Real-time Character Recognition System Using Associative Memory Based Hardware

Ali Ahmadi, Yoshinori Shirakawa, Md.Anwarul Abedin, Kazuhiro Takemura, Kazuhiro Kamimura, Hans Jürgen Mattausch, and Tetsushi Koide

Research Center for Nanodevices Systems, Hiroshima University, 1-4-2 Kagamiyama, Higashi-Hiroshima, 739-8527, Japan

Phone: +81-82-424-6265 Fax: +81-82-422-3499

Email: {ahmadi, shirakawa, abedin, takemura, kamimura, hjm, koide}@sxsys.hiroshima-u.ac.jp

1. Introduction

Optical character recognition (OCR) systems have been widely used in recent years and various approaches are applied for developing their hardware and processing algorithms [1]. As for a small mobile OCR system, ex. a cognitive pen, usually an ideal model is thought as a system with high accuracy and speed, and minimum hardware size at the same time. Different movable OCR products are presently in the market [2] but they hardly ever afford the desired robustness and hardware size, simultaneously.

In this research we propose an associative memory based system for real-time character recognition and evaluate its performance with real data samples of English texts. The associative memory we use here as the main classifier is already designed in our lab [3] and has a mixed analog-digital fully-parallel architecture for nearest Hamming/ Manhattan-distance search. The OCR system proposed here may be used ultimately in a cognitive pen product for online text recognition task.

2. Major System Steps

The major steps of the system are as shown in the block diagram of Fig. 1. It is worth-noting that all processes are achieved online.

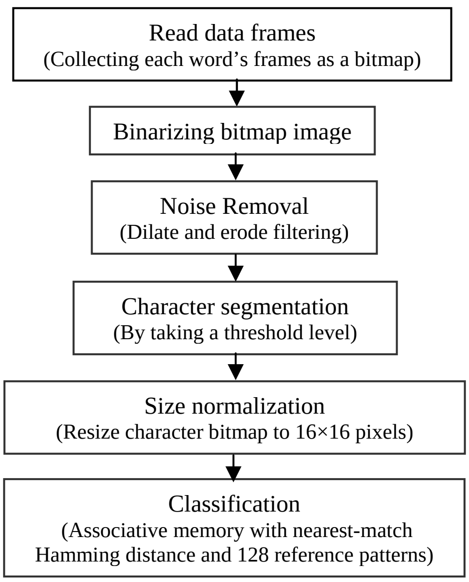


Fig. 1: Block diagram of major system steps.

For simplicity we suppose that at this phase the system is used only for recognition of printed texts. The first step is data reading where the reading device (scanner sensor) moves on each line of the text with an appropriate speed and scans the data continuously as a sequence of thin frames. The frames between each two word spaces are collected and form a large frame as a gray-scale bitmap array which contains all the word characters. By taking a proper threshold value the image is binarized to a simple black-white bitmap (including noise). In order to remove noise from the frame, two different filters are applied for dilate and shrink of the image. Next, by employing a simple segmentation method and taking a threshold level the black segments within each frame are recognized and each one is considered as a single character.

To have an accurate classification each character size is normalized to 16×16 pixels before classification. We use a simple linear algorithm for resizing the character bitmap. The last and main step of the process is character classification which is carried out by a nearest-distance search algorithm applying the associative memory. The normalized segmented character is matched as a 256 bits vector to a number of reference patterns using the Hamming distance measure and the reference pattern with minimum distance is considered as the winner class. More explanations about the associative memory characteristics are given in Section 3.

3. Associative Memory Classifier

Figure 2 shows a simple architecture of the compact associative memory.

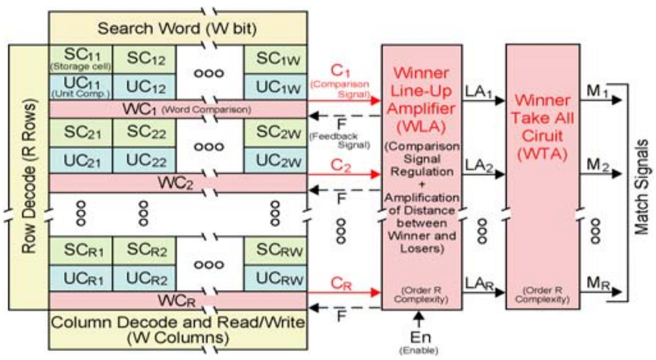


Fig. 2: Associative memory architecture.

A number of k-bit digital subtraction and absolute-value calculation units compare the W binaries in all rows of the memory field in parallel with the reference data. The WLA achieves a large regulation range for feedback stabilization and eliminates the inefficient possibilities of under- or over-regulation by a maximum-gain region which self-adapts to the winner input C_{win}. A signal follower provides the necessary high driving current for scaling to an increased number of reference patterns R. Low power dissipation of the system is achieved by an individual power regulation from the signal-regulation units for each input-signal source.

The transistor-count is only 6 per row. A modified version of the fast minimum circuit proposed by Opris et al. [4] is applied for combined feedback generation and distance amplification. The minimum function is used in the feedback loop and an intermediate node in each row circuit is used for the distance-amplified WLA-output LA_i. Table 1 shows the performance data of designed associative memory depending on the Hamming and Manhatan distance measure. More detailed information about the associative memory performance can be found in [3].

Table 1: Performance data of designed associative memory test chips.

Distance Measure	Hamming	Manhattan (5 bit)
Memory Field	32 x 768	128 x 80
Technology	0.6 µm CMOS	0.35 µm CMOS
Area	9.11 mm²	8.6 mm²
Search Range	0 - 400 bit	0 - 480 bit
Winner-Search Time (Measured)	< 70 nsec	< 190 nsec
Performance	1.34 TOPS	160 GOPS
Power Dissipation	43 mW	91 mW
Supply Voltage	3.3V	3.3V

4. Experimental Results

The system was simulated with a Matlab program. Different samples of scanned data including different fonts, noisy data, color background data, slightly rotated data, and data with different resolution were selected as the input samples. A total number of 80 samples for each character type were gathered. Figure 3 shows some data samples.

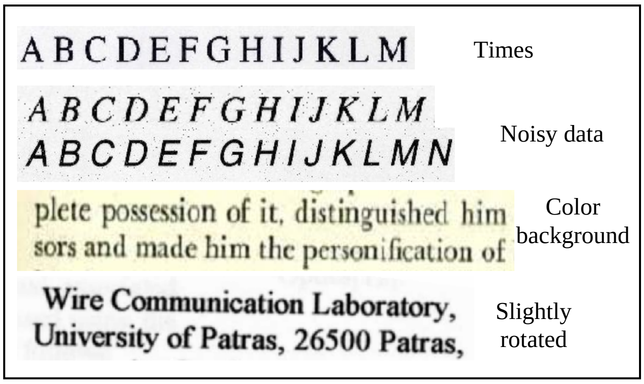


Fig. 3: Some text samples used as input data.

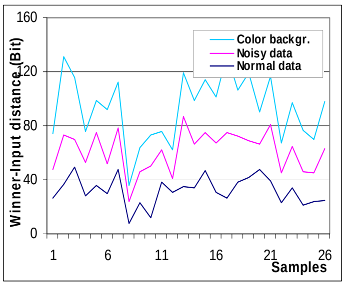
As mentioned in Section 2, each data sample is considered as a 256 bits vector. The experimental results of distance-matching between data vectors and reference patterns are reported in Table 2. As can be seen from the Table, excepting for the noisy data, the number of misclassified samples in other cases is zero. The minimum distance between winner and nearest-loser over all the data samples is 9 which is reliable enough. Figure 4(a) indicates the winner-input distance for different data samples. The average winner-input distance for all the input samples was calculated and found as 31 bits. Having this distance and referring to plot of Fig. 4(b) which gives the typical winner search time of the associative memory according to winner-input distance, we can find the average search time of 45 ns for classification of each test sample. This is the search time within 128 reference patterns and will be changed in case of increase in reference patterns number.

5. Conclusions

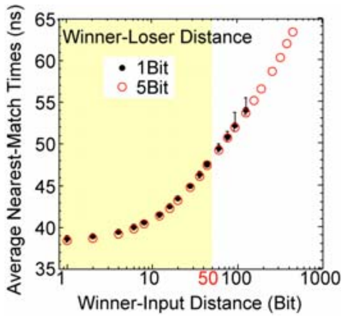
An associative memory based system for online character recognition is proposed in this paper. Taking an associative memory of 128 reference patterns size and 256 bits per pattern designed in 0.35 µm technology we could get an average nearest-search time of 45 ns for classification of different samples of characters written in Times and Arial fonts.

Table 2: Experimental results of data classification.

Data type	Normal		Noisy		Color Background		Slightly rotated	
Font	Times	Arial	Times	Arial	Times	Arial	Times	Arial
Sample no.	10	10	10	10	10	10	10	10
Scan resolution	70/100	70/100	70/100	70/100	70/100	70/100	70/100	70/100
Misclassified samples	0	0	1	1	0	0	0	0
Classified to different font (but still correct)	1	1	1	0	2	0	2	1
Min. distance of winner & nearest loser	11	11	9	10	10	9	8	9
Average distance of winner-input	19	22	32	34	45	47	51	53



(a)



(b)

Fig. 4: (a) Winner-Input distance for different types of data. (b) Winner average search times in associative memory as a function of Winner-Input distance.

The initial classification results taken from real samples are acceptable however we still need to test the system with larger number of real data and also apply more effective algorithms for image preprocessing and noise removal. Comparing to OCR products existing in the market, however this prototype model is not yet robust enough but is advantageous in terms of classification time and hardware size. We are also planning to develop a system with a learning algorithm for optimizing the reference-pattern selection process.

References

[1] S.V. Rice, et al., *Optical Character Recognition: An Illustrated Guide to the Frontier*, Kluwer Academic Publishers, USA, 1999.

[2] For example Wizcom *Quicktionary* which is a mobile scanner dictionary, and *IrisPen* a handheld scanner pen.

[3] Y.Yano, T. Koide, H.J. Mattausch, *Associative Memory with Fully Parallel Nearest-Manhattan-distance Search for Low-power Real-time Single-chip Applications*, Proc. of ASP-DAC'2004, pp. 543 – 544, Japan, 2004.

[4] I. O. Opris, *Rail-to-Rail Multiple-Input Min/Max Circuit*, IEEE Trans. on Circ. and Systems II, Vol. 45, No. 1, pp. 137–141, 1998.

Real-time Character Recognition System Using Associative Memory Based Hardware

Hiroshima University
Research Center for Nanodevices and Systems (RCNS)
System Design and Architecture Research Division

Ali Ahmadi, Yoshinori Shirakawa, Md.Anwarul Abedin, Kazuhiro Takemura, Kazuhiro Kamimura, Hans Jürgen Mattausch, and Tetsushi Koide

Research Center for Nanodevices and Systems, Hiroshima University

Background & Motivations

Optical character recognition (OCR) systems have been widely used in recent years and various approaches are applied for developing their hardware and processing algorithms*.

As for a small mobile OCR system, ex. a cognitive pen, usually an ideal model is thought as a product with high accuracy and speed, and minimum hardware size at the same time.

Different movable OCR products are presently in the market** but they hardly ever afford the desired robustness and hardware size simultaneously.

* S.V. Rice, et al., *Optical Character Recognition: An Illustrated Guide to the Frontier*, Kluwer Academic Publishers, USA, 1999.

**For example Wizcom *Quicktionary* which is a mobile scanner dictionary, and *IrisPen* a handheld scanner pen.

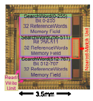
1

Research Objective

An associative memory-based OCR with qualifications:

1- Recognition of printed characters & words
2- Robustness to Noise, Rotation, Color
3- Applicable to different Fonts
4- High Speed

5- Learning capability



Associative memory: A mixed analog-digital fully-parallel architecture for nearest Hamming/ Manhattan-distance search which is already fabricated in our lab.

2

Flowchart of System

Read data frames
(Collecting each word's frames as a bitmap)

Binarizing bitmap image

Noise Removal
(Dilate and erode filtering)

Character segmentation
(By taking a threshold level)

Size normalization
(Resize character bitmap to 16×16 pixels)

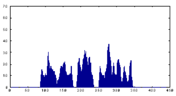
Classification
(Associative memory with nearest-match Hamming distance and 128 reference patterns)

3

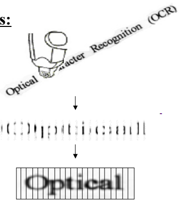
Data Reading

Real-time data reading problems:

-Rotation & shift
-High rate of noise
-High speed of recognition needed



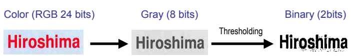
A Vertical Projection Profile (VPP) of the characters is used to find the space between word.



Data is read as a sequence of thin frames

4

Binarizing

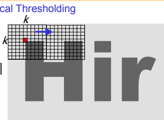


Thresholding Method:

Algorithm 1: Image Equalization → Threshold definition from image histogram (*OTSU method*)

Algorithm 2: Local Thresholding

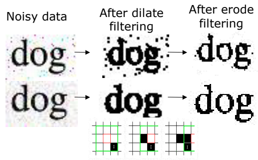
$$Th_i = \frac{1}{nb}\sum I_j$$
$$j \in [i-k, i+k]$$
$$nb = k \times k$$



For each pixel of image we take a neighborhood of size *k* and find a local threshold by calculating the mean of pixels intensities.

5

Noise Removal



Salt & pepper noise

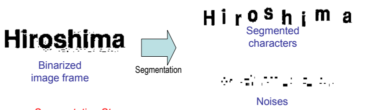
Gaussian noise

We use morphological opening/closing operators for reducing noise: Dilate filter & Erode filter

These filters can be easily implemented in hardware.

6

Character Segmentation

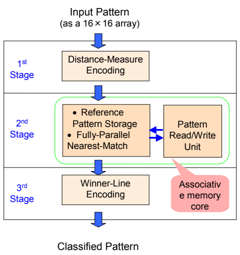


Segmentation Steps:

1- Run-length encode the input image.
2- Scan the runs, assigning preliminary labels and recording label equivalences in a local equivalence table.
3- Resolve the equivalence classes.
4- Relabel the runs based on the resolved equivalence classes.

7

Classification



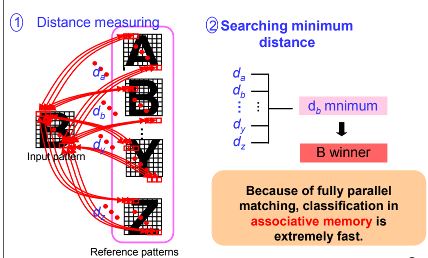
Encoding Table for Hamming Distance

Decimal	Binary (3 bits)	Hamming Encoding (7 bits)
0	000	0000000
1	001	0000001
2	010	0000011
3	011	0000111
4	100	0001111
5	101	0011111
6	110	0111111
7	111	1111111

(2^k -1 bits are needed for encoding k-bits binary)

8

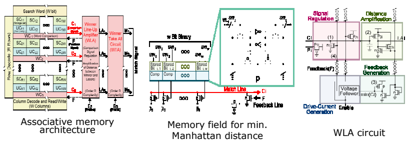
Parallel Pattern-Matching



Because of fully parallel matching, classification in associative memory is extremely fast.

9

Associative Memory Architecture

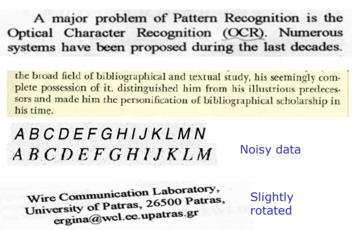


Distance Measure	Hamming	Manhattan (5 bit)
Memory Field	32 x 768	128 x 80
Technology	0.6 m CMOSμ	0.35 m CMOSμ
Area	9.11 mm2	8.6 mm2
Search Range	0 - 400 bit	0 - 480 bit
Winner-Search Time (Measured)	< 70 nsec	< 190 nsec
Performance	1.34 TOPS	160 GOPS
Power Dissipation	43 mW	91 mW
Supply Voltage	3.3V	3.3V

Characteristic performance data of designed chip

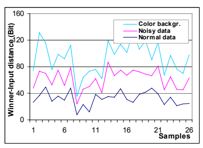
10

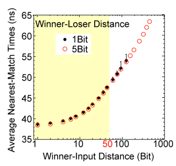
Data Samples



11

Experimental Results





Average winner-input distance for all samples: 31 bits

Average search time: 45 ns

12

Experimental Results (2)

The classification results obtained from real scanned data. The system is simulated by Matlab programming.

Data type	Normal		Noisy		Color Background		Slightly rotated	
Font	Times	Arial	Times	Arial	Times	Arial	Times	Arial
Sample no.	10	10	10	10	10	10	10	10
Scan resolution	70/100	70/100	70/100	70/100	70/100	70/100	70/100	70/100
Misclassified samples	0	0	1	1	0	0	0	0
Classified to different font (but still correct)	1	1	1	0	2	0	2	1
Min. distance of winner & nearest loser	11	11	9	10	10	9	8	9
Average distance of winner-input	19	22	32	34	45	47	51	53

13

Conclusions

1- The proposed associative memory based OCR is advantageous in terms of classification speed and hardware size.

2- Due to fully parallel pattern-matching used in the associative memory the average search time for each character is obtained as 45 ns which is very much faster than other existing OCRs.

3- More robust algorithms are still required for noise removing however complicated algorithms generally are needed to be simplified for hardware implementation.

4- Larger number of reference patterns with larger memory banks are necessary to improve system application to wider range of data samples. Also, larger number of input samples are needed to verify the system performance more adequately.

14

Future Plan

- **We are planning to equip system with a learning algorithm which adapts and verifies reference patterns automatically over the time.**

- **Besides Hamming & Manhattan distance we are studying use of Euclidean distance which gives more effective results in some pattern-matching cases.**

- **The system can be improved as an independent portable OCR for using in professional tasks.**

15