Low-Hardware-Cost Motion Estimation with Large Search Range for VLSI Multimedia Processors

Seongsoo LEE†a), Min-Cheol HONG†, and Jae-Kyung WEE†, Nonmembers

SUMMARY In this paper, we propose new low-hardware-cost motion estimation with a large search range for VLSI multimedia processors. It reduces the hardware amount required for pixel comparison by reducing both the spatial-resolution and bit-resolution of pixel values. Low-hardware-cost block-matching criterion is also employed. To avoid performance degradation from low resolution, we introduce an "outlier" pixel with large overload quantization error in the search window, and a search position is excluded from the motion estimation if its corresponding search window block contains one or more outliers. The proposed motion estimation is easy to implement in VLSI multimedia processors, and it significantly reduces the hardware amount when the search range is larger than ±64 × ±64. In MPEG2 MP@ML video compression with ±128 × ±128 search range, it reduces the hardware cost to 1/144 that of the full search algorithm, while its degradation of peak signal-to-noise ratio is 0.32 dB.

key words: motion estimation, block matching, low hardware cost, outlier exclusion, VLSI

1. Introduction

Over the past decade, multimedia and digital video applications have had a great impact all over the world. Due to the limited transmission capacity, video compression is one of the key technologies, but it suffers from tremendous computation when picture size and frame rate increase. Many applications such as digital multimedia broadcasting and mobile videophone require real-time video compression with huge computation, which can be achieved by VLSI processors. In most video compression algorithms such as MPEG-2 [1], the most computation-intensive process is motion estimation [2], which occupies 50–90% of total computation [3]. For each macroblock in the current frame, it searches for a best matched block in the previous frame. Motion vector is defined as a displacement between the current block and the best matched block. By sending the block difference and the motion vectors instead of whole current block, it eliminates the temporal redundancy of the image sequences. Due to tremendous computation, motion estimation is usually realized in dedicated hardwired VLSI implementation.

Among various motion estimation algorithms, the full search [2] provides the best performance, but it suffers from huge computation if its search range is large. Reasonable search ranges that can maintain good performance for the MPEG-2 P-prediction without a B-picture or with two B-pictures are ±64 × ±64 and ±128 × ±128, respectively [4]. With these search ranges, the required computation of the full search is $5.1 \times 10^{11}$ and $2.0 \times 10^{12}$ operations per second, respectively, which is too high for VLSI implementation. Many fast algorithms [3], [5]–[10] have been proposed to reduce this computation. These algorithms can be categorized into three groups according to how they reduce computation: (1) reduction in the number of search positions based on the assumption of unimodal error surfaces [5]–[7] (2) reduction in spatial-resolution by subsampling [8] or averaging [9] pixel values, and (3) reduction in bit-resolution by truncating or quantizing pixel value [3], [10], [11]. However, even these fast algorithms are often impractical for VLSI implementation if its search range is large. In some fast algorithms, severe degradation of image quality is far from satisfactory. Other fast algorithms yet suffer from enormous computation, since they failed to reduce it enough. Consequently, new low-hardware-cost motion estimation is indispensable for VLSI implementation of real-time multimedia processors.

In this paper, we propose novel hardware-oriented motion estimation for VLSI multimedia processors called outlier pixel exclusion motion estimation (OPEME). To reduce hardware cost, it reduces both the spatial-resolution and bit-resolution of pixel values, and employs a novel low-hardware-cost block-matching criterion. We introduce an "outlier" pixel as a new concept to reduce performance degradation due to the quantization error. An outlier for a current block is defined as a pixel in its search window such that the pixel value is outside the enlarged range of pixel values in the given current block. A search position is excluded from the motion estimation if its corresponding search window block contains one or more outliers, because outliers usually result in large overload quantization errors. To improve performance, we also introduce minimum mean-absolute-error (MMAE) quantization in bit-resolution reduction.

2. Reduction in Temporal-Resolution

As illustrated in the introduction, most fast motion estimation algorithms reduce one of the followings: (1) number of search positions, (2) spatial-resolution of pixel values, and (3) bit-resolution of pixel values. In this paper, both spatial-resolution and bit-resolution are reduced. We didn’t con-
consider reducing the number of search positions because of its complex control and irregular dataflow. Furthermore, it suffers from severe performance degradation due to local minima when the search range is large. To reduce the spatial-resolution, we exploited $2 \times 2$ pixel averaging [8] based on the performance and hardware cost.

There are two methods of bit-resolution reduction, i.e., truncating least significant bits [10], [11] and adaptive quantization [3]. In order to optimize bit-resolution, the hardware costs and peak signal-to-noise ratios (PSNR’s) of these methods are compared in Fig. 1. As shown in Fig. 1, 2-bit adaptive quantization is optimal in the hardware cost and PSNR.

For all simulations throughout this paper, the first 45 frames are selected from each of the following CCIR601 sequences (720 $\times$ 480 pixels, 30 frames/sec): “PRL CAR,” “football,” and “flower garden”. As described in the introduction, MPEG-2 P-prediction without a B-picture is used when the search range is $\pm 64 \times \pm 64$, and MPEG-2 P-prediction with two B-pictures is used when the search range is $\pm 128 \times \pm 128$. The full search is used unless the method of motion estimation is explicitly specified. We modified the MPEG-2 simulator v.1.1 by MPEG software simulation group (MSSG), where MPEG-2 Test Model 5 (TM5) was exploited in the encoder.

Development and optimization of motion estimation should consider not only the performance but also the hardware cost of VLSI implementation. The most practical measure of hardware cost is the gate count of VLSI implementation. However, it is usually difficult to estimate total gate count at the time of algorithm development, and it strongly depends on the architectures, applications, throughput, and operation speed. Most motion estimation algorithms exploit multiple processing elements, and they dominate total gate count. In this paper, we define the hardware cost as (total NAND-equivalent gate count of all processing elements) $\times$ (required cycle time per current block). It is easy to calculate and flexible to cover various motion estimation algorithms. It is a good measure of algorithm complexity for hardware implementation [3], because it is an estimated gate count of processing elements, normalized with throughput and operation speed.

To improve performance, we propose a 2-bit minimum mean-absolute-error (MMAE) quantization. For each current block, three thresholds, $m-\delta$, $m$, and $m+\delta$, are selected so as to minimize the average quantization error of the current block. Figure 2 shows the pseudo-codes of the binary search algorithm to select these thresholds. The maximum number of mean-absolute-error (MAE) calculation is 11, and the maximum computation of the binary search is $8.6 \times 10^7$ operations per second, which is less than $1/6000$ of the full search. As shown in Table 1, the proposed 2-bit MMAE quantization improves the performance of original 2-bit adaptive quantization.

### 3. Reduction in Temporal-Resolution

As a block-matching criterion, the sum of absolute differ-
Fig. 3 Computation of 2-bit PSAD. (a) Metric table. (b) Processing element.

ence (SAD) [2] is widely used. To reduce the hardware cost of the SAD, various block-matching criteria such as the multi-level pixel difference classification (MPDC) [10], the reduced-bit mean absolute difference (RBMAD) [11], and the different pixel count (DPC) [3] have been proposed. However, these criteria suffer from severe performance degradation for $2 \times 2$ pixel averaging. In this paper, we propose the pseudo-SAD (PSAD) defined in Eq. (1), which is an approximation of the SAD with low bit-resolution.

$$\text{PSAD}(u, v) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |\tilde{p}_k(i, j) - \tilde{p}_{k-1}(i+u, j+v)|$$

(1)

where $\tilde{p}_k(x, y)$ is the reconstruction value of a quantized pixel in the $k$th frame, and the current block size is $N \times N$. The reconstruction value $\tilde{p}_k(x, y)$ is determined so that the average quantization error of the pixel value is minimized, and it is given as Fig. 3 (a) in 2-bit resolution.

Figure 3 shows the metric table and the implementation of 16-pixel processing element (PE) in 2-bit PSAD computation. As compared in Fig. 4, the proposed 16-pixel PSAD PE has the following advantages in VLSI implementation.

1. It requires several logic gates for pixel comparison, while conventional PE’s of 8-bit resolution require an 8-bit subtractor and 8 exclusive OR gates.

2. It can be implemented by ripple carry adders instead of area-consuming fast adders, because its critical path is shorter than that of conventional PE’s of 8-bit resolution.

3. By using adder tree, it requires only one accumulator for 16 pixel comparison, while conventional PE’s of 8-bit resolution require 16 accumulators because 16 PE’s are required for the same number of pixel comparisons.

Figure 5 shows the PSNR’s and hardware costs of four block-matching criteria for $2 \times 2$ pixel averaging. From Fig. 5, it can be seen that the proposed 2-bit PSAD successfully reduces the hardware cost compared to the 8-bit SAD and improves the performance of the 2-bit MPDC and the 2-bit DPC.
4. Analysis on Quantization Error and Outlier Pixel Exclusion

In MMAE quantization, it is desirable to determine thresholds so as to minimize the average quantization error for both the current block pixels and the search window pixels. However, it requires huge hardware cost to minimize the quantization error of search window pixels if the search window is large. Therefore, as shown in Fig. 2 (b), MMAE quantization thresholds are determined to minimize the quantization error of only the current block pixels.

There are two kinds of quantization error, namely overload error and granular error [12], as shown in Fig. 6 (a) and (b) for 2-bit resolution. Overload error occurs in the two outermost quantization intervals and granular error occurs in all the other quantization intervals. In general, the range of pixels in the search window is much wider than that in the current block, because the search window is much larger than the current block. Since quantization thresholds are optimized only for the narrow range of pixels in the current block, the overload error in the search window can be much larger than that in the current block, as shown in Fig. 6 (b). This may result in substantial degradation of performance, especially when the pixel value distribution of the search window is quite different from that of the current block.

We introduce the “outlier” concept to solve this problem. For a current block, an outlier is defined as a pixel in its search window such that the pixel value exceeds a certain range \( (\lambda_{\text{min}}, \lambda_{\text{max}}) \), as shown in Fig. 6 (c). Here, \( (\lambda_{\text{max}} - m) = K (p_{\text{max}} - m) \) and \( (m - \lambda_{\text{min}}) = K (m - p_{\text{min}}) \), where \( K = \frac{(\lambda_{\text{max}} - m)}{(p_{\text{max}} - m)} \) is a constant, \( m \), \( p_{\text{max}} \) and \( p_{\text{min}} \) are the average, maximum and minimum values of the current block, and \( s_{\text{max}} \) and \( s_{\text{min}} \) are the maximum and minimum values of the search window, respectively.

Outliers have large overload error, and they usually result in performance degradation. Therefore, in the proposed motion estimation, a search position is excluded from the motion vector search if its corresponding search window block contains one or more outliers. We can determine the constant \( K \) from simulation results. If \( K \) is too large, severe performance degradation occurs due to the large overload error. On the contrary, if \( K \) is too small, too many search positions are excluded from the motion estimation, which also results in severe performance degradation. As shown in Fig. 7, we found that the optimal value for \( K \) is \( \frac{3}{2} \) for 2-bit resolution. Note that the PSNR is improved by 0.84 – 1.40 dB after excluding outlier pixels.

5. Outlier Pixel Exclusion Motion Estimation

In this paper, we propose novel motion estimation called outlier pixel exclusion motion estimation (OPEME) that uses low bit-resolution quantization, pixel averaging, and the “outlier” concept. As shown in Fig. 8, the proposed motion estimation employs two search steps, namely a low-resolution search and a full-resolution search. In the low-resolution search, a set of candidate motion vectors (CMV set) is determined, which has a lower PSAD and contains no outlier at all. In the full-resolution search, the motion vector is found by calculating the SAD on the positions of the CMV set. The outline of the proposed motion estimation is described as follows.

1. Preprocessing of current and previous frames
   - For the current and previous frame, \( 2 \times 2 \) pixel averaging is performed.

2. Low-resolution search
   - For each current block, the MMAE quantization thresholds \( m - \delta, m, m + \delta \) are determined using a binary search, and outlier thresholds \( \lambda_{\text{min}}, \lambda_{\text{max}} \) are also determined.
   - With selected thresholds, 2-bit MMAE quantization is performed for the \( 2 \times 2 \) averaged images of both the current block and the search window. If an averaged pixel in the search window does not lie in the range \( (\lambda_{\text{min}}, \lambda_{\text{max}}) \), it is classified as an outlier.
The PSAD for every search position is calculated. If the search window block for a search position contains one or more outliers, such a search position is excluded from the motion estimation.

- $N$ search positions that have a lower PSAD and contain no outliers are determined to be the CMV set.

3. Full-resolution search

- The SAD for each of $4N$ search positions corresponding to $N$ CMV’s is calculated using the original full-resolution image (8-bit resolution, without $2 \times 2$ pixel averaging). Note that a search position in the $2 \times 2$ averaged image corresponds to 4 search positions in the original full-resolution image.

- The search position with the minimum SAD is determined as the motion vector.

In determining the number of CMV’s ($N$), we used the simulation results shown in Fig.9. Considering the PSNR and hardware cost, $N = 8$ and $N = 16$ are selected for the search range of $±64 \times ±64$ and $±128 \times ±128$, respectively.

The PSNR’s and hardware costs are compared in Fig.10 for the full search (FS) [2], the 4:1 alternate subsampling (4:1AS) [9], the one-dimensional full search (1DFS) [7], the 3-step hierarchical block-matching algorithm (3HBMA) [5], and the proposed motion estimation. The hardware architecture of the proposed motion estimation is based on [3]. How to calculate the hardware cost is illustrated in [3] in detail. The hardware cost of the proposed motion estimation includes all miscellaneous steps such as MMAE quantization, pixel averaging, and the exclusion of search positions.

The hardware cost of the proposed motion estimation is reduced to 1/122 and 1/144 that of FS, while its PSNR degradation is 0.37 dB and 0.32 dB for the search range of $±64 \times ±64$ and $±128 \times ±128$, respectively. The PSNR of the proposed motion estimation is similar to that of 4:1AS, while its hardware cost is reduced to 1/30.6–1/36.1 of 4:1AS. Compared to 3HBMA, the proposed motion estimation shows similar hardware cost while its PSNR is superior by 0.89 – 1.21 dB.

Along with the hardware cost, the memory bandwidth should be also considered. To reduce the huge memory bandwidth, large internal buffer is often exploited to store the whole search window data [3], [4]. In this case, all motion estimation algorithms compared in this paper require same memory bandwidth and same internal buffer size. When the search range is $±64 \times ±64$ and $±128 \times ±128$, the required memory bandwidth is $9.3 \times 10^7$ bit/s and $1.7 \times 10^8$ bit/s, respectively. Internal buffer size can be reduced by exploiting data reuse techniques [4]. When the search range is $±64 \times ±64$ and $±128 \times ±128$, the internal buffer size is 38.0 kbits and 71.8 kbits, respectively. These values are quite acceptable for state-of-the-arts VLSI motion estimators.

6. Conclusion

In this paper, we proposed novel low-hardware-cost motion estimation for VLSI multimedia processors with large search range. Pixel values are averaged with $2 \times 2$ pixels, and the result is quantized into 2-bit resolution using minimum mean-absolute-error quantization. The proposed motion estimation performs a two-step search, employing new hardware-oriented block-matching criterion. A search position is excluded from the motion estimation to reduce performance degradation if its corresponding search window block contains one or more outliers. The hardware cost of the proposed motion estimation is 1/122 and 1/144 that of the full search algorithm with acceptable performance degradation, when the search range is $±64 \times ±64$ and $±128 \times ±128$, respectively.
References


Seongsoo Lee received B.S, M.S, and Ph.D degrees in E.E. from Seoul National University, Korea in 1991, 1993, and 1998, respectively. In 1998–2000, he was a research associate in Institute of Industrial Science, University of Tokyo, Japan. In 1998–2002, he was a research professor in Department of Information Electronics Engineering, Ewha Womans University, Korea. He joined School of Electronics Engineering at Soongsil University, Korea in 2002, where he is currently an assistant professor. His research interests include low-power VLSI systems, multimedia signal processing, and system-in-package design.

Min-Cheol Hong received B.S and M.S degrees in E.E. from Yonsei University, Korea in 1988 and 1990, and Ph.D degree from Northwestern University, in 1997, respectively. He worked for LG Information and Communication Research Center and LG Electronics Corporate Research Center in 1990–1991 and 1998–2000, respectively. He worked as a postdoctoral research fellow at Northwestern University in 1997–1998. He joined School of Electronic Engineering at Soongsil University, Korea in 2000, where he is currently an assistant professor. His research areas include image/video restoration, nonlinear video processing and filtering, video coding, motion analysis and modeling, generalized Gaussian noise modeling, and blind image deconvolution.

Jae-Kyung Wee received the B.S degree in physics from Yonsei University, Korea in 1988. He received M.S degree in physics and Ph.D degree in E.E. from Seoul National University, Korea in 1990 and 1998, respectively. In 1990–2001, he worked for Hynix Semiconductor. In 2001–2004, he was an assistant professor in College of Information and Electronics Engineering, Hallym University, Korea. He joined School of Electronics Engineering at Soongsil University, Korea in 2004, where he is currently an assistant professor. His research areas include system-in-package design, high-speed circuits, and CAD tools.