Hindawi Publishing Corporation
International Journal of Reconfigurable Computing
Volume 2012, Article ID 813023, 10 pages
doi:10.1155/2012/813023

Research Article

Algorithm and Hardware Design of a Fast Intra Frame Mode Decision Module for H.264/AVC Encoders

Daniel Palomino,[1] Guilherme Corrêa,[1] Cláudio Diniz,[1] Sergio Bampi,[1] Luciano Agostini,[2] and Altamiro Susin[1]

[1] *Microelectronics Group, INF, Federal University of Rio Grande do Sul, Avenida Bento Gonçalves 9500, P.O. Box 15064, 91501-970 Porto Alegre, RS, Brazil*
[2] *Group of Architectures and Integrated Circuits, CDTEC, Federal University of Pelotas, Campus Universitário s/n, P.O. Box 354, 96001-970 Pelotas, RS, Brazil*

Correspondence should be addressed to Daniel Palomino, dmvpalomino@inf.ufrgs.br

Received 19 January 2012; Revised 29 May 2012; Accepted 21 July 2012

Academic Editor: Massimo Conti

In the rate-distortion optimization (RDO), the process of choosing the best prediction mode is performed through exhaustive executions of the whole encoding process, increasing significantly the encoder computational complexity. Considering H.264/AVC intra frame prediction, there are several modes to encode a macroblock (MB). This work proposes an algorithm and the hardware design for a fast intra frame mode decision module for H.264/AVC encoders. The application of the proposed algorithm reduces in more than 10 times the number of encoding iterations for choosing the best intramode when compared with RDO-based decision. The architecture was synthesized to FPGA and achieved an operation frequency of 98 MHz processing more than 300 HD1080p frames per second. With this approach, we achieved one order-of-magnitude performance improvement compared with RDO-based approaches, which is very important not only from the performance but also from the energy consumption perspective for battery-operated devices. In order to compare the architecture with previously published works, we also synthesized it to standard cells. Compared with the best previous results reported, the implemented architecture achieves a complexity reduction of five times, a processing capability increase of 14 times, and a reduction in the number of clock cycles per MB of 11 times.

1. Introduction

H.264/AVC is the state-of-art video compression standard proposed by ITU-T and ISO-IEC [1]. It has shown significantly better coding performance than existing video coding standards, for example, about 50% bit-rate reduction compared with MPEG-2, and, for this reason, it has been adopted by most of electronic devices that encode digital video, such as camcorders and mobile phones. A huge amount of coding options have been included in the H.264/AVC encoder to achieve a better coding efficiency for different video sequences. Since all video codecs, as well as H.264/AVC, include lossy compression schemes, the video encoder must know how much the video quality is degraded for a given target quality, that is, there is a direct relation between the amount of bits of the coded video and its visual quality.

Figure 1 shows how the video quality behaves for a target bit-rate of a coded video. The video quality is measured in peak signal-to-noise ratio (PSNR), and the bit-rate is measured in bits per second (bps). The point **a** in Figure 1 shows a very low bit-rate video with very degraded quality, which is not desirable for most applications. The point **c** in Figure 1 shows a higher quality video but with very large bit-rate, which is not useful for practical applications with restricted bandwidth. The point **b** in Figure 1 shows a video with balanced relation between quality and bit-rate.

Those various coding tools, included mainly on the prediction modules (intra frame prediction and interframes prediction) of an H.264/AVC encoder, must be evaluated

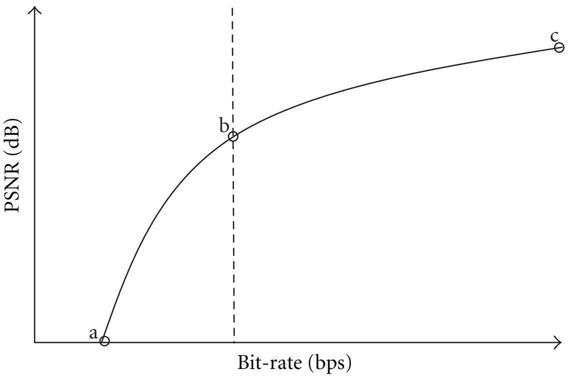


Figure 1: Rate-distortion behavior of an encoded video.

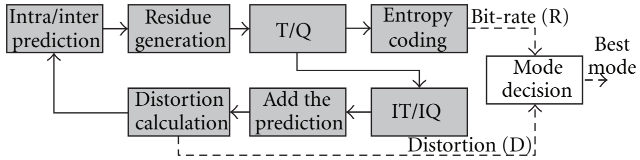


Figure 2: Diagram of RDO-based decision.

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Figure 3: Neighbor samples used in the prediction of a 4 × 4 block.

in terms of their rate-distortion results to be selected, and then the residual data between the coding possibility that generates the best predicted block and the original block is encoded to form the final video stream. The rate-distortion optimization (RDO) [2] is a well-known technique for maximizing coding quality and minimizing the amount of coded bits. It is usually employed in video encoder implementations to achieve the best coding result. However, the RDO technique demands a high computational complexity in H.264/AVC encoders, since it exhaustively examines all intra- and interprediction modes, performing a complete encoding process for each mode. Up to 90% of total encoding time may be spent in the mode decision stage [3].

Figure 2 shows the diagram of the RDO-based mode decision to better explain its complexity. Grey blocks are performed once for each prediction mode, while the mode decision block (in white) receives all bit-rates and distortions (dashed lines) of the candidate modes and the best mode selected is the one that presents a better tradeoff among bit-rate (R) and distortion (D).

In the intra frame prediction for the luminance layer, there are two possible block size partitions to encode one macroblock (MB): (1) I16MB, with four possible prediction modes applied to the whole MB (16 × 16 samples) and (2) I4MB, with nine possible prediction modes applied to the sixteen blocks of 4 × 4 samples which compose the MB. For the chrominance layer there are also four possible modes to predict each 8 × 8 block (Cr and Cb) in a MB. Considering an HD1080p video sequence (1920×1080 pixels), 138.720 iterations of prediction, forward transform and quantization, inverse transform and quantization and entropy coding (called in this work as *encoding loop*) (Figure 2) are needed to encode each intra frame using the RDO technique. This way, it is clear that the RDO-based decision is hard to be used when high resolution and real-time applications are considered. Besides that, an encoder architecture that uses the RDO-based decision will spend a lot of clock cycles and energy to perform the encoding loop for all those candidate modes that will not be included in the bit stream at the end.

Due to this complexity, some works as [4–6] have proposed fast intra frame mode decision algorithms and hardware designs to decrease the encoding time for one MB. However, all these works focus only on reducing the number of modes that will be evaluated by the RDO-based decision. The work [4] proposes an intradecision based on the dominant edge strength. The authors use a filtering technique to perform the edge detection. This way, the number of modes is reduced from nine to four in a 4 × 4 luma block. For 16 × 16 luma or 8 × 8 chroma block, only the detected mode and the DC mode are selected to be evaluated by the RDO technique. In the work [5], the authors proposed a modified low complexity mode decision algorithm based on a cost function composed by distortion and an estimated rate. The work [6] proposes a modified three-step algorithm [7] to perform the intradecision. As well as the work in [4], the main goal is to decrease the number of I4MB candidate modes (from nine to seven) to be evaluated by the RDO technique.

The main drawback of the related works [4–7] is that the proposed techniques can only reduce the number of candidate modes to be evaluated by the RDO process. This is a good technique but the gains in computational complexity are limited, since the RDO process is still executed for some modes. In this work, our approach is very different. There are other relevant works [8–12] in the literature proposing fast mode decision algorithms to improve the intra prediction performance. These works present interesting techniques to reduce the computational complexity of the intra prediction process. However, none of these works takes the hardware implementation issues into account and all of them still use the RDO-based decision.

We propose a fast algorithm and its hardware architecture in order to completely eliminate the RDO-based decision of the encoding process, thoroughly decreasing the time needed to encode one MB. The algorithm uses a threshold value to choose the partition type (I16MB or I4MB), and for each partition type the encoding mode is selected with a simple distortion metric, for example, sum of absolute differences

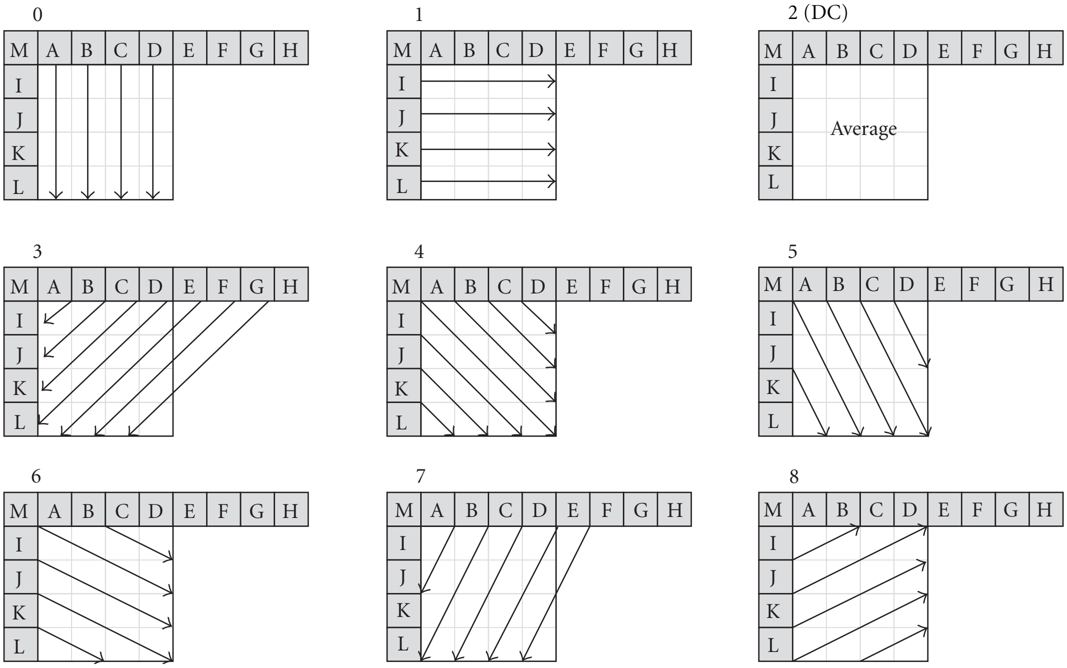


Figure 4: Nine possible prediction modes for I4MB macroblocks.

(SAD). The threshold value is based on the difference of distortions (DD) and is determined by offline simulations with several video sequences which represent a variety of illumination and texture patterns.

The paper is organized as follows. Section 2 presents the intra frame prediction and all possible modes to perform the prediction. Section 3 shows our fast intradecision algorithm. Section 4 shows the designed architecture of the intradecision and compares it with state of the art. Finally, Section 5 concludes this work.

2. Intra Frame Prediction

The intra frame prediction module exploits the spatial redundancy inside a video frame. It is performed by an interpolation of the neighbor pixels previously coded and decoded. In the H.264/AVC standard, the intra prediction is applied either for luminance or chrominance samples. Considering the luminance samples, there are two possible partition sizes: (1) 4 × 4 (I4MB) and (2) 16 × 16 (I16MB). When the I4MB partition is chosen, the 16 × 16 luminance block is divided into 16 4 × 4 blocks and, then, they are all individually predicted. In the I16MB partition, the entire 16 × 16 luminance block is predicted. The chrominance samples are always predicted using the same size (8 × 8), when the 4 : 2 : 0 subsampling is used.

2.1. 4 × 4 *Partition Size (I4MB).* For the I4MB partition size, there are nine different ways to generate the prediction of one 4 × 4 block. Figure 3 shows the neighbor samples that can be used to perform the prediction over a 4 × 4 block. Samples A to M were previously encoded and reconstructed. Samples a to p represent the current block that is being predicted.

The interpolation of the samples A to M is performed according to an angle direction that defines which and how the samples will be interpolated to generate the predicted block. Figure 4 shows the nine prediction modes for I4MB macroblocks.

In the modes 0 and 1, only a copy is performed (following the presented directions) to generate the predicted block. In the mode 2, an average over the neighbor samples is performed to generate all pixels of the predicted block. In the modes 3, 4, 5, 6, 7, and 8, an interpolation following the directions is performed to generate the predicted block [13].

2.2. 16 × 16 *Partition Size (I16MB).* When the partition size used is 16 × 16 (I16MB), there are four different ways to perform the prediction over a block. There are 32 neighbor samples (H and V in Figure 5) to be used in the prediction modes. Figure 5 shows the four possible modes to the I16MB partition.

In modes 0 (VERTICAL, in Figure 5) and 1 (HORIZONTAL, in Figure 5), the samples H and V are copied in the vertical and horizontal directions, respectively, for the whole 16 × 16 block. In mode 2 (DC, in Figure 5), the samples in the predicted block are generated using the average value between all neighbor samples (H and V).

The mode 3 (PLANE, in Figure 5) is the most complex, since its calculation uses a linear function where three parameters are used for each sample in the predicted block [13]. The four chrominance modes are the same as these used for I16MB partitions, but for smaller blocks (8 × 8).

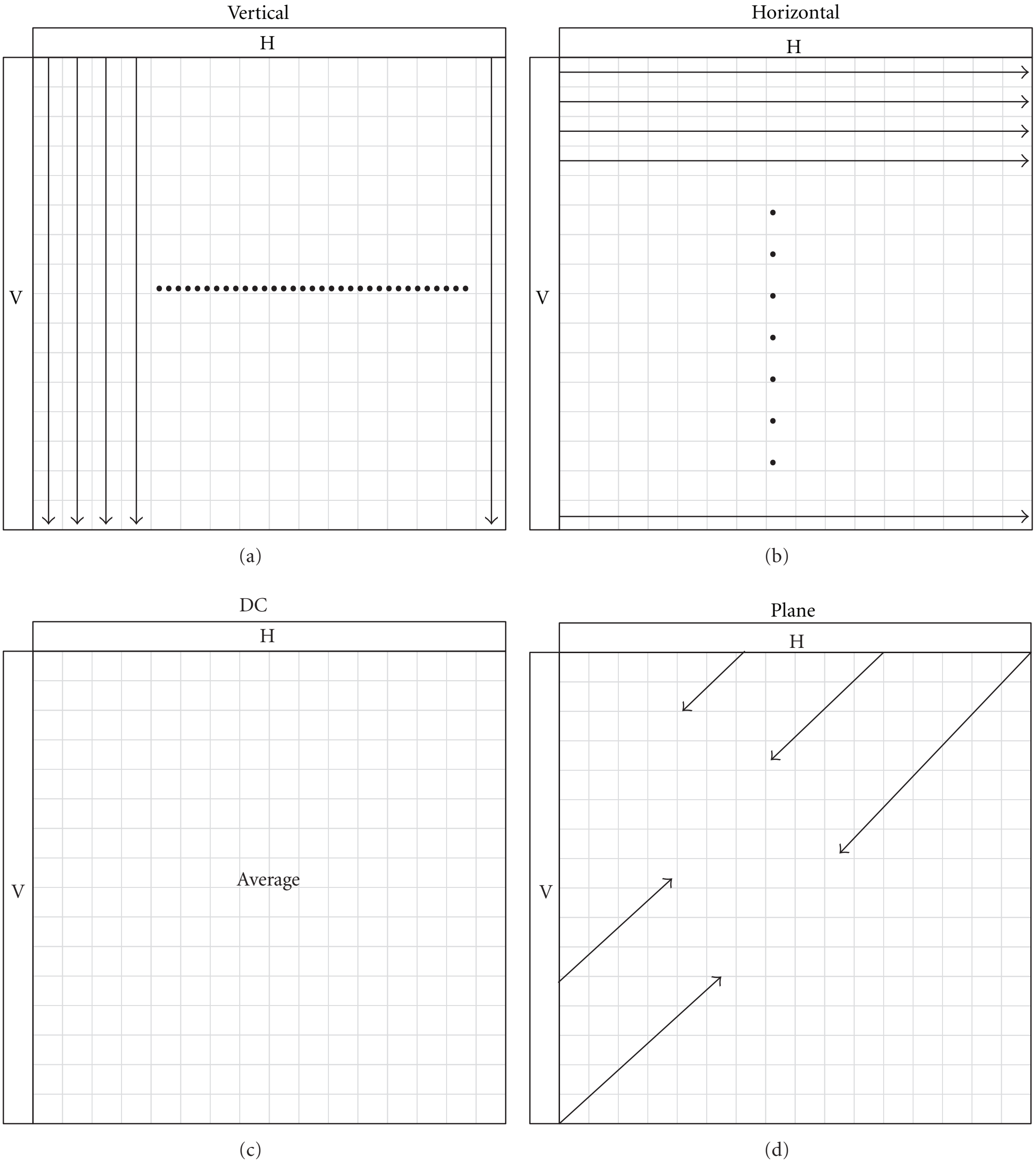


Figure 5: Four possible modes for I16MB macroblocks.

3. Fast Intra Frame Mode Decision Algorithm

Our fast intramode decision algorithm is based only on distortion calculation. The decision is performed in a hierarchical way in two steps: (1) decision among equal partition sizes and (2) decision among different partition sizes. In the first step, distortion calculation for different partition types (I4MB and I16MB) is calculated independently. In the second step, we applied a metric that we called difference of distortion (DD) and a heuristic to choose which partition shall be coded to obtain the best RD result. The next subsections will better explain our approach.

3.1. Decision among Equal Partition Sizes. The first decision step is based on distortion calculation to choose the best I16MB partition considering the four possible modes and the best I4MB partition considering the nine possible modes. Several simulations considering three different distortion metrics were performed: sum of absolute differences (SAD), sum of squared differences (SSD), and sum of absolute transformed differences (SATD). The results (for bit-rate and video quality) obtained using these three metrics were compared among each other, and a comparison was performed considering computational complexity measured by the number of sums (see Table 1).

Table 1: Comparison among SAD, SSD, and SATD.

SSD versus SAD			SATD versus SAD		
PSNR (dB)	Bit-rate (%)	Sums (%)	PSNR (dB)	Bit-rate (%)	Sums (%)
+0.008	−0.63	+361.29	+0.079	−1.13	+348.39

SATD and SSD metrics show better RD results (bit-rate and video quality). However, the computational complexity of these two metrics is extremely much larger than SAD (about 361% larger when the SSD metric is considered). As the main goal of this work is to design a faster intramode decision, we decided to use SAD as the distortion metric. In addition to that, the hardware architecture design for SAD calculators is simpler than one for SATD (which includes a 4 × 4 Hadamard transform) and SSD (which includes a multiplier and a square root). So in this step, a SAD calculation is performed for all modes considering each 4 × 4 block and the partial results are accumulated to obtain the SAD of the entire MB in I4MB modes. A SAD calculation is also performed for the four I16MB modes and four chrominance modes.

3.2. Decision among Different Partition Sizes. The second step of the proposed intradecision is to choose which partition size (I4MB or I16MB) will be used to encode the luminance channel of MB. This decision is made using the information generated by the first step: the distortion of the best I4MB partition and the distortion of the best I16MB partition measured with SAD. A simple comparison between these two values would cause, in most cases, the choice for I4MB partitions, because of their finer prediction granularity and more coding modes. However, analyzing the behavior of the difference between these two distortion values when RDO-based decision is applied, it is possible to make a good choice on which partition shall be used for each MB.

Simulations were performed with various video sequences using JM H.264 reference software [14] set in full-RDO-based decision and intra-only MB modes (to choose only between I4MB or I16MB modes). It was possible to notice that in most cases when I16MB partition had been chosen the distortion values of the best I4MB partition and the best I16MB partition were very close. It means that most of the 16 4 × 4 modes were the same, so it is more beneficial to choose a 16 × 16 partition (I16MB mode), since it will generate less modes information, that is, less bits in the bit stream to signalize mode information. On the other hand, when I4MB partitions were chosen, the distortion values of the best I4MB and the best I16MB partition were very different. It means that most of the 16 4 × 4 modes were different and even choosing only one 16 × 16 mode it will generate a lot of residual information.

Then, we created a metric called difference of distortions (DD) and several simulations were performed to classify in which situation each intra prediction mode is selected by the RDO technique. Equation (1) shows the difference of distortion (DD) calculation where the $\mathrm{SAD}_{\mathrm{I4MB}}$ is the sum of all residual generated by the 16 best 4 × 4 modes and the $\mathrm{SAD}_{\mathrm{I16MB}}$ is the residual generated by the best 16 × 16 mode:

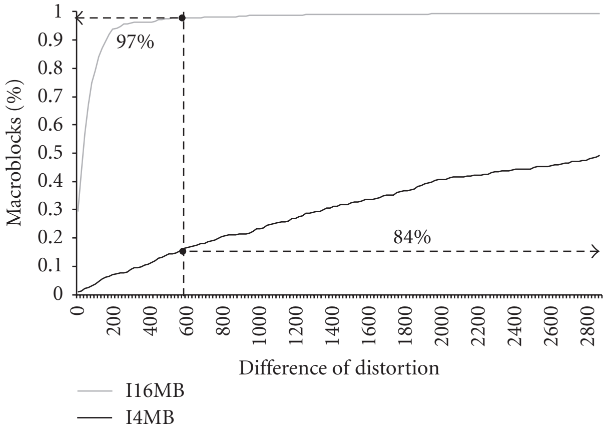


Figure 6: Percentage of modes chosen by the RDO technique and difference of distortion (DD) for the MBs.

$$\mathrm{DD} = \mathrm{SAD}_{\mathrm{I4MB}} - \mathrm{SAD}_{\mathrm{I16MB}}. \tag{1}$$

Figure 6 shows a graph where the percentage of chosen modes (I4MB and I16MB) for each MB selected by the RDO technique is compared with the difference of distortion generated by the best I4MB and I16MB partitions (first decision step). With this metric, the partition (I4MB or I16MB) is chosen and then the rate-distortion calculation is performed.

With the difference of distortion set to 600, for example, it is possible to see that in most cases when the I16MB partition is chosen (97%) the difference of distortion is very small (lesser than 600), while when the I4MB partition is chosen the difference of distortion is very large (84% are bigger than 600). This way, it is possible to use this information in comparison with a threshold value to choose the partition size for intra frame MBs.

The threshold value that presents the best results in terms of PSNR and bit-rate was obtained as follows: (1) all videos used in the simulations were first encoded using the RDO technique for the decision among different block sizes. During that, the distortion values and the chosen partition were saved. Then, the differences of distortion were compared with the chosen partition to define the threshold value. Simulations were performed with a threshold ranging from 0 to 1000, adjusting it to bit-rate and video quality. Threshold value set to 600 generated the best results considering the bit-rate and video quality relation for the video sequences evaluated.

The results obtained by using the proposed intradecision algorithm are presented in Table 2. The first columns present the results using RDO-based decision. The central columns present the results obtained by the proposed decision. Finally, the last columns show a comparison between the two approaches in terms of bit-rate increase and image quality

Table 2: Results of the proposed intramode decision and comparison with RDO-based decision.

Video	RDO		Proposed decision		RDO versus proposed decision (threshold 600)	
	PSNR (dB)	Bit-rate	PSNR (dB)	Bit-rate	PSNR loss	+Bit-rate (%)
Station 2	39.643	34022 k	39.338	35729 k	0.305	5.02
Sunflower	42.766	33030 k	42.413	35391 k	0.353	7.15
Tractor	39.400	58208 k	39.035	60939 k	0.365	4.69
Traffic	39.643	50884 k	39.308	53070 k	0.335	4.30
Manincar	42.609	8818 k	42.518	9018 k	0.091	2.27
Pedestrian area	40.878	24949 k	40.699	26875 k	0.179	7.72
Riberbed	38.652	56735 k	38.280	58962 k	0.372	3.93
Rolling tomatoes	40.475	12170 k	40.387	12537 k	0.088	3.02
Rushhour	41.694	20013 k	41.484	21306 k	0.210	6.46
Average	40.640	33203 k	40.385	34870 k	0.255	5.02

Table 3: Comparison with related works.

Works	PSNR loss (dB)	Bit-rate increase (%)	Number of iterations reduction (times)
Wang et al. [4]	0.021	2.92	2.6
Kao et al. [5]	0.010	—	—
Lin et al. [6]	0.110	1.03	1.1
This work	0.255	5.02	13

(PSNR) difference. These results were obtained through simulations performed with HD1080p video sequences [15] which represent a variety of illumination and texture patterns.

The application of the proposed heuristic resulted in an average increase of 5.02% in the bit-rate and an average decrease of 0.255 dB in the image quality (PSNR). The increase of bit-rate and the decrease of image quality are very small and are justifiable by the enormous computational complexity reduction achieved in the decision process. As presented in Figure 2, the RDO-based encoding process is finished only after the execution of all possible intra frame prediction modes by the encoding loop. The decision proposed in this work is performed after the generation of the predicted blocks by the intra prediction followed by the SAD-based distortion calculation and then the difference of distortion operation. This way, the encoding loop presented in Figure 2 is completely eliminated resulting in enormous gain in terms of computational complexity reduction of the intra frame decision process. When RDO-based decision is performed, four I16MB modes and nine I4MB intra frame modes must be evaluated, totalizing 13 encoding iterations per MB. Considering the proposed decision method, the encoding process is performed only once for each MB. Table 3 presents a comparison with related works in terms of bit-rate, image quality (PSNR), and reduction in RDO calculations.

While other works have shown a reduction of coding iterations from 1.1 to 2.6 times in comparison with RDO-based decision, the proposed decision allows a reduction of 13 times (one order-of-magnitude). The cost of this gain resides, however, in the bit-rate increase of 5.02% and image quality loss 0.255 dB which do not compromise coding efficiency when the enormous gain in terms of computational complexity reduction is considered. Moreover, reducing the number of encoding loop iterations, it is possible to reduce the number of clock cycles and energy consumption needed by the hardware architecture to perform the whole prediction of one MB.

4. Designed Architecture and Comparisons

In order to further improve the performance of our fast intra frame decision scheme, a hardware architecture was designed. The fast intramode decision architecture is shown in Figure 7, which consists of 17 SAD calculators (nine for I4MB modes, four for I16MB modes, and four for chroma modes), three comparators, and the DD mode decision module. The distortion calculation is performed by the SAD calculators using as input the predicted block and the original block. Considering the distortion calculation for I4MB partitions, the SAD value of each mode is compared and then the 16 lowest SADs of each 4 × 4 block are accumulated to generate the total distortion for the I4MB partition. For I16MB partitions, the SAD values among the four modes are compared and then the lowest one is chosen as the best I16MB partition. As chrominance samples are predicted considering only one partition size (8 × 8), the decision among these samples is easier. The SAD values for each mode are compared, and the lowest one is chosen as the best.

Figure 8 shows the RTL diagram of each SAD calculator. The SAD calculator consumes eight samples (two lines of a 4 × 4 block) per cycle. It was designed with two pipeline stages,

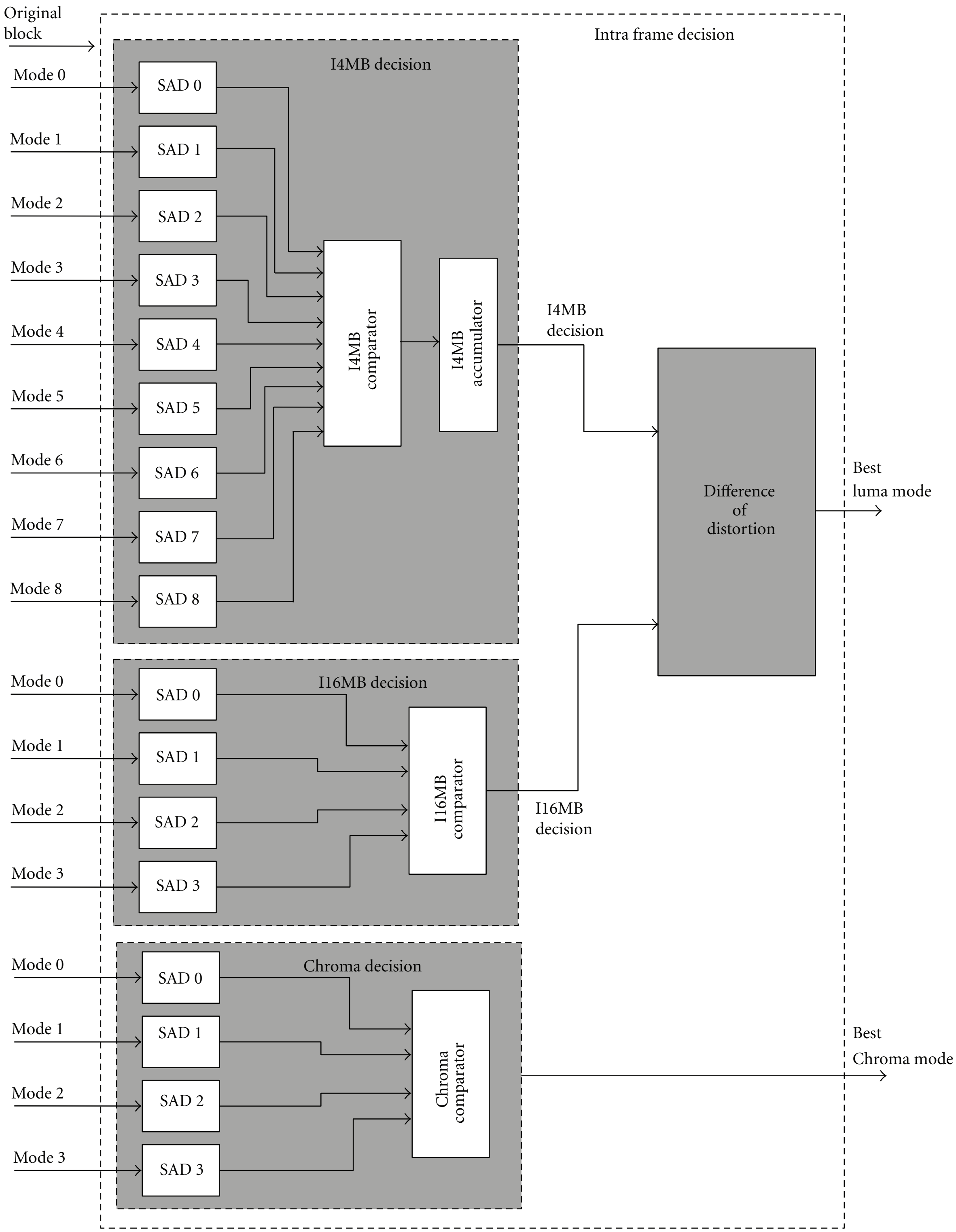


Figure 7: Fast mode decision architecture.

that is, it takes two clock cycles to deliver the first result. There is a little difference between the SAD calculator used in the I4MB distortion decision and the SAD calculator used in the I16MB distortion decision. Considering I4MB partition, the accumulated value is used by the comparator each time that one 4 × 4 block was processed, since the prediction is performed over 4 × 4 blocks. After that, the lowest SAD of the nine 4 × 4 blocks is chosen as the best, and then it is

Table 4: Synthesis results and comparison with related works.

Works	Technology	Hardware resources	Number of cycles	Max. frequency	Number of HD1080p frames/s	Frequency to process HD1080p@30 fps
Wang et al. [4]	UMC 0.18 μm	10,302 gates	416	66 MHz	31	62.21 MHz
Kao et al. [5]	TSMC 0.13 μm	11,229 gates	672	75 MHz	Not supported	Not supported
Lin et al. [6]	TSMC 0.13 μm	*94,700 gates	560	140 MHz	30	140 MHz
This work	FPGA	3,267 ALUTs 2,312 DLRs	36	98.43 MHz	335	8.26 MHz
	TSMC 0.18 μm	28,518 gates	36	129.1 MHz	439	8.26 MHz

* Hardware resources considering the complete intra-coder.

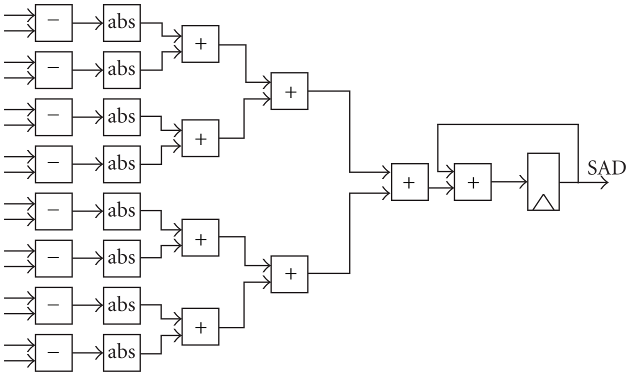


Figure 8: RTL diagram of one SAD calculator.

accumulated again until all the 16 4 × 4 blocks are read. On the other hand, when the best I16MB and the best 8 × 8 chroma partitions are considered, only one accumulator is needed, since the prediction is generated over the whole block. Finally, the comparator chooses the best mode.

Figure 9 shows the time diagram of the designed architecture. As the SAD calculators were designed with two pipeline stages and with eight samples of parallelism, the architecture takes 3 clock cycles to deliver the first valid SAD value of a 4 × 4 block. When the pipeline is filled, there is a valid SAD value every two clock cycles. This way, the architecture takes 34 clock cycles to evaluate all the nine I4MB modes and one more clock cycle to accumulate the last best SAD. Considering the I16MB partition decision, the SAD values are accumulated in the SAD calculator itself. This way, the architecture takes 33 clock cycles to accumulate the SAD for the four modes and one more clock cycle to compare them. The chroma decision is similar to the I16MB decision; however, the block size is of 8 × 8 samples. Then, it only takes 10 clock cycles to perform the whole chroma decision. After the calculation of the best I4MB distortion and the best I16MB distortion, the difference of distortion is evaluated in one clock cycle to decide which partition size will be used.

The architecture was described in VHDL and synthesized targeting the EP2S130F1508C3 Stratix II FPGA [16]. Even though we have performed an extensive research in the technical literature, no works using FPGA as device target in intra prediction mode decision were found. This way, the architecture was also synthesized for TSMC 0.18 μm standard cells [17] to allow for a fairer comparison. Table 4 presents the synthesis results for both technologies and compares them with previous works. The results are presented in terms of hardware resources usage, number of clock cycles needed to process one MB, maximum operation frequency achieved, and throughput measured in HD1080p frames per second.

When synthesized for FPGA, the architecture used 3.267 ALUTs (look-up tables) and 2.312 DLRs (dedicated logic register), totalizing 4% of the resources in the device. The FPGA synthesis achieved 98.43 MHz as maximum operation frequency, being able to process up to 335 HD1080p frames per second. When synthesized to TSMC 0.18 μm, the total gate count was 28.518. The maximum operation frequency achieved was 129.1 MHz. This way, the architecture is able to process 439 HD080p frames per second.

Compared with previous works [4–6], the designed architecture consumes the lowest number of cycles to process an intra frame: more than 11X reduction compared with [4], 18X reduction compared with [5]. It also results in the highest throughput among the related works: more than 11X and 14X increase in number of HD1080p frames encoded per second for FPGA and TSMC 0.18 μm versions, respectively, compared with [4]. All these results were obtained considering our architecture operating at maximum frequency. An interesting result is that with higher throughput we can reduce the operating frequency down to 8.26 MHz and still process HD1080p videos at 30 fps, which is the target frame rate normalized by H.264/AVC standard for real-time systems [1]. With this really low frequency, we can achieve very low power using our architecture, which is an excellent alternative for battery-powered devices. However, if the whole encoder design is considered, this low frequency

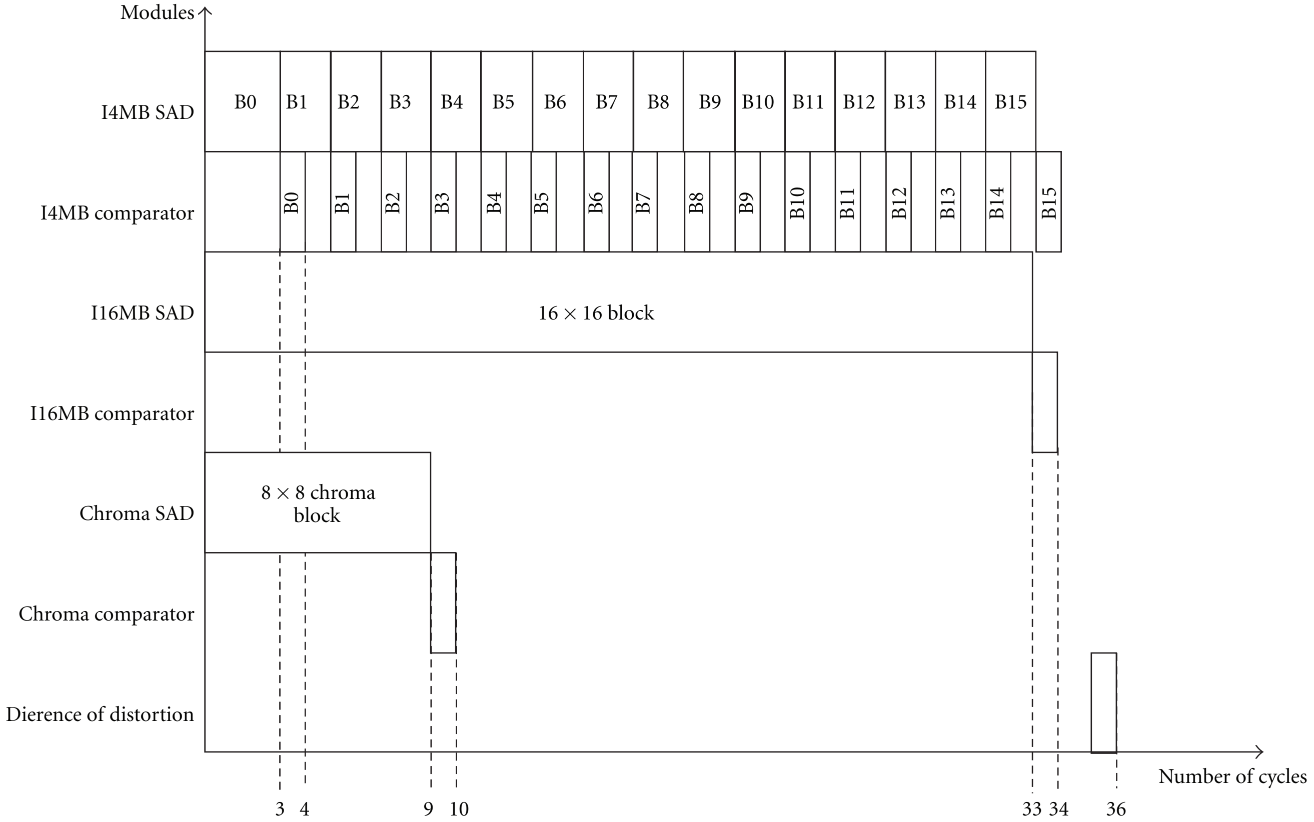


Figure 9: Schedule of the designed architecture.

could not allow HD1080p processing in real time if the other encoder modules do not achieve such performance.

5. Conclusions

This work has presented a fast intramode decision algorithm and its hardware architecture design for an H.264/AVC video encoder. The proposed algorithm allows the complete elimination of the encoding loop present in RDO-based mode decision, which is substituted by simple distortion calculations and comparisons, thoroughly decreasing the complexity of the video encoder. The number of encoding iterations was reduced in 13 times when compared with RDO-based decision, at the cost of relatively small bit-rate increase (5.02% in average) and image quality loss (0.255 dB in average).

When compared to other works, the proposed algorithm achieves a complexity reduction more than five times higher, while the bit-rate increase and the image quality loss are slightly higher and still similar to the compared works. The designed architecture of the fast intradecision algorithm was described in VHDL and synthesized targeting two technologies: (1) Stratix II FPGA and (2) TSMC 0.18 μm standard cell library. The synthesis results have shown that the architecture is able to process up to 439 HD1080p frames per second.

References

[1] ITU-T Recommendation H.264/AVC (05/03): advanced video coding for generic audiovisual services, 2003.

[2] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for: video compression," *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[3] S. B. Solak and F. Labeau, "Complexity scalable video encoding for power-aware applications," in *Proceedings of the International Green Computing Conference*, pp. 443–449, Chicago, Ill, USA, August 2010.

[4] J.-C. Wang, J.-F. Wang, J.-F. Yang, and J.-T. Chen, "A fast mode decision algorithm and its vlsi design for H.264/AVC intra-prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 10, pp. 1414–1422, 2007.

[5] Y.-C. Kao, H.-C. Kuo, Y.-T. Lin et al., "A high-performance VLSI architecture for intra prediction and mode decision in H.264/AVC video encoding," in *Proceedings of IEEE Asia Pacific Conference on Circuits and Systems (APCCAS '06)*, pp. 562–565, Singapore, December 2006.

[6] Y.-K. Lin, C.-W. Ku, D.-W. Li, and T.-S. Chang, "A 140-MHz 94 K gates HD1080p 30-frames/s intra-only profile H.264 encoder," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 3, pp. 432–436, 2009.

[7] C. C. Cheng and T. S. Chang, "Fast three step intra prediction algorithm for 4x4 blocks in H.264/AVC," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 769–772, May 2005.

[8] M. Parlak, Y. Adibelli, and I. Hamzaoglu, "A novel computational complexity and power reduction technique for H.264 intra prediction," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 4, pp. 2006–2014, 2008.

[9] Y. H. Huang, T. S. Ou, and H. H. Chen, "Fast decision of block size, prediction mode, and intra block for H.264 intra prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 8, pp. 1122–1132, 2010.

[10] J. H. Min, S. Lee, I. K. Kim, W. J. Han, J. Lainema, and K. Ugur, "Unification of the directional intra prediction methods in TMuC," in *Proceedings of the 2nd Meeting of Joint Collaborative Team on Video Coding*, JCTVC-B100, Geneva, Switzerland, July 2010.

[11] J. H. Min, S. Lee, I. K. Kim, W. J. Han, J. Lainema, and K. Ugur, "TE4: Results for Simplification of Unified Intra Prediction," JCTVC-C042, Guangzhou, China , 2010.

[12] L. Zhao, L. Zhang, S. Ma, and D. Zhao, "Fast mode decision algorithm for intra prediction in HEVC," in *Proceedings of IEEE Visual Communications and Image Processing*, pp. 1–4, Tainan, Taiwan, November 2011.

[13] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[14] JM H.264 reference software version 16.2, http://iphome.hhi.de/suehring/tml/.

[15] "Video Library and Tools," http://nsl.cs.sfu.ca/wiki/index.php/Video_Library_and_Tools#1080p.

[16] Altera Corporation, "Altera: The Programmable Solutions Company," http://www.altera.com.

[17] Artisan Components, "TSMC 0.18μm 1.8-Volt SAGE-XTM Standard Cell Library Databook," 2001.

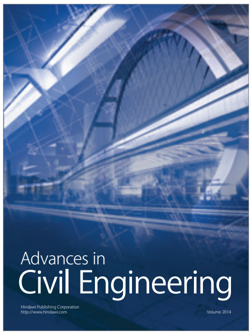
Journal of
Engineering

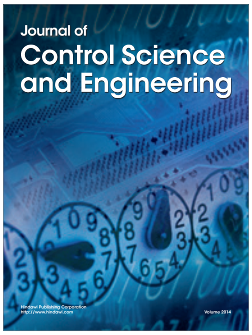
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

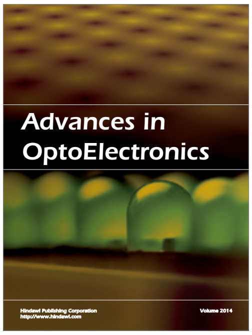
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

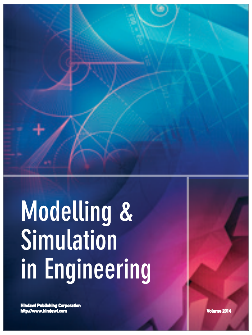
Journal of
Robotics

Journal of
Electrical and Computer Engineering

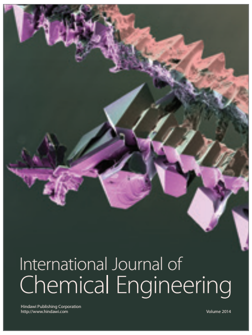

Advances in
OptoElectronics

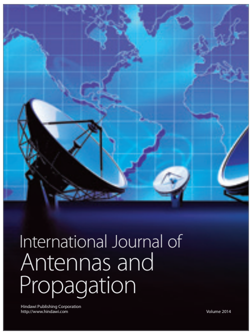
VLSI Design

International Journal of
Navigation and Observation

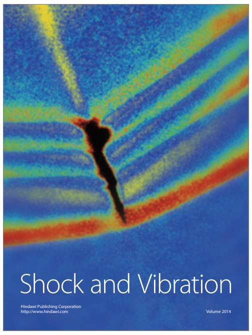

Modelling &
Simulation
in Engineering

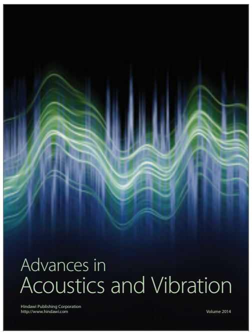
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration