A BINARY BLOCK MATCHING ARCHITECTURE WITH REDUCED POWER CONSUMPTION AND SILICON AREA REQUIREMENT

Marcelo M. Mizuki Ujjaval Y. Desai Ichiro Masaki Anantha Chandrakasan

Massachusetts Institute of Technology *
Cambridge, MA USA

ABSTRACT

Motion estimation is essential for reducing bit rate by exploiting the temporal redundancy existent in image sequences. MPEG, one of the current standards for video coding, specifies the use of block matching (BM) for motion estimation. Conventional block matching is based on the Mean-Absolute Difference (MAD) distortion metric, which requires a large number of 8-bit arithmetic computations and thereby limits wider usage. This paper examines the possibility of using a binary distortion metric based on contour data to reduce the silicon area and power consumption of the block matching chip by a factor of 5 or more. Our simulation results indicate that the performance of the proposed binary system for a restricted class of sequences, such as "head-and-shoulder" images, is very close to that of conventional gray level methods. Detailed design of the binary block matching (BBM) chip is currently underway. Potential applications include low power, portable video devices and machine vision applications such as stereo matching and template matching.

1. INTRODUCTION

Efficient coding of image sequences relies on motion estimation to reduce interframe redundancy. Motion estimation involves the analysis of successive video frames to obtain the displacement of pixels or blocks of pixels. Spatio-temporal differential algorithms, which determine the motion on a pixel-by-pixel basis, have the disadvantage that several computations need to be performed for each pixel. Algorithms involving translational motion for entire objects are hindered by difficulties encountered in segmentation. The most widely used method for motion compensation is block matching [1].

Block matching (BM) involves partioning the current frame into several blocks. The size of these blocks is specified to be 16 x 16 by the Motion Picture Experts Group (MPEG) standard and the ITU-T (formerly CCITT) H.261 standard. For every block in the current frame, there is a corresponding search window in the previous (reference) frame. The block in the current frame is matched to a block in the search window according to some distortion criterion. The most commonly used criterion is the Mean-Absolute Difference (MAD) between blocks. For a candidate block at position (m,n) in the search window, the distortion is

$$D(m,n) = \sum_{i,j} |x(i,j) - y(i+m, j+n)| \quad (1)$$

$$v = [m,n] | D_{min} \quad (2)$$

The search window block with the minimum distortion is matched to the block in the current frame. The motion displacement between the current block and the selected block determines the motion vector for the current block.

A potential drawback of block matching is its high computational requirement. If the maximum block displacement is allowed to be in the range [-p, +p] in both horizontal and vertical directions, there will be $(2p+1)^2$ candidate blocks in the search window. Over 50 million operations on 6-16 bit data are required for a 256 x 256 image with 16 x 16 block size and $p = 8$.

2. BINARY BLOCK MATCHING

There are three approaches to deal with the computationally intensive nature of BM. The first approach is to make use of specialized VLSI architectures to implement exhaustive search block matching [2, 3]. The second alternative is to reduce the number of pixel-level distortion operations as described in [4, 5, 6, 7]. The third approach is to propose a different pixel-level distortion operation and a VLSI architecture suited to the new operation.

The objective of this paper is to follow the third approach by combining contour based motion estimation techniques [8] and the block matching algorithm. The block diagram for the proposed VLSI chip is shown in figure 1. Edge detection is performed on both the current image and the reference image. A modified form of block matching is performed on the binary edge maps. Instead of using the MAD criterion, the correlation between blocks is used to perform matching [9]. The distortion criterion becomes:

$$D(m,n) = \sum_{i,j} x(i,j) \oplus y(i+m, j+n) \quad (3)$$

where x(i,j) and y(i,j) are binary edge maps and $\oplus$ is the XOR operation. In special blocks where the number of contour pixels is less than some threshold, interpolation of motion vectors of adjacent blocks is done. In practical architectures, a 2-D array of processing elements (PE's) and shift registers is used to minimize I/O bandwidth. Reduction in computation for BBM arises from the fact that the PE's operate on binary data instead of 8-bit gray scale data.

*This research was sponsored in part by NSF.

0-7803-3192-3/96 $5.00©1996 IEEE

3. EXPERIMENTAL RESULTS

In order to evaluate the performance of BBM, tests were performed on the "susie" image sequence, shown in figure 2 (a) and (b). BBM and conventional BM were used to predict the 4th frame of this sequence from the 1st frame, the results of which are shown in figure 2 (c) and (d), respectively. To further illustrate the differences between the motion compensated frames, the error images in figure 2 (e) and (f) were created by subtracting the motion compensated frames from the original frame. Although there are some differences in the results of the two methods, the subjective quality appears to be similar. In addition to subjective comparison, a quantitative comparison was also made. The plot in figure 3 (a) is a histogram of the intensity error between the motion compensated frames and the original frame. The two curves are close to each other, reflecting the similarity in performance of the two methods. The histogram in figure 3 (b) shows the match between the motion vectors obtained from the two methods. As can be seen from this plot, 90% of the motion vectors are off by at most 2 pixels. These preliminary results suggest that for "head-and-shoulder" sequences, the performance of BBM is visually as well as quantitatively similar to that of conventional BM.

4. HARDWARE COMPARISON

As discussed in Section 2, conventional full search block matching makes use of a 2-D array architecture to meet the throughput constraints. In the arrangement shown in Figure 4, each PE is responsible for computing the distortion of one candidate block. Tri-directional registers that can shift search window pixels up, down, and right reduce the required I/O bandwidth (reduce memory accesses) by allowing adjacent PE's to transfer data among themselves. The distortion computation module for the conventional PE computes the MAD criterion and includes an 8-bit subtractor, the absolute value circuit, and a 16-bit accumulator. For the binary PE, the distortion module consists of an XOR gate and an 8-bit accumulator. Preliminary layouts indicate that the physical area of a binary PE is approximately 4.5 times smaller than the area of a conventional PE. In practice, a reduction in size by a factor of 8 can be achieved by realizing that the binary PE's can be time-multiplexed (the binary PE's are about twice as fast as the conventional PE's). Switch level simulations indicate that the effective capacitance of the binary PE is lower than that of the conventional PE by a factor of approximately 4, hence the power dissipation will decrease even if the binary PE's were to run at twice the clock speed of the conventional PE's. Even after considering the edge detection circuitry overhead, the total silicon area can be reduced by a factor of five. Figure 5 gives a breakdown of the area distribution for the entire chip.

5. CONCLUSIONS

An inexpensive, low power, reduced area motion estimation chip is crucial for many image processing applications. Due to its computational requirements, the conventional block matching algorithm is unsuitable for the development of such a chip. In this paper, we present a binary block matching architecture which significantly reduces computation. Instead of performing block matching on pixel intensity values, the proposed algorithm uses binary edge data to obtain motion vectors, allowing us to scale down the processing hardware by a factor of 5 or more. Simulation results indicate that this improved implementation does not have a significant impact on the quality of motion compensation. Detailed design of the BBM chip is in progress.

REFERENCES

[1] T. Komarek and P. Pirsch, *Array architectures for block matching algorithms,* IEEE Trans. on Circuits and Systems, Vol. 36, No. 10, 1989, pp. 1301-1308.

[2] M. Sun, *Algorithms and VLSI architectures for motion estimation,* In VLSI implementations for image communications, Editor: P.Pirsch, 1993, pp. 251-282.

[3] P. Pirsch, N. Demassieux, and W. Gehrke, *VLSI arhitectures for video compression - a survey,* Proc. of the IEEE, Vol. 83, No. 2, Feb. 1995, pp. 220-246.

[4] M. Ghanbari, *The cross-search algorithm for motion estimation,* IEEE Trans. Comm., Vol. 38,July 1990, pp. 950-953.

[5] A. Puri, H. Hang, and D. Schilling, *An efficient block matching algorithm for motion compensated coding,* Proc. IEEE ICASSP, 1987, pp. 25.4.1-25.4.4.

[6] M. Bierling, *Displacement estimation by hierarchical block matching,* Proc. SPIE Visual Comm. and Image Proc. II, Vol. 845, 1987, pp. 942-951.

[7] J. Jain and A. Jain, *Displacement measurement and its application in interframe image coding,* IEEE Trans. Comm., vol. COM-29, Dec. 1981, pp. 1799-1808.

[8] S. Carlsson and C. Reillo, *Contour based representations of the displacement field for motion compensated image coding,* ICASSP, Tokyo, 1986, pp. 161-164.

[9] I. Masaki, *Industrial vision systems based on application-specific IC chips,* IEICE Trans., Vol.E 74, No. 6, June 1991, pp. 1728-1734.

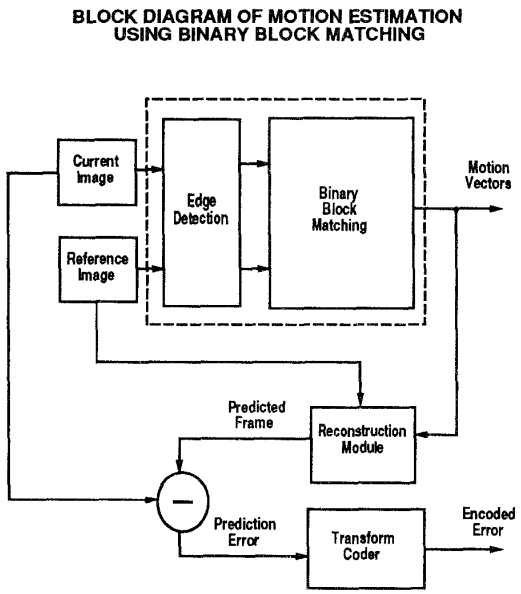


Figure 1. A conceptual block diagram of motion compensation using binary block matching (BBM).

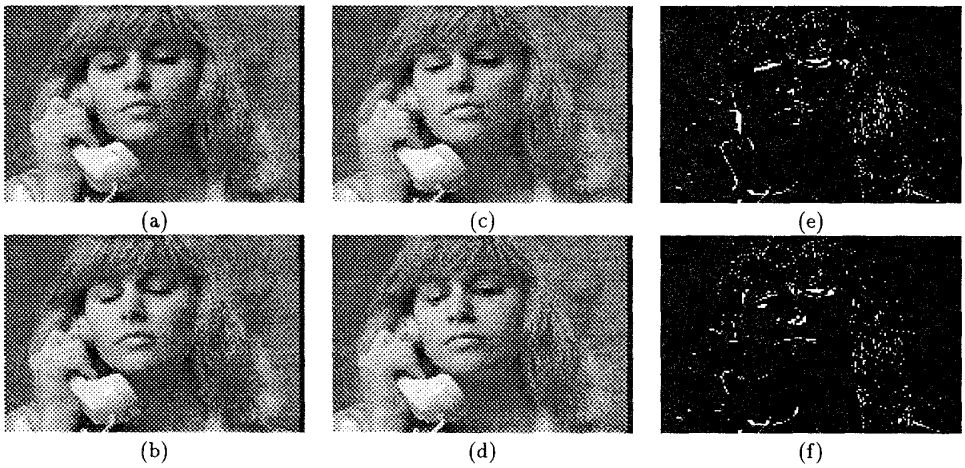
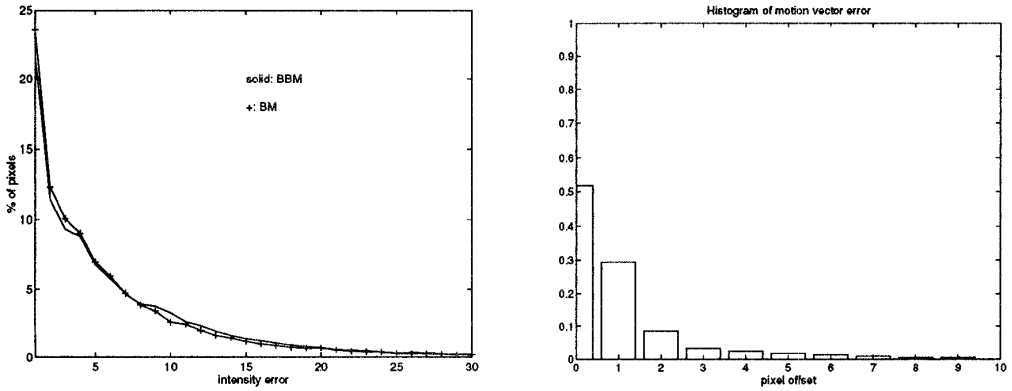


Figure 2. (a) Original frame 1; (b) Original frame 4; (c) Motion compensated frame 4 from binary block matching (BBM); (d) Motion compensated frame 4 from conventional block matching (BM); (e) Error image from BBM; (f) Error image from BM (Error images histogram-modified for display purposes).



(a) (b)

Figure 3. (a) Histogram of intensity error for binary block matching (BBM) and conventional block matching (BM); (b) Histogram of motion vector error between BBM and BM.

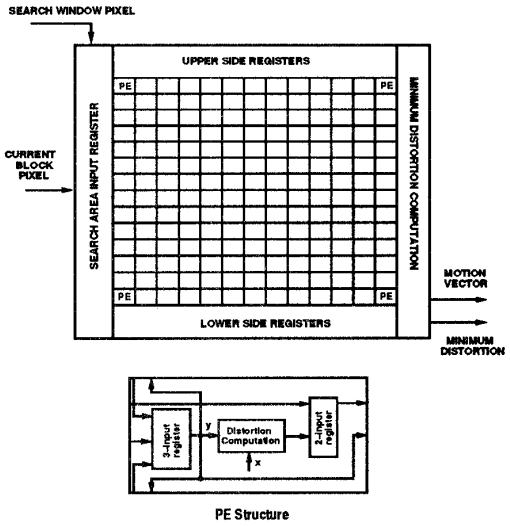


Figure 4. A parallel 2-D array architecture for block matching.

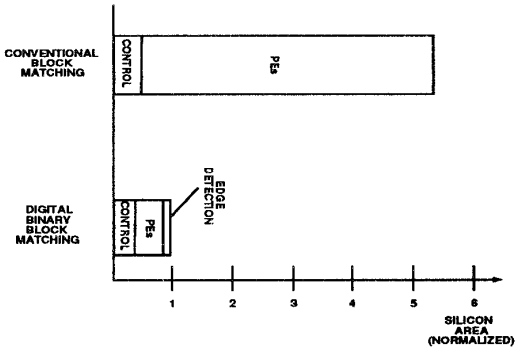


Figure 5. Comparison of the total area of the conventional and binary block matching chip.