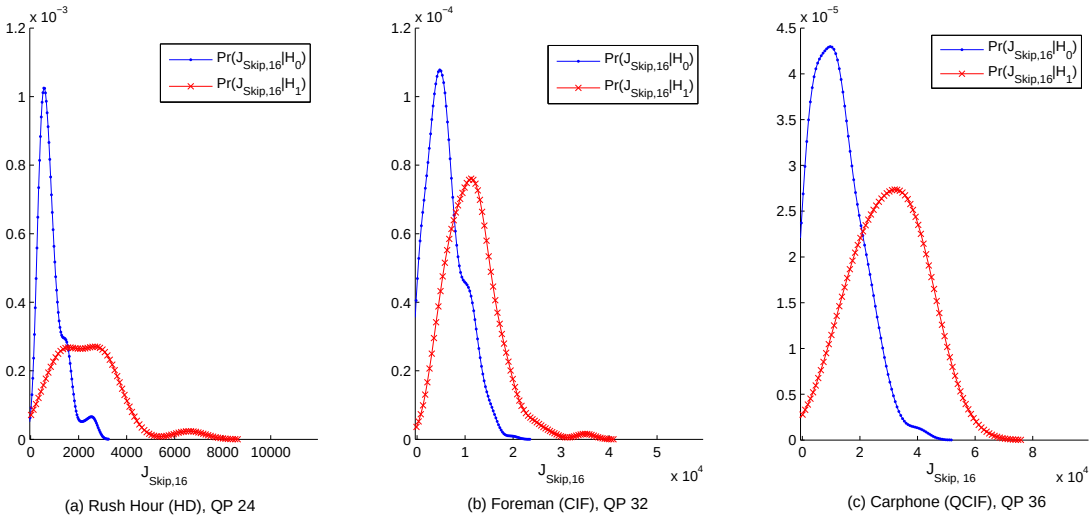


Fig. 2. Examples of $\Pr(J_{SKIP,16}|H_0)$ and $\Pr(J_{SKIP,16}|H_1)$. a) "Rush Hour" (HD) at QP 24; b) "Foreman" (CIF) at QP 32; and c) "Carphone" (QCIF) at QP 36.

transmission application, or the available CPU resources in non-dedicated multi-task systems.

The TC is converted into an equivalent parameter that is directly managed by the proposed algorithm: the number of MBs encoded in low complexity mode, MB_{low}. Actually, each time the hypothesis testing decides D_0, a low complexity MB is encoded. In this way, if the TC is low, MB_{low} should be high and vice-versa.

Given a target complexity TC, MB_{low} is computed as follows. Let us define μ_{high} and μ_{low} as the average time spent for encoding an MB at high- or low-complexity, respectively. These two parameters are computed by simply averaging the real encoding time spent on each type of MB over several MBs, and are initialized using the first high- and low-complexity samples, respectively. Let us define now the target time that should be spent per frame, TT, to meet the TC:

$$TT = time_{per-frame-full} \times \frac{TC}{100}, \tag{12}$$

where $time_{per-frame-full}$ denotes the time spent encoding a whole frame at full complexity. We rewrite the previous equation by expressing the time per frame as a function of the number of MBs in a frame, $MB_{per-frame}$:

$$TT = (\mu_{high} \times MB_{per-frame}) \times \frac{TC}{100}. \tag{13}$$

Likewise, the target time TT can be expressed in terms of the number of MBs encoded at high complexity, MB_{high}, the number of MBs encoded at low complexity, MB_{low}, and the corresponding average coding times per MB, μ_{high} and μ_{low}:

$$TT = (\mu_{high} \times MB_{high}) + (\mu_{low} \times MB_{low}). \tag{14}$$

When equations (13) and (14) are combined, the number of MB encoded at low complexity can be easily found as a function of the TC:

$$MB_{low} = \frac{(\mu_{high} \times MB_{per-frame})\left(1 - \frac{TC}{100}\right)}{\mu_{high} - \mu_{low}}. \tag{15}$$

Once the TC is converted into MB_{low}, we can tackle the problem of selecting a specific value for the threshold ϵ so that a given MB_{low} is met. The relationship between ϵ and MB_{low} has been studied experimentally. Figure 3 illustrates the result by means of two examples. One of the curves is derived from "Paris" and the other from "Foreman", both with CIF resolution, at QP=28. It can be observed that ϵ increases with MB_{low} (the number of early stops) until saturation. The saturation of the curve indicates that $MB_{low} = MB_{per-frame}$, i.e., all the MBs (396 for the CIF sequences of our example) are encoded at low complexity, reaching the lowest complexity level achievable by the proposed method.

It is worth noting that the number of early stops obtained for a given ϵ actually depends on the video content. For example, $\epsilon = -2$ produces $MB_{low} = 182$ for "Paris" and $MB_{low} = 63$ for "Foreman". Furthermore, the differences between curves are more significant for low values of ϵ due to the low slope of the curve. In general, the statistics in (10) are time-variant; therefore, fixing a specific value of ϵ would produce meaningful differences in the number of early stops MB_{low} from frame to frame.

Because of these reasons, ϵ must be adjusted *on-the-fly* to follow the time-variant statistics and achieve the target MB_{low}. Specifically, we propose to update ϵ on a frame-by-frame basis by means of a feedback algorithm, as shown in the following equation:

$$\epsilon_f = \epsilon_{f-1} + \nu \times \Delta MB_{low}, \tag{16}$$

where ϵ_f and ϵ_{f-1} are the thresholds applied to the $f-th$ and $(f-1)-th$ frames, respectively; ΔMB_{low} is the difference between the MB_{low} target for the $f-th$ frame and the actual MB_{low} obtained for the $(f-1)-th$ frame; and ν

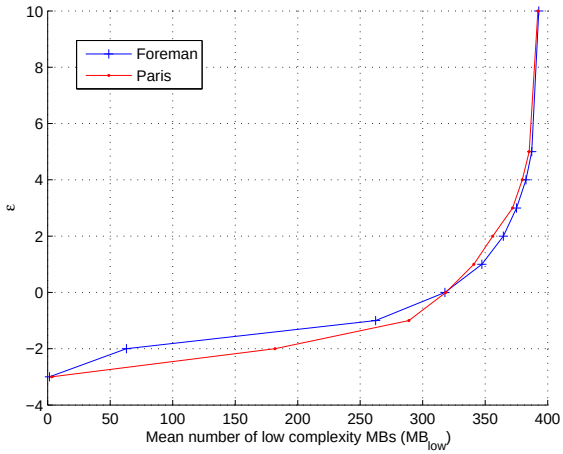


Fig. 3. An illustration of the relationship between the number of MBs encoded at low complexity MB_{low} and the threshold ϵ in two sequences.

is a parameter experimentally determined as a function of ΔMB_{low} and the frame size.

The ν value allows for choosing an application-specific operating point that properly balances the adaptation speed versus the amplitude of the oscillations around the target complexity. If a high value of ν is used, the target time per frame, TT, will be reached faster, but a larger oscillation around this TT will be observed, and vice-versa. Figure 4 illustrates this behavior for "Mobile" (QCIF) at QP 28. The resulting time evolution of MB_{low} (the number of MBs encoded at low complexity) is shown for two values of ν. As can be seen, for $\nu = 0.005$ (left part of the figure), some frames are needed to reach the desired value of MB_{low}, but the oscillations around the desired value are moderated. In contrast, for $\nu = 0.1$ (right part of the figure), the desired value of MB_{low} is reached much faster, but at the expense of larger oscillations.

To properly manage this trade-off, the value of ν is varied adaptively according to the magnitude of ΔMB_{low}: the higher the ΔMB_{low}, the higher the ν. In this manner, when encoding time is far from TT, ϵ is adapted faster, and vice-versa. Furthermore, different ν values are used for each spatial resolution (QCIF, CIF, and HD), specifically:

QCIF: $|\Delta MB_{low}| > 20 \Rightarrow \nu = 0.05$; $|\Delta MB_{low}| < 5 \Rightarrow \nu = 0$; other case: $\nu = 0.05$.

CIF: $|\Delta MB_{low}| > 50 \Rightarrow \nu = 0.025$; $|\Delta MB_{low}| < 5 \Rightarrow \nu = 0$; other case: $\nu = 0.01$.

HD: $|\Delta MB_{low}| > 80 \Rightarrow \nu = 0.001$; $|\Delta MB_{low}| < 5 \Rightarrow \nu = 0$; other case: $\nu = 0.0005$.

E. Summary of the Algorithm

Algorithm 1 summarizes the complete algorithm.

Algorithm 1 Proposed coding process of the complexity control algorithm.

Require: N: number of frames.
Require: M: number of MBs in a frame.
for $\forall n_i \in N$ **do**
  Calculate MB_{low} based on the mean time measures and the demanded encoding time (15).
  Calculate the threshold ϵ based on the feedback algorithm (16).
  for $\forall m_i \in M$ **do**
    Evaluate SKIP, Inter 16x16, and Intra 16x16 modes.
    Calculate the input feature to the hypothesis testing $J_{SKIP,16}$.
    Apply the hypothesis testing (11).
    if $\theta < \eta + \epsilon$ **then**
      Decide the best mode between SKIP, Inter 16x16, and Intra 16x16.
    else
      Calculate all remaining modes.
      Decide the best mode.
    end if
    Update μ_{high} and μ_{low}, and statistics in (10).
  end for
end for

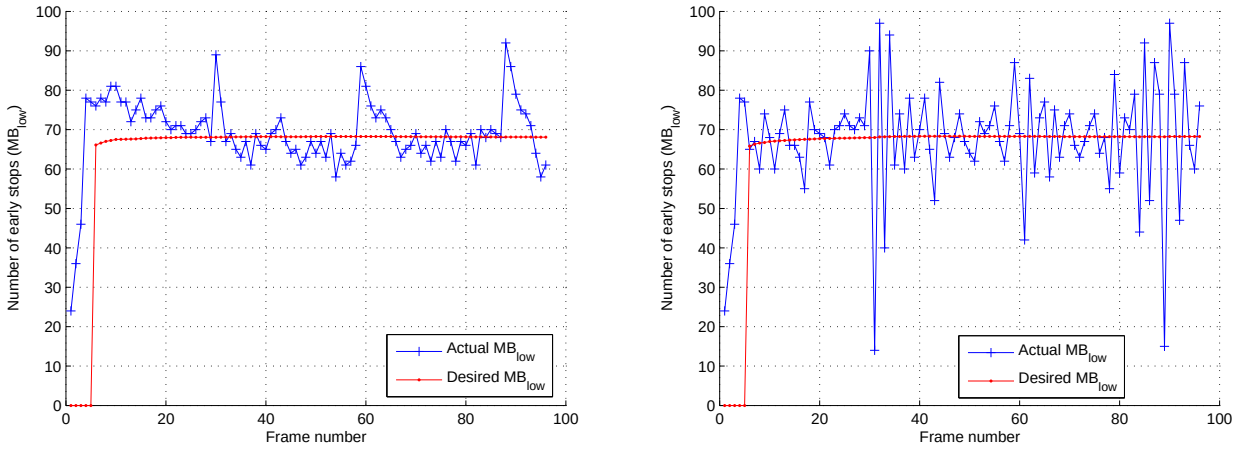
IV. Experimental Results

A. Experimental Protocol

To assess the performance of the proposed method, it was integrated into the H.264 reference software JM10.2 [30]. The main test conditions were selected according to the recommendations of the JVT [31], namely: main profile, ± 32 pixel search range for QCIF and CIF and ± 64 pixels for HD, 5 reference frames, Hadamard transform, CABAC, and RDO. The experiments were conducted using an IPPP GOP pattern, five QP values (24, 28, 32, 36 and 40), and 100 frames per sequence. Table IV summarizes these conditions.

TABLE IV
Test conditions.

Coding options	
Profile	Main
RD Optimization	Enabled
Use Hadamard	Enabled
Symbol Mode	CABAC
Search Range (CIF, QCIF)	± 32
Search Range (HD)	± 64
QP	$24, 28, 32, 36, 40$
Number of Reference Frames	5
Frames to be encoded	100
GOP pattern	IPPP



(a) $\nu = 0.005$. (b) $\nu = 0.1$.

Fig. 4. Illustration of the role of the ν parameter, which controls the balance between complexity adaptation velocity and oscillation amplitude. This results have been obtained for "Mobile" (QCIF) at QP 28.

The experiments involved a large set of sequences of different resolutions covering a wide variety of contents. These sequences are listed in Tables V, VI, and VII for QCIF, CIF, and HD resolutions, respectively.

To evaluate the capability of the algorithm to meet a certain target complexity TC, a measurement of computational time saving TS was calculated as follows:

$$TS = \frac{Time(JM10.2) - Time(Proposed)}{Time(JM10.2)} \times 100. \quad (17)$$

Thus, the higher the measured computational time saving, the lower the reached complexity. In particular, the proposed algorithm was assessed for seven different target complexities, $TC(\%) = \{80, 70, 60, 50, 40, 30, 20\}$, in our experiments.

Furthermore, to evaluate the coding efficiency losses incurred by the proposed method due to the complexity control, average bit rate differences (ΔBR) with respect to the reference software were computed, as described in [32].

B. Performance Assessment

Tables V, VI, and VII show the results for QCIF, CIF, and HD resolutions, respectively. Specifically, for each of the TCs considered, the mean values of $TS(\%)$, and $\Delta BR(\%)$ across the five considered QP values are given. Furthermore, the last row of each table shows the average results for all the sequences.

As can be observed, the achieved complexity was very close to the TC. Therefore, the method is successful in fulfilling the main goal of having a precise complexity control. Moreover, the coding efficiency was maintained very close to that of the reference implementation when medium or high TCs were sought. Obviously, when low TCs were demanded, these were achieved in exchange for more significant losses in coding efficiency.

It is worth mentioning that, exceptionally, bit rate reductions were found. These unexpected results were achieved because the encoder decisions are sub-optimum in the sense that they are made assuming independence between MBs. Thus, in some cases, a decision that is not locally- optimum (in the sense that only explores a subset of modes) could produce better overall performance.

To illustrate how the coding efficiency depends on the TC, Figs. 5, 6, and 7 show the R-D performance for *Coastguard* (QCIF), *Tempete* (CIF), and *Rush hour* (HD) for every other of the considered TCs, respectively (not all of the TCs are depicted to make the graph clearer). The left part of each figure presents the complete R-D curves, while the right part presents a zoom of a selected area. As can be observed, the coding efficiency is very close to that of the reference software for high and medium TCs and degrades gracefully as the TC decreases.

Although the results in terms of objective R-D measurements are good, we also checked that the proposed method does not have negative effects on the subjective quality. To this end, we carefully watched some of the resulting encoded sequences and concluded that there are not perceptual differences with respect to those generated by the reference encoder. Moreover, we labeled the MBs according to the complexity level assigned by the algorithm (low or high) to visually check whether its decisions are as expected. Figure 8 shows an illustrative example where the encoder must comply with a tough complexity constraint ($TC = 30$). As can be observed, only a few MBs are encoded with high complexity (light-colored in the figure) and are those related to moving objects.

Moreover, the proposed algorithm was assessed in comparison with the complexity control algorithm proposed in [23]. Table VIII shows the average results achieved by the compared algorithms for several target complexities ($TC(\%) = \{80, 70, 60, 50, 40, 30, 20\}$). In particular, for each one of the image resolutions considered (QCIF, CIF, and HD), an average result was computed taking into account the five QP values and all the test video sequences. As can be seen, for low complexities (20, 30, and 40), the proposed algorithm generates a complexity closer to the target. The same happens for high

TABLE V
PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHM RELATIVE TO JM10.2 FOR QCIF SEQUENCES. TS STANDS FOR TIME SAVING AND ΔBR STANDS FOR BIT RATE INCREMENT.

TC	20%		30%		40%		50%		60%		70%		80%	
Sequence	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**
Akiyo	76.1	5.1	68.2	1.0	57.7	0.4	48.4	0.3	39.5	0.2	30.2	0.0	22.0	0.0
Bridge close	75.3	3.3	65.5	2.0	55.8	1.0	46.4	0.5	37.9	0.3	28.9	0.3	20.1	0.2
Bridge far	72.0	1.1	64.6	0.8	54.8	0.5	44.1	0.3	37.7	0.2	28.8	0.1	21.1	0.1
Carphone	79.8	11.3	67.9	5.6	57.1	2.8	46.7	1.7	37.2	0.9	28.0	0.0	19.0	0.3
Claire	77.9	5.4	65.0	0.6	54.6	0.2	45.0	−0.2	36.4	−0.1	28.1	−0.1	20.6	−0.3
Coastguard	82.1	9.8	74.8	5.4	62.2	3.2	50.4	1.9	39.9	1.3	30.5	0.8	21.5	0.4
Container	76.0	6.9	66.7	2.7	55.4	1.2	44.4	0.3	35.2	0.1	26.3	0.2	18.2	0.1
Foreman	82.2	17.7	67.7	8.8	56.5	4.7	45.8	2.5	36.4	1.4	27.7	0.6	20.2	0.1
Grandma	77.6	5.7	69.8	1.7	58.4	0.8	48.3	0.5	36.7	0.3	27.6	0.2	18.4	−0.2
Hall	73.5	5.6	65.2	1.13	56.5	0.9	47.0	0.1	38.3	0.2	30.5	−0.1	22.4	0.1
Highway	75.8	12.0	64.3	4.6	53.1	2.5	42.6	1.3	33.5	1.1	26.4	1.0	19.2	0.9
Miss America	74.5	4.2	64.8	1.3	53.7	0.2	42.0	0.0	33.5	−0.1	25.6	−0.3	18.8	−0.4
Mobile	83.8	15.5	71.3	10.2	60.0	7.3	49.5	5.2	39.2	3.5	29.9	2.4	20.7	1.4
M&D	78.2	6.9	67.1	2.2	54.6	1.0	42.7	0.2	32.3	0.3	24.3	−0.2	16.7	0.0
News	76.5	8.3	67.1	2.8	55.8	0.9	45.9	0.3	37.2	0.3	29.1	0.2	20.8	0.3
Salesman	79.1	9.0	71.0	2.8	59.9	0.9	49.4	0.0	39.1	0.1	29.5	−0.1	20.1	0.0
Silent	77.5	8.6	68.4	2.8	58.8	1.4	49.4	1.0	40.0	0.7	31.9	0.5	23.5	0.0
Suzie	78.8	10.7	69.8	5.5	55.6	3.0	44.5	1.7	33.6	0.8	24.1	0.3	16.4	0.3
Average	**77.6**	**8.2**	**67.7**	**3.5**	**56.7**	**1.8**	**46.3**	**1.0**	**36.9**	**0.6**	**28.2**	**0.3**	**20.0**	**0.2**

TABLE VI
PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHM RELATIVE TO JM10.2 FOR CIF SEQUENCES. TS STANDS FOR TIME SAVING AND ΔBR STANDS FOR BIT RATE INCREMENT.

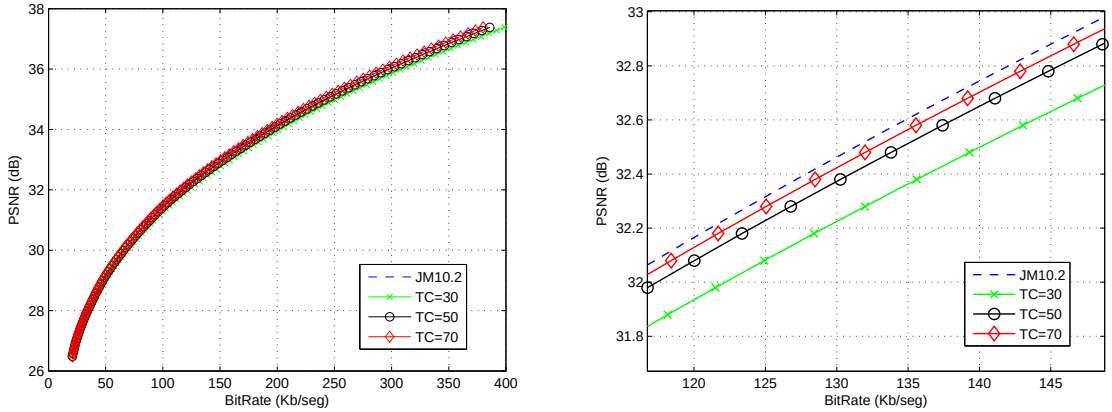
TC	20%		30%		40%		50%		60%		70%		80%	
Sequence	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**
Akiyo	76.8	3.5	66.6	0.7	55.6	0.0	46.0	0.0	37.8	0.0	30.5	0.0	22.2	0.0
Bus	82.1	17.9	70.5	8.0	60.3	4.4	51.0	2.9	42.6	2.5	34.0	2.4	24.5	1.4
Coastguard	83.5	6.2	74.2	3.9	62.2	2.6	50.3	2.0	40.8	1.4	32.6	1.1	22.4	0.5
Container	79.0	5.2	70.1	1.9	59.4	0.5	48.7	0.3	38.8	0.2	30.0	0.0	21.8	−0.1
Football	80.5	21.7	67.6	13.7	53.8	6.7	42.8	2.9	32.7	1.1	24.2	0.7	16.2	0.3
Foreman	80.5	15.2	68.7	5.5	57.9	3.2	47.9	1.7	41.5	2.1	35.1	2.3	23.8	0.8
Garden	82.7	16.8	71.0	11.6	54.7	5.9	42.4	3.6	28.7	2.0	17.6	0.8	9.7	0.3
Highway	75.4	8.5	65.1	3.7	51.5	1.4	42.1	0.6	35.7	0.8	31.2	1.3	20.4	0.3
Mobile	81.0	16.9	67.4	10.6	54.2	6.9	42.6	4.5	32.8	2.9	23.9	1.9	14.3	0.7
M&D	78.9	3.8	66.8	0.9	54.5	0.3	42.6	0.2	33.0	−0.2	24.8	−0.2	17.6	−0.1
News	76.8	6.7	66.6	2.5	56.3	1.0	46.4	0.4	39.1	0.3	32.6	0.3	22.0	0.1
Paris	79.6	14.8	64.8	4.6	54.5	2.0	45.6	0.8	38.5	1.1	31.4	1.1	22.6	0.3
Silent	79.5	6.5	70.0	2.0	60.4	1.3	51.5	0.8	43.1	0.8	35.0	0.7	24.5	0.3
Stefan	76.2	13.9	67.2	10.0	54.8	6.5	40.6	3.1	32.0	1.6	24.5	0.9	16.5	0.6
Tempete	83.3	11.1	69.1	7.0	57.0	4.9	45.9	3.2	37.2	2.2	32.3	1.8	21.3	1.0
Waterfall	82.1	8.0	72.9	3.5	62.0	1.8	52.1	1.4	43.9	0.9	36.3	0.6	25.5	0.5
Average	**79.9**	**11.0**	**68.7**	**5.6**	**56.8**	**3.1**	**46.2**	**1.8**	**37.4**	**1.2**	**29.8**	**1.0**	**20.3**	**0.4**

TABLE VII
PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHM RELATIVE TO JM10.2 FOR HD SEQUENCES. TS STANDS FOR TIME SAVING AND ΔBR STANDS FOR BIT RATE INCREMENT.

TC	20%		30%		40%		50%		60%		70%		80%	
Sequence	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**
Blue Sky	63.8	2.2	62.9	1.9	55.7	1.4	46.2	0.8	36.7	0.5	28.3	0.4	19.7	0.3
Pedestrian	75.7	5.4	63.8	2.4	52.6	1.1	43.0	0.7	34.6	0.4	26.9	0.3	19.6	0.2
Riverbed	82.0	12.4	72.6	10.0	61.0	7.2	50.3	5.1	40.6	3.6	31.5	2.5	23.2	1.7
Rush Hour	77.7	5.4	66.7	2.6	55.7	1.3	46.2	0.7	37.9	0.4	30.0	0.2	22.4	0.2
Station2	78.4	2.6	71.6	0.9	61.4	0.3	51.6	0.2	42.3	0.1	33.0	0.0	23.4	0.2
Sunflower	76.2	1.8	67.5	1.3	58.2	0.9	49.7	0.5	41.6	0.5	33.4	0.2	25.0	0.2
Tractor	80.6	9.1	70.0	3.8	59.9	1.9	49.9	1.3	40.8	0.8	32.9	0.6	24.3	0.5
Average	**76.4**	**5.6**	**67.9**	**3.3**	**57.8**	**2.0**	**48.1**	**1.3**	**39.2**	**0.9**	**30.9**	**0.6**	**22.5**	**0.5**

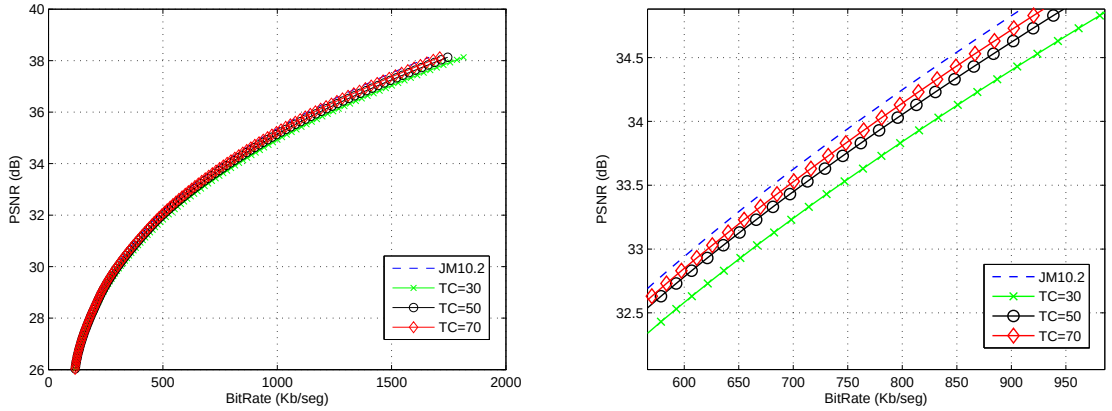
complexities (70 and 80), where the algorithm in [23] generates lower complexities than those actually demanded (because it works by selecting a subset of modes and, sometimes, this procedure does not allow for finer complexity control), usually in exchange for a higher increment of bit rate. Furthermore, in general, the proposed algorithm produces significantly lower bit rate increments for the same TC.

To gain an insight into the differences between the performance of the compared algorithms, some graphical examples are shown for several representative sequences. In particular, we show the bit rate increments of the compared algorithms with respect to the reference software as a function of the computational TS. Obviously, for higher TSs, the losses in coding efficiency and, consequently, the bit rate increments are



(a) R-D performance. (b) Zoom of the Figure on the left.

Fig. 5. R-D performance for a representative subset of the target complexities considered. *Coastguard* at QCIF resolution.



(a) R-D performance. (b) Zoom of the Figure on the left.

Fig. 6. R-D performance for a representative subset of the target complexities considered. *Tempete* at CIF resolution.

more relevant. Figure 9 shows these results for two QCIF sequences, *Coastguard* and *Mother & Daughter*; Fig. 10 shows the results for two CIF sequences, *Foreman* and *Waterfall*; and Fig. 11 shows the results for two HD sequences, *Pedestrian* and *Rush Hour*. As can be observed, the proposed algorithm clearly outperformed that proposed in [23], especially for high computational TSs, where the bit rate increment generated by the proposed algorithm was significantly lower.

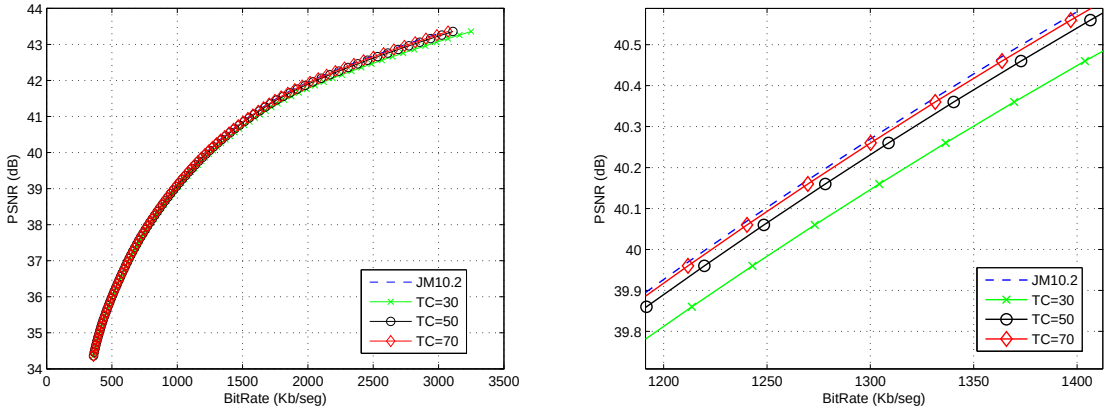
To provide an additional reference, we also compared the proposed algorithm with a fixed mode reduction, i.e., a method that simply explores a predetermined subset of modes. Specifically, we tested three different subsets of Inter modes (Intra modes are always available), namely:

- SKIP and Inter 16×16;
- SKIP, Inter 16×16, Inter 16×8, and Inter 8×16; and
- SKIP, Inter 16×16, Inter 16×8, Inter 8×16, and Inter 8×8.

The results achieved by this method have been added to Figs. 9, 10, and 11. In particular, each subset of modes generates a $(Bit\ rate\ increment,\ Time\ saving)$ point in these figures (these points have been linked by straight lines to improve visualization). As can be observed, the proposed method achieved better performance for QCIF and CIF resolutions, especially for high time savings. On the other hand, for HD resolution, the results were slightly better for the fixed mode reduction method. This last result was expected, since the impact on the R-D performance of the small modes (8×4, 4×8, and 4×4) is not significant for HD, and the proposed method explores all of them for high-complexity MBs. Finally, although this fixed mode reduction is provided as an alternative benchmark, it should be noticed that, actually, it is not a complexity control algorithm (a fixed subset of modes are explored in all the MBs and, therefore, the encoder is not capable of adapting to any target complexity).

C. Performance Assessment: Baseline Profile

In contrast to other approaches that act on the encoder configuration (number of references, search range, ...) to adapt to different complexity levels ([18], [24]), the proposed method aims to control the complexity by dynamically selecting one of two possible subsets of modes at the MB level. The goal of this subsection is to prove that the suggested algorithm can successfully work with different encoder configurations and profiles (which should be selected *a priori* according to the application demands). In particular, we show that it works properly in a configuration very different from that of



(a) R-D performance.

(b) Zoom of the Figure on the left.

Fig. 7. R-D performance for a representative subset of the target complexities considered. *Rush Hour* at HD resolution.

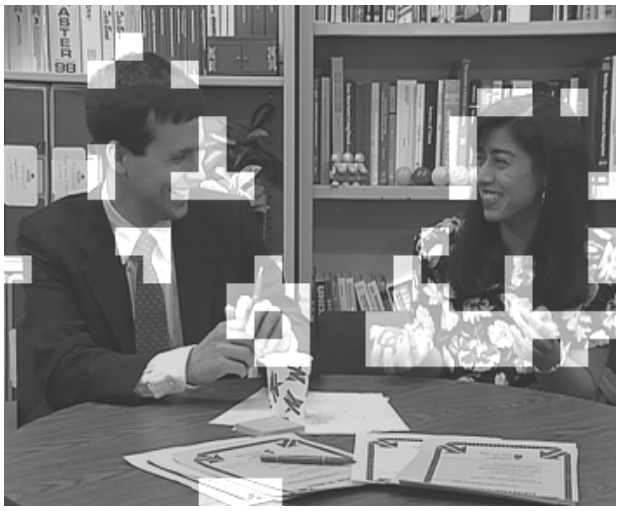


Fig. 8. Illustration of the decisions made by the proposed algorithm. For a tough target complexity, Paris (CIF) with $TC = 30$, we have highlighted those MBs encoded with high complexity. As expected, in general, these MBs belong to moving objects.

the previous experiment. Instead of using the main profile, 5 references, ± 32 pixel search range, and CABAC, we tested our algorithm on a much simpler configuration, more suitable to fit low-capacity devices: baseline profile, 1 reference frame, ± 16 pixel search range, and CAVLC. Table IX shows the complete experimental setup.

For this new configuration, we conducted the same kind of experiments as for the first one. TS and ΔBR were computed with respect to the reference software for the same sets of sequences in QCIF, CIF, and HD resolutions. Table X shows the average results considering all the sequences and QP values. The results obtained for the main profile, denoted as "Main", are also included in the table for reference, together with the new results, denoted as "Baseline".

As can be observed, the algorithm performance is also good for this "Baseline" configuration. It is worth noticing, in particular, how the bit rate increments are lower than those of the "Main" configuration when high complexity reductions are considered.

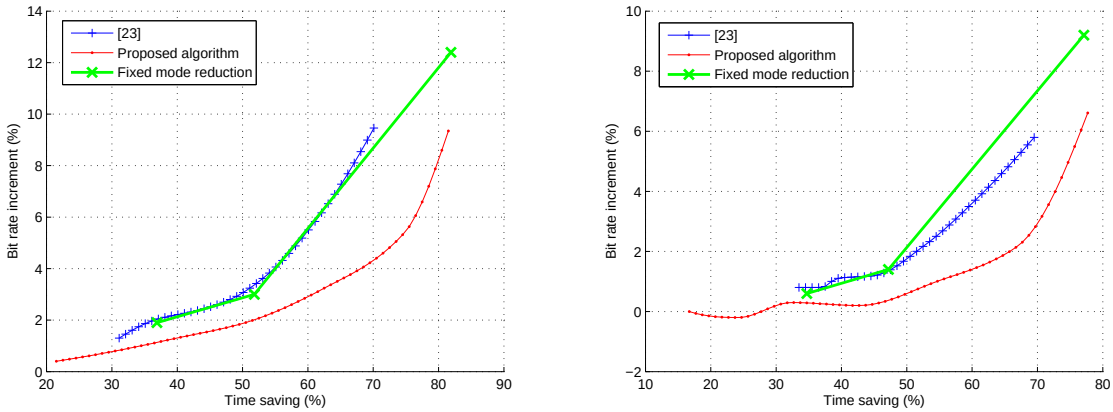
TABLE IX
BASELINE TEST CONDITIONS.

Coding options	
Profile	Baseline
RD Optimization	Enabled
Use Hadamard	Enabled
Symbol Mode	CAVLC
Search Range (CIF, QCIF)	± 16
Search Range (HD)	± 32
QP	$24, 28, 32, 36, 40$
Number of Reference Frames	1
Frames to be encoded	100
GOP pattern	IPPP

Furthermore, since the proposed method worked success-

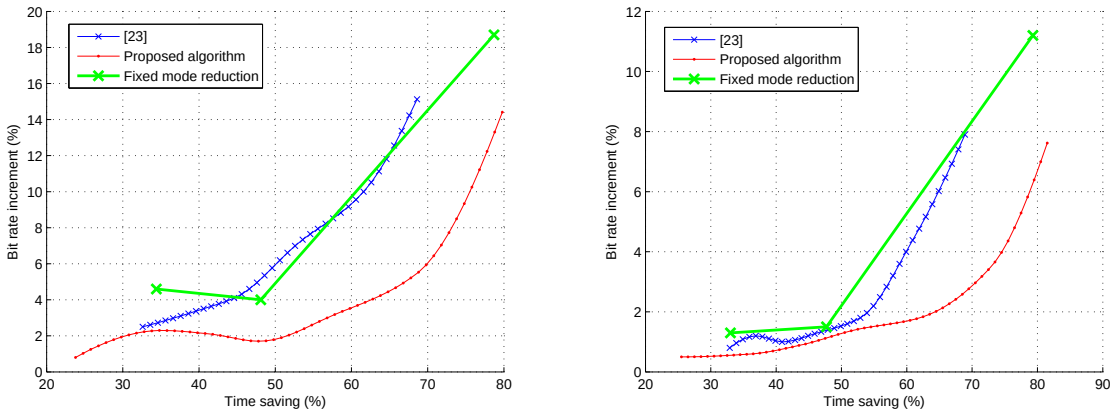
TABLE VIII
PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHM IN COMPARISON WITH [23]. AVERAGE RESULTS. TS STANDS FOR TIME SAVING AND ΔBR STANDS FOR BIT RATE INCREMENT.

TC	20%		30%		40%		50%		60%		70%		80%	
Sequence	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**
Proposed QCIF	77.6	8.2	67.7	3.5	56.7	1.8	46.3	1.0	36.9	0.6	28.2	0.3	20.0	0.2
[23] QCIF	70.1	7.6	61.5	5.2	52.7	3.6	45.7	2.4	39.8	2.0	35.9	1.4	32.0	1.1
Proposed CIF	79.9	11.0	68.7	5.6	56.8	3.1	46.2	1.8	37.4	1.2	29.8	1.0	20.3	0.4
[23] CIF	69.9	10.4	60.6	6.8	52.6	4.6	46.1	3.2	39.8	2.4	33.5	1.6	27.4	1.1
Proposed HD	76.4	5.6	67.9	3.3	57.8	2.0	48.1	1.3	39.2	0.9	30.9	0.6	22.5	0.5
[23] HD	70.2	7.0	62.8	4.2	54.7	2.2	47.6	1.0	41.4	0.5	39.2	0.4	38.4	0.4



(a) Coastguard (QCIF). (b) Mother & Daughter (QCIF).

Fig. 9. Performance evaluation of the proposed method in comparison to that in [23] and to that of a fixed mode reduction for two representative QCIF sequences. The graphs show bit rate increment as a function of the computational time saving.



(a) Foreman (CIF). (b) Waterfall (CIF).

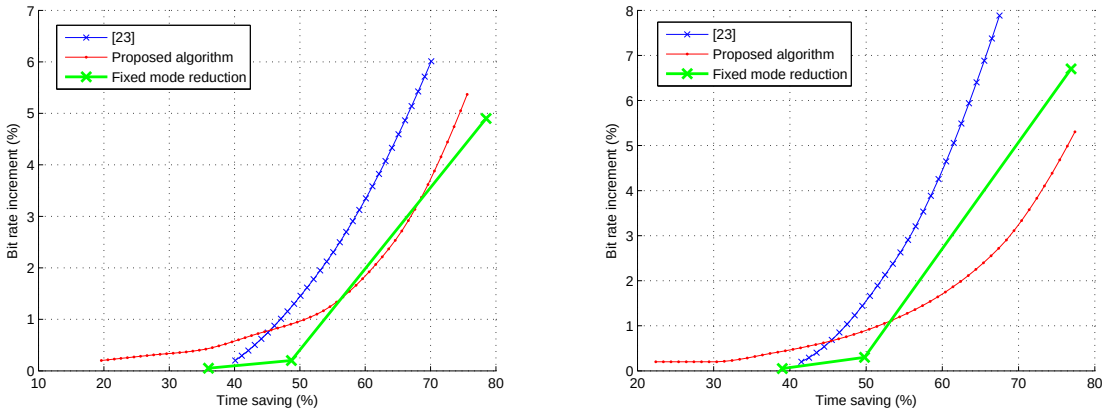
Fig. 10. Performance evaluation of the proposed method in comparison to that in [23] and to that of a fixed mode reduction for two representative CIF sequences. The graphs show bit rate increment as a function of the computational time saving.

fully on two quite different configurations, it is also conceivable that it could work in combination with methods that act on the encoder configuration, such as [18], [24].

D. Illustrations of the algorithm convergence properties

Since the capability to adapt to a time-variant complexity target and to the video content is one of the main goals of the proposed algorithm, some illustrations regarding the algorithm convergence properties are in order.

First, we provide two graphical examples of the capability of the algorithm to converge to a certain TC. Specifically, Fig. 12 illustrates, for *Carphone* (QCIF) at QP = 28, how the number of low-complexity MBs evolves with time (frame number) for two different TC*s*: 20 (Fig. 12a) and 50 (Fig. 12b). As can be observed, when the TC was set to low value, 20 on Fig. 12a, the actual number of early stops (MB_{low}) reached a value very close to the desired one in just a few frames. Furthermore, the variance with respect to the desired value was low. When the TC was set to a higher value, 50 in Fig. 12b, the convergence time was again very small, but



(a) Pedestrian (HD). (b) Rush Hour (HD).

Fig. 11. Performance evaluation of the proposed method in comparison to that in [23] and to that of a fixed mode reduction for two representative HD sequences. The graphs show bit rate increment as a function of the computational time saving.

TABLE X
PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHM FOR THE "BASELINE" ENCODER CONFIGURATION. THE CORRESPONDING RESULTS FOR THE "MAIN" CONFIGURATION ARE ALSO GIVEN FOR REFERENCE. TS STANDS FOR TIME SAVING AND ΔBR STANDS FOR BIT RATE INCREMENT.

TC	20%		30%		40%		50%		60%		70%		80%	
Sequence	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**	**TS (%)**	**ΔBR (%)**
Main QCIF	77.6	8.2	67.7	3.5	56.7	1.8	46.3	1.0	36.9	0.6	28.2	0.3	20.0	0.2
Baseline QCIF	73.9	4.7	64.8	2.2	54.0	1.2	45.8	0.7	38.0	0.4	30.6	0.3	23.5	0.1
Main CIF	79.9	11.0	68.7	5.6	56.8	3.1	46.2	1.8	37.4	1.2	29.8	1.0	20.3	0.4
Baseline CIF	75.5	7.6	65.0	4.0	55.7	2.3	47.4	1.6	40.2	1.5	32.7	1.0	24.7	0.5
Main HD	76.4	5.6	67.9	3.3	57.8	2.0	48.1	1.3	39.2	0.9	30.9	0.6	22.5	0.5
Baseline HD	73.4	6.0	63.9	3.6	54.8	2.4	44.3	1.5	36.6	1.1	29.3	0.8	21.8	0.5

in this case the variance around the desired value of MB_{low} was higher. A very similar behavior was observed for almost all the sequences.

Second, Fig. 13 shows two illustrative examples of a time-variant TC for *Paris* (CIF) at QP=28. On the left part of the figure we illustrate the behavior of the proposed algorithm when the TC changed from 50 to 20 at frame 50. On the right part of the figure, two changes happened: TC went from 20 to 50 at frame 25 and to 30 at frame 50. As shown, the proposed algorithm was able to reach the desired complexity quickly even when fast changes in TC happened.

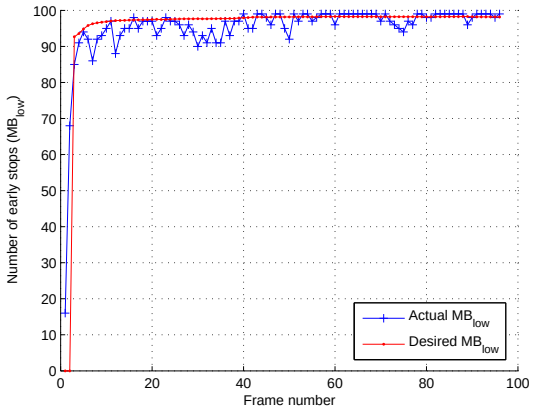
Finally, to provide a more solid proof of the convergence properties of the algorithm than the previous illustrative examples, we computed average results for several sequences covering all the image resolutions considered. Specifically, Table XI shows, for some listed sequences and three different target complexities ($TC(\%) = \{20, 50, 80\}$), the actual value of MB_{low} and the desired value of MB_{low} averaged over all the encoded frames. It is worth noticing that these measurements are totally independent of the implementation. These results allow us to conclude that, on average, the proposed algorithm is able to reach TC with a remarkable precision.

V. CONCLUSIONS

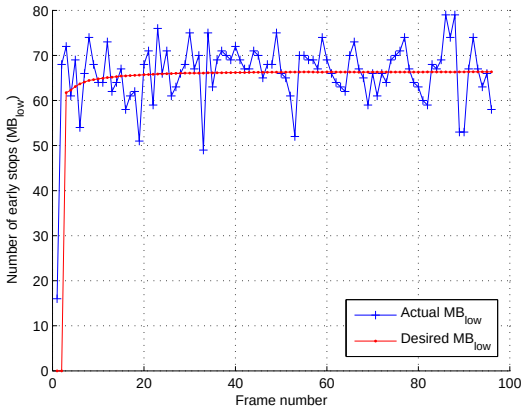
In this paper we have proposed a novel algorithm to control the complexity of an H.264/AVC encoder. The proposed method relies on the application of a hypothesis testing to meet a target complexity with minimum losses in coding efficiency. Assuming Gaussian distributions, the hypothesis testing paradigm allows us to formulate the problem in a simple form that depends on some statistics that can be estimated *on-the-fly*. As a result, the proposed algorithm is capable of adapting to the content and to time-variant target complexities and is able to operate on a large range of target complexities. Furthermore, the proposed algorithm is computationally simple.

To assess its performance, the proposed algorithm was implemented on the reference software JM10.2. The experimental evaluation was carried out on a large set of sequences of several spatial resolutions, a comprehensive set of potential target complexities, and two different profiles and coding configurations. The results obtained allow us to conclude that the proposed algorithm can reach any target complexity with remarkable precision, adapt to time-variant target complexities, and work properly with any spatial resolution, having insignificant bit rate increments for high and medium complexities and acceptable bit rate increments for very low complexities. When compared with the complexity control method in [23], the proposed method was able to reach complexities closer to the target and to provide a better trade-off between complexity reduction and coding efficiency, especially for low and medium target complexities.

An interesting future research line would focus on developing the ideas of the proposed algorithm for the future high efficiency video coding (HEVC) standard [33], which is

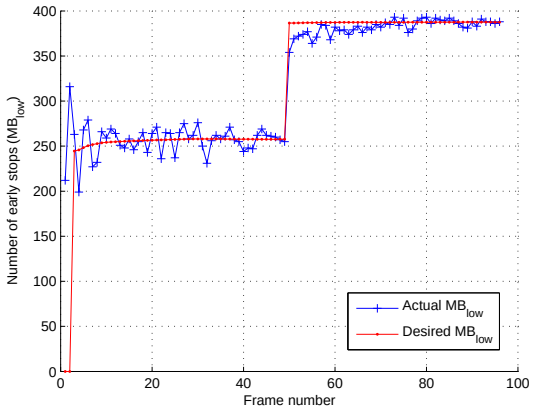


(a) Time evolution of MB_{low} for $TC = 20$.

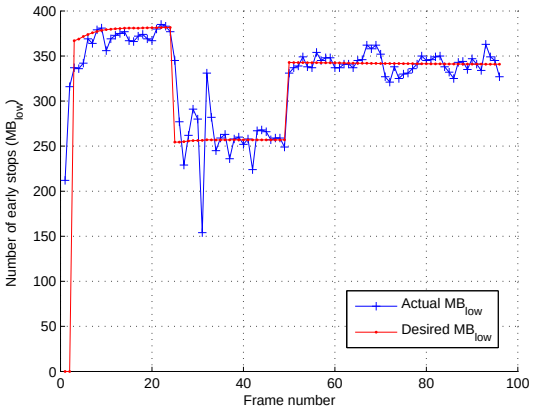


(b) Time evolution of MB_{low} for $TC = 50$.

Fig. 12. Illustrative example of the algorithm convergence properties for *Carphone* (QCIF) at QP 28.



(a) Time evolution of MB_{low} for a time-variant TC, which changes from 50 to 20 at frame 50.



(b) Time evolution of MB_{low} for a time-variant TC, which changes from 20 to 50 at frame 25 and to 30 at frame 50.

Fig. 13. Illustrative example of the algorithm convergence for a time-variant TC, for *Paris* (CIF) at QP 28.

TABLE XI
ASSESSMENT OF THE CONVERGENCE PROPERTIES OF THE PROPOSED ALGORITHM.

	$TC = 20$		$TC = 50$		$TC = 80$	
Sequence	**Desired** MB_{low}	**Actual** MB_{low}	**Desired** MB_{low}	**Actual** MB_{low}	**Desired** MB_{low}	**Actual** MB_{low}
Carphone QP 28 (QCIF)	97	96	66	66	34	34
Container QP 32 (QCIF)	99	98	68	69	34	35
M&D QP 36 (QCIF)	99	94	66	64	32	36
Akiyo QP 28 (CIF)	396	387	271	271	139	139
Mobile QP 36 (CIF)	392	388	261	260	132	131
Silent QP 40 (CIF)	396	394	281	282	141	141
Pedestrian QP 28 (HD)	3528	3453	2368	2377	1189	1191

expected to be submitted in January 2013 for final standardization approval. This work would require a solid knowledge of the mode decision process in HEVC and the corresponding adaptation of the proposed method to the new coding tools.

REFERENCES

[1] H.-Y. Tourapis and A. Tourapis, "Fast motion estimation within the H.264 codec," in *Multimedia and Expo, 2003. ICME '03. Proceedings. 2003 International Conference on*, vol. 3, Jul. 2003, pp. III – 517–20 vol.3.

[2] C. Zhu, X. Lin, and L.-P. Chau, "Hexagon-based search pattern for fast block motion estimation," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 12, no. 5, pp. 349 –355, May. 2002.

[3] J. Zhang, Y. He, S. Yang, and Y. Zhong, "Performance and complexity joint optimization for H.264 video coding," in *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, vol. 2, May. 2003, pp. II–888 – II–891 vol.2.

[4] W. I. Choi, B. Jeon, and J. Jeong, "Fast motion estimation with modified diamond search for variable motion block sizes," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 2, Sept. 2003, pp. II – 371–4 vol.3.

[5] G.-L. Li, M.-J. Chen, H.-J. Li, and C.-T. Hsu, "Efficient search and mode prediction algorithms for motion estimation in H.264/AVC," in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, May. 2005, pp. 5481 – 5484 Vol. 6.

[6] I. Gonzalez-Diaz and F. Diaz-de Maria, "Adaptive multipattern fast block-matching algorithm based on motion classification techniques,"

Circuits and Systems for Video Technology, IEEE Transactions on, vol. 18, no. 10, pp. 1369 –1382, Oct. 2008.
[7] C. Grecos and M. Yang, "Fast mode prediction for the baseline and main profiles in the H.264 video coding standard," *Multimedia, IEEE Transactions on*, vol. 8, no. 6, pp. 1125 –1134, Dec. 2006.
[8] J. You, W. Kim, and J. Jeong, "16x16 macroblock partition size prediction for H.264 P slices," *Consumer Electronics, IEEE Transactions on*, vol. 52, no. 4, pp. 1377 –1383, Nov. 2006.
[9] T.-Y. Kuo and C.-H. Chan, "Fast variable block size motion estimation for H.264 using likelihood and correlation of motion field," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 16, no. 10, pp. 1185 –1195, Oct. 2006.
[10] A. Saha, K. Mallic, J. Mukherjee, and S. Sural, "SKIP prediction for fast rate distortion optimization in H.264," *Consumer Electronics, IEEE Transactions on*, vol. 53, no. 3, pp. 1153 –1160, Aug. 2007.
[11] C. Zhou, Y. Tan, J. Tian, and Y. Lu, "3 σ-rule-based early termination algorithm for mode decision in H.264," *Electronics Letters*, vol. 45, no. 19, pp. 974 –975, Sept. 2009.
[12] E. Martinez-Enriquez, A. Jimenez-Moreno, M. Angel-Pellon, and F. Diaz-de Maria, "A two-level classification-based approach to inter mode decision in H.264/AVC," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 21, no. 11, pp. 1719 –1732, Nov. 2011.
[13] E. Martinez-Enriquez, M. de Frutos-Lopez, J. Pujol-Alcolado, and F. Diaz-de Maria, "A fast motion-cost based algorithm for h.264/avc inter mode decision," *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 5, pp. V –325–V –328, 16 2007-Oct. 19 2007.
[14] H. Ates and Y. Altunbasak, "Rate-distortion and complexity optimized motion estimation for H.264 video coding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 2, pp. 159 –171, Feb. 2008.
[15] X. Gao, K. M. Lam, L. Zhuo, and L. Shen, "Complexity scalable control for H.264 motion estimation and mode decision under energy constraints," *Signal Processing*, vol. 90, no. 8, pp. 2468 – 2479, 2010.
[16] C. Kannangara, I. Richardson, and A. Miller, "Computational complexity management of a real-time H.264/AVC encoder," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 9, pp. 1191 –1200, Sept. 2008.
[17] E. Huijbers, T. Ozcelebi, and R. Bril, "Complexity scalable motion estimation control for H.264/AVC," in *Consumer Electronics (ICCE), 2011 IEEE International Conference on*, Jan. 2011, pp. 49 –50.
[18] R. Vanam, E. Riskin, S. Hemami, and R. Ladner, "Distortion-complexity optimization of the H.264/MPEG-4 AVC encoder using the gbfos algorithm," in *Data Compression Conference, 2007. DCC '07*, Mar. 2007, pp. 303–312.
[19] R. Vanam, E. Riskin, and R. Ladner, "H.264/MPEG-4 AVC encoder parameter selection algorithms for complexity distortion tradeoff," in *Data Compression Conference, 2009. DCC '09.*, Mar. 2009, pp. 372 –381.
[20] L. Su, Y. Lu, F. Wu, S. Li, and W. Gao, "Complexity-constrained H.264 video encoding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, no. 4, pp. 477 –490, Apr. 2009.
[21] Y. H. Tan, W. S. Lee, J. Y. Tham, S. Rahardja, and K. M. Lye, "Complexity scalable H.264/AVC encoding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 9, pp. 1271 –1275, Sept. 2010.
[22] C. Kannangara, I. Richardson, M. Bystrom, and Y. Zhao, "Complexity control of H.264/AVC based on mode-conditional cost probability distributions," *Multimedia, IEEE Transactions on*, vol. 11, no. 3, pp. 433 –442, Apr. 2009.
[23] T. da Fonseca and R. de Queiroz, "Complexity-constrained H.264 hd video coding through mode ranking," in *Picture Coding Symposium, 2009. PCS 2009*, May. 2009, pp. 1 –4.
[24] T. A. da Fonseca and R. L. de Queiroz, "Complexity-constrained rate-distortion optimization for h.264/avc video coding," in *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, May. 2011, pp. 2909 –2912.
[25] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *Signal Processing Magazine, IEEE*, vol. 15, no. 6, pp. 23 –50, Nov. 1998.
[26] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers [speech coding]," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 36, no. 9, pp. 1445 –1453, Sept. 1988.
[27] E. Martinez-Enriquez, A. Jimenez-Moreno, and F. Diaz-de Maria, "A novel fast inter mode decision in H.264/AVC based on a regionalized hypothesis testing," in *Picture Coding Symposium, 2009. PCS 2009*, May. 2009, pp. 217–220.
[28] ——, "An adaptive algorithm for fast inter mode decision in the H.264/AVC video coding standard," *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 2, pp. 826 –834, May. 2010.
[29] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E*, vol. 69, no. 6, p. 066138, Jun. 2004.
[30] JVT H.264/AVC reference software v.10.2 [online], "http://iphome.hhi.de/suehring/tml/download/old_jm/".
[31] G.Sullivan, "Recommended simulation common conditions for H.26L coding efficiency experiments on low-resolution progressive-scan source material," ITU-T, VCEG-N81, Sept. 2001.
[32] G.Bjontegaard, "Calculation of average PSNR differences between RD-curves," ITU-T, VCEG-M33, Apr. 2001.
[33] High Efficiency Video Coding [online], "http://hevc.hhi.fraunhofer.de/".

Amaya Jiménez-Moreno received the Telecommunications Engineering degree from Universidad Carlos III de Madrid, Madrid, Spain in 2012. She is currently working toward a Master's degree in the same university. Her research interests include video coding optimization and high definition video coding.

Eduardo Martínez-Enríquez (SM'07) received the Telecommunications Engineering degree from Universidad Politécnica de Madrid, Madrid, Spain, in 2006. He is currently working toward the Ph.D. degree at the Universidad Carlos III de Madrid, Madrid, Spain. His research interests include lifting transforms on graphs, wavelet-based video coding and video coding optimization. He received the Best Paper Award of ICIP in 2011 for his paper on video coding based on lifting transform on graphs, co-authored with Fernando Díaz-de-María and Antonio Ortega.

Fernando Díaz-de-María (M'97) received the Telecommunication Engineering degree and the Ph.D. degree from the Universidad Politécnica de Madrid, Madrid, Spain, in 1991 and 1996, respectively. Since October 1996, he has been an Associate Professor in the Department of Signal Processing and Communications, Universidad Carlos III de Madrid, Madrid, Spain. His primary research interests include robust speech recognition, video coding, and video analysis. He has led numerous projects and contracts in the fields mentioned. He is co-author of several papers in peer-reviewed international journals, two book chapters, and has presented a number of papers in national and international conferences.