

H.263 MOBILE VIDEO CODEC BASED ON A LOW POWER CONSUMPTION DIGITAL SIGNAL PROCESSOR

Yukihiro NAITO and Ichiro KURODA

C&C Media Research Laboratories, NEC Corporation
1-1, Miyazaki 4-chome, Miyamae-ku, Kawasaki 216 Japan

ABSTRACT

This paper describes an H.263 video codec implementation based on a low power consumption general purpose DSP. Fast algorithms, such as a fast motion estimation algorithm and a low complexity noise reduction filter, are proposed to implement the video codec on a single DSP chip maintaining sufficient picture quality. By using a 50MIPS, 100mW DSP, the developed codec encodes and decodes 7.5 QCIF frames per second, which is sufficient performance for low bit-rate video compression, typically below 64kbps.

1. INTRODUCTION

The H.263 video codec[1] is a ITU-T recommendation for low bit-rate video compression, typically below 64kbps. The current target application is PSTN videotelephony, ie. videotelephony on normal analogue telephone lines. In addition, it will be used for mobile networks, combined with the error resilient multiplexing protocol H.223[2].

Low power implementation of the H.263 video codec is important for the portable videotelephony over mobile networks. In terms of low power consumption, a dedicated LSI is better than a programmable processor. On the other hand, programmability is also important for sophisticated coding control for better image quality as well as future changes of standards, such as the H.263+ and the MPEG4. Under this contradictory circumstance, a processor architecture should be decided based on performance requirements. In the case of the video codec, they involve target frame rate and image size as well as power consumption. The bit-rate around 32kbps is now available in Japan by using the personal handyphone system (PHS) high speed data transmission service, and 64kbps mobile communication is also going to be available in a few years, which sets up the target performance as 5 to 10 QCIF (176 × 144 pixels) frames per second.

To achieve the performance mentioned above, we selected an approach of using a general purpose 100mW DSP with fast algorithms as a performance enhancement method. As opposed to application specific DSPs, general purpose

DSPs can efficiently implement various kinds of applications, such as a speech codec with echo canceler[3] as well as the video codec.

This paper describes an H.263 video codec implementation based on a 100mW DSP. Fast algorithms, such as a fast motion estimation (ME) algorithm and a low complexity noise reduction (NR) filter, are proposed to achieve the required performance.

2. LOW POWER DSP μ PD7701x

As the low power consumption general purpose DSP, we selected μ PD7701x, which is a 16 bits fixed-point DSP family[5]. The performance ranges from 33MIPS to 60MIPS depending on the processor type. The power consumption is typically around 100mW at 50MIPS and 3V. The internal circuit consists of 8 general registers (40bits), a multiplier (16bits × 16bits + 40bits → 40bits), a data ALU (40bits) and a barrel shifter (40bits). All instructions are executed in one cycle except for branch instructions.

3. H.263 VIDEO CODEC

Figure 1 shows the block diagram of the H.263 video encoder. The H.263 coding algorithm is similar to that used by H.261[4], which is used in the ISDN teleconference, but there are some changes in order to enhance its performance. The H.263 uses half pixel precision for motion compensation, whereas the H.261 uses full pixel precision and a loop filter. In addition to the core H.263 coding algorithm, four negotiable coding options are included, which improve the coding performance.

As for the four negotiable coding options, the developed codec supports two options that significantly improve the coding performance: the advanced prediction mode (Annex F) and the PB-frames mode (Annex G). NR filters, which are not specified in the recommendation, are also important for the low bit-rate coding. The developed encoder includes a low complexity NR filter, which has newly been developed for better coding performance and an efficient implementation.

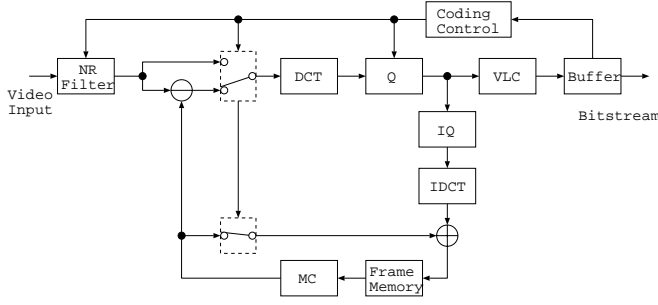


Figure 1: Block diagram of the H.263 encoder

4. FAST ME ALGORITHMS

The ME is used in video codecs in order to remove interframe redundancy. It searches the best matched reference block with the current block using a block matching algorithm. Because of a number of candidate reference blocks to search and the calculation of the matching criterion, it requires huge computation power. Several fast algorithms have been proposed. In the developed encoder, two types of fast algorithms are incorporated. The first one is based on a two-step tree search using a sub-sampling filter[6], which slightly degrades motion tracking capability. The second one is the numerical fast algorithm using a multiply accumulator[7], which does not degrade motion tracking capability at all.

4.1. Two-step search

In the first step, sub-sampled image are used to obtain an approximate motion vector with 2-pel accuracy. The search area is horizontally ± 14 and vertically ± 10 . In order to remove aliasing effects due to sub-sampling, a 2-D separable filter, whose component transfer function is $H(Z) = (1 + Z^{-1})/2$, is applied before sub-sampling.

In the second step, a motion vector is estimated with half-pel accuracy. The search area is ± 1 around the estimated motion vector in the first step. By applying the two-step search, the computation cost reduces to 1/20 compared to full-search.

4.2. Fast ME using a multiply accumulator

In a block matching algorithm, both the the mean absolute error (MAE) and the mean square error (MSE) between a current block and a reference block are feasible as the block matching criterion. When the ME is implemented on processors with a fast multiply accumulator, using the MSE as the matching criterion is equal to or faster than using the MAE.

The MSE is defined as:

$$e_{u,v} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (x_{i,j} - y_{i+u,j+v})^2, \quad (1)$$

where $x_{i,j}$ and $y_{i,j}$ represent (i,j) pixels of the current block and the search window, respectively. N represents a block size, and (u, v) represents the position of a reference block to search.

The fast ME method[7] computes the MSE by following equation, which is equivalent to (1), to reduce the computation cost further.

$$e_{u,v} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x_{i,j}^2 + \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} y_{i+u,j+v}^2 - 2 \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x_{i,j} y_{i+u,j+v}. \quad (2)$$

The first term of (2) is the power of the current block, which is constant for each reference block. Therefore, when comparing $e_{u,v}$ for every candidate (u, v) to find the minimum value, this term is not necessary.

The second term of (2) is the power of the reference block. Reference blocks in the search window have overlapped area for different (u, v) . Therefore, when computing the power of reference blocks, the computation of the overlapped area can be reduced.

The third term of (2) can be considered as 2-D FIR filtering. By applying a fast 2-D FIR filtering algorithm, the computation cost can be reduced. However, combining with the two-step search, the current block is down-sampled (16×16 to 8×8), and the number of candidate reference blocks is also reduced, which implies to the reduction of the 2-D filter order and the number of input samples. As a result, the ratio of pre and post processing cost of the fast 2-D FIR filter increases, and little gain are achieved from applying the fast 2-D FIR filter.

In the developed encoder, the fast 2-D FIR filter is not applied. By the effects of the first and second term computation reduction, the amount of computation reduces to 60% compared to the two-step search ME while maintaining motion tracking capability the same. As a result, the ratio of ME computation to the total computation of the encoder decreases to 40%.

5. LOW COMPLEXITY NR FILTER

The noise reduction is important for low bit-rate compression to achieve better image quality. Two types of noise reduction filters have been utilized in practice[8]. The first one is the temporal IIR filter reducing the noise in the temporal direction. The typical order of the IIR filter is one

because of the required frame memory. The second one is the spatial FIR filter, typically the order of 3×3 , which reduces the noise inside each frame. Implementing these filters costs more than 20% of the total encoder computation amount. Therefore, a low-complexity NR algorithm is necessary to develop.

In the developed encoder, noise reduction is performed after the DCT, and combined with the quantization as shown in fig.2. The noise in the temporal direction decreases by attenuating the difference value between the input and the predicted image, which implies to multiply the difference value and the factor less than one. Multiplying each difference value and the factor implies to multiplying each DCT output and the factor, because of the linearity of the DCT.

The spatial FIR filter requires the convolution in the spatial domain. the convolution in the spatial domain implies to the multiplication in the frequency domain. Therefore, multiplying the filter transfer function and the DCT outputs have a similar effect with the spatial FIR filtering to the input image.

To combine these two filtering effects, the NR filter in fig.2 provides several sets of a multiplication coefficient table. Each table has 8×8 coefficients. The coefficients for high frequencies are smaller than those for low frequencies to provide low pass filtering characteristics. The DC level of the coefficient reflects the filtering characteristics in the temporal direction. The coding controller selects the best one among the provided sets of a table.

Replacing the convolution with the multiplication reduces computation cost. Moreover, combining the NR filter with the quantization process can further reduces the computation. In the quantization, DCT outputs are divided by the quantization step-size, which is implemented by multiplication with the reciprocal of the step-size. By combining the reciprocal of each quantization step-size and the multiplication coefficient table for the NR filter to create new sets of a multiplication coefficient table, the NR filtering and the quantization can be combined. As a result, the NR filtering are incorporated into the quantization procedure with a slight increase of the computation.

6. H.263 EVALUATION BOARD DESIGN

In order to evaluate the video codec, the H.263 evaluation board has been developed. Figure 3 shows a picture of the H.263 experimental system. A notebook PC controls the H.263 evaluation board, which encodes and decodes video sequences. By connecting the PC and a mobile telephone with a PC card, video transmission over the mobile network can be realized.

Figure 4 shows a block diagram of the H.263 evaluation board. In the encoding process, the input video frames are

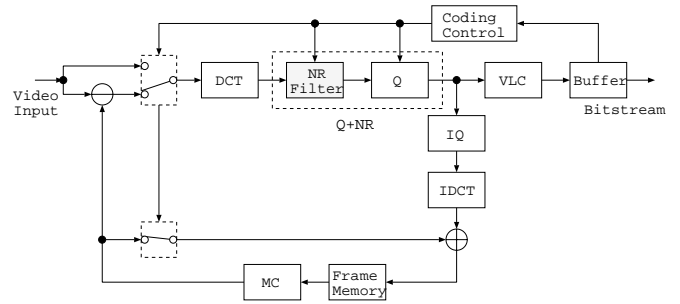


Figure 2: Block diagram of the H.263 encoder using the low complexity NR filter



Figure 3: Picture of the H.263 experimental system

Y/C decoded and stored on the video input FIFO (VIFIFO). The DSP compresses the input video frames and sends the compressed bitstream to the PC through the PCMCIA bus. In the decoding process, the PC sends the compressed bitstream to the DSP through the PCMCIA bus. The DSP decodes it and the decoded lines are stored on the video output FIFO1 (VOFIFO1). After completion of one frame decoding, the frame in the VOFIFO1 is transferred to the VOFIFO2, Y/C encoded, and displayed on a TV.

7. PERFORMANCE OF THE H.263 CODEC

To evaluate the performance of the developed codec, two types of evaluation are carried out. The first one is the worst case evaluation. Required cycles for each modules in the codec are counted up assuming the worst case. The second one is the real case evaluation using the developed evaluation board.

Table 1 shows the worst case performance of the H.263 codec, when using a DSP(μ PD7701x) of 50MIPS. The

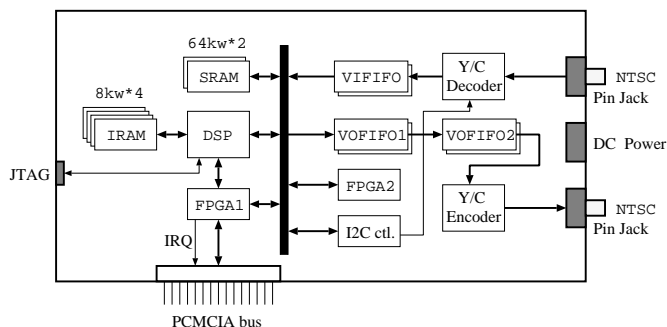


Figure 4: Block diagram of the H.263 evaluation board

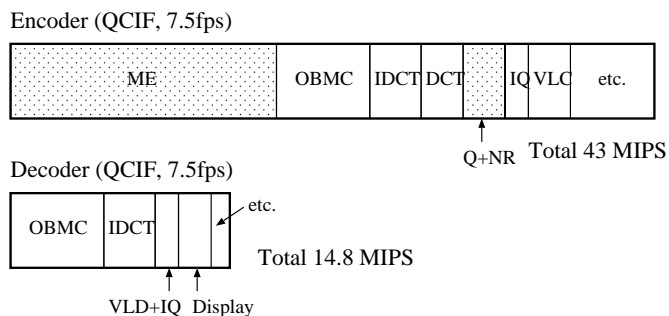


Figure 5: Required cycles for the codec

Table 1: Worst case codec performance (frame/sec)

Options (Annex)	Encoder		Encoder & Decoder	
	sub-QCIF	QCIF	sub-QCIF	QCIF
None	19.0	9.48	14.8	7.39
F	17.2	8.61	12.6	6.32
G	21.0	10.5	15.5	7.73
F,G	19.8	9.92	14.2	7.01

numbers are in frames per second. The Annex F includes the overlapped block motion compensation (OBMC), which increases the computation cost due to its weighted averaging operations. However, by using a fast multiply accumulator in the DSP, the performance decreases only a few frames per second. As for Annex G, the performance are evaluated assuming all the frames PB-frames. Processing a B-picture in the PB-frames requires less computation than a P-picture, because the local decoding procedure is not necessary.

Figure 5 shows the worst case cycles for the encoder and the decoder to process QCIF images at the rate of 7.5 frames per second. The applied option is the Annex F, which requires the most computation. The target bit-rate is around 64kbps, which affects the computation cost of the variable length coding (VLC) and decoding (VLD). The shaded areas are modules employing the described fast algorithms. The motion estimation requires 40% of the total computation of the encoder. The NR filters are incorporated into the quantization procedure with a slight increase of quantization computation.

These results are based on the worse case. In actual encoding and decoding, several skipped blocks appear, which reduce the amount of computation. When using a 50MIPS DSP, developed codec stably encodes and decodes 7.5 QCIF frames or 15 sub-QCIF(128 × 96 pixels) frames per second at the rate of 64kbps, applying the Annex F.

8. CONCLUSION

We have presented an H.263 video codec implementation based on a 100mW general purpose DSP. By using the fast ME algorithm and the low complexity NR filter, the developed codec encodes and decodes 7.5 QCIF frames or 15 sub-QCIF frames per second with a single 50MIPS, 100mW DSP, which is sufficient performance for low bit-rate video compression, typically below 64kbps.

9. REFERENCES

- [1] ITU-T Recommendation H.263, "Video coding for low bit rate communication", (1996)
- [2] ITU-T Draft Recommendation H.223/Annex A~C, "Multiplexing protocol for low bitrate mobile multimedia communication", (1997)
- [3] H. Sano, Y. Ono, M. Ikekawa, and S. Ono, "One chip 16-bit fixed-point dsp for PDC dual-mode speech codec with a high performance echo canceler", Int. Conf. on Telecommunications, pp.186-189, (1996)
- [4] ITU-T Recommendation H.261, "Video codec for audiovisual services at p×64kbit/s", (1993)
- [5] NEC Corp., "μPD77018A, 77019 data sheet", (1996)
- [6] H. Honma, M. Ohta, and T. Nishitani, "A simplified method of motion vector estimation for VLSI implementation", Proc. Int. Conf. on Syst. Eng., pp. 216-219, (1992)
- [7] Y. Naito, T. Miyazaki, and I. Kuroda, "A fast full-search motion estimation method for programmable processors with a multiply-accumulator", ICASSP96, Vol.6, pp. 3222-3225, (1996)
- [8] S. Sabri, and B. Prasada, "Video conferencing systems", Proc. of the IEEE, vol. 73, no. 4, pp.671-688, (1985)