Available online at www.sciencedirect.com

SCIENCE DIRECT®

Signal Processing: *Image Communication* 18 (2003) 801–811

SIGNAL PROCESSING: IMAGE COMMUNICATION

www.elsevier.com/locate/image

Macroblock classification for complexity management of video encoders

Yafan Zhao*, Iain E.G. Richardson

School of Engineering, The Robert Gordon University, Schoolhill, Aberdeen AB10 1FR, UK

Received 7 January 2003; received in revised form 6 May 2003; accepted 13 June 2003

Abstract

Typically, many macroblocks (MBs) are skipped during encoding of H.263 or MPEG-4 SP video data, particularly at low bit-rates. In this paper, we describe an algorithm that predicts the occurrence of skipped MBs prior to encoding, making it possible to save significant computational effort by not coding these MBs. The algorithm estimates the energy of low-frequency quantized coefficients in order to classify each MB as 'skipped' or 'not skipped'. Results show that the algorithm can deliver substantial computational savings at the expense of a small reduction in rate-distortion performance.

1. Introduction

Video CODECs based on the H263 [5] and MPEG-4 [3] video coding standards are used in a wide range of applications. Software-only CODECs are becoming particularly popular, offering advantages such as flexibility, ease of upgrading and distribution. In real-time and/or power-constrained applications, the performance of a video CODEC may be limited by the amount of processing power available as well as, or rather than, the available transmission bandwidth. In a desktop video conferencing system, the CODEC runs on a general-purpose PC and has to share processing resources with other applications. In a mobile video handset, power consumption is closely related to processor utilization and it may be necessary to restrict computational processing in order to maximize battery life. Current software video applications typically control processor utilization by dropping frames during encoding, leading to intermittent and 'jerky' motion in the decoded video sequence. Hence, computational complexity can be a major constraint on coding performance. It is therefore important to develop methods of managing the computational complexity of video CODECs.

Previous work on reducing computational complexity of video CODECs has included many proposals for 'fast search' motion estimation algorithms [6,7]. Adaptive motion estimation algorithms such as Nearest Neighbour Search [1] provide coding performance that is close to that of Full Search with greatly reduced complexity. However, the computational cost of this type of algorithm can vary significantly depending on the scene characteristic of the video sequence. Several

---

*Corresponding author. Fax: +44-1224-262-444.
E-mail address: 9905854@rgu.ac.uk (Y. Zhao).

0923-5965/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0923-5965(03)00072-9

methods [10,2] have been proposed to reduce the computational complexity in the discrete cosine transform (DCT) by calculating a subset of the DCT coefficients. Applying these methods to all blocks in an image will significantly reduce the quality of the decoded image and this has led to proposals for algorithms that selectively calculate the DCT based on sequence statistics [11]. In [9], we describe an algorithm that enables flexible and accurate management of DCT complexity in a software video encoder and a similar approach to motion estimation complexity is presented in [14]. The initial calculation of sum of absolute differences (SAD) for zero motion vector (MV) is used to reduce motion estimation complexity in [13] and the authors report that this method performs well together with DCT computation reduction. The results presented in these papers show that variable-complexity algorithms can reduce computational complexity, usually at the expense of increased distortion.

Many coded macroblocks (MBs) in an inter-coded frame have zero MVs and/or quantized coefficients (Q_{Coeff}). An MB with zero MV and no non-zero coefficients is skipped. If these skipped MBs could be accurately predicted prior to encoding, all subsequent operations on these MBs (motion estimation and compensation, DCT, quantization, etc.) could be avoided, saving considerable computational effort.

This paper presents a pre-classification algorithm that categorizes MBs into two types, 'skipped' and 'not skipped', prior to encoding. Computational complexity can be reduced by not processing MBs that are expected to be skipped. The proposed algorithm provides a simple, controllable and robust method of managing computational resources. It can control the amount of reduced computation whilst minimizing distortion due to occasional incorrect MB classification.

This paper is organized as follows. Section 2 introduces the concepts of MB classification and the trade-off between computational complexity and rate-distortion performance. Section 3 describes the development of a pre-classification algorithm based on measured characteristics of an input MB and Section 4 presents simulation results for the proposed algorithm. Conclusions and suggestions for further work are given in Section 5.

2. Distribution of macroblock types

A CODEC that conforms to one of the popular DCT-based video coding standards (such as H.263 or MPEG-4 Simple Profile) processes each frame in units of an MB. In an inter-coded picture, motion estimation is carried out in order to find a suitable prediction for the current MB from a reference frame. Each block of the motion-compensated residual MB is coded using the DCT, quantization, reordering and entropy coding. MVs and quantized coefficients are encoded together with side information and the MB is reconstructed for prediction of further pictures.

Four video sequences ('Carphone', 'Mother and Daughter', 'Foreman' and 'Claire') were encoded using an H.263+encoder (corresponding to the low complexity mode of test model TMN10 [12]; hereafter described as 'TMN10') with a fixed quantizer step size. Coded MBs in P-pictures were categorized into four types: 'skipped' (zero MV, no non-zero Q_{Coeff}), '$\mathrm{MV}=0$' (zero MV, some non-zero Q_{Coeff}), '$Q_{\mathrm{Coeff}}=0$' (non-zero MV, no non-zero Q_{Coeff}) and 'other' (non-zero MV and non-zero Q_{Coeff}). Fig. 1 shows the distribution of the four categories (a) for 'Carphone' encoded with two quantizer step sizes and (b) for the remaining three sequences encoded with a quantizer step size of 8. Fig. 1(a) demonstrates that the proportion of skipped MBs increases with quantizer step size and Fig. 1(b) shows that low-motion sequences such as 'Claire' and 'Mother and Daughter' contain a higher proportion of skipped MBs than high-motion sequences such as 'Carphone' and 'Foreman'. With the exception of 'Foreman', the majority of MBs in each sequence (i) contain no MVs, (ii) contain no coefficients or (iii) are skipped.

It is clear from Fig. 1 that a significant proportion of MBs are skipped (not coded), particularly in low-motion sequences and/or at higher quantizer step sizes (and hence lower bit-rates). Predicting the presence of a skipped MB prior to coding could make it possible to save

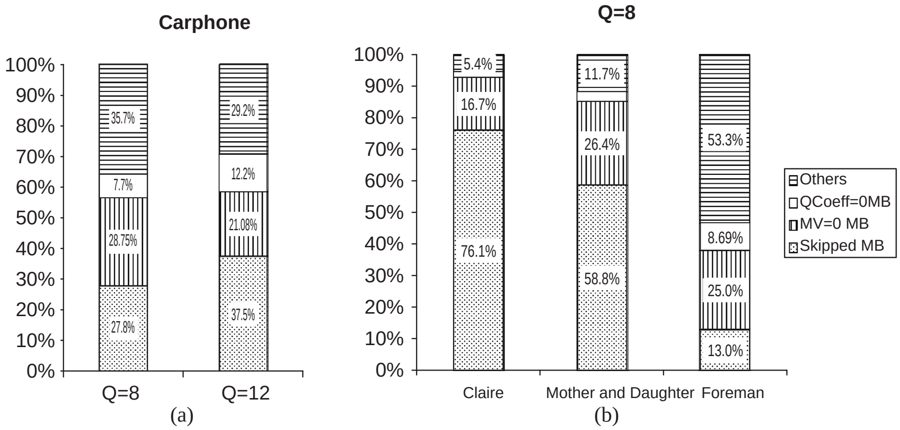


Fig. 1. Distribution of four types of MBs in video sequences: (a) Carphone; and (b) $Q = 8$.

considerable computational resources by not carrying out computationally intensive functions. We propose the following approach to MB classification:

1. Prior to encoding, classify each inter-coded MB as ‘skipped’ or ‘not skipped’ by prediction from local sequence statistics.
2. If the MB is predicted as ‘not skipped’, carry out the usual encoding functions (motion estimation and compensation, DCT, quantization, rescaling, IDCT, reconstruction, reordering, run-level coding, entropy coding).
3. If the MB is predicted as ‘skipped’, indicate the presence of a skipped MB in the bitstream; no further processing is carried out.

If the prediction of MB type is correct, computational complexity is reduced without any effect on decoded video quality. If an MB that should have been encoded (i.e. an MB that contains non-zero MV and/or Q_{Coeff} after encoding) is predicted as ‘skipped’, a reduction in decoded quality is likely to occur. Computational complexity reduction may therefore lead to increased distortion. We have argued previously [9] that a small reduction in PSNR is an acceptable penalty for reduced computation. A limited degradation in PSNR (less than 1 dB) is difficult to distinguish subjectively [8]. Furthermore, in a real-time video application, maintaining a consistent video frame rate through computational complexity management (at the expense of limited reduction in PSNR) is likely to be preferable to a ‘jerky’ decoded video sequence due to an encoder dropping frames.

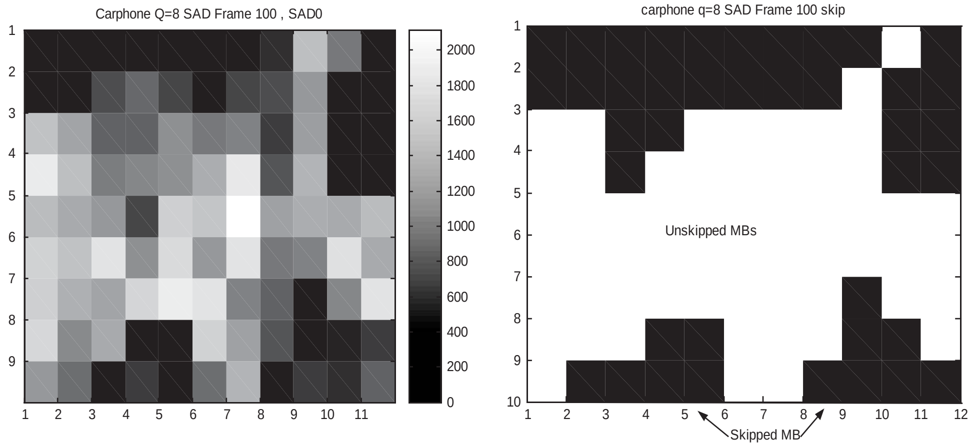
3. Macroblock classification algorithms

3.1. Correlation between residual energy and probability of skipped macroblock

MBs that are skipped have zero MV and Q_{Coeff}. This means that (a) the closest matching region is in the same position in the reference frame and (b) the energy of the residual MB (after subtracting the reference region) is low, such that there are no non-zero DCT coefficients after quantization. Both of these conditions are likely to be met if there is a strong similarity between the current MB and the same MB position in the reference frame. The energy of the residual MB formed by subtracting the reference MB (without motion compensation) from the current MB is approximated by $\mathrm{SAD0}_{\mathrm{MB}}$ (SAD for luminance part of MB, zero displacement):

$$\mathrm{SAD0}_{\mathrm{MB}} = \sum_{i=0}^{15} \sum_{j=0}^{15} |C_{\mathrm{C}}(i,j) - C_{\mathrm{P}}(i,j)|, \tag{1}$$

$C_{\mathrm{C}}(i,j)$ and $C_{\mathrm{P}}(i,j)$ are luminance samples from an MB in the current frame and in the same position in the reference frame, respectively.



(a) (b)

Fig. 2. SAD0_{MB} and skipped MB (Carphone, $Q = 8$, Frame 100): (a) SAD0_{MB}; and (b) skipped MB.

SAD0_{MB} represents the total energy of a residual MB. A skipped MB has low residual energy and a zero MV and so we might expect to find a correlation between SAD0_{MB} and $P(\text{skip})$, the probability of skipping the current MB. Fig. 2(a) plots SAD0_{MB} for frame 100 of the 'Carphone' sequence (QCIF format, encoded with a fixed quantizer step size of 8) and Fig. 2(b) plots the distribution of skipped MBs after coding this frame using TMN10. It is clear that skipped MBs often have low values of SAD0_{MB}. It may therefore be possible to use SAD0_{MB} to predict whether an MB is likely to be skipped.

3.2. Classifying skipped MBs using $SAD0_{MB}$

The 'Carphone' sequence was coded using TMN10 with fixed quantizer step size of 8. The dotted region of Fig. 3 plots SAD0_{MB} (x-axis) against $P(\text{skip})$, the probability that an MB with a given SAD0_{MB} will be skipped (y-axis). The solid line plots the cumulative density (CDF) of SAD0_{MB}.

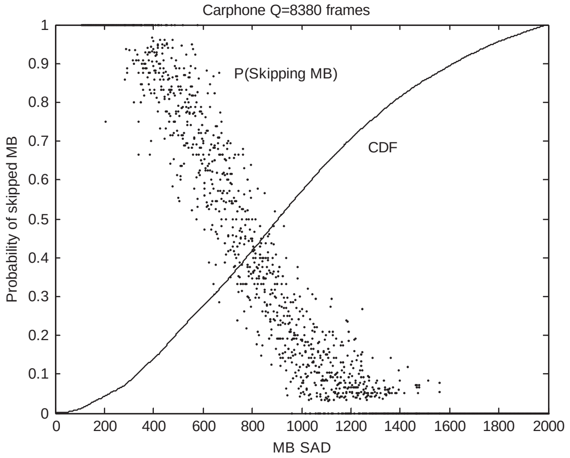


Fig. 3. Probability of skipping MB with $Q = 8$.

Fig. 3 indicates that low SAD0_{MB} correlates with high $P(\text{skip})$. The relationship between SAD0_{MB} and $P(\text{skip})$ also depends on the quantizer step size since a higher step size results in an increased proportion of skipped MBs (as shown in Fig. 1). Fig. 4 plots $P(\text{skip})$ against $(\text{SAD0}_{\text{MB}}/Q)$ for the 'Carphone' sequence with various quantizer step sizes Q. The plots of $P(\text{skip})$ in this figure are smoothed with a moving average operator to facilitate comparison of the different sequences.

It may be possible to use $\text{SAD0}_{\text{MB}}/Q$ as a predictor to determine whether a given MB is likely to be skipped. An algorithm for MB classification can be described in +code as follows:

```
IF $\text{SAD0}_{\text{MB}}/Q < T_{\text{sad0}}$
THEN skip coding this MB and set MV and $Q_{\text{Coeff}}$
  to zero
ELSE continue coding
```

When $SAD0_{MB}/Q$ is less than a pre-determined threshold, the encoder skips coding this MB and sets MV and Q_{Coeff} to zero directly, hence reducing the processing required for the MB.

3.3. Improving the accuracy of the classification algorithm

The classification algorithm described in the previous section may not produce the correct classification for every MB. Fig. 3 indicates that P(skip) is less than 1.0 in many cases. Whatever threshold value T_{sad0} is chosen, there may be some MBs that have a value of $SAD0_{MB}$ below this threshold but should not be skipped. The classification algorithm will fail in these cases, causing MBs to be erroneously skipped and increasing distortion in the decoded video sequence. Fig. 5 shows one example of an MB falling into this category. MB(8,4) in Frame 18 of 'Carphone' has a low value of $SAD0_{MB}$ (5 3 5) but is not skipped.

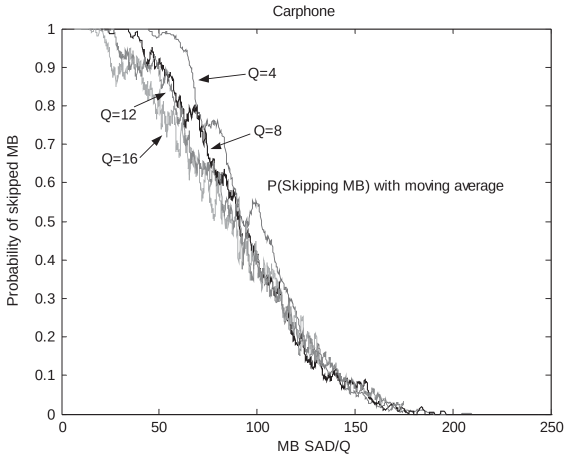
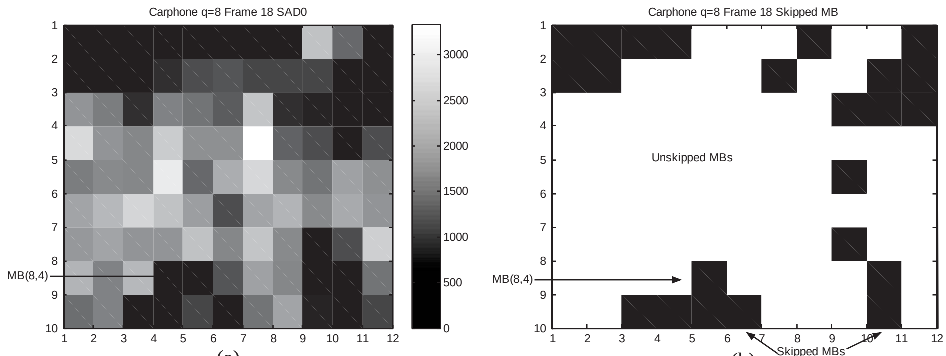


Fig. 4. Probability of skipping MB with normalized $SAD0_{MB}$.

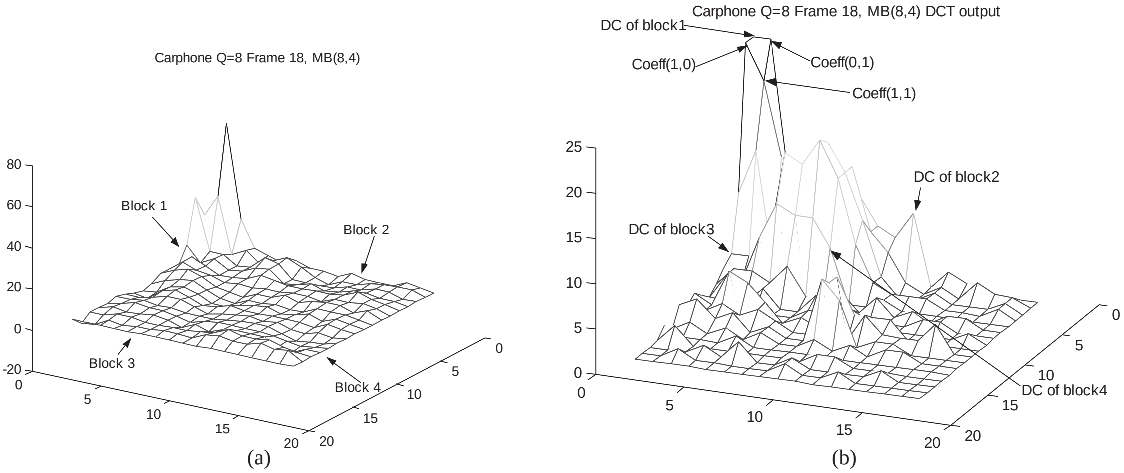
Examination of MBs that fall into this category shows that these MBs often occur on the edges of moving areas. The residual (without motion compensation) typically contains a small number of high-valued samples, not enough to cause a significant increase in SAD but enough to produce non-zero quantized coefficients. Fig. 6 shows the residual and DCT output for the luminance part of MB(8,4) in frame 18 of 'Carphone'. The residual (a) contains a single high-valued region (corresponding to a moving edge in the video sequence). The unquantized DCT coefficient magnitude of the four luminance blocks (b) shows that the top-left block has a number of significant coefficients. In particular, the four low-frequency DCT coefficients (the DC coefficient and the three lowest AC coefficients) have large magnitudes.

A high-magnitude DC coefficient tends to produce a correspondingly high $SAD0_{MB}$ (since the DC coefficient is proportional to the mean sample value of each block). However, the three lowest-frequency AC coefficient magnitudes are not reflected in the calculation of $SAD0_{MB}$. We therefore propose a low-complexity method of estimating the magnitude of these coefficients (without actually carrying out the DCT) and this is illustrated in Fig. 7.



(a) (b)

Fig. 5. Example of low-$SAD0_{MB}$ value MB (Carphone, $Q = 8$, Frame 18, MB (8, 4)): (a) $SAD0_{MB}$; and (b) skipped MB.



(a) (b)

Fig. 6. Example of DCT process of (Carphone, $Q = 8$, Frame 18, MB (8, 4)): (a) residual value; and (b) absolute DCT output.

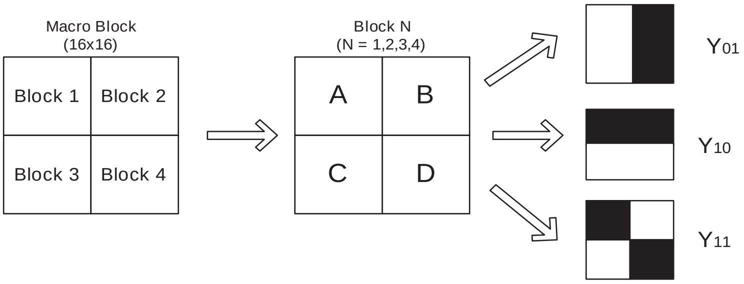


Fig. 7. Predicting the three low-frequency components.

Each 8×8 luminance block is divided into four 4×4 blocks. A, B, C and D (Eq. (2)) are the SAD values of each 4×4 block and $R(i,j)$ are the residual pixel values without motion compensation

$$A = \sum_{i=0}^{3}\sum_{j=0}^{3}|R(i,j)| \quad B = \sum_{i=0}^{3}\sum_{j=3}^{7}|R(i,j)|$$
$$C = \sum_{i=4}^{0}\sum_{j=3}^{7}|R(i,j)| \quad D = \sum_{i=4}^{7}\sum_{j=4}^{7}|R(i,j)| \tag{2}$$

$Y_{01,}$ Y_{10} and Y_{11} (Eq. (3)) provide a low-complexity estimate of the magnitudes of the three low-frequency DCT coefficients coeff(0,1), coeff(1,0) and coeff(1,1), respectively. If any of these coefficients is large then there is a high probability that the MB should not be skipped. We therefore use $Y4 \times 4_{\text{block}}$ (Eq. (4)) to predict whether each block may be skipped. The maximum for the luminance part of an MB is calculated using Eq. (5)

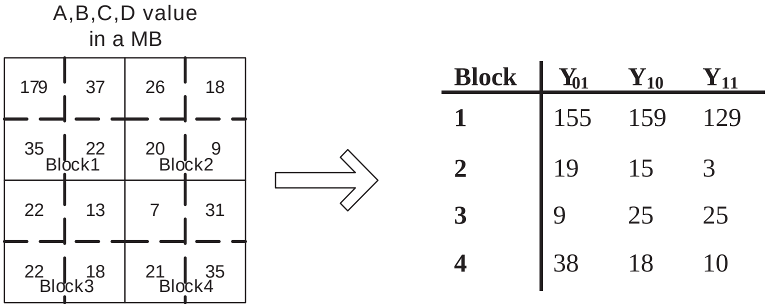
$$\begin{aligned} Y_{01} &= \text{abs}(A + C - B - D), \\ Y_{10} &= \text{abs}(A + B - C - D), \\ Y_{11} &= \text{abs}(A + D - B - C), \end{aligned} \tag{3}$$

$$Y4 \times 4_{\text{block}} = \max(Y_{01}, Y_{10,} Y_{11}), \tag{4}$$

$$Y4 \times 4_{\max} = \max(Y4 \times 4_{\text{block1}}, Y4 \times 4_{\text{block2}}, Y4 \times 4_{\text{block3}}, Y4 \times 4_{\text{block4}}). \tag{5}$$

The calculated value of $Y4 \times 4_{\max}$ is compared with a threshold $T_{4\times4}$ to improve the accuracy of skip prediction. The MB classification algorithm becomes:

```
IF $\text{SAD0}_{\text{MB}}/Q < T_{\text{sad0}}$ and $Y4 \times 4_{\max} < T_{4\times4}$
THEN skip coding this MB and set MV and $Q_{\text{Coeff}}$
  to zero
ELSE continue coding
```



Block	Y_{01}	Y_{10}	Y_{11}
1	155	159	129
2	19	15	3
3	9	25	25
4	38	18	10

Fig. 8. Example of predicting the three low-frequency components (Carphone, $Q = 8$, Frame 18, MB(8, 4)).

Fig. 8 shows the results of calculating Y_{01}, Y_{10} and Y_{11} for MB(8,4) in frame 18 of 'Carphone'. Using Eqs. (3)–(5), $Y4 \times 4_{\max}$ of this MB is calculated to be 159. If it is greater than the threshold $T_{4\times4}$, it will not be skipped during encoding.

3.4. Choosing threshold $T_{4\times4}$

The video sequences 'Carphone', 'Foreman' and 'Mother and Daughter' were coded using the modified classification algorithm with various values of quantizer stepsize Q and $T_{4\times4}$. For each fixed value of Q, a range of values of $T_{4\times4}$ were tested and the choice of $T_{4\times4}$ resulting in minimum distortion was recorded. The results of this experiment indicate that the optimal choice of $T_{4\times4}$ is approximately linearly related to Q (Eq. (6))

$$T_{4\times4} = 10Q + 70. \tag{6}$$

Incorporating Eq. (6) into the pre-classification algorithm gives the following:

```
IF SAD0_MB/Q < T_sad0 and (Y4 × 4_max − 70)/Q < 10
THEN skip coding this MB and set MV and Q_Coeff
  to zero
ELSE continue coding
```

Applying this algorithm to MB(8,4) in frame 18 of 'Carphone' sequence, where $Y4 \times 4_{\max} = 159$ and $Q = 8$:

$$(Y4 \times 4_{\max} - 70)/Q = (159 - 70)/8 = 11.125.$$

This is greater than 10 and so the MB is not skipped, regardless of the value of $\mathrm{SAD0}_{\mathrm{MB}}$.

4. Experimental results

4.1. Performance of MB prediction

Video sequences 'Carphone', 'Mother and Daughter' and 'Forman' were coded using a modified TMN10 encoder, incorporating our MB classification algorithms (a) T_{sad0} only and (b) $T_{\mathrm{sad0}} + T_{4\times4}$ with 280 frames and $Q = 8$. Fixed thresholds T_{sad0} and $T_{4\times4}$ were chosen for the two algorithms to achieve the same target percentage of skipped MBs. For each algorithm, the total number of skipped MBs, the number of 'missed' MBs (MBs that are skipped by the unmodified encoder but were not predicted as skipped by our algorithm) and the number of MBs wrongly skipped by our algorithm are summarized in Table 1. The average luminance PSNR drop relative to TMN10 encoder with no complexity reduction is also listed in Table 1 for each case. In order to evaluate the performance of the proposed classification algorithms, each sequence was coded by TMN10 and MPEG-4 VM18 [4] (Simple Profile) without any classification algorithms. The percentage of skipped MBs and average luminance PSNR of each sequence are shown in Table 2(a), (b), and (c) respectively.

From Table 2(a), 7712 MBs are skipped during encoding of 'Carphone' using TMN10 (27.82% of total MBs). Using MPEG-4 VM18 with the same coding parameters, 7409 MBs (26.72% of total MBs) are skipped during encoding. For the sequences listed in Table 2, TMN10 and MPEG-4 skip a similar number of MBs but MPEG-4

Table 1
Performance of skipping MBs prediction using two classification algorithms

Video sequence	Target skipped MBs (%)	Classification algorithms	Total skipped	Missed	Wrongly predicted	Average Y PSNR drop (dB)
Carphone	30	T_{sad0} only	8312	1585	2185	0.22
		$T_{\text{sad0}}+T_{4\times4}$	8329	1532	2149	0.19
	45	T_{sad0} only	12,413	523	5224	0.63
		$T_{\text{sad0}}+T_{4\times4}$	12,389	431	5108	0.51
Mother and daughter	45	T_{sad0} only	8298	8209	222	0.14
		$T_{\text{sad0}}+T_{4\times4}$	8261	8226	202	0.08
	70	T_{sad0} only	19,411	1041	4167	1.05
		$T_{\text{sad0}}+T_{4\times4}$	19,368	1011	4094	0.58
Foreman	30	T_{sad0} only	8288	318	4998	0.64
		$T_{\text{sad0}}+T_{4\times4}$	8264	280	4936	0.41

Table 2
Number of MBs skipped by TMN10 and MPEG-4

	TMN10	MPEG-4
(a) ‘*Carphone*’ *sequence*		
Skipped MB	7712 (27.82%)	7409 (26.72%)
Average Y PSNR(dB)	34.10	34.32
(b) ‘*Mother and daughter*’ *sequence*		
Skipped MB	16285 (58.75%)	16288 (58.75%)
Average Y PSNR(dB)	34.95	35.28
(c) ‘*Foreman*’ *sequence*		
Skipped MB	3608 (13%)	3347 (12.07%)
Average Y PSNR(dB)	33.37	33.65

VM18 outperforms TMN10 in terms of average PSNR.

Table 1 shows that the algorithm using $T_{\text{sad0}}+T_{4\times4}$ ‘misses’ fewer MBs than the algorithm using T_{sad0} alone for all the video sequences, i.e. $T_{\text{sad0}}+T_{4\times4}$ provides a more accurate prediction than T_{sad0} only. For the ‘Carphone’ sequence, the maximum percentage of skipped MBs by TMN10 is 27.82% (from Table 2(a)). Hence, when the skipped MB target is set to 45% for ‘Carphone’, an extra 5224 MBs are skipped by $T_{\text{sad0}}+T_{4\times4}$. These extra skipped MBs cause a degradation of video quality. However, because our algorithms select MBs with low residual energy, the loss of video quality is minimized.

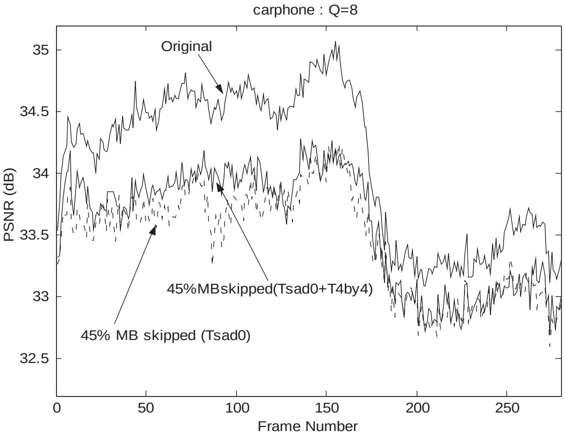


Fig. 9. PSNR against frame number for ‘Carphone’ with same complexity reduction by two types of classification algorithms.

4.2. *Video quality*

4.2.1. *Objective quality*

Fig. 9 shows the PSNR of each frame of ‘Carphone’ with $Q=8$, 45% skipped MBs and fixed thresholds T_{sad0} and $T_{4\times4}$. The thresholds were chosen to give a 45% mean reduction in computational complexity (i.e. 45% skipped MBs) in each case. This figure shows that $T_{\text{sad0}}+T_{4\times4}$ consistently achieves better video quality than T_{sad0} alone, for a comparable reduction in computational complexity.

The ‘Carphone’ video sequence was coded using two types of MB classification algorithms. At each fixed quantizer, T_{sad0} was varied from low to high to parameterize the performance of the algorithm at a range of computational complexity values. Fig. 10 plots the percentage of MBs classified as 'skipped' by our algorithm (x-axis) against the mean drop in PSNR compared with the same sequence encoded without complexity reduction (y-axis). This drop in PSNR is caused by occasional incorrect skipping of MBs that should have been encoded. As T_{sad0} increases, the graph moves to the right and up: more MBs are skipped but the PSNR drop increases.

This figure shows that classifying MBs using two thresholds, $T_{\text{sad0}} + T_{4\times4}$, produce better video quality than T_{sad0} alone. For example, using only T_{sad0} to classify MBs, the algorithm skips 40% of MBs at the expense of a PSNR drop of 0.5 dB (for a fixed quantizer step size $Q = 8$). Using $T_{\text{sad0}} + T_{4\times4}$, approximately 44% of MBs can be skipped without any further PSNR drop. When $Q = 16$, skipping approximately 46% of MBs produces a PSNR drop of 0.5 dB using only T_{sad0}; approximately 56% may be skipped for the same PSNR drop using $T_{\text{sad0}} + T_{4\times4}$.

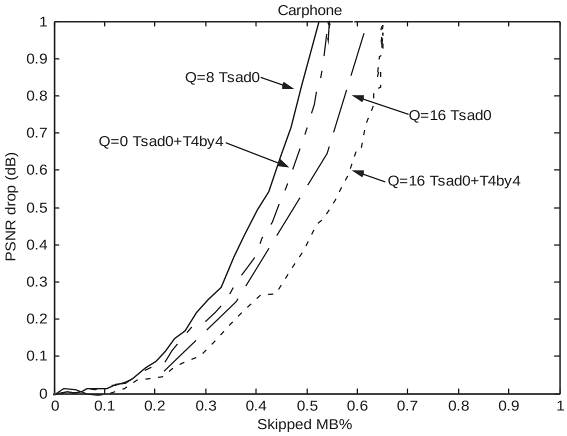


Fig. 10. Performance comparison of various Q and two types of classification algorithms.

4.2.2. Subjective quality

Fig. 11 compares frame 87 of the 'Carphone' sequence after encoding and decoding using three methods: (a) without any complexity reduction, (b) after skipping 45% of the MBs using T_{sad0} only and (c) after skipping 45% of the MBs using $T_{\text{sad0}} + T_{4\times4}$. Fig. 11(b) shows some visible degradation compared with Fig. 11(a), with blocking artefacts around the top of the head introduced by the complexity reduction algorithm (note that these artefacts occur at the boundary of a moving object, as reported in Section 3.3). Fig. 11(c) is very similar to Fig. 11(a) with little apparent increase in distortion. When played back at 30 frames per second, there is a slight degradation in sequence (b) compared with (a) but no obvious difference between (c) and (a). An interesting feature of the ($T_{\text{sad0}} + T_{4\times4}$) algorithm is that MBs classified as 'skipped' are typically in low-activity regions of the scene (such as the background of 'Carphone') and incorrectly skipped MBs in this region tend not to be noticed by the viewer. Hence the subjective quality is often better than the PSNR results might imply.

4.3. Rate-distortion performance

Fig. 12 plots the rate-distortion performance of the 'Carphone' sequence after encoding and

Fig. 11. Sample frame 87 of decoded 'Carphone' sequence with $Q = 8$ and 45% skipped MB: (a) original; (b) with T_{sad0} only; and (c) $T_{\text{sad0}} + T_{4\times4}$.

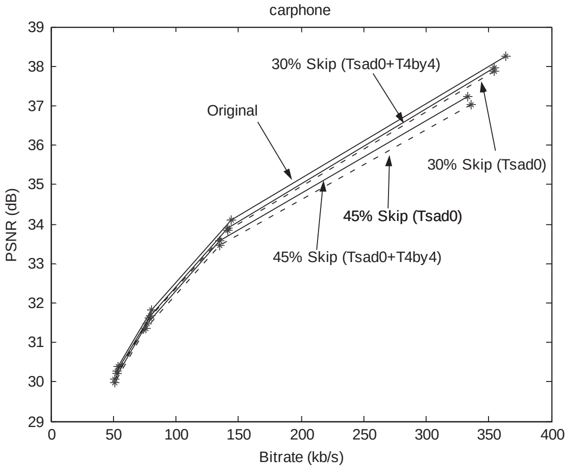


Fig. 12. Rate-distortion performance.

decoding using the H.263 TMN-8 reference model operating in Baseline mode (Original). Two methods (T_{sad0} only and $T_{\mathrm{sad0}} + T_{4\times4}$) are used to achieve two target complexity reduction levels (30% skipped MBs and 45% skipped MBs) and the corresponding rate-distortion curves are plotted in Fig. 12. The ($T_{\mathrm{sad0}} + T_{4\times4}$) algorithm outperforms the T_{sad0} algorithm at both complexity targets. Using the ($T_{\mathrm{sad0}} + T_{4\times4}$) method we can reduce complexity by 30% with a very slight degradation in rate-distortion performance.

4.4. Computational complexity

The purpose of our proposed algorithm is to reduce the computational complexity during encoding and so it is important to identify any additional computational costs due to the classification algorithm. $\mathrm{SAD0_{MB}}$ is normally computed in the first step of any motion estimation algorithm and so there is no extra calculation required. Furthermore, the SAD values of each 4×4 block (A, B, C and D in Eq. (2)) may be calculated without penalty if $\mathrm{SAD0_{MB}}$ is calculated by adding together the values of SAD for each 4×4-sample sub-block in the MB.

The additional computational requirements of the classification algorithm are the operations in Eqs. (3)–(5). For each MB consisting of four 8×8 blocks, there are $12(=4 \times 3)$ additions, $24(4 \times 6)$ subtractions and 12 comparisons. This is negligible compared with the cost of motion estimation, DCT, etc. for each MB and so the computational saving of our proposed algorithm significantly outweighs the small computational cost of classifying MBs.

5. Conclusions

In this paper we have described an MB classification algorithm that has the potential to significantly reduce computational complexity in a software video encoder. The goal of this algorithm is to avoid expensive processing of MBs that are destined to be skipped during encoding, whilst minimizing distortion due to incorrect classification predictions. The initial SAD calculation $\mathrm{SAD0_{MB}}$ can give an approximate prediction of MB type (as reported in [13]) but incorrectly predicts non-skipped MBs that contain significant low-frequency AC coefficients after quantization. Adding a prediction of low-frequency residual energy improves the accuracy of MB classification, particularly for MBs at the edge of moving regions.

Experimental results show that our classification algorithm can deliver substantial computational savings (40–50% for a high-activity sequence such as ‘Carphone’, more for low activity sequences) with only a small reduction in rate-distortion performance. The reduction in subjective quality is actually smaller than the PSNR results imply, since our algorithm tends to classify skipped MBs in static, visually unimportant regions of the scene. The classification process is computationally simple and most of the operations may be integrated into the normal motion estimation process.

In this paper, the classification algorithm was applied to H.263 but it is equally applicable to MPEG-4 Simple Profile and other MB-based CODECs. It is planned to incorporate other classification methods with this algorithm (e.g. the DCT block classification algorithm described in [9]) in order to avoid unnecessary computation for MBs in other categories such as ‘$Q_{\mathrm{Coeff}} = 0$’ and ‘MV = 0’ (Fig. 1) and to examine the effect of adding a control algorithm such as the feedback-based computational complexity control mechanism described in [9].

References

[1] M. Gallant, G. Côté, F. Kossentini, An efficient computation-constrained block-based motion estimation algorithm for low bitrate video coding, IEEE Trans. Image Process. 8 (12) (December 1999) pp. 1816–1822.

[2] A. Hossen, U. Heute, Fast approximate DCT, in: Proceedings of the ICASSP97, Munich, 1997.

[3] ISO/IEC 14496-2, Information technology—generic coding of audio-visual objects—Part 2: Visual, MPEG4, 1999.

[4] ISO/IEC JTC1/SC29/WG11, MPEG-4 video verification model version 18, January 2001.

[5] ITU-T Recommendation H.263, Video coding for low bit rate communication, 1998.

[6] J. Jain, A. Jain, Displacement measurement and its application in interframe image coding, IEEE Trans. Commun. COM-29 (December 1981) pp. 1799–1808.

[7] K.I.T. Koga, A. Hirano, Y. Iijima, T. Ishiguro, Motion-compensated interframe coding for video conferencing, in: Proceedings of the NTC 81, Vol. 28, New Orleans, December 1992.

[8] I. Rhee, Error control techniques for interactive low-bit rate video transmission over the internet, ACM SIGCOMM'98, Vancouver, Canada.

[9] I.E.G. Richardson, Y. Zhao, Adaptive management of video encoder complexity, J. Real-time Imag. 8 (4) (August 2002) 291–301.

[10] A. Skodras, Fast discrete cosine transform pruning, IEEE Trans. Signal Process. 42 (1994) pp. 1833–1838.

[11] M.-T. Sun, I.-M. Pao, Statistical computation of discrete cosine transform in video encoders, J. Visual Commun., Image Repr. 9 (1998) pp. 163–170.

[12] Video Coding Experts Group, ITU, Video codec test model, Version 10 (TMN10), April 1998.

[13] J.-F. Yang, S.-C. Chang, C.-Y. Chen, Computation reduction for motion search in low rate video coders, IEEE Trans. Circuits Systems Video Technol. 12 (10) (October 2002) pp. 948–951.

[14] Y. Zhao, I.E.G. Richardson, Computational complexity management of motion estimation in video encoders, IEEE DCC2002, Snowbird, UT.