

# A Still Image Encoder Based on Adaptive Resolution Vector Quantization Employing Needless Calculation Elimination Architecture

M.Fujibayashi, T.Nozaawa, T.Nakayama, K.Mochizuki, K.Kotani, S.Sugawa<sup>1</sup> and T.Ohmi<sup>2</sup>

Department of Electronic Engineering, Graduate School of Engineering, Tohoku University

<sup>1</sup>Management of Science and Technology Department, Graduate School of Engineering

<sup>2</sup>New Industry Creation Hatchery Center, Tohoku University

Fluctuation Free Facility for New Information Industry, Aza-Aoba 04, Aramaki, Aoba-ku, Sendai 980-8579, JAPAN

Phone: +81-22-217-3977, FAX: +81-22-217-3986, E-mail: bayashi@fff.niche.tohoku.ac.jp

**Abstract** - We have developed an advanced vector quantization (VQ) encoding hardware for still image encoding systems. By utilizing needless calculation elimination method, computational cost of VQ encoding is reduced to 40% or less, while maintaining the accuracy of full-search VQ. We have successfully implemented the advanced encoding method and Adaptive resolution VQ (AR-VQ), which realizes compression ratio over 1/200 while maintaining image quality, into a still image encoding processor. The processor can compress still image of 1600 x 2400 pixels within one second, which is 60 times faster than software implementation on current PCs.

## I. INTRODUCTION

Image compression technologies become more important in consequence of spreading digital still camera and document data processing applications. In this area, vector quantization (VQ) [1] attracts considerable interest because it is an effective technique to compress images.

In image coding system using VQ, an input vector which is typically a block of 4x4 pixels divided from the input image is approximated by the most similar template vector in the codebook. And only the index of that template vector is transmitted. Since searching the most similar template vector is computationally heavy for general-purpose microprocessors, several VQ encoding processors have been developed to accelerate VQ encoding [2-4]. Some of them employ fully parallel architecture[2], but they require increased hardware volume far from practical use. Other VQ processors employ hierarchical VQ encoding algorithm, such as two-step search[3] or tree-search[4]. But, these algorithms decrease computational complexity by sacrificing the exactness of the VQ operation, resulting in deterioration of compressed image quality. To overcome these problems, we have developed an advanced VQ encoding method, which can reduce computational cost down to 40% or less as compared with the conventional full-search method by eliminating needless calculation while guaranteeing optimum VQ result.

We have successfully implemented this algorithm into a still image encoding processor[5] based on adaptive resolution VQ (AR-VQ) method, which can realize compression ratio over 1/200 without severe degradation in image quality.

The chip is fabricated using 0.35μm embedded-SRAM gate array. This processor can compress A4 @ 200dpi image (1600x2400 pixels) within 1 sec, which is 60 times faster than currently PC, with the compression ratio over 1/200.

## II. VQ ENCODING METHOD

We have developed an advanced VQ encoding method to eliminate needless calculation without losing the exactness of VQ. This method utilizes the following explicit inequality.

$$d_j = \sum_{l=0}^{15} |i_l - t_{lj}| \geq \left| \sum_{l=0}^{15} i_l - \sum_{l=0}^{15} t_{lj} \right|$$

In this inequality,  $i_l$  is l-th element of input vector  $I$ ,  $t_{lj}$  is l-th element of j-th template vector  $T_j$  and  $d_j$  is Manhattan distance (MD) between  $T_j$  and  $I$ , which is employed as a distance measure in our VQ encoding system. This inequality means that the absolute difference between the sum of j-th template vector  $T_j$  and that of input vector  $I$  (RHS) is always smaller than the MD between  $T_j$  and  $I$  (LHS). Let us suppose that distance calculations between an input vector and lots of template vectors are carried out sequentially and temporal minimum MD are always held. When RHS for current template vector exceeds the stored temporal minimum MD, MD for the current template (LHS) never become the minimum and it is found that the current template is not the answer without calculating the real MD. Therefore, using this inequality in template vector search process, we can eliminate needless MD calculation, which is complicated vector operation, by simple scalar operation. (Note that sum of elements of a template vector  $\sum t_{lj}$  is calculated and stored in advance and that of an input vector  $\sum i_l$  is calculated once at the beginning of the search process.)

## III. VQ ENCODING HARDWARE ARCHITECTURE

We have introduced this encoding method into parallel VQ encoding processor architecture. Fig.1 shows the architecture of VQ encoding hardware. Codebook memory is organized by 16-columns x 32-rows matrix. Template vectors are firstly sorted and then stored from the bottom row to the top row according to their sum of elements. In addition, the minimum and the maximum sum of elements in each row are stored in lower tag and upper tag memories, respectively. VQ encoding is carried out as follows. First, the sum of input vector elements is calculated and it is compared with a certain

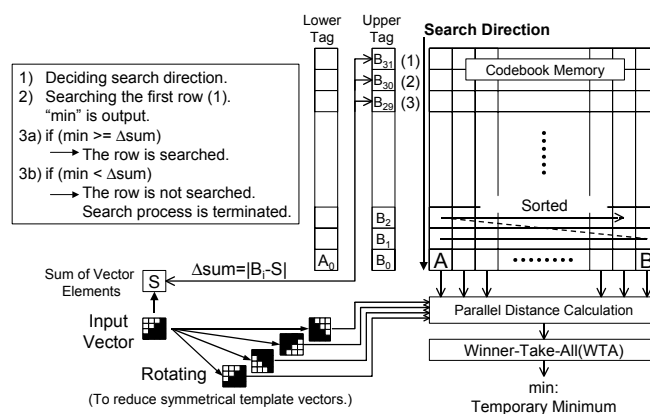


Fig. 1 Block diagram of the VQ encoding engine.

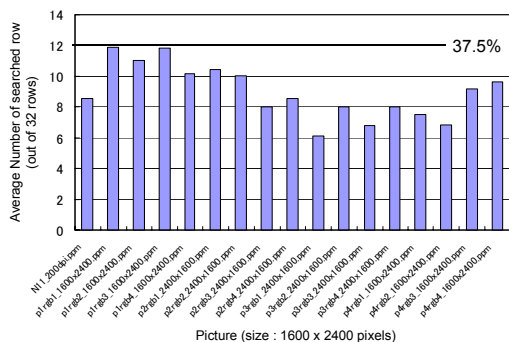


Fig. 2 Effectiveness of the VQ encoding algorithm.

threshold value to decide search direction (bottom-to-top or top-to-bottom). Next, MD between template vectors in the first row (bottom or top row) and the input vector are calculated in parallel, and winner-take-all (WTA) circuit finds the minimum distance among the calculated MDs of the row. The temporal minimum distance is obtained at this point. While searching the remaining row, if the absolute difference between the sum of input vector elements and the value of tag is larger than the temporal minimum distance, MD calculation is skipped since MD of the row is never smaller than the temporal minimum distance. In other words, the needless calculation can be reduced. In this search process, upper (lower) tag is used in the case of searching from top (bottom). This architecture can take full advantage of parallel calculation while it eliminates needless calculations and guarantees the optimum VQ result.

Fig.2 shows the effectiveness of this architecture. It is seen from this graph that the developed VQ encoder can find the optimum result by 40% or less computational cost as compared with full-search VQ encoder[2]. Computational cost reduction contributes to hardware volume reduction and faster operation.

#### IV. STILL IMAGE ENCODING PROCESSOR BASED ON AR-VQ

We have developed a still image encoding processor employing the AR-VQ algorithm and the advanced VQ encoding method. The chip was designed and fabricated with 0.35 $\mu$ m embedded-SRAM Gate-Array process. The chip micrograph is shown in Fig.3. Table I summarizes the features of the still image encoder.

We have measured the performance of the chip. The result of measurement shows that the encoding time of the VQ processor is 1.1 second and is constant for any picture (1600x2400 pixels), because PCI bus data rate limit the overall performance of VQ encoding system. Assuming that there is no bus bandwidth limitation, it is expected that this VQ processor can encode a still image of 1600x2400 pixels within one second. This encoding performance is about 60-fold faster than software implementation on ordinary PCs (PentiumIII 750MHz is assumed).

#### V. CONCLUSION

We have developed an advanced VQ encoding method. It can reduce computational cost to 40% or less as compared with conventional full-search algorithm by utilizing needless calculation eliminating method, while guaranteeing optimum

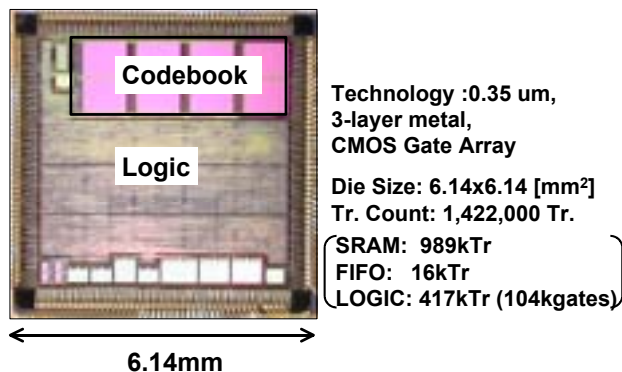


Fig. 3 Die micrograph of the still image encoder.

Table. I Still image encoder characteristics

Max. Picture Size	5440x8191 pixel @ 2Mb Line Buffer
Template Vector/ Input Vector	16 element / vector 8bit / element
Codebook Size	512 template vectors x 2 (equivalent to 2048 template vectors x 2)
Max. Operating Frequency	74.1 MHz @ 2.5V
Power Dissipation	660 mW @ 2.5V,66MHz

VQ result. We have designed and fabricated AR-VQ still image compression processor employing the advanced VQ encoding algorithm. The chip can compress 1600x2400 pixel still image within 1 second, which is 60 times faster than currently PCs.

#### ACKNOWLEDGEMENT

The VQ processor was fabricated by Rohm Co. Ltd. The authors express sincere thanks to K.Marumoto of Rohm Co. Ltd. The AR-VQ algorithm is collaborative study with SIPEC Corporation in Japan. The authors express sincere thanks to M. Konda of SIPEC Corporation.

#### REFERENCES

- [1] A.Gersho and R.M.Gray, *Vector Quantization and Signal Compression*. Norwel., MA: Kluwer, 1992.
- [2] A.Nakada, T.Shibata, M.Konda, T.Morimoto and T.Ohmi, "A fully parallel vector-quantization processor for real-time motion-picture compression", *IEEE J. Solid-State Circuits*, Vol.34, pp.822-829, June 1999.
- [3] T.Nozaawa, M.Konda, M.Fujibayashi, M.Imai, K.Kotani, S.Sugawa and T.Ohmi, "A Parallel Vector-Quantization Processor Eliminating Redundant Calculations for Real-Time Motion Picture Compression", *IEEE J. Solid-State Circuits*, Vol.35, pp.1744-1751, Nov.2000
- [4] C.-Y. Lee, S.-C. Juan and Y.-J. Chao, "Finite state vector quantization with multipath tree search strategy for image/video coding", *IEEE Trans. Circuit Syst. Video Technol.*, vol.6, pp.287-294, June 1996.
- [5] M.Fujibayashi, T.Nozaawa, T.Nakayama, K.Mochizuki, M.Konda, K.Kotani, S.Sugawa and T.Ohmi, "A Still Image Encoder Based on Adaptive Resolution Vector Quantization Realizing Compression Ratio over 1/200 Featuring Needless Calculation Elimination Architecture", *VLSI Circuits Dig. Tech. Papers*, pp.262-265, 2002