

# The Implementation of Fractional Pixel ME in H.264 Video Encoder

Shikai Zuo, Mingjiang Wang, Zejun Wu, Liyi Xiao<sup>1</sup>

**Abstract.** This paper first introduce the H.264 video codec standard briefly, then analyze the inter-frame motion estimation in H.264. After that we propose an optimized algorithm and the corresponding architecture. A pipeline-architecture was proposed with scaled searching algorithm based on the analysis to the previous work and the consideration of the system requirement between chip area and performance. A pipelined architecture with two 1-dimension interpolation for fractional pixel ME was proposed to meet the needs of SHD (2160P) video encoding, resulting in lower complexity and more throughput. Synthesized with Charmed 0.13 $\mu$ m process, the design consumed 174.3K logic gates, with peak frequency of about 200MHz, and consumed less than 20mW at the power supply of 1.20V.

**Keywords:** H.264. inter-frame. fractional pixel ME.

## 1.1 Introduction

As the new generation of video compression standard, H.264[1] can reduce 64% bitrates on average at the same picture quality as MPEG-2[2]. H.264 only

---

<sup>1</sup> Shikai Zuo  
Harbin Institute of Technology , 518055 Harbin, Chinese

Mingjiang Wang (✉)  
Key Laboratory of IOT Terminal Pivotal Technology  
Harbin Institute of Technology ShenZhen Graduate School,518055 ShenZhen, China  
e-mail: mjwang@hit.edu.com (corresponding author)

Zejun Wu  
Key Laboratory of IOT Terminal Pivotal Technology  
Harbin Institute of Technology ShenZhen Graduate School,518055 ShenZhen, China

Liyi Xiao  
Harbin Institute of Technology , 518055 Harbin, Chinese

Funded by: The Key Laboratory Projects of Shenzhen (CXB201104220018A)

specifies the syntax framework of the bitstream and the decoder architecture. It consists of a hybrid of temporal and spatial prediction, in conjunction with transform coding[3].

In March 2003, the H.26L new video coding standard develops by ISO / IEC and ITU-T formed a joint video Team (JVT), that is, the H.264 standard, and was formally included in the standard MPEG-4 as part 10, formally known as H.264/AVC [4]. And early compression standard H.263, MPEG-2, MPEG-4 compared to, H.264, under the same visual quality can save half the bit rate coding [5]. As a result of a series of advanced coding techniques, H.264 with very good compression rate and network adaptability, can be very effective in a variety of video applications: low-latency mode of the video conferencing, high-definition digital TV broadcasts, digital video storage, streaming media applications.

As shows the coding complexity of each module in H.264 encoder in the analysis which is gave by the JM8.0 reference software based on official JVT[6], the complexity can be seen from the highest, most time-consuming and resource part of the motion estimation / compensation, the module accounts for more than 80% of system resources. Especially in the H.264/AVC inter frame coding process, because of the use of variable block-matching motion estimation (Variable Block Sizes Motion Estimation, VBSME), multiple reference frame motion compensation (Motion Compensation, MC) and Lagrange rate-distortion optimization (Rate Distortion Optimization, RDO) and other advanced coding techniques, making the integer pixel motion estimation (IME) and fractional pixel motion estimation (FME) consisting of inter-frame motion estimation process takes up more than 70% of the entire encoder encoding computing time. FME operations which amounted to more than 45% of the entire coding [7]. To achieve 30 frames/sec real-time encoding requirement, it is necessary to optimize the motion estimation algorithm, reducing the computational complexity of motion estimation, hardware-accelerated encoding to reduce the computation time.

## **1.2 Algorithm Analysis**

### ***1.2.1 Selection of Variable Size Block Motion Estimation Mode***

In this design, in order to reducing the complexity of the algorithm to reduce variable size block choose the amount of computation, can map out the specific hardware implementation, and several other considerations, the only guarantee of image encoding quality, but also be able to chip area and power consumption limited within a certain range, the introduction of aspect types in block mode, some simplification. In paper[8], seven horizontal split mode is divided into two groups, according to the results of the IME, FME only two of the best model within the group. For this design, taking into account the large size of video coding, detailed

block into the role of video quality improvement is small, but a huge amount of computing, where the type of model can be simplified, when the IME's compute the seven sizes block at the same time, and therefore no longer for FME block mode cut.

### ***1.2.2 Selection of Multiple Reference Frame***

H.264 coding standard to obtain good coding performance, largely due to its enhanced motion estimation algorithm, multiple reference frame motion estimation which played an important role. H.264 encoder can be set from the front or the back of the encoded image, select the multiple correlation with the current frame image as a larger reference image inter frame coding. By expanding the scope of the block to find matching images, and then after several more results, after matching can significantly improve the prediction accuracy. H.264 allows maximum use of 16 reference frame. In fact, the vast majority of image sequences, using five reference frames motion estimation coding can get very good results. JM reference software, with full search approach to multi-reference frame motion estimation, that the first reference frame for each motion vector and determine the best matching block, and then compares each frame corresponding to the optimal matching block distortion, select one of the most optimal matching block as the final match, and the corresponding motion vector prediction residual and reference frame index code, and finally write the code word bit stream. Can be seen in the use of the premise of full search algorithm, multiple reference frames technique will allow the introduction of computation as the reference frame increases linearly with increasing.

Taking into account the specific circumstances to implement the hardware for such a large 2160P video, the introduction of three or more reference frames, in computing the amount of image data storage size and other aspects will be difficult to accept, so usually reduce the reference frame from to think in terms of number of multi-reference frame.

### ***1.2.3 Selection of Rate-distortion Optimization Mode***

H.264 encoding process needs to face the block is divided in a variety of ways to select the best mode, motion estimation, how to choose the best motion vector and other issues, mainly based on Lagrangian rate-distortion theory of rate-distortion optimization algorithms in different modes to choose.

These algorithms are essentially the model for each possible first to calculate the number of bits and distortion, and then use a standard to measure the integrated model, the minimum value of the final evaluation of the mode selected. Highly complex model is the most time-consuming way to search, but the recon-

struction of the video quality is the best. It uses the SAD, SATD, SSD and other descriptions of the distortion and the actual rate used for mode selection, and low-complexity models for the different models give a tendentious different initial SAD0, only to describe the distortion using SAD , do not calculate the actual rate. Analysis shows that despite the high complexity of the model code to better results, but requires a lot of computing, and low complexity model does not calculate the SAD value outside the other, the mode selection operation is more simple and easy hardware implementation.

### 1.3 Hardware Architecture and Analysis

#### 1.3.1 The architecture of fractional ME

Fractional ME step closer to the best in the vicinity of an integer to find the best match point. Hardware architecture includes the following main components: the interpolation array, on-chip buffer memory, SAD generator, comparator, controller. Shown in Figure 1.

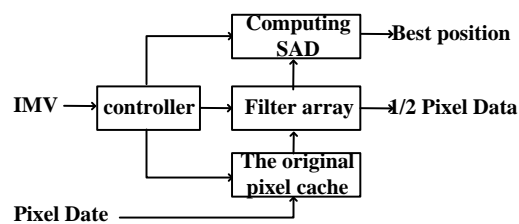


Fig. 1 The architecture of the system of fractional ME

IME module  $16 \times 16, 16 \times 8, 8 \times 16, 8 \times 8, 4 \times 8, 8 \times 4, 4 \times 4$  seven kinds of block mode to find the best block mode and the corresponding motion vector, while the 1/2ME need to use the interpolated data and the current block level data is stored in IME-1/2ME between the cache and store the address information is also available to 1/2ME module. Since memory cell is  $4 \times 4$  block size as a unit, the actual valid data may be any  $4 \times 4$  in a position, therefore, IME module also provides a valid data start position information. If the integer search time to search the image border, no more data outside of its boundaries expand the use of copying data to achieve. Therefore, the matching point of the boundary information is also available to 1/2ME module.

### 1.3.2 The pipeline of fractional ME

1/2 interpolation involves two dimensions, while the interpolation is not only difficult to control, and will consume significant resources. The design of the two-dimensional interpolation, broken down into two one-dimensional interpolation. Taking into account processing a macro block to be sixteen 4×4 blocks processing, it will be two one-dimensional interpolation into the middle of water, not only easy to control, simple structure, but also shorten the macro block processing time.

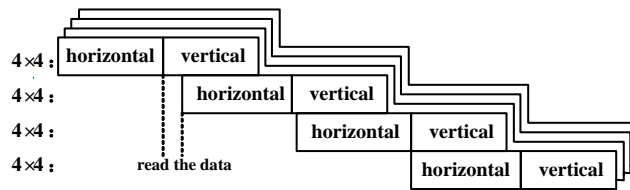


Fig. 2 The architecture of fractional ME

Pipeline shown in Figure 2, the horizontal interpolation module of work to complete horizontal interpolation, vertical interpolation to complete vertical interpolation module, and select the best match point after interpolation. The current block and reference block data are stored in the same memory, and during the vertical interpolation, the first to read the current block of data, so in order to solve the access conflicts, before the completion of a level of 4×4 interpolation, the need to wait a few cycles until the end of the current block of data read, and then to the next level of 4×4 block interpolation.

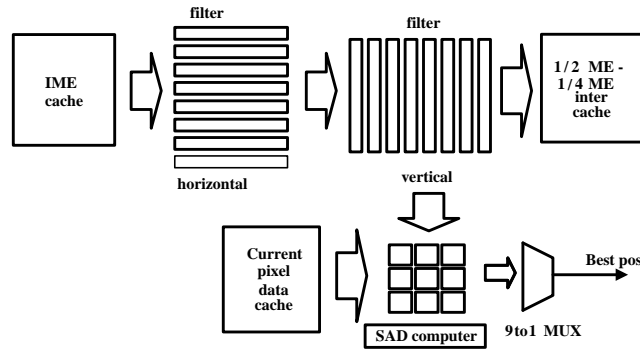


Fig. 3 The data pass 1/2ME

The 1/2ME the data path shown in Figure 3. Data between the cache from IME-1/2ME level, enter the level of interpolation module level data interpolated into 1/2ME internal cache, which is the first water level. Out from 1/2ME internal cache to the data by vertical interpolation between the cache into 1/2ME-1/4ME level; the same time, the current macro block data is read, the nine candidate

points and the calculation of the SAD accumulate, and ultimately compare the optimum matching point. The current macro block data is also stored in the cache between 1/2ME-1/4ME level, this is the second water level. The second stage will give a signal the end of the current block is read, the first level of access to avoid conflict.

Horizontal interpolation control module is designed to process one 4×4 block. 1/2 6-tap interpolation filter, the actual boundary extended to deal with because 10×10 pixels. The design of the storage unit width of 128bit, for a 4×4 sub-block size data, a line 14 to read pixel values stored up to five visits (1,4,4,4,1), at least four times. Similarly, a total of 14 rows of pixels, up to read 5, a total of up to 25 times to access the storage.

The design uses eight filters to take full advantage of the bandwidth, available 8 week period interpolated pixels, interpolated pixel and another eight of the original 4×4 pixels in the form of still kept in the cache. Module set 14×7 pixels size of registers, read the data (block data around) boundary conditions, and according to valid data start position and then cut into the register. Read the data in the register in accordance with certain rules for the left and on the move, making the filter input to be updated. The shaded area of the 7×4 pixels that are 8 filter input. Exports immediately after filtering the data within the cache.

Reference pixel effectively a different starting position Y to begin effectively filtering the data processing time is also different. Filter designed to last for 4 rows of data, is a starting position is 0, read the first line of 4×4 block (Horizontal of 7 pixels) you can start filtering. Other position, to be read the second line (up to at least 4 rows of pixels) before starting the filter. Starting position is 0 and 3, the read data at the same time, the interpolation has come to an end. Starting position of 1 and 2, read the data need to wait for several cycles, until the end of the interpolation, so adding a few wait states.

Vertical interpolation and horizontal interpolation control module control module similar as you do not cut blocks of data, it is relatively easy. Perpendicular to the vertical interpolation of data read, and each column to be read four times, a total of five. In the vertical interpolation module, containing the SAD calculation and accumulation module, while the selection of the best point. The current block data stored in the integer and 1/2 modules of RAM, so the data can be read simultaneously with interpolation. After the current block of data read is stored in the 8 × 8 pixel size of register set, read the interpolation pixel is stored in the 4×10 of the register group. Reference pixel registers according to certain rules on the move, so that the input filter has been updated. Shaded in Figure 4-7 is the input filter. After each insert a reference pixel register is updated.

Interpolation filter is the same set 8, a plug out of the eight pixels and read in by the level of the interpolated pixels 4×4 block form, sent to 1/2ME-1/4ME inter-stage buffer. The current block of data, also sent to inter-stage buffer 1/2ME-1/4ME to prepare 1/4ME use. SAD calculation accumulator receives the current block and the interpolated data points with the completion of nine candidate SAD calculations. Nine candidate SAD in the interpolation points by the end of a comparator to get the minimum, and this corresponds to the location of points is the best match point.

### 1.3.3 The data analysis of 1 / 2 pixel interpolation

1 / 2 pixel interpolation diagram shown in Figure 4. The corresponding half-pixel by pixel for the entire 6-tap filter come with a weight of (1/32, -5/32, 5/8, 5/8, -5/32, 1/32). Capital letters represent integer pixels, lower-case letters 1 / 2 pixel interpolation.

				A	aa	B				
				C	bb	D				
E		F		G	b	H		I		J
cc		dd		h	j	m		ee		ff
K		L		M	s	N		P		Q
				R	gg	S				
				T	hh	U				

Fig. 4 Parallel structure of intra 4\*4 mode

From the near six vertical pixel interpolation, pixel h is gave:

$$h = \text{round}(A - 5C + 20G + 20M - 5R + T) / 32)$$

Transform to get the following formula:

$$h = \text{round}(A - C - 4C + 16G + 4G + 16M + 4M - 4R - R + T) \gg 5)$$

Each input factor is an integer power of 2, the shift can be done by multiplication of the input and series. For negative items, the use of take and add 1 to achieve. Necessary because some addends left, so it's right side can be added to add the item. Similarly, the rounding operation, that is combined with 16 parts, also added to the left of the item right. Wallace tree using the form available filtering results.

### 1.3.4 The SAD architecture

SAD calculation is 1/2 accuracy of the data, produce 4×4 week period of the data, the corresponding data point is the location of 2×2, that is four, so, SAD module is designed to generate four pixels each part of SAD value, to accumulate.

Absolute value for the part, directly after the highest demand is poor, that is the sign bit, as the select signal to strobe the difference of the original code or the anti-code, add 1 part for the transfer to the Accumulator. Accumulator using Wallace tree adder plus a register implementation as show in Figure 5.

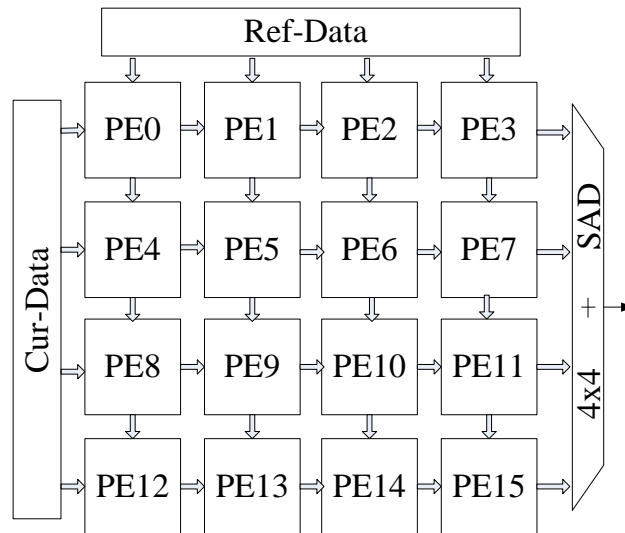


Fig. 5 The process of Sub-Mb in intra 4\*4

Comparing module is a multi-choice values from the nine SAD, SAD selected the smallest one, the output of the corresponding point location information. Design carved two compared with three first-class 3-to-1 comparators selected three candidate points, the second level and then choose from three points in the smallest one. For these nine points, taking into account the direction of motion of objects in real video probability, set some priorities. Highest priority is the horizontal direction of movement, followed by vertical movement, and finally the 45-degree corners along the direction of movement.

### 1.3.5 Analysis of dynamical simulation data

Figure 6 shows the simulation waveform of control module. When the halfMe\_bgn\_in is activated, the 1/2ME module starts working. The value of state[2:0] means the 1/2ME module working state. When the value of state[2:0] is 0, the state of 1/2ME module is idle state and when the value of state[2:0] is 1~5, the state of 1/2ME module is interpolation states including horizontal interpolation and vertical interpolation. The halfMeHIntpl\_begin is controlling signal of the horizontal interpolation, when it is high level the horizontal interpolation module start to work. When the halfMeHIntpl\_end is low level, the horizontal interpola-



tion is ending. The halfMeOthr\_begin uses as halfMeHIntpl\_begin, when the vertical interpolation is high level, the vertical interpolation start to work and when the halfMeOthr\_end is low level the vertical interpolation ending. When the halfMeCurRd\_end is high level, the current block data is loaded in memery. The simulation waveform illustrates that when the horizontal interpolation is finished, the vertical interpolation starts to work. When the data of current block is completely read, the next horizontal interpolation begins to work. The simulation waveform shows the control module achieves the design object.

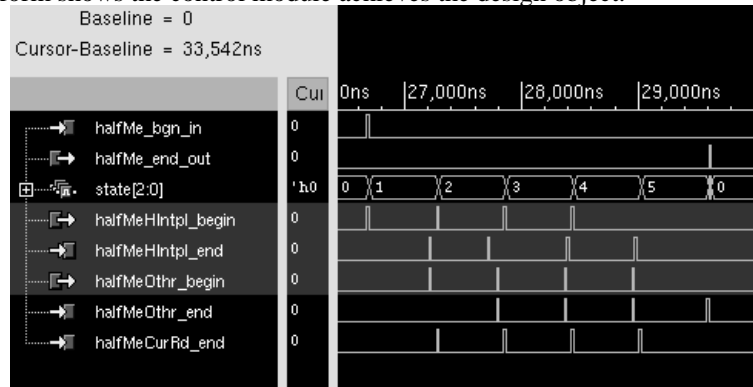


Fig. 6 The simulation analysis of control module

The simulation waveform of whole 1/2 ME is shown in figure 7. The category of video frame have two for I frame and P frame. If the value equals to 1 means the current frame is I frame when 0 is P frame. The halfmv[15:0] is the motion vector signal. halfME\_bgn\_in is the starting signal and halfMe\_end\_out is the ending signal. The quarterme\_sramwaddr and quarterme\_sramwdata represent the quarter ME data access operation. When the signal quarterme\_sramwe\_out is active , data writing process is finished. The simulation waveform shows that the ME module is not at working state when the frame is I frame. The simulation data prove that the 1/2ME module can achieve fraction pixel motion estimation.

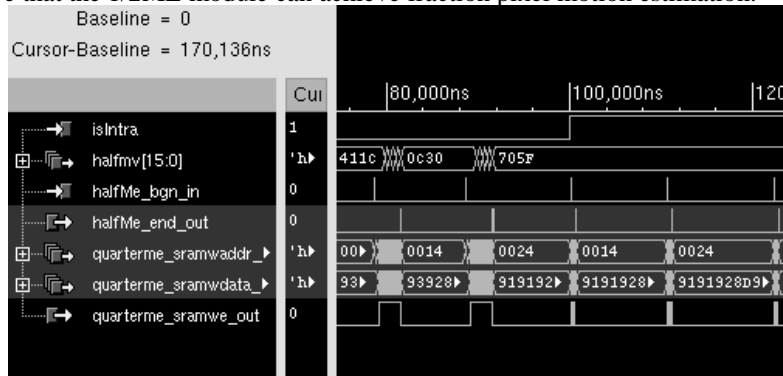


Fig. 7 The simulation analysis of 1/2 ME module

## 1.4 Conclusion

In this paper, we designed a 1/2ME hardware acceleration module to implement the motion estimation and pixel interpolation of video encoder. The simulation results show that the throughput of this module is high. Also the controlling structure of this module is simple. The time that this circuit takes to process a whole macro block is less than 150 clock cycles. The module is synthesized with Charter 0.13um process library. The synthesis results shows that the area of this module is 174.3K gates, power dissipation is less than 20 mW. Post simulation results show that this circuit can work at the maximum frequency of 200 MHz.

## 1.5 References

1. ITU-T Series H -Advanced video coding for generic audiovisual services 2009.
2. ITU-T Recommendation H.262 . ISO/IEC 13818-2 (MPEG-2): Generic coding of moving pictures and associated audio information . Part 2: Video ITU-T and ISO/IEC JTC1, November 1994.
3. Sehafer Ralf, Wiegand Thomas, Heiko Schwarz. The emerging H.264/AVC Standard. EBU Technical Review. 2003:1~12.
4. Joint Video Team (JVT) of ITU-T VCEG and ISO/IEC MPEG. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC). 7th Meeting, Pattaya, Thailand, 2003: I
5. B. Meng, O. C. Au. C.-W. Wong, H.-K. Lam, "Efficient Intra-Prediction Mode Selection for 4x4 Blocks in H.264" Proceedings of IEEE ICME, pp: 521-524, 2003.
6. J. Zhang, et al. Performance and Complexity Joint Optimization for H.264 Video Coding. Proc. of ISCAS'03. 2003,2: 888~891
7. T.-C. Chen, et al. Analysis and Design of Macroblock Pipelining for H.264/AVC VLSI Architecture. Proc. of ISCAS'04. 2004, 2: 273~276.
8. Y. Lin, et al. A Hardware-efficient H.264/AVC Motion Estimation Design for High-definition Video. IEEE TCSI. 2008,55(6): 1526~1535