Extended Abstracts of the 2006 International Conference on Solid State Devices and Materials, Yokohama, 2006, pp. 584-585

P-5-1

Hardware Architecture for Pseudo-2D Hidden-Markov-Model-Based Face Recognition System Employing Laplace Distribution Functions

Yasufumi Suzuki and Tadashi Shibata

Department of Frontier Informatics, School of Frontier Sciences, The University of Tokyo
701 Kiban Bldg. 5-1-5 Kashiwanoha, Kashiwa, Chiba, 277-8561, Japan
email: yasufumi@else.k.u-tokyo.ac.jp, shibata@ee.t.u-tokyo.ac.jp

1 Introduction

The development of robust face recognition systems is quite essential in a variety of applications such as intelligent human-computer interfaces, security systems, and so forth. Real-time human face recognition, in particular, plays an important role in establishing user-friendly interfaces between humans and computers. Pseudo two dimensional (2D) Hidden Markov Models (HMM) are one of the statistical classifiers successfully applied to face recognition. In our previous work [1], the robust nature of the pseudo-2D HMM-based face recognition system employing the edge-based feature vectors has been demonstrated. In order to realize a real-time-responding low-power embedded face recognition system, the development of VLSI hardware is quite essential. We have already developed the dedicated VLSI chip [2] which takes only 0.23ms for edge-based feature vector generation of an entire image. Therefore, a hardware implementation of the pseudo-2D HMM classifier is essential for realizing a real-time-responding face recognition system.

Several researchers have already realized 1D HMM-based speech recognition systems on VLSI hardware since the Viterbi algorithm, which obtains the maximum-likelihood path among all state transitions within the HMM to the input vector sequence, is suitable for parallel processing. These works implement multiple processor elements (PE's) in a chip, each of which contains a Viterbi decoder and a calculator of the observation probability function. In Ref. [3], a Gaussian distribution function which is most widely utilized in various applications is employed for the observation probability function. This is not appropriate for our purpose because calculating the Gaussian distribution requires multipliers for every PE's, making their area quite large. Ref. [4] utilizes a discrete density function. Although this reduces the area of PE, it makes the accuracy of recognition worse. Therefore, another type of distribution function which is suitable for hardware implementation and approximates the Gaussian distribution function is required.

The purpose of this paper is to present the hardware architecture for a pseudo-2D HMM-based image recognition system. For reducing the area of PE's, the Laplace distribution function which can be calculated without multipliers is introduced to the observation probabilities. In order to prove the concept, the face recognition system has been implemented on a Field Programmable Gate Array (FPGA) and the performance of face recognition has been demonstrated.

2 Pseudo-2D Hidden Markov Models

The pseudo-2D HMM, which has been introduced by Kuo and Agazzi [5] to process 2D images with HMM, consists of a set of super states each of which contains a 1D HMM within themselves. The pseudo-2D HMM is utilized for modeling facial images in a hierarchical manners as in the following. Several super states correspond to the vertical facial features, such as forehead, eyes, nose, and mouth as illustrated in Fig. 1. Each state within the super state is utilized for modeling the horizontal sequence of localized features. As shown in Fig. 1, a 6×6-state left-right model is utilized in this work. The block diagram of the whole system is illustrated in Fig. 2. The system composes six sets of the embedded HMM module and Viterbi decoder. The embedded HMM consists of six sets of PE which corresponds to an embedded sate and contains the observation probability calculator and Viterbi decoder.

Viterbi Algorithm

The Viterbi algorithm computes the state sequence which maximizes the output probability efficiently. We define the value $\delta_t(j)$ which is the maximum probability in j-th state at time t. For each time the feature vector is inputed, $\delta_t(j)$ is computed recursively as follows:

$$\delta_t(j) = \max_{i=j-1,j} [\delta_{t-1}(i) a_{ij}]\, b_j(\mathbf{o}_t) \tag{1}$$

where a_{ij} is the transition probability from i-th state to j-th state and $b_j(\mathbf{o})$ is the observation probability function at j-th state. By taking the logarithm of (1), $\delta_t(j)$ is calculated as

$$\log\delta_t(j) = \max_{i=j-1,j} [\log\delta_{t-1}(i) + \log a_{ij}] + \log b_j(\mathbf{o}_t). \tag{2}$$

This means that $\log\delta_t(j)$ can be computed by only accumulators and a comparator with $\log a_{ij}$ stored in the memory and $\log b_j(\mathbf{o}_t)$ provided by the observation probability calculator as shown in Fig. 3.

Observation Probability Function

A Gaussian distribution function is often utilized for the observation probability function. For uncorrelated single m-dimension Gaussian distribution, the probability function $b_j(\mathbf{o}_t)$ and the logarithm of it are given by

$$b_j(\mathbf{o}_t) = \frac{1}{\left(\sqrt{2\pi}\right)^m \prod_{i=1}^m \sigma_i} \exp\left(-\frac{1}{2}\sum_{i=1}^m \frac{|x_i-\mu_i|^2}{\sigma_i^2}\right) \tag{3}$$

$$\log b_j(\mathbf{o}_t) = -0.5m\log(2\pi) - \sum_{i=1}^m \sigma_i - \frac{1}{2}\sum_{i=1}^m \frac{|o_i-\mu_i|^2}{\sigma_i^2} \tag{4}$$

where $\mu = \{\mu_i\}$ and $\sigma = \{\sigma_i\}$ are the mean vector and standard deviation vector obtained by learning, respectively. In this work, a Laplace distribution function is introduced to the observation probability. The Laplace distribution function and the logarithm of it are expressed as follows:

$$b_j(\mathbf{o}_t) = \frac{1}{\left(\sqrt{2}\right)^m \prod_{i=1}^m \sigma_i} \exp\left(-\sqrt{2}\sum_{i=1}^m \frac{|o_i-\mu_i|}{\sigma_i}\right) \tag{5}$$

$$\log b_j(\mathbf{o}_t) = -0.5m\log 2 - \sum_{i=1}^m \sigma_i - \sqrt{2}\sum_{i=1}^m \frac{|o_i-\mu_i|}{\sigma_i}. \tag{6}$$

Assuming all standard deviations of feature-vector elements are equal, (6) is simplified with constant values α and C as

$$\log b_j(\mathbf{o}_t) = \alpha \sum_{i=1}^{m} |o_i - \mu_i| + C. \tag{7}$$

In (7), the constant value C can be eliminated since it does not affect the result of Viterbi path search. Hence, the observation probability is obtained by only calculating the Manhattan distance between the feature vector $\mathbf{o}$ and the mean vector μ. The parameter α is determined to maximize the dynamic range of the output metrics under finite precession.

3 FPGA Implementation and Performance Evaluation

The proposed architecture has been implemented on an Altera Cyclone FPGA (EP1C12Q240C6). The recognition performance of the system was evaluated on the AT&T face database [6] which contains 10 different images for each of 40 people. For each person, the face model is learned from nine images of the same person. The recognition rate was evaluated on the images excluded in the training. The details of face recognition algorithm are described in Ref. [1]. The recognition rates with different bit widths of data were shown in Fig. 4. When no less than 8-bit data precision was employed, the recognition rate of over 97% was obtained. Table 1 shows the specification of the recognition system when the data bit width is 8. The processing time for identifying an face image from 40 people was 44.2ms at a 100MHz clock frequency. The chip size was about 62 kilo logic gates and reduced by 47% from the one employing the Gaussian distribution function.

4 Conclusion

The hardware architecture of the face recognition system employing the pseudo-2D HMM has been proposed. In order to reduce the area of PE, the Laplace distribution function has been introduced to the observation probabilities. The system has been implemented on the FPGA, and the performance of the system has been successfully demonstrated.

References

[1] Y. Suzuki and T. Shibata, to be published in *Proc. European Signal Processing Conf.* (2006).

[2] H. Yamasaki and T. Shibata, *Proc. European Solid-State Circuits Conf.* (2005) 121.

[3] S. Yoshizawa, N. Wada, N. Hayasaka and Y. Miyanaga, IEEE Trans. Circuits Syst. I **53** (2006) 70.

[4] F. Vargas, R. Fagundes and D. Barros, *Proc. Int'l Conf. Accoustics, Speech and Signal Processing* (2001) 1217.

[5] S. Kuo and O. Agazzi, IEEE Trans. Pattern Anal. Machine Intell. **16** (1994) 842.

[6] F. Samaria and A. Harter, *Proc. 2nd IEEE Workshop on Applications of Computer Vision* (1994) 138.

Table 1 Specification of FPGA implementation.

# of PE's	36	# of Gates	62,000
Data Precision	8bits	Processing Time	4.42ms
Clock Freq.	100MHz	Power	475mW

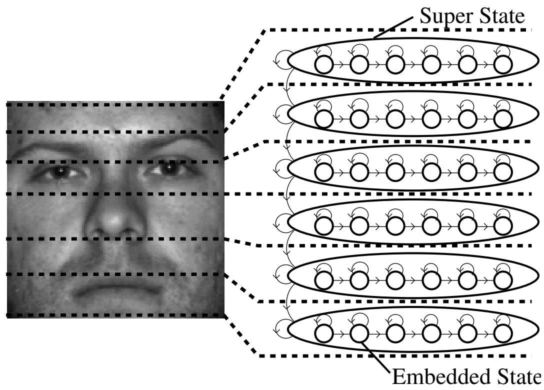

Fig. 1 Pseudo-2D Hidden Markov Model.

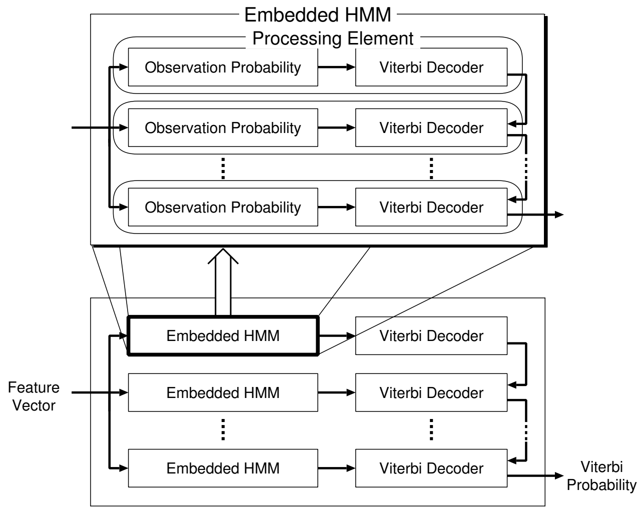

Fig. 2 Block diagram of pseudo-2D HMM system.

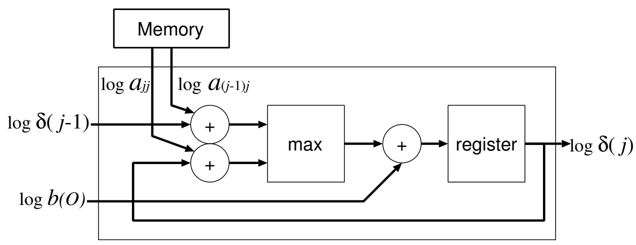

Fig. 3 Block diagram of Viterbi decoder.

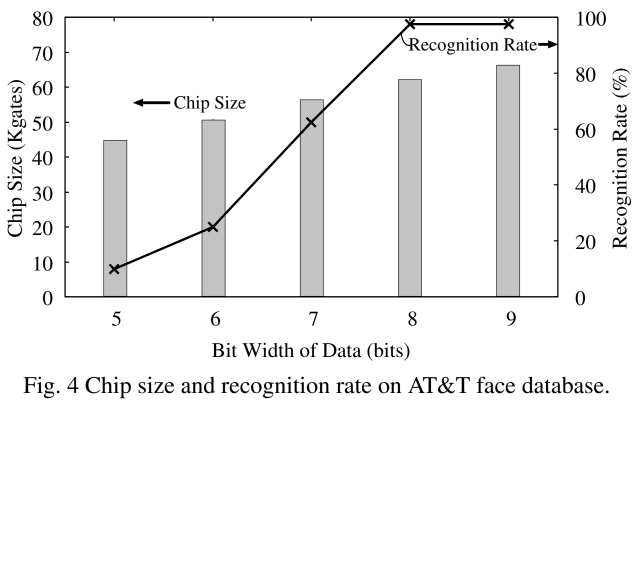

Fig. 4 Chip size and recognition rate on AT&T face database.