Low-Complexity Rate-Distortion Optimal Transcoding of MPEG I-Frames

R.L. Lagendijk[(1)], E.D. Frimout[(2)] and J. Biemond[(1)]

[(1)] Information and Communication Theory Group
Faculty of Information Technology and Systems
Delft University of Technology, Delft, The Netherlands

[(2)] Philips Consumer Electronics
Advanced Systems and Applications Lab
Eindhoven, The Netherlands

Summary

The transcoding of MPEG video streams is important in a variety of situations such as in rate reduction for small bandwidth downstream channels and in extracting a fast visual playback stream for digital video cassette recorders. This paper addresses transcoding of MPEG intra-coded frames by selectively copying DCT AC coefficients from the primary MPEG stream into the secondary MPEG stream. A rate-distortion optimal transcoding approach is presented, as well as several computationally more attractive suboptimal approaches. The proposed solutions outperform the more traditional codeword extraction methods from literature.

Correspondence

For all correspondence concerning this paper, please contact:
Dr. Reginald L. Lagendijk
Information and Communication Theory Group
Faculty of Information Technology and Systems
Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands
phone: +31 15 278 3731 fax: +31 15 278 1843
E-mail: R.L.Lagendijk@its.tudelft.nl

I. Introduction

A typical MPEG encoder that compresses digital video for broadcasting purposes has a target bit rate of between approximately 4 and 10 Mbit/sec. The choice of this rate is primarily determined by the available or allocated capacity of the terrestrial or satellite transmission link. In some cases these primary wide-band channels feed into secondary downstream channels of lower capacity. For instance, in a cable head-end a network operator may decide to allocate a smaller bandwidth to a specific video service than the bandwidth the service occupies on reception. Clearly this is at the advantage of providing more different services on the cable network, however at the cost of a lower quality per channel. Another example is the provision of fast visual playback facilities in digital video cassette recorders. Fast visual playback, i.e., a fast forward or reverse playback mode which displays video pictures, is particularly complicated for MPEG compressed digital video stored on a helical scan video cassette recorder [With93,Frim95a,Frim95b]. One solution to fast visual playback is to store (and display on fast forward or reverse) a low bit rate stream on the digital cassette in addition to the original MPEG bit stream. This requires the real-time extraction of a low bit rate MPEG stream when recording the primary MPEG video bit stream

The problem of converting a compressed video stream such as MPEG into another (MPEG) compressed video stream of lower bit rate is known as transcoding [Kees96,Naka95]. Though with transcoding a wide class of format conversions for compressed video is meant, we confine ourselves in this paper to the efficient bit rate reduction of the intra-coded frames (I-frames) of a given MPEG-2 video bit stream, without fully decoding and re-encoding (or re-quantization) the video signal. This last requirement is particularly important for cost-effective real-time systems. A typical configuration under consideration in this paper is the real-time conversion of a 10 Mbit/sec primary MPEG-2 video stream into a 1 Mbit/sec secondary I-frame only MPEG-2 stream.

The solution strategy that we propose in this paper is that of selectively copying encoded non-zero AC coefficients or codewords from the primary stream into the secondary stream. Since no re-quantization nor inverse DCT operations are necessary, this method is inherently fast. Similar approaches have been suggested in [Lane93,Yana93,Okam93]. The transcoding methods that we propose aim at rate-distortion optimality, and outperform earlier proposed techniques without increasing the complexity of the transcoder significantly

In Section II various approaches to transcoding are discussed. Section III concentrates on rate-distortion optimal methods for coefficient or AC codeword selection, while Section IV proposes a control strategy that governs the selection mechanism. In Section V an experimental evaluation is given for the transcoding methods described in this paper. The paper concludes with a discussion in Section VI.

II. Low Complexity Approaches to Transcoding

In extracting a secondary stream only intra-coded information (I-frame) of the primary stream is used. The reason for this is that we focus on fast and low-complexity transcoding. Transcoding of predicted video frames (P-frames) would require motion-compensation of the additional distortion introduced by transcoding of previous I- or P-frames [Kees96,Naka95]. The complexity of this operation is considered too high in the context of this paper.

Figure 1 gives two examples of how intra-frames from the primary MPEG stream are used in the secondary stream. The center of the figure shows a concatenation of 3 MPEG groups of pictures (GOPs) with different length in the primary stream. Directly above and below the primary stream we show the frames that are used in the secondary stream, depending on the desired frame rate. In Figure 1(a) the secondary stream has a frame rate that is one-third of the primary stream; in Figure 1(b) the frame rate is one-sixth. The frame rate of the secondary stream can be artificially increased by including so-called *dummy P-frames*. These dummy P-frames contain only codes for *not-coded* macroblocks and are represented by a fixed bit pattern [MPEG96]. Effectively this results in repeating the previous I-frame in the decoder. Note that depending on the original group of pictures (GOP) structure, not all intra-coded frames of the primary stream may be used in the secondary stream.

Though a significant bit rate reduction is already achieved by selecting only I-frames from the primary stream, the resulting bit rate is usually still too high. Some information fields from the primary stream, such as header information, must be copied directly into the secondary stream, possibly after small modifications (for instance the bit rate entry). Furthermore, the DC coefficients per DCT block are also copied directly into the secondary stream. It is therefore only possible to achieve a further bit rate reduction by *not* using all information contained in the AC coefficients of the primary stream in the secondary stream.

Figure 2 shows the different domains in which a quantized DCT block can be represented. In the *coefficient domain* (*cd*) a block contains $N*N$ (N=8) integer entries that correspond with the *quantized* DCT coefficients. Many of the entries are usually zero, especially those entries that correspond with the spatial high frequencies. In the *run-level domain*, the non-zero AC coefficients are re-ordered in a zigzag scan fashion and are subsequently represented by a tuple (r,l), where the run (r) is equal to the number of zeros preceding a certain coefficient and the level (l) is equal to the value of the coefficient. The transition from the coefficient domain to the tuple domain is called *tuple coding*. In the *bit domain* (*bd*) the (r,l) tuples are represented by variable length coded (VLC) codewords. The codewords for a single DCT block are terminated by an end of block (EOB) marker.

For the selective use of non-zero AC coefficients from the primary into the secondary stream, three approaches are possible as illustrated in Figure 3 [Frim95b].

Full Transcoding

Figure 3(a) shows the straightforward transcoding approach for a single DCT block. The codewords in the bit domain are variable length decoded (VLD), yielding run-level (r,l) tuples. The tuple decoding (TD) yields quantized coefficients in the coefficient domain. These coefficients are de-quantized (DQ) to obtain the DCT coefficients (dctd). From this point the encoding can be done as in a normal intra-frame encoder, namely *quantizing* (Q) subject to a rate control, tuple encoding (TC), and VLC encoding. We will not elaborate on this *full transcoding* approach but we will use it as a performance reference requiring a relatively high complexity.

Coefficient Selection

In Figure 3(b) the DCT blocks from the primary stream are decoded up to the coefficient domain. In the coefficient domain a *selection* is made of those non-zero AC coefficients that need to be retained for the corresponding DCT block of the secondary stream. No re-quantization is needed in this case, only tuple encoding and VLC encoding have to be carried out. The objective is to select those AC coefficients from the primary stream such that an overall good quality picture results in the secondary stream. As a consequence, this *coefficient selection* (CCS) mechanism must be governed by a rate control mechanism. This may require de-quantization of the DCT coefficients.

Codeword Selection

The approach with the lowest complexity is the *codeword selection* (CWS) in the bit domain (Figure 3(c)). Selected VLC codewords from the primary stream are simply copied into the secondary stream. No decoding or re-quantization of the copied codewords is carried out. In view of the run-level encoding, this approach is equivalent to a *constrained* coefficient selection where only a number of *consecutive* (r,l) tuples of the DCT block can be retained for the secondary stream. Effectively this means that for a certain DCT block we always start with the first AC coefficient and retain a certain number of consecutive non-zero AC coefficients. We will denote the number of codewords selected as the cut-off level c. In order to determine the cut-off level, various approaches are possible. The simplest methods, here denoted by *zonal codeword selection* (ZCS), uses a fixed cut-off level independent of the image contents [Lane93, Yana93, Okam93]. Alternatively, the cut-off level is chosen adaptively per DCT block such that the distortion can be made dependent on the local spatial content. In both cases in the cut-off level has to be governed by a rate control. As with the coefficient selection, rate control for codeword selection may require tuple decoding and de-quantization.

III. Coefficient/Codeword Selection Mechanism

For the *coefficient selection* and the *codeword selection* approaches a mechanism is required to determine the set of optimal non-zero AC coefficient and the cut-off level, respectively, per DCT block. We assume that the target bit rate for the frame to be transcoded is known. Clearly then the selection mechanism needs to deliver the maximum quality of the decoded picture, given this bit rate constraint. This section addresses how to solve this rate-distortion optimization problem.

III.A Rate-Distortion Optimal Selection

The problem of the optimal rate constrained *coefficient selection* (Figure 3(b)) can be formulated as follows. If X_p is the intra-coded frame from the primary stream and X_s is the corresponding frame in the transcoded secondary stream, the objective is minimize the distortion $D(X_p,X_s)$ between X_p and X_s subject to a total coding bit budget R_{budget} for X_s, i.e.

$$\min_{AC_s}[D(X_p, X_s)] \quad \text{subject to} \quad R(X_s) \le R_{\text{budget}} \tag{1}$$

The result of the minimization is the optimal set of selected non-zero AC coefficients AC_s per DCT block that will be copied from the primary into the secondary bit stream. The distortion measure we propose to use is the mean squared error (MSE) calculated on the *quantized* DCT coefficients:

$$D(X_p, X_s) = \sum_{n \in \text{all DCT blocks}} \sum_{q=1}^{N^2-1} (\hat{X}_p^{n,q} - \hat{X}_s^{n,q})^2 \qquad (2)$$

Here $\hat{X}^{n,q}$ refers to the q-th quantized AC coefficient in the n-th DCT block in a frame. Since the q-th AC coefficient is not re-quantized, but either retained or discarded in the secondary stream, the quadratic term in (2) is either equal to zero or equal to $(\hat{X}_p^{n,q})^2$. The summation in (2) is over N^2-1 AC terms since the DC coefficient is not taken into account. Due to the standard MPEG I-frame weighting matrix for DCT coefficients [MPEG96], the above distortion measure is essentially a perceptually weighted MSE (WMSE).

The formulation in (1) also applies to *codeword selection* (Figure 3(c)), with an additional constraint on which sets of non-zero AC coefficients can be considered. The only free parameter in the codeword selection is the cut-off level, which implicitly determines the set of selected AC coefficient. In this respect codeword selection yields an optimization problem of lower complexity, but the principle remains the same.

The constrained minimization problem (1) can be solved by minimizing the following unconstrained cost function [Ever63,Shoh88,Ramc94]:

$$J(\lambda) = D(X_p, X_s) + \lambda R(X_s) \qquad (3)$$

Here λ>0 is the Lagrange multiplier that has to be chosen such that the condition:

$$R(X_s) = R_{\text{budget}} \qquad (4)$$

is satisfied. The Lagrange multiplier λ is equal to the (negative) gradient of the rate-distortion function at the point where the distortion is minimal given the coding bit budget (4). Alternatively λ can be interpreted as a rate control parameter. Increasing λ will increase the bit rate of the optimal solution, and vice versa.

It was shown in [Shoh88] that, for a quadratic distortion criterion D, the optimization of (1) can be carried out independently for every $N*N$ DCT block for the fixed slope λ. In other words, given a certain λ, the minimization of $J(\lambda)$ in (3) is identical to minimization of

$$J(\lambda) = \sum_{n \in \text{all DCT block}} J^n(\lambda) = \sum_{n \in \text{all DCT block}} D(X_p^n, X_s^n) + \lambda R(X_s^n) \qquad (5)$$

where X^n refers to the n-th DCT block. The consequence of (5) is that we can now solve the constrained optimization problem for each DCT block independently, as long as for each DCT block the same value for λ is used.

III.B Optimal Coefficient Selection (CCS)

Finding the optimal solution for CCS is a linear programming problem. For DCT block n we need to select a set of non-zero AC coefficients such that $J^n(\lambda)$ is minimized. The differential costs of adding or removing a certain AC coefficient depends on the presence of other AC coefficients through the coding of the run-level tuples. The method that we use is inspired by the approach of Ramchandran and Vetterli for rate-distortion optimal zero-thresholding [Ramc94].

Let us assume that we have decided to retain the first AC coefficient in DCT block n. If we subsequently retain the q-th AC coefficient in this DCT block, the *decrease* in distortion will be $(\hat{X}_p^{n,q})^2$. The *increase* in bit rate depends on the length of the tuple VLC that represent a run of q-2 zeros followed by the amplitude $\hat{X}_p^{n,q}$ of the q-th AC coefficient. The difference in the cost function $J^n(\lambda)$, denoted by $\Delta J_{1,q}^n(\lambda)$ indicating that AC coefficient q is retained next to AC coefficient one, is therefore:

$$\Delta J_{1,q}^n(\lambda) = -(\hat{X}_p^{n,q})^2 + \lambda R_{1,q}^n \tag{6}$$

where $R_{1,q}^n$ denotes the additional bits used for the tuple coding.

The above illustrative case can be generalized as follows. The costs for retaining no AC coefficients at all in DCT block n, denoted by $J_0^n(\lambda)$, is

$$J_0^n(\lambda) = \sum_{q=1}^{N^2-1} (\hat{X}_p^{n,q})^2 \tag{7}$$

while the retaining of AC coefficient q following the already retained AC coefficient k yields the following differential costs for DCT block n:

$$\Delta J_{k,q}^n(\lambda) = -(\hat{X}_p^{n,q})^2 + \lambda R_{k,q}^n \tag{8}$$

As in (6), the differential costs are composed of a decrease in distortion and an increase in bit rate $R_{k,q}^n$ that is determined by the length of the VLC codes used for the tuple encoding.

Using the above relations, the optimal set of non-zero AC coefficients to retain can be determined. To this end a cost matrix is constructed as illustrated in Figure 4 for the case of five non-zero AC coefficients. In this matrix, the vertical axis indicates the index of the retained non-zero AC coefficient. The horizontal axis numbers the individual non-zero AC coefficients. A path through this matrix now represents a set of retained AC coefficients. For instance, in Figure 4 the bold line indicates a path representing {1,4,5} as set of retained AC coefficients.

The total costs associated with a particular path are easily calculated by (7) and (8). The *optimal* path through the cost matrix minimizes the total cost $J^n(\lambda)$. The total number of possible paths through the cost matrix is equal to $2^{(\text{number of non-zero AC coefficients})}$. Fortunately, the optimal path, and therefore the optimal CCS, can be found efficiently by a dynamic programming approach [Ever63].

III.C Optimal Codeword Selection (CWS)

Since codeword selection is equivalent to coefficient selection with an additional constraint on the coefficients that can be retained, a similar approach as for CCS can be used. However, instead of considering all possible paths through the cost matrix (Figure 4), only those paths have to be considered which include for a given cut-off level *c* all preceding non-zero AC coefficients from the primary stream. The complexity of finding the optimal path is now significantly lower compared to CCS, since the number of possible paths is equal to the number of non-zero AC coefficients in the primary stream plus one. For instance, in Figure 4 there are only 6 possible paths (corresponding to the 6 horizontal arrows), namely: no AC coefficients retained, {1}, {1,2}, ...,{1,2,3,4,5}. The costs of all paths can be calculated easily using (7) and (8); the minimization of (5) then becomes trivial.

III.D Comparison of CCS, CWS and ZCS

In this section we experimentally compare three methods for selecting non-zero AC coefficients, namely:

1. zonal codeword selection (ZCS) using a fixed cut-off level for all DCT blocks,
2. optimal codeword selection (CWS) using a fixed λ for all DCT blocks,
3. optimal coefficient selection (CCS) using a fixed λ for all DCT blocks.

For a wide range of λ-values and cut-off levels we determined the bit rate and the WMSE distortion calculated according to (2) for an I-frame.

The I-frame is taken from a standard MPEG-2 stream with rate 9.6 Mbit/sec and a “IBBPBBPBBPBB” group of pictures structure. Figure 5(a) compares ZCS with CWS for a particular frame. It is clear that the optimization of the cut-off level as proposed in this section greatly outperforms approaches that use a constant cut-off level [Lane93,Yana93,Okam93]. Figure 5(b) compares CWS with CCS, showing only a marginal difference between the two. Since similar results were obtained for other sequences, we conclude that since codeword selection has only a slightly worse performance than coefficient selection, it is greatly preferred over CCS because of its computational simplicity.

IV. Control of the Lagrange Parameter λ

As indicated in Section II, all selection methods require control of the Lagrange parameter λ. For CWS controlling λ implicitly means controlling the cut-off level c. For ZCS only the cut-off level c is controlled without rate-distortion considerations. In all cases the objective is ensure that the target bit rate (4) for a given frame is reached. One possibility is to find λ by an intelligent iterative search technique. Figure 6(a) show the block diagram of the resulting transcoder. Observe that the coefficient/codeword selection mechanism uses the same value of λ in all DCT blocks. Though this *feed-forward control* will result in the optimal value for λ, it is rather computationally expensive. This section presents three sub-optimal alternatives that are computationally more attractive in a real-time context.

IV.A Lagrange Feedback Control

The optimal control of λ requires an iterative search algorithm in which all DCT blocks of the frame under consideration have to be evaluated for each iteration step. A suboptimal but more efficient approach is to replace these iterations by a recursion over the DCT blocks. Starting off with an initial estimate λ_0 in the first DCT block, the recursion updates λ_n for a next DCT block based upon the achieved rate and distortion in the already transcoded DCT blocks in the frame. Figure 6(b) shows the block diagram of this recursive or feed-back λ-control. The recursive updating of λ has to be sufficiently slow to avoid spurious spatial oscillations of λ. Besides the computational advantage, another advantage of this *feed-back control* over feed-forward control is the reduced memory requirement.

IV.B Simplified Lagrange Feedback Control

Though for the actual transcoding based on CCS and CWS de-quantization of the AC coefficients is not necessary, the de-quantized AC coefficients are still necessary in computing the WMSE distortion measure (2) and (8). This can also been seen in Figures 3(b) and 3(c) where the selection mechanism requires input from the de-quantization *DQ*. An approximation to (2) is obtained is we assume that all DCT blocks have been quantized using the same quantizer step size. This simplification has two effects. In the first place (2) can now be computed without de-quantizing the DCT coefficients, reducing the computational complexity. Secondly, the distortion computed will deviate somewhat from the actual distortion defined in (2), influencing the cost function (5) and therefore the solution of CCS and CWS. The significance of this effect depends greatly on the variations of the quantizer step size in the frame: the larger the variations in the step size are, the larger the deviation from the optimal solution will be.

IV.C Zonal Codeword Selection with Feedback Control

In ZCS the cut-off level c is controlled instead of the Lagrange parameter λ. As in the previous subsections, the objective is to control c such that (4) is satisfied. The control strategy for c can be the same as the "Lagrange Feedback Control" in Figure 6(b), with the further simplification that no tuple decoding nor de-quantization has to take place since ZCS takes place without distortion consideration, i.e. the distortion defined by (5) does not need to be computed. The resulting transcoder represents essentially a further computational simplification of the "Simplified Lagrange Feedback Control".

V. Performance Comparison

This section presents a performance comparison from two points of view. In the first place we evaluate the most promising transcoding methods experimentally. In this way we can compare the numerical performance differences between the various proposed techniques and their simplifications, and the differences with the known methods "Full Transcoding and "Zonal Codeword Selection". Secondly, we compare the transcoding techniques with regard to their complexity. Though ideally complexity is measured in throughput and silicon area (gate count), we belief that an operations count is sufficient to show the attractiveness of some of the proposed transcoding techniques.

V.A SNR Comparison

We experimentally compare the following transcoding methods:

1. Full transcoding (FT),
2. Coefficient selection with λ feed-forward control (CCS-FF),
3. Codeword selection with λ feed-forward control (CWS-FF),
4. Codeword selection with λ feed-back control (CWS-FB),
5. Codeword selection with simplified λ feed-back control (CWS-SFB),
6. Zonal codeword selection with cut-off level *c* feed-back control (ZCS-FB)

We have used an original sequence that consisted of 240 frames, format 576 lines by 720 pixels per line, progressive scan, 50 frames per second. This sequence was MPEG-2 encoded at a rate of 9.6 Mbit/sec. The GOP structure used was "IBBPBBPBBPBB", so that in total 20-I frames were available for copying from the primary into the secondary stream. The target bit rate of the secondary stream was 1.25 Mbit/sec. Since each I-frame in the primary stream consumes about 2 Mbit and the secondary stream has 2 I-frames per second, the transcoding needs to achieve an additional compression factor of approximately 3.

The quality of the transcoded I-frames is measured using the commonly used signal-to-noise ratio (SNR):

$$\text{SNR} = 10\log_{10}\left(\frac{\sigma^2_{\text{original I-frame}}}{\text{MSE}}\right) \text{ (dB)} \qquad (9)$$

Here the MSE is computed with reference to the *original* sequence, so that the above SNR reflects the quality of the transcoded sequence with reference to the original sequence.

Figure 7(a) shows the I-frame transcoding results for CCS-FF, CWS-FF, CWS-FB, ZCS-FB. As in Section III.D was already found, the SNR differences between coefficient and codeword selection are very small. As could be expected, feed-forward control outperforms feed-back control (compare CWS-FF and CWS-FB), but the differences are fairly small and visually insignificant. Due to the dynamic nature of the feed-back control, the results for CWS-FB are occasionally better than CWS-FF, c.f. frame 13 and 14. Finally, zonal codeword selection using a feed-back control (ZCS-FB) is on the average by 3 dB worse than the other transcoding approaches. The visual differences between zonal codeword selection and any of the other transcoding techniques described in this paper as visually very significant.

Figure 7(b) shows the transcoding results for FT, CWS-FB, CWS-SFB, and ZCS-FB. The full transcoding outperforms any of the methods proposed in this paper by at least 2 dB. From time to time this difference is visually significant. Figure 7(b) also shows that simplifying the λ feed-back control strategy (CW-SFB) results in a small quality loss compared to CWS-FB. As in Figure 7(a), the zonal codeword selection is not only numerically but also visually inferior to all other transcoding methods.

V.B Complexity Comparison

The objective of this section is to compare the computational complexity of transcoding techniques that were discussed in this paper. We do this by comparing the type and number of operations needed for the various methods. We assume that parsing of the incoming MPEG-stream has already been done (this step is the same in all techniques), and start at the point where compressed data can be input to the variable length decoding (VLD). Similarly, at the output of the transcoder we end our comparison just prior to the remultiplexing of the VLC encoded bit patterns per DCT block.

The complexity of the following three basic algorithmic components of transcoders are evaluated:

- the operations on the MPEG I-frames to extract a subset of DCT coefficients from the incoming stream. We have FT, CCS and CWS as alternatives (see Section II);
- the operations to determine which DCT coefficients to extract from the incoming stream. Here we have ZCS, CCS and CWS as possible alternatives (see Section III);
- the operations to carry out the rate control: we will restrict ourselves to the three proposed suboptimal feed-back procedures namely λ feed-back and its simplified version, and cut-off level feed-back (see Section IV).

Table I lists the number of operations for the building blocks operating on DCT data (see Figure 3), which are needed in the comparison. In this and all following tables we assume that the incoming DCT block contains N_1 non-zero DCT coefficients and the transcoded DCT block contains N_2 non-zero DCT coefficients. Practical values are N_1=5-15 and N_2=2-3. At this point we make a distinction between the operation complexity/time needed for memory access (O_{mem}), for address calculations (O_{address}), and for integer multiplications (O_{multiply}). Table II summarizes the total number of operations – using the numbers in Table I – for the "Extraction of DCT coefficients".

In CCS, the optimal set of DCT coefficients to be retained needs to be determined by finding the optimal path in a cost matrix. The largest complexity involved is the calculation of the differential costs per node, as indicated by Eq. (8). The number of operations are roughly $2(O_{\text{mem}} + O_{\text{multiply}})$. If the incoming DCT block contains N_1 non-zero DCT coefficients, and if we use dynamic programming for minimizing the sum of the terms (8) for each possible set of retained DCT coefficients, then (8) needs to be evaluated approximately $N_1(N_1+1)/2$ times. All other factors in dynamic programming are of order N_1, and are therefore negligible. In case we use CWS, we do not need the dynamic programming algorithm at all. The number of times Eq. (8) needs to be evaluated is N_1+1. The resulting numbers are summarized in Table II under the category "DCT coefficient selection".

In the rate control, the dominant factor is determined by the type of information that is used to calculate the value for λ or c. As mentioned before, we consider λ feed-forward control too complex and do not consider this. For λ feed-back control we need VLD, TD and DQ in order to be able to evaluate Eq. (8). This adds computational complexity depending on whether CCS or CWS is considered. In case the simplified λ feed-back is used, no DQ is needed, while for cut-off level feed-back only VLD is needed. For full transcoding also a rate control will be necessary. The simplest rate control using feed-back on the quantizer step size has a complexity similar to cut-off level feed-back. The resulting numbers are again summarized in Table II under "Rate control".

Using Table II, we can now compare the computational complexity some of the transcoding methods that were numerically evaluated in the previous section. If we totally ignore the potential performance gain that can be achieved by combining certain operations, then we obtain the results that are summarized in Table III. From these number we can draw several conclusions.

In the first place we see that integer multiplications are always necessary except for zonal codeword selection (ZCS). However, this computationally attractive approach has a poor performance as seen in the previous section. Among the other four transcoding techniques, clearly the full transcoding requires the largest number of integer multiplications. Although the coefficient selection transcoding (CCS) has a much more attractive data path than full transcoding because fewer operation are to be carried out on the actual DCT coefficients, the dynamic programming adds significantly to the complexity of this method, yielding a multiplication complexity of order N_1^2. Both codeword selection methods (CWS-FB and CWS-SFB) have a complexity which is only of order N_1, while their numerical performance is similar to CCS.

The above complexity analysis may change depending on the actual implementation decided upon. Especially the CCS method has a great potential for exploiting the sharing operations in different parts of the algorithm (memory access and address calculation). Furthermore, the integer multiplication complexity in this algorithm is different from the one in the full transcoding, namely a multiply and add (accumulation) operation in CCS versus individual multiplications per DCT coefficient in FT. However, although in both CCS and FT a significant degree of parallelism exists in the multiplication operations, the difference between on the one hand the 128 multiplications for FT and the order N_1^2 multiplications for CCS and on the other hand the order N_1 multiplications in CWS, remains significant. We therefore believe that over different implementations that exist for the various transcoding techniques, the CWS method will always prevail.

VI. DISCUSSION

This paper has proposed several transcoding methods that are based on copying selected AC DCT coefficients or codewords from a primary MPEG stream into a secondary MPEG stream. We have shown that the methods proposed outperform the more traditional codeword extraction methods from literature with various degrees of increase in computational complexity. The codeword selection method in combination with a (simplified) Lagrange feed-back control has a low complexity and qualifies for application in real-time systems. The other proposals require more computation, especially for the calculation of the distortion in Eq. (2) or the minimization of Eq. (5) using dynamic programming.

A limitation of the transcoding concept presented in this paper is that it uses only I-frames from the primary MPEG stream. For some applications this may result in a frame rate that is too low. Nevertheless, the proposed transcoding concept can find direct application in the extraction of fast visual playback streams for digital video cassette recorders [Frim95b].

REFERENCES

[Berg71] T. Berger, *Rate distortion theory: A mathematical basis for data compression,* Prentice-Hall, Englewood Cliffs, NJ, 1971.

[Ever63] H. Everett III, “Generalized Lagrange multiplier method for solving problems of optimum allocation of resources”, *Operations research*, vol. 11 pp 399-417, 1963.

[Frim95a] E.D. Frimout, R.L. Lagendijk, and J. Biemond, “Extraction of a dedicated fast playback MPEG bit stream”, *Proc. SPIE VCIP’95*, Taipei, Taiwan, May 1995.

[Frim95b] E.D. Frimout, *Fast playback of helical-scan recorded MPEG video*, Ph.D. thesis Delft University of Technology, Department of Electrical Engineering, 1995.

[Kees96] G.J. Keesman, R. Hellinghuizen, F. Hoeksema, and G. Heidenman, “Transcoding of MPEG bitstreams”, *Signal Processing: Image Communciations*, vol. 8, pp. 481-500, 1996.

[Lane93] F. Lane and J. Boyce, “Fast Forward/Fast Reverse for a Digital HDTV VCR”, *Intl. Workshop. on HDTV '93*, Nov 1993, Ottawa Canada.

[MPEG96] ISO/IEC International standard 13818-2, *Information technology - Generic coding of moving picctures and associated audio information: Video*, 1996.

[Naka95] Y. Nakajima, H. Hori, and T. Kanoh, “Rate conversion of MPEG coded video by requantization process”, *IEEE Int. Conf. On Image Processing 1995,* Washington DC, pp. 408-411, 1995.

[Okam93] H. Okamoto et. al., “A Consumer Digital VCR for Advanced Television”, *IEEE Transactions on Consumer Electronics*, Vol. 39, No. 3, August 1993, pp.199-204.

[Ramc94] K. Ramchandran and M. Vetterli, “Rate-Distortion Optimal Fast Thresholding with complete JPEG/MPEG Decoder Compatibility”, *IEEE Trans. Image Processing*, Vol. 3, No. 5, September 1994.

[Shoh88] Y. Shoham and A. Gersho, “Efficient bit allocation for an arbitrary set of quantizers”, *IEEE Trans. Acoust. Speech, Signal Proc*. vol. 36, pp. 1445-1453, Sept. 1988.

[With93] P.H.N. de With, A.M.A. Rijckaert, H. -W. Keesen, J. Kaaden, and C. Opelt, “An Experimental Digital Consumer HDTV Recorder using MC-DCT Video Compression”, *IEEE Trans. on Consumer Electronics*, Vol 39, No 4, Nov 1993, pp. 711-722.

[Yana93] N. Yanagihara, C. Chan, and Y. Kubota, “A Recording Method of ATV data on a Consumer Digital VCR”, *Intl Worksh. on HDTV '93*, Nov 1993, Ottawa Canada.

TABLES

Table I: Number of operations for building blocks in Figure 3, if the incoming DCT block contains N_1 non-zero DCT coefficients and the transcoded DCT block contains N_2 non-zero DCT coefficients.

Algorithm	#Operations	Remarks
Variable Length Decoding (*VLD*)	$N_1\ O_{mem}$	One table lookup is counted as one memory access
Tuple Decoding (*TD*)	$N_1\ O_{address}$	Decoding is actually writing data a pre-calculated addresses in 64 units data buffer
De-quantization, excl. weightmatrix (*DQ*)	$64\ O_{multiply}$	
Re-quantization, excl. weightmatrix (*Q*)	$64\ O_{multiply}$	
Tuple Coding (*TC*)	$64\ O_{address}$	
Variable Length Coding (*VLC*)	$N_2\ O_{mem}$	

Table II: Total Number of operations for the basic algorithmic components of transcoders per 8x8 DCT block. Row numbering (left most column) is for reference purposes.

	Algorithmic component	**Possibilities**	**Number of Operations**
II.a	Extraction of DCT coefficients	Full transcoding	$(N_1 + N_2)\ O_{mem} + (N_1 + 64)\ O_{address} + 128\ O_{multiply}$
II.b		Coefficient selection	$(N_1 + N_2)\ O_{mem} + (N_1 + 64)\ O_{address}$
II.c		Codeword selection	$1\ O_{address}$
II.d	DCT coefficient selection	ZCS	0
II.e	(for given λ or cut-off level *c*)	CCS	$N_1(N_1+1)\ (O_{mem} + O_{multiply})$
II.f		CWS	$2\ (N_1+1)\ (O_{mem} + O_{multiply})$
II.g	Rate control (control of λ or *c*)	λ feed-forward	*Not considered*
		λ feed-back per DCT block	
II.h		*CSS*	$N_1\ O_{multiply}$
II.i		*CWS*	$N_1\ (O_{mem} + O_{address} + O_{multiply})$
II.j		Simplified λ feed-back per DCT block	$N_1\ (O_{mem} + O_{address})$
II.k		Cut-off level *c* feed-back	$N_1\ O_{mem}$

Table III: Complexity comparison of some transcoding techniques that are numerically compared in Section V.A. The left most column shows how the operation count was obtained from the entries in Table II.

Transcoding technique		**Operation count**
II.a+II.k	Full transcoding (FT)	$(2N_1 + N_2)\ O_{\text{mem}} + (N_1 + 64)\ O_{\text{address}} + 128\ O_{\text{multiply}}$
II.b+II.e+II.h	Coefficient selection with λ feed-back control (CCS-FB)	$(N_1^2 + 2N_1 + N_2)\ O_{\text{mem}} + (N_1 + 64)\ O_{\text{address}} + (N_1^2 + 2N_1)\ O_{\text{multiply}}$
II.c+II.f+II.i	Codeword selection with λ feed-back control (CWS-FB)	$(3N_1+2)\ O_{\text{mem}} + N_1\ O_{\text{address}} + (3N_1+2)\ O_{\text{multiply}}$
II.c+II.f+II.j	Codeword selection with simplified λ feed-back control (CWS-SFB)	$(3N_1+2)\ O_{\text{mem}} + N_1\ O_{\text{address}} + (2N_1+2)\ O_{\text{multiply}}$
II.c+II.d+II.k	Zonal codeword selection with cut-off level *c* feed-back control (ZCS-FB)	$N_1\ O_{\text{mem}} + 1\ O_{\text{address}}$

FIGURES

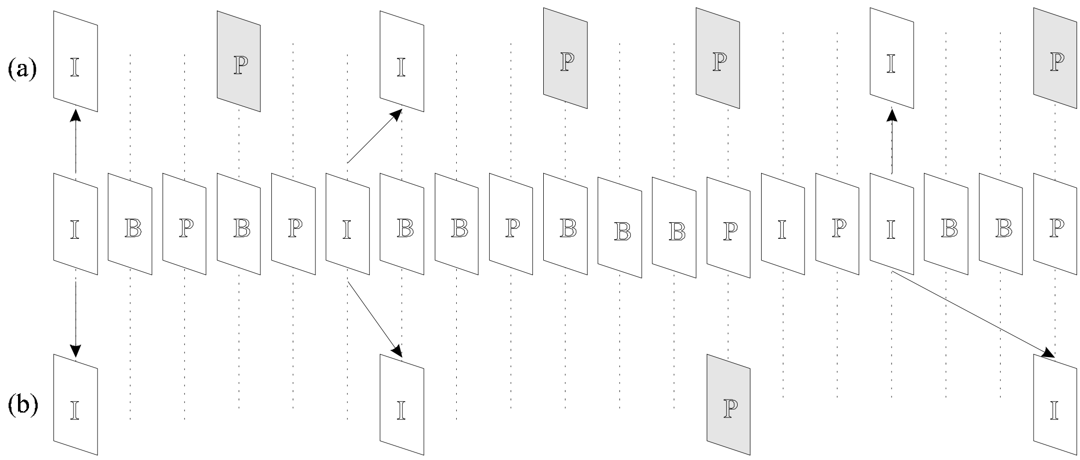


Figure 1: Two examples of how I-frames of the primary stream can be used in the secondary stream. Dummy P-frames can be included to increase the frame rate of the secondary stream. In (a) the frame is of the secondary stream is one-third of the primary stream, in (b) this is one-sixth.

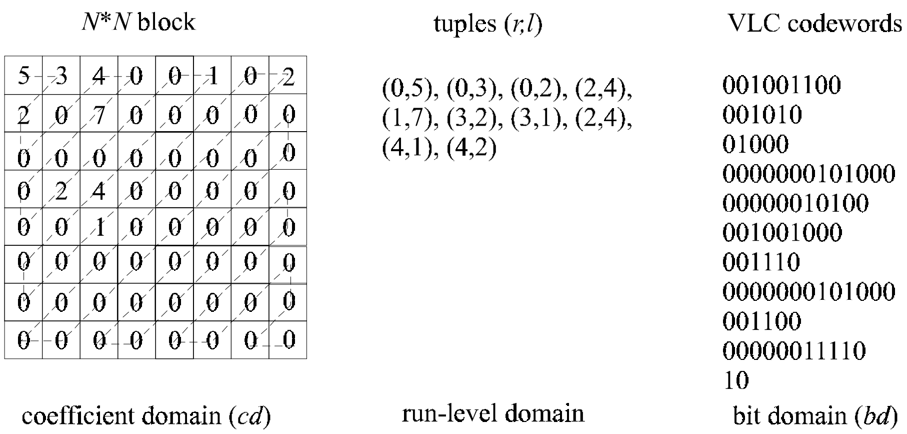


Figure 2: DCT block representation domains

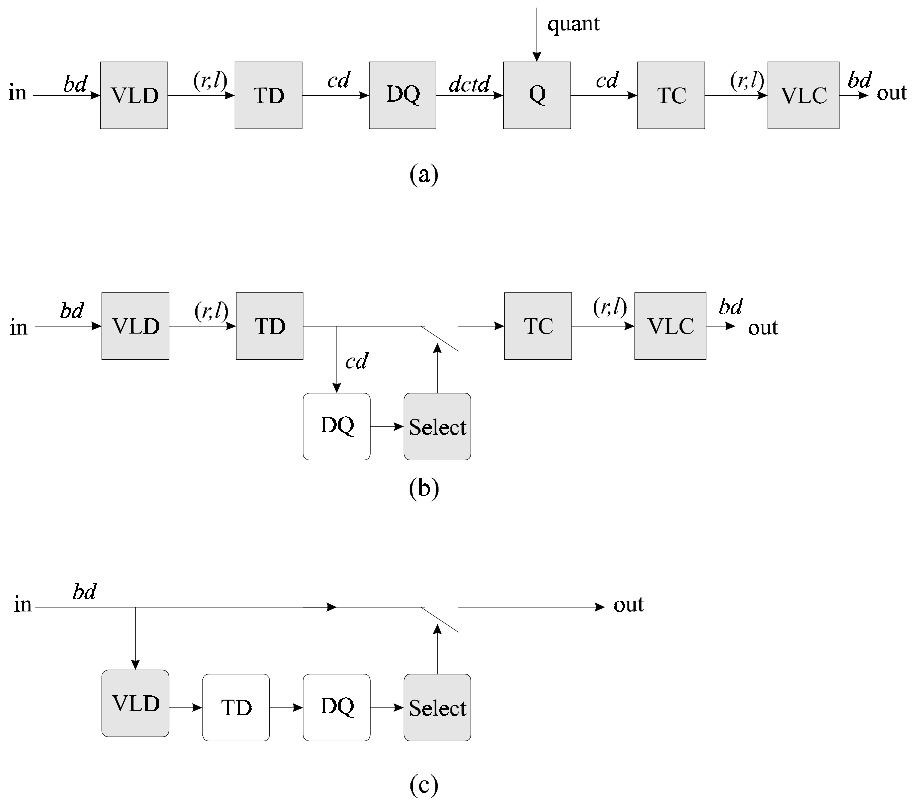


Figure 3: Three possible block transcoding architectures for I-frames: (a) Full transcoding (FT), (b) Coefficient selection (CCS), (c) Codeword selection (CWS).

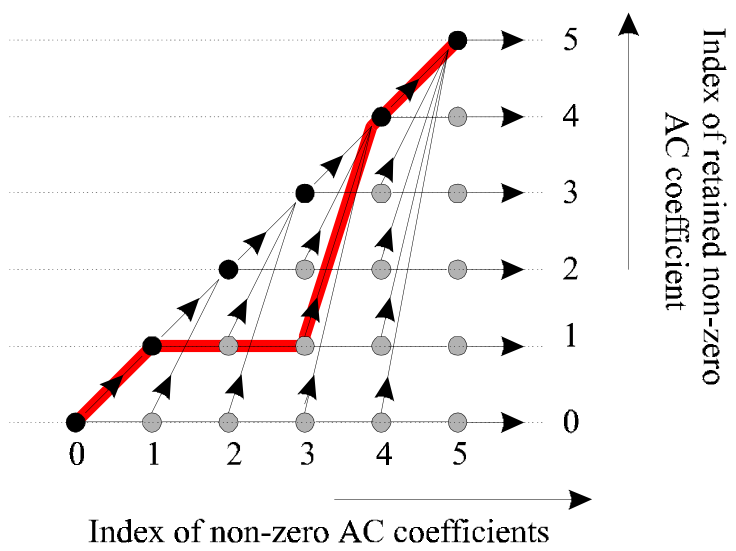
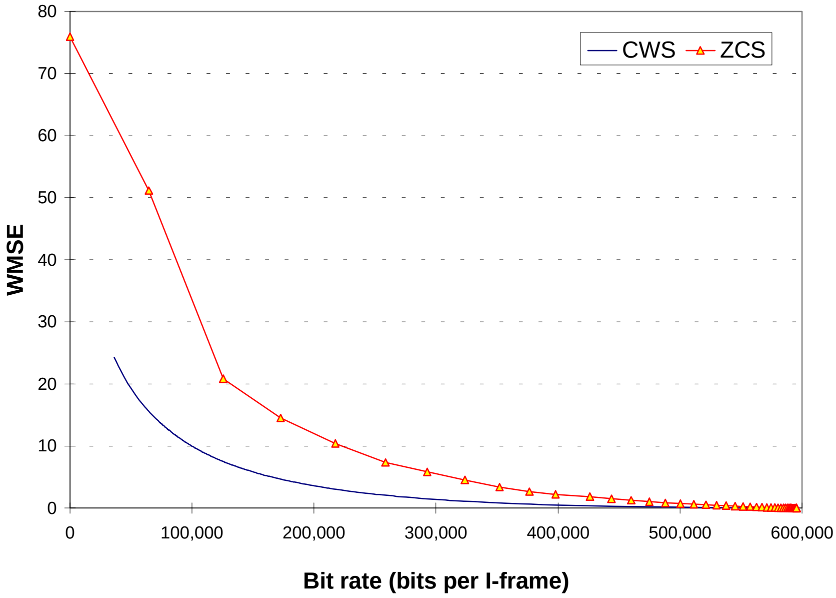
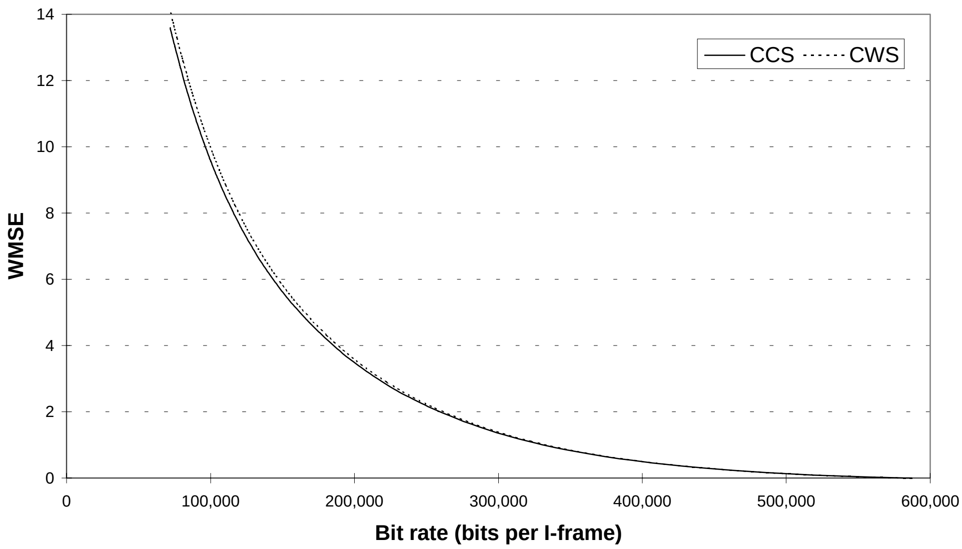


Figure 4: Cost matrix for CCS. The arrows indicate possible decisions. The bold line is a path through the cost matrix representing the set of retained coefficient {1,4,5}.

)

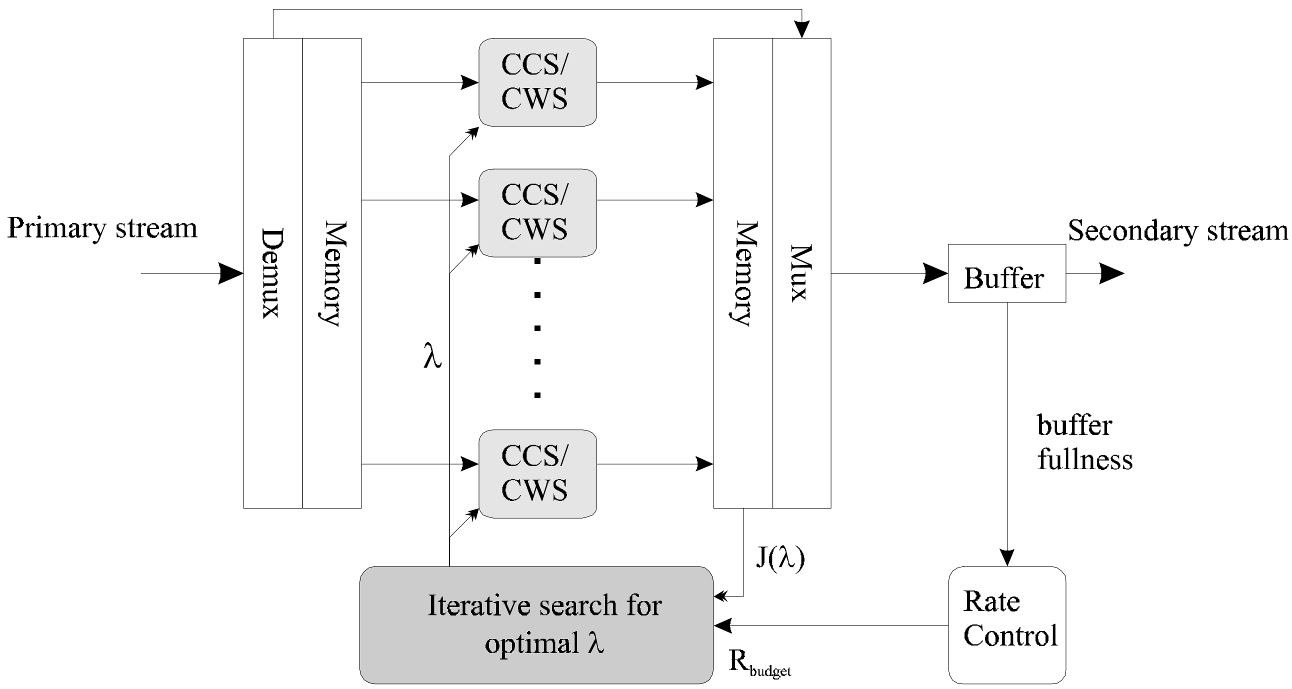


(a)

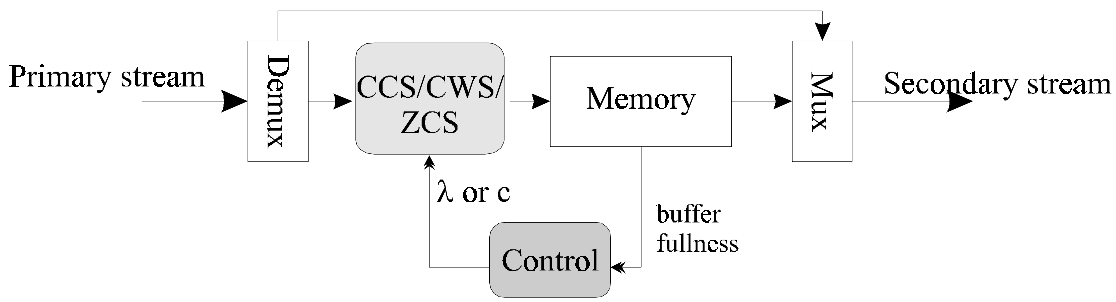


(b)

Figure 5: Experimentally determined rate-distortion curves for CCS, CWS, and ZCS, (a) compares CWS and ZCS, (b) compares CCS and CWS. Observe the difference in scale of the vertical axes.

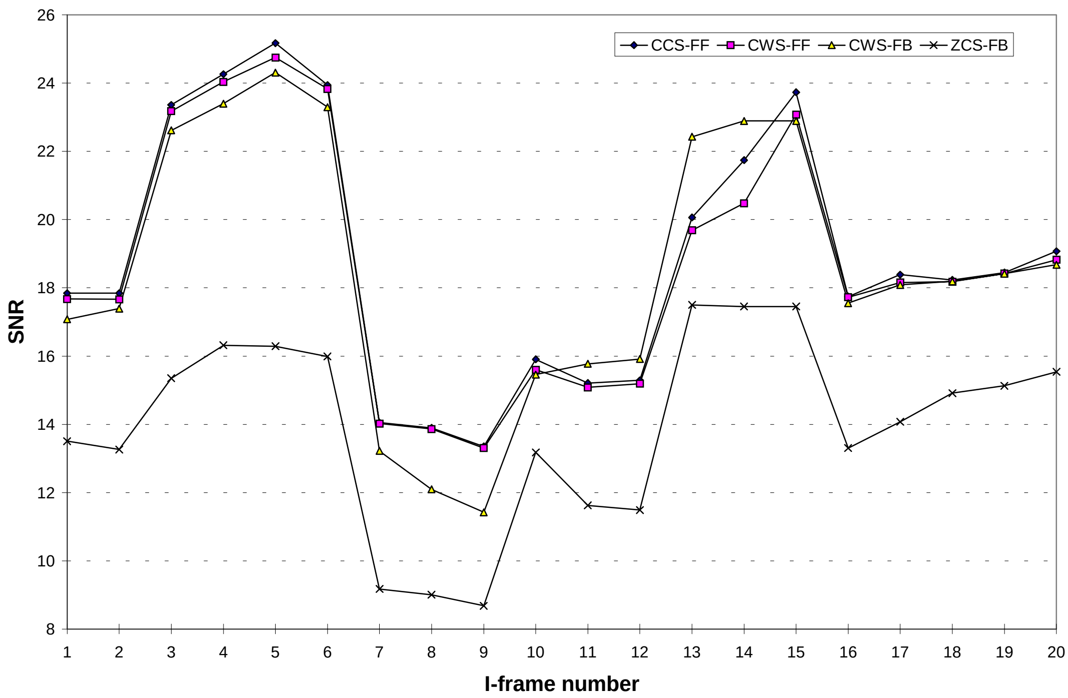


(a)

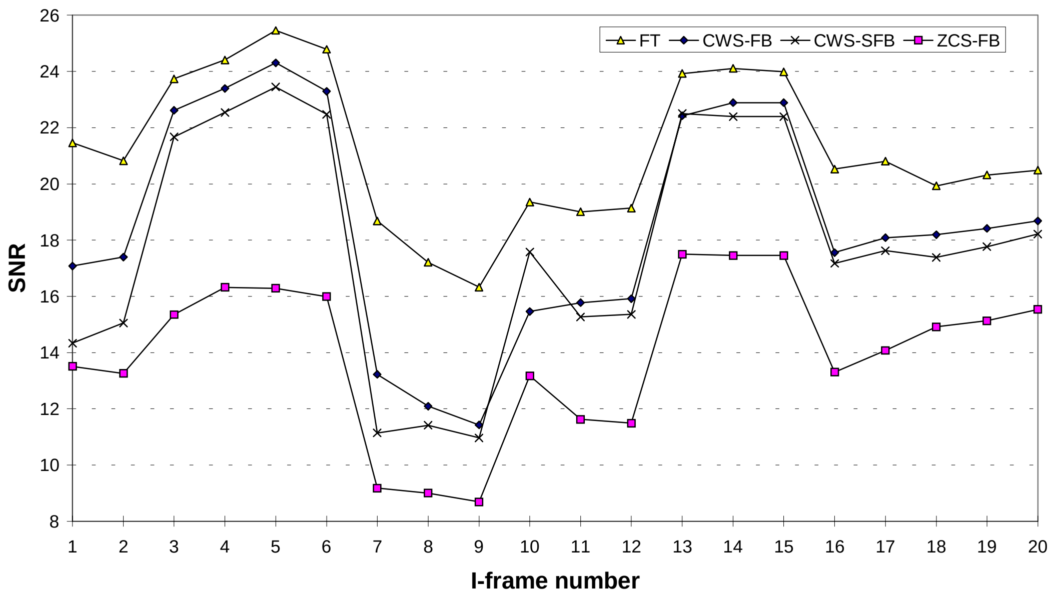


(b)

Figure 6: Rate control mechanisms, (a) Feed-forward control of λ, (b) Feed-back control of λ or the cut-off level *c*.



(a)



(b)

Figure 7: Comparison of transcoding techniques, (a) compares CCS-FF, CWS-FF, CWS-FB and ZCS-FB, (b) compares Ft, CWS-FB, CWS-SFB, and ZCS-FB.