A Parallel Vector Quantization Processor Featuring an Efficient Search Algorithm for Real-time Motion Picture Compression

Toshiyuki Nozawa, Makoto Imai, Masanori Fujibayashi and Tadahiro Ohmi

Department of Electronic Engineering, Graduate School of Engineering, Tohoku University
4New Industry Creation Hatchery Center, Tohoku University
Aza-Aoba 05, Aramaki, Aoba-ku, Sendai 980-8579, JAPAN
Phone: +81-22-217-7124, Fax: +81-22-263-9395, e-mail: nozawa@sse.ecei.tohoku.ac.jp

Abstract - A parallel vector quantization (VQ) processor has been developed aiming at real-time compression of motion pictures using 0.35μm CMOS technology. The chip employs a new simple search algorithm for VQ encoding. As a result, the die area of the chip is decreased to 33% of that of conventional fully parallel design. This chip can handle 2048 template vectors by a single chip and is applicable to real-time compression of 30 frames/sec full-color VGA (640x480 pixels) images. The chip operates at 33MHz with power dissipation of 790mW under 2.5V power supply.

I. INTRODUCTION

Vector quantization (VQ) [1] is a classical but an effective technique to compress data and attracts considerable interest in image/video compression [2, 3]. In image compression using VQ, a macro block of pixels in the input image is approximated by the nearest pattern in the codebook, where typical patterns seen in images are stored as template vectors, and only the index of that pattern is outputted and transmitted. To decompress image, all decoder have to do is only to put the corresponding pattern at each location. In contrast to this simple decoding algorithm, to search out the nearest template vector in encoding process is extremely heavy operation. As a result, real-time encoding of motion pictures cannot be realized by a general-purpose microprocessor.

Several VQ encoding processors have been developed aiming at real-time compression of motion pictures [4-6] by employing fully parallel design. These processors, however, have the same problem: hardware volume. In other words, a single processor can handle limited number of template vectors, which prevents practical use of VQ.

In this paper, we present a parallel VQ processor based on single-instruction multiple-data (SIMD) architecture for real-time motion picture compression [7]. This processor employs two-step search algorithm to reduce computation and hardware. As a result, this chip can handle eight times as many template vectors as the existing fully parallel, SIMD-based VQ processor [4]. The developed VQ processor can handle as much as 2048 template vectors by a single chip and is applicable to real-time compression of 30 frames/sec full-color VGA (640x480 pixels) images.

II. TWO-STEP SEARCH ALGORITHM FOR VQ ENCODING

As is mentioned in the previous section, searching for the nearest template vector is computationally intensive operation. We have newly introduced a rough search procedure before detailed search for the nearest template vector in order to reduce computational complexity. Since a rough search can prevent a VQ encoder from searching unpromising templates, redundant distance calculations can be eliminated. We utilized the average intensities of input and template vectors to execute the rough search because the difference of average intensity between an input and a template vector is highly correlated with the real distance between them.

The procedure of two-step search algorithm is as follows. The template vectors are sorted in advance according to their average intensity of vector elements. First, the average intensity of the input vector is calculated. Then a certain number of template vectors that have the average intensity close to that of the input vector are selected as candidates. Finally, Manhattan distances, sum of absolute value of vector elements, between the input and the selected templates are calculated, and the nearest template is searched out from the previously selected templates.

By introducing this two-step search sequence, computation for distance is drastically reduced. Up to about 90% of distance calculation can be eliminated without degrading image quality.

III. VQ PROCESSOR EMPLOYING TWO-STEP SEARCH ALGORITHM

A. Hardware Implementation

Hardware implementation of the two-step search VQ encoding process is shown in Fig. 1 as a block diagram. Memory for the codebook is matrix-shaped. The template vectors are stored and sorted from the bottom row to the top row according to their average intensity. So, a lower low contains darker templates and an upper row contains brighter templates, and template vectors stored in a row have similar average intensity. The boundary intensity value of each row is stored in the corresponding register named “tag”. In Fig. 1, for example, average intensity values of template vectors stored in the bottom row do not exceed “132”; the value stored in the bottom tag. VQ is carried out as follows. The average intensity of the input vector is compared with the values stored in tags. And then a single row that contains the template vectors having similar average intensity to the input vector is selected. Template vectors in the selected row are downloaded to distance calculation units and Manhattan
distances are calculated in parallel. The distance data are then transferred to winner-take-all (WTA) circuits in bit serial format. A WTA circuit finds the minimum distance datum and outputs the minimum distance and its address. Thus the column address of the nearest template vector to the input vector in the selected row is output from WTA. In this design, both parallelism and elimination of distance calculations can be achieved.

B. The VQ Chip

We have designed and fabricated a VQ processor based on this architecture using 0.35µm CMOS process. Fig. 2. shows the chip micrograph. The chip consists of eight 32-parallel matching blocks, which are shown in Fig. 1. The chip has two-stage pipeline composition. The pipeline cycle is 26 clocks. In this VQ processor, each codebook memory has eight rows and only one is selected at a VQ operation. Therefore, this VQ processor requires only one-eighth as many distance calculation units as a fully parallel one [4] to handle the same number of template vectors, decreasing hardware complexity drastically.

We evaluated the encoding quality of the VQ chip. Fig. 3. shows quality of decoded images which are encoded with full search VQ and the VQ processor. Degradation of image quality is negligible. Eliminating template vectors does not degrade encoding quality in this chip.

The chip can operate up to 55MHz at 2.5V power supply. This meets the requirement of 25MHz operation, which guarantees real-time encoding of full-color motion pictures (VGA, Y:U:V=4:1:1, 30frames/sec). The shmoo plot of the fabricated chip is shown in Fig. 4. Table I summarizes the features of the chip.

The die area of the VQ processor of this work is only 33% of that of fully parallel design [4]. This value is derived by scaling the fully parallel VQ processor [4] and comparing the fully parallel one with the VQ processor of this work. This drastic reduction is brought by reducing number of distance calculation units.

Fig. 2. Die micrograph of the VQ chip.

Fig. 3. Evaluation of the developed VQ processor on decoded image quality.

IV. CONCLUSIONS

A vector quantization processor has been developed for real-time encoding of motion pictures. This chip employs a new search algorithm for VQ encoding to reduce computational complexity and hardware volume. The chip can handle 2048 template vectors (16 elements, 12bit/element) by a single chip. This chip can encode 30frames/sec of full-color VGA motion pictures on the real-time basis.

In this chip, real-time encoding capability is obtained by massively parallel operation and the capacity of handling large codebook is brought by the new algorithm. Effective combination of the parallel processing hardware and the new search algorithm enhances the chip performance.

ACKNOWLEDGEMENT

The chip was fabricated in Rohm Co. Ltd. The authors wish to thank A. Kawamura, K. Marumoto and H. Honda of Rohm Co. Ltd. for help in chip design.

REFERENCES