Fast mode decision algorithm for H.264 inter-prediction

L.-J. Pan and Y.-S. Ho

A fast mode decision algorithm for H.264/AVC inter-prediction to reduce computational complexity of the H.264 encoder is presented. Experimental results show that the algorithm can save the entire encoding time by 77% on average while introducing only negligible loss in PSNR value and small increment of bit rate.

Introduction: The latest H.264 video coding standard can greatly outperform other existing coding standards in both PSNR and visual quality. This efficiency is achieved by using several powerful coding approaches [1]. One important approach is the variable block-size macroblock (MB) mode. There is a problem when we choose the best mode among all other modes. To take full advantage of all modes, H.264 provides a rate distortion optimisation (RDO) technique to select the best mode [2]. In this technique, by searching all combinations of modes for each MB exhaustively, we can achieve the best coding quality while minimising the bit rate. However, the RDO technique increases complexity and computation load drastically. This makes H.264 unsuitable for real-time applications. Thus a fast mode decision method is required to reduce the encoding time.

This Letter presents a fast mode decision algorithm for inter-prediction with early SKIP mode decision in the first stage of mode decision. To avoid calculating J_{mode} (16 × 16) in the first stage, we estimate SKIP mode cost by using SATD (sum of absolute Hadamard transform differences) value. Comparing SKIP mode cost with a threshold which considers the quantisation parameter (QP), we can decide whether SKIP mode is the best mode in the first stage of the mode decision procedure. We also use both the early 16 × 16 mode decision and the fast P8 × 8 mode decision method to reduce the searching time. Experimental results show that the fast inter-mode decision algorithm increases the speed of coding significantly.

Inter-mode decision: H.264 supports inter-prediction to reduce the temporal redundancy. H.264 uses seven different block sizes in total (16 × 16, 16 × 8, 8 × 16, 8 × 8, 8 × 4, 4 × 8, 4 × 4) for interframe motion estimation/compensation. These different block sizes actually form a two-level hierarchy tree structure inside a MB. The first level includes block sizes of 16 × 16, 16 × 8, and 8 × 16. The second level is specified as the P8 × 8 type. In the P8 × 8 type, each 8 × 8 block can be sub-divided into a smaller block size, such as 8 × 8, 8 × 4, 4 × 8, or 4 × 4. There is also a SKIP mode in P slice referring to the block size of 16 × 16, where no motion and residual information is encoded. To decide the best motion vector (MV), reference frame, and mode, H.264 uses the RDO method which is based on the Lagrangian function to minimise motion cost J_{motion} and mode cost J_{mode} [2].

Proposed algorithm: It is observed that there exist many homogenous regions in natural video sequences and when an object moves, we expect that most parts of the object move in the same or closely the same direction. In general, a homogeneous region or area with similar motions is more likely to be coded using a large block size, such as SKIP or 16 × 16 mode. If we detect these areas at an early stage, significant time could be saved for the motion estimation and RDO computations of small size modes. In our algorithm, we differentiate the SKIP mode from other block types and give it the highest priority. In H.264, since ultimately the transformed coefficients are coded, we can achieve a better estimation for the mode cost by estimating the effect of the DCT transform with the Hadamard transform. SATD (sum of absolute Hadamard transform differences) in H.264 is defined as:

$$SATD = \sum_{i}^{N} \sum_{j}^{N} |c_{ij}| \qquad (1)$$

where C_{ij} denotes the (i, j)th element of C, which is the Hadamard transform of the residual block. The performance of SATD is close to the Lagrangian function while the computational load is much lower. From simulation results, we find that the SATD cost of the SKIP mode is always smaller than a threshold when the SKIP mode is the best choice for the current macroblock. Therefore, we can consider a threshold for the early selection of the SKIP mode. Since for larger QP, the SKIP mode is preferred, the threshold should vary with QP to reflect quantisation effect. The linear equation of QP which is defined as (2) is found to give good performance. Parameters 'a' and 'b' in the equation are decided by exhaustive experiments

$$\mathrm{Th} = a^{*}\mathrm{QP} + b \qquad (2)$$

We also borrow an early 16 × 16 mode decision scheme [3]. After performing motion estimation of the 16 × 16 block, calculating J_{mode} (16 × 16) and finding the SKIP motion vector, we determine the best mode as the 16 × 16 mode when the following conditions are satisfied. (1.) J_{motion} (16 × 16) is the smallest among J_{motion} (16 × 16), J_{motion} (16 × 8) and J_{motion} (8 × 16). (2.) CBP (16 × 16) is zero. (3.) The SKIP motion vector is the same as the 16 × 16 motion vector. Even though an MB is not determined as 16 × 16 mode, if the conditions (1.) and (2.) are both satisfied, we exclude the P8 × 8 mode for the best mode decision process.

Since the P8 × 8 mode is the most complex mode among all the modes and its frequency increases at small QPs, it is necessary to have a fast P8 × 8 mode decision in the algorithm. Observations show that the best prediction mode of a block is most likely to have the minimum SATD value. To this end, we compare SATD of all P8 × 8 modes. The mode with the smallest SATD value is then selected and other P8 × 8 modes are inactivated.

Fig. 1 shows the flowchart of the proposed inter-mode decision algorithm. The aforementioned procedure can be summarised as:

Step 1: Find SKIP motion vector and estimate SKIP cost by using SATD value.

Step 2: Compare SKIP cost with a threshold. If SKIP cost is less than the threshold, the best mode is determined as SKIP and mode decision procedure stops. Otherwise, go to the next step.

Step 3: Decide the motion vector and reference frame for 16 × 16, 16 × 8 and 8 × 16 modes. Then choose the mode with minimum J_{motion} value among J_{motion} (16 × 16), J_{motion} (16 × 8) and J_{motion} (8 × 16) as sub-optimal best mode.

Step 4: If the following three conditions are all satisfied, inactivate 16 × 8, 8 × 16 and P8 × 8 modes. If only condition (1.) and (2.) are satisfied, inactivate P8 × 8 mode. Otherwise, go to the next step. (1.) Sub-optimal best mode is 16 × 16; (2.) CBP value of 16 × 16 is zero; (3.) Motion vectors of SKIP and 16 × 16 modes are equal.

Step 5: Decide the motion vector and reference frame for P8 × 8 mode. Then calculate SATD value of P8 × 8 mode. Choose the mode with minimum SATD value and inactivate the other P8 × 8 modes.

Step 6: Calculate J_{motion} and J_{mode} for the activated modes and determine the best mode.

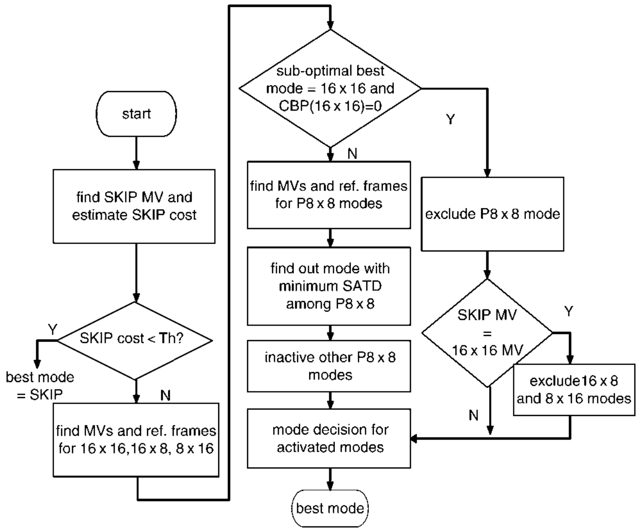

Fig. 1 *Flowchart of fast inter-mode decision algorithm*

Results: The proposed algorithm was implemented on JM 9.5. We examined a variety of CIF video sequences, adopted as test sequences in the MPEG standard. For each sequence, we encoded 100 frames. The simulation conditions are shown in Table 1 [4]. For performance comparison, we used the Bjonteggard delta PSNR and Bjonteggard delta bit rates [5]. Table 2 shows the simulation results where we applied the early SKIP mode decision method only. Table 3 gives the results for the fast inter-mode decision method. All the results are relative to results by the original H.264/AVC without fast motion estimation and fast mode decision. The simulation results show that the proposed early SKIP mode decision method can efficiently reduce the encoding time. The proposed fast inter-mode decision algorithm can achieve 77% time saving on average with negligible loss in PSNR and increment in bit rate. Fig. 2 shows the rate-distortion curves of 'Coastguard' test sequences. The RD performance of the proposed algorithm is almost the same as the H.264 standard.

Table 1: Encoding parameters for fast inter-mode decision algorithm

RDO mode	Fast high complexity mode
GOP structure	IPPP...
Hadamard transform	Used
Search range	± 16
Reference frames	5
Quantisation parameters	28, 32, 36, 40
FME algorithms	UMHexagonS, CBFPS

Table 2: Performance comparison using early SKIP mode decision only

Sequence	Performance	Early skip	Fast method
Foreman	ΔTime (%)	−69.83	−80.95
	ΔPSNRY (dB)	−0.095	−0.17
	ΔBits (%)	−0.0595	1.255
Coastguard	ΔTime (%)	−60.2	−71.11
	ΔPSNRY (dB)	−0.015	−0.08
	ΔBits (%)	0.0105	−0.68

Table 3: Performance comparison for fast inter-mode decision algorithm

Sequence	ΔPSNRY (dB)	ΔBits (%)	ΔTime (%)
Foreman	−0.17	1.255	−80.95
Akiyo	−0.12	0.01	−91.86
Mobile	−0.12	0.397	−70.88
City	−0.08	0.456	−78.55
Crew	−0.12	0.838	−74.68
Bus	−0.11	1.256	−71.43
Soccer	−0.09	1.771	−78.40
Coastguard	−0.08	−0.68	−71.11
Average	−0.11	0.663	−77.23

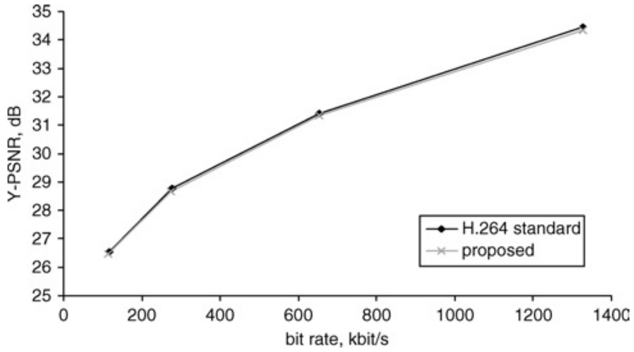

Fig. 2 *RD performance of 'Coastguard' for inter-mode decision*

Acknowledgments: This work was supported in part by IITA, in part by MIC through RBRC, and in part by MOE through the BK21 project.

29 July 2007
Electronics Letters online no: 20072214
doi: 10.1049/el:20072214

L.-J. Pan and Y.-S. Ho (*Department of Information and Communications, Gwangju Institute of Science and Technology, Gwangju 500-712, South Korea*)

E-mail: hedy@gist.ac.kr

References

1 Wiegand, T., Sullivan, G.J., Bjontegaard, G., and Luthra, A.: 'Overview of the H.264/AVC video coding standard', *IEEE Trans. Circuits Syst. Video Technol.*, 2003, **13**, (7), pp. 560–576
2 Sullivan, G.J., and Wiegand, T.: 'Rate-distortion optimization for video compression', *IEEE Signal Process. Mag.*, 2003, **15**, pp. 74–90
3 Kim, G.Y., Yoon, B.Y., and Ho, Y.S.: 'Fast inter mode decision algorithm in H.264/AVC for IPTV broadcasting server'. VCIP, 2007, SanJose, California, USA
4 Sullivan, G.J., and Bjontegaard, G.: 'Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive Scan Source Material', ITU-T Q.6/16, Doc. VCEG-N81, 2001
5 Bjontegaard, G.: 'Calculation of average PSNR difference between RDCurve', ITU-T Q.6/16, Doc. VCEG-M33, 2001