

二進 映像의 正規化 시스템에 관한 研究

金 赫 九* · 黃 勝 郁**

A Study on the Hardware System for Normalization of Binary Images

H.K.Kim, S.W.Hwang

Abstract

The binary image normalization method in image processing can be used in several fields. Especially, its high speed processing method and its hardware implementation is more useful. A normalization process of each character in character recognition requires a lot of processing time. Therefore, this research was done as a part of high speed process of optical character reader implementation as a pipeline structure with host computer to give temporal parallelism. For normalization process, We propose the special hardware for rotating the character image and use the general purpose CPU, MC68000, in order to implement the system controller.

1. 서 론

광학적 문자인식(Optical Character Recognition)시스템은 처리속도나 인식율에 있어서 고속(高速), 고정도(高程度)가 요구된다. 특히 한글이나 한자를 대상으로 하는 인식시스템은 인식대상이 되는 문자의 종류가 수천자에 이르므로 상용성을 갖는 시스템을 구현하는 데는 많은 제약이 따른다. 최근의 연구 동향을 살펴보면 처리속도를 고속화하기 위하여 병렬성을 갖는 문자인식 시스템을 구현하고 있으며, 인식율을 높이기 위한 다양한 알고리즘이 제시되고 있다^{6~9)}.

* 고려대학교 전자공학과 박사과정

** 한국해양대학교 제어계측과

본 논문에서는 OCR 연구 분야중 고속 문자인식 시스템을 구현하기 위한 연구의 일환으로 문자인식의 여러 단계중에서 문자크기의 정규화 과정을 고속화하는 방안에 대하여 기술하고자 한다. 정규화는 임의의 크기($H \times W$)를 갖는 2차원의 입력화상을 일정한 크기($H \times M$)의 화상이 되도록 선형 또는 비선형적으로 축소 및 확대시키는 작업이다. 정규화 과정은 인식알고리즘의 전처리 단계에서 문자화상의 크기 변화에 의한 문자 특징의 변화를 줄이기 위하여 이용된다. 문자의 인식과정이 문자의 크기 정보를 기준으로 하여 단계적으로 국소적 특징들을 추출하는 과정을 거치므로 특징량의 정규화를 위한 문자 화상의 정규화 과정은 알고리즘에 따라 중요한 부분이 된다. 특히, 특징추출 과정을 systolic array등을 이용하여 하드웨어로 구현할 경우에는 규격화된 화상을 제공하기 위한 정규화 과정은 필수적이다.

그림 1은 스캐너를 통하여 입력된 문서화상으로부터 각 문자화상의 추출, 정규화, 특징추출, 분류 및 인식단계로 나누어지는 통계적 문자인식 과정을 나타내고 있다. 그 동안의 연구결과^{3,6)}를 토대로 살펴보면, 문자화상의 추출시간이 IBM-PC 486에서 약 4msec이고, 정규화 소요시간이 약 15msec정도 소요되고 있다. 즉, 문자인식 과정에 있어 정규화 소요시간은 문자영역 추출시간의 약 4배로 정규화 과정이 문자인식의 고속처리에 제한요소로 작용하고 있다.

본 논문에서 제안하고 있는 문자화상의 정규화 시스템은 모토롤라의 MC68000 MPU를 중심으로 자체 메모리, 중형 변환회로 및 FIFO(First-In First-Out)등으로 구성된다. 이 시스템에서의 정규화 과정은 인식대상 문서로부터 추출된 문자화상을 받아서 행방향의 정규화를 수행한 다음, 한 방향으로 정규화된 문자화상을 중형변환회로를 이용하여 90° 회전시켜서 다시 한번 행방향의 정규화를 수행함으로써 이루어진다. 행방향 정규화 알고리즘은 메모리에 저장되어 있으며, 인접시스템과의 통신은 FIFO(First-In First-Out)를 통하여 이루어진다. 이 시스템의 장점은 중형변환회로를 이용하여 행방향으로만 데이터처리를 수행하므로 화소단위(비트)의 처리를 16비트의 화소열 단위의 처리로 대체할 수 있어 처리시간을 대폭적으로 줄일 수 있다. 제안된 시스템을 이용한 문자화상의 정규화 결과, 약 50×50 크기의 문자 화상을 기준으로 할 경우에 정규화 시간이 0.22msec로서 처리시간을 약 7~8배 단축시킬 수 있었다.

이 시스템은 파이프라인 방식의 다중프로세서로 구성되는 인식시스템의 구현을 위한 연구의 일환으로 구현되었으며, 구현결과, 문자 추출및 정규화 과정의 파이프라인 가능성을 확인할 수 있었다.

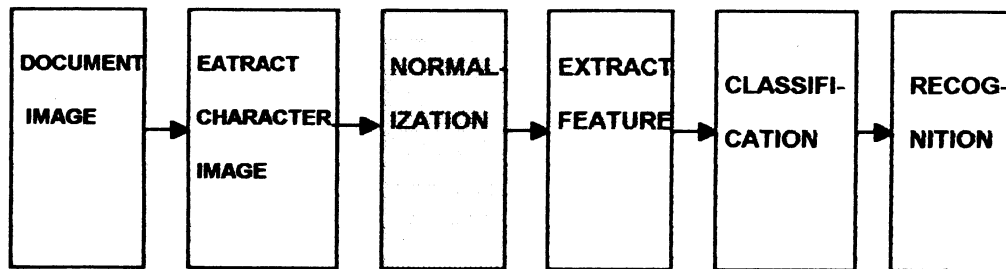


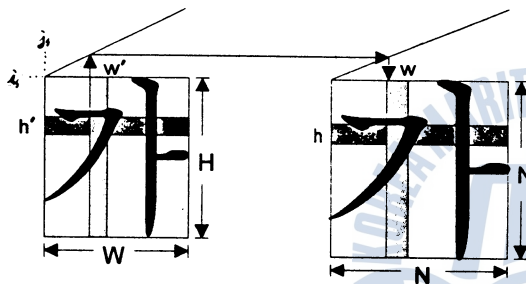
Fig. 1. The Process for character recognition.

2. 정규화 시스템의 설계

이 장에서는 먼저 정규화 알고리즘에 대하여 설명하고, 정규화 알고리즘을 고속화하기 위한 중형 변환 회로에 대하여 기술한다. 특별한 설명이 없는 한 입력화상은 크기가 $W \times H$ 인 2진화상으로써 $N \times N$ 의 크기로 정규화되는 것으로 한다.

2.1 정규화 알고리즘

정규화는 원래의 화상을 일정한 크기로 확대 또는 축소하여, 문자영역의 위치와 크기의 차이에 의한 영



(a) Input character image. (b) normalized character image

Fig. 2. An example of normalization.

향을 흡수하는 처리과정이다. 정규화는 선형 정규화와 비선형 정규화로 나눌수 있으며, 이 중에서 선형정규화는 입력화상을 일정한 크기로 선형 신축하는 작업이고, 비선형정규화는 입력화상의 특징을 변환함수로 하여 사상하는 방법이다.

본 논문에서는 정규화 알고리즘으로 선형 및 비선형 정규화가 가능한 알고리즘¹⁰⁾을 선정하여 이를 이용하였다. 이 알고리즘에서는 행방향 및 종방향의 선밀도 $f(i)$, $g(j)$ 를 구하여 변환함수 $F(i)$, $G(j)$ 를 식 ①, ②과 같이 정의하고 이를 이용한다.

여기서 선밀도란 백화소에서 흑화소로의 반전수이다. 식 ①, ②에서 i_s , j_s 는 흑화소가 존재하는 영역의 첫 번째 행 및 열의 위치를 나타내고, b 는 비선형의 정도를 정하는 무계계수(weight coefficient)이다. 이를 그림을 이용하여 설명하면, 확대할 경우에는 그림 2의 a)의 j' 열을 b)의 $G(j') \times D/W$ 열부터 $(g(j'+1) \times D/W - 1)$ 열에 대응시키고, 축소할 경우에는 b)의 j 열을 a)의 $G^{-1}(\lceil j \times W/D \rceil)$ 로 정해지는 연속된 복수열에 대응시킨다. 여기서 ' $\lceil \rceil$ '는 round up을 의미한다.

$$F(i) = \begin{cases} \sum_{i'=i-1}^i \{ f(i')b + 1 \} & (i \geq i_s) \\ 0 & (i < i_s) \end{cases} \dots\dots\dots ①$$

$$G(j) = \begin{cases} \sum_{j'=j-1}^j \{ g(j')b + 1 \} & (j \geq j_s) \\ 0 & (j < j_s) \end{cases} \dots\dots\dots ②$$

2.2 중형 변환회로의 제안

정규화 과정은 행방향 및 열방향으로 데이터를 처리하게 되는데, 이 과정에서 시스템에 따라 데이터의 액세스 수가 행방향에 비하여 열방향 처리시 8배~16배 정도 많게 된다. 이는 정규화 알고리즘 수행시 행

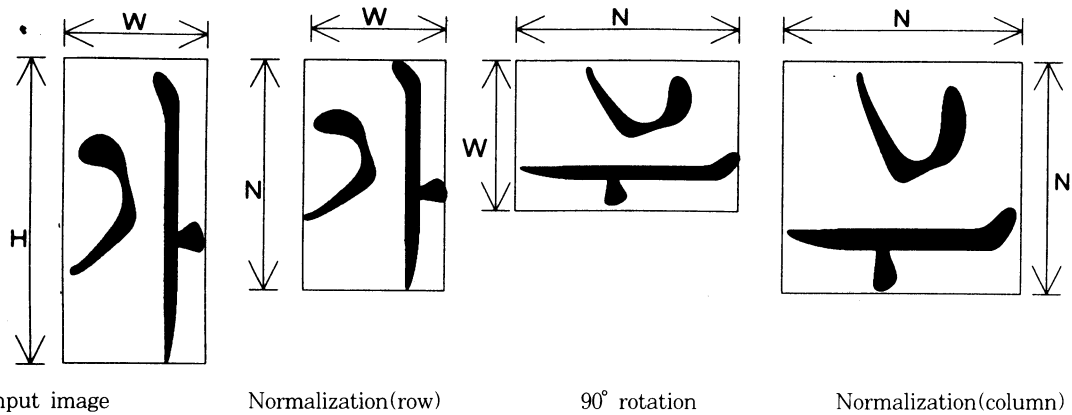


Fig. 3. Normalization process.

방향 처리는 화소를 워드단위(16비트)로 처리할 수 있는 반면, 종방향 처리는 화소단위(1비트)로 밖에 처리할 수 없기 때문이다. 본 연구에서 이러한 점에 착안하여 종횡 변환회로를 제안하여 행방향 정규화 후에 화상을 90° 변환시켜 열방향 정규화를 수행하도록 하였다.

종횡 변환회로는 메모리에 어드레스 라인을 2중화한 구조로서, 화소 데이터를 변환회로의 메모리에 기록한 후에 다시 읽게 되면 90° 변환된 화상을 얻을 수 있도록 설계하였다. 종횡변환회로는 열방향의 정규화를 워드 단위로 수행하기 위하여 이용된 하드웨어로서, 이를 이용하면 행방향 및 열방향 정규화 모두 화소 데이터를 워드 단위로 처리할 수 있어, 소프트웨어에 의한 화소단위의 정규화 과정에 비해 정규화 속도를 대폭 향상시킬 수 있다. 이같은 문자화상의 정규화 과정을 그림 3에 나타내었다. 이상의 과정을 거쳐 정규화가 완료된 문자화상은 소영역으로 분할되어 FIFO를 통하여 특징 추출부에 전달된다.

2.3 정규화 시스템

정규화 시스템은 그림 4와 같이 모토롤라의 MC68000(10MHz)을 중심으로 호스트 컴퓨터와의

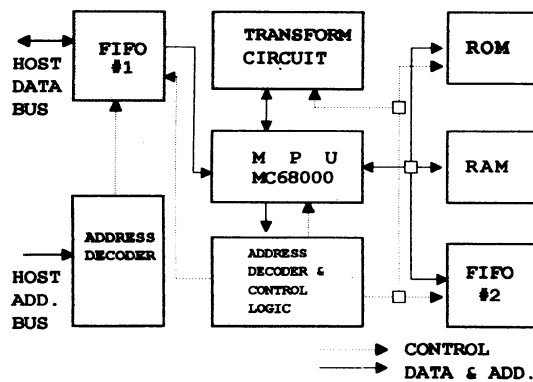


Fig. 4. Block diagram of proposed normalization system.

데이터의 전송을 위한 FIFO #1, ROM, RAM, 종횡 변환회로 및 정규화된 문자화상 데이터를 특징추출 하드웨어로 전송하는 FIFO #2로 구성한다.

문자화상 데이터의 전송은 호스트 컴퓨터의 입출력 포트 및 FIFO를 통하여 이루어지며, 데이터의 전송여부는 각 FIFO의 플래그를 이용하여 전송상태를 파악하도록 하였다. 호스트 컴퓨터에서 문자 화상의 너비와 높이 정보 및 이에 해당되는 데이터의 갯수를 FIFO에 기록하면, 정규화 시스템에서는 FIFO의 상태를 플래그를

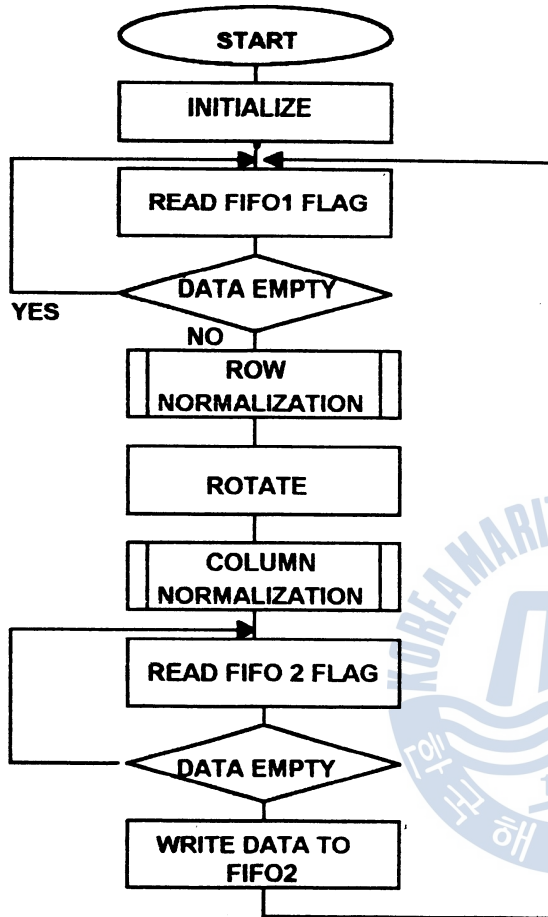


Fig. 5. Flowchart of normalization program.

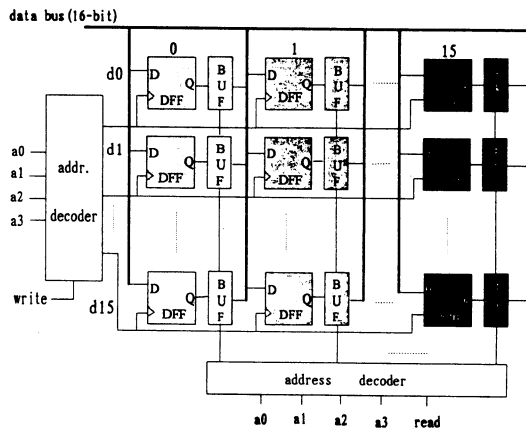


Fig. 6. The block diagram of column-row rotating circuit.

통하여 확인 한 후, 전송된 문자화상의 데이터를 읽어온다. 문자화상의 전송은 문자열 단위로 이루어질 수 있도록 FIFO의 크기를 32KByte (400Bytes/Character X 80characters/line X 1 line=32KBytes)로 설정하였다. 시스템 ROM은 정규화 프로그램을 저장하고 있으며, RAM은 중간 처리 과정의 데이터를 저장한다. 정규화 과정을 수행하기 위한 프로그램의 처리과정에 대한 순서도를 그림 5에 나타내었다.

3. 정규화 시스템의 구현 및 결과

문자 화상의 종횡 변환회로는 DPRAM (Dual Port RAM)구조와 유사한데, 내부적으로 어드레스디코더와 플립플롭을 이용하여 16×16 크기의 화상 정보를 90° 회전 시키는 기능을 수행한다. 워드(16비트)단위로 16개의 데이터를 내부의 플립플롭에 기록한 후, 읽으면 90° 회전된 데이터를 얻었 수 있다. 그러나 1차 행방향으로 정규화된 문자화상의 크기가 N(42)×W이므로 이를 16×16 단위의 크기로 나누어 몇번의 종횡 변환을 수행하여야 한다. 종횡 변환 회로의 구성도를 그림 6에, 데이터의 처리 과정을 그림 7에 나타내었다.

FIFO #2는 정규화가 완료된 문자 화상데이터를 특징추출 하드웨어로 전송하기 위하여 사용하였고, 한 문자에 해당되는 화상에 대해 정규화가 완료되면 제어조직은 인터럽트 신호를 발생시켜 종료를 알린다. 인터럽트 신호를 인지한 다음단계의 하드웨어에서 정규화 데이터를 인출하기 시작하면 정규화 하드웨어는 다음 정규화 대상 문자를 FIFO #1에서 인출하여 정규화를 수행한다.

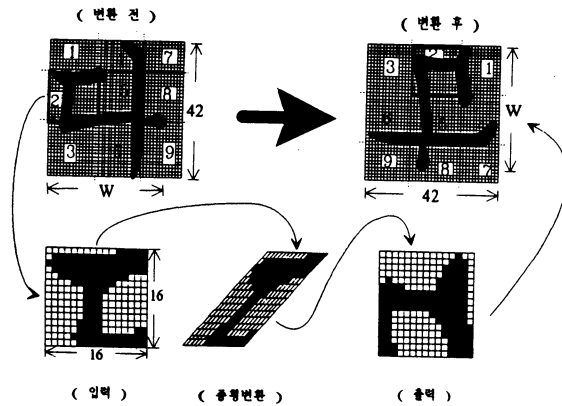


Fig. 7. Rotation of character image.

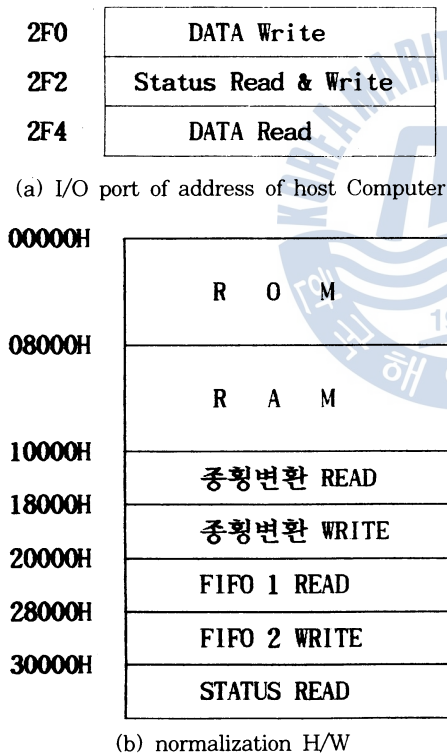


Fig. 8. Address map of normalization hardware.

Table 1. Comparison of processing time.

Items	Host computer	Implemented H/W
Normalization	14.6ms	2.2ms
Extraction of character image	4.6ms	---

정규화 회로중 중형 변환회로와 에드레스 디코더, 제어로직, 플립플롭 및 정규화 회로의 게이트 레벨의 모든 회로를 2개의 UCIC(User Configured) 구현된 하드웨어에서 처리하는 정규화 크기는 입력문자 화상의 크기에는 제한을 두지 않았으며, 정규화 후의 출력의 크기도 임의로 설정이 가능하다. 본 논문에서는 일반적으로 많이 통용되는 출력물을 300dpi로 스캔한 크기에 많은 변형을 가하지 않는 크기로 설정(42×42 화소)하여 실험 하였다. 실험을 통해 호스트 컴퓨터(IBM PC486-33MHz)에서 정규화에 소요되는 평균 시간과 구현된 하드웨어에 의해 수행된 소요 시간을 표 1에 나타내었으며, 병렬 처리를 위한 소요시간의 비교를 호스트에서의 개별 문자 추출에 소요되는 시간을 함께 기록하였다.

위의 결과로부터 하나의 개별 문자에 대하여 평균적으로 호스트 컴퓨터에서 정규화하는데에는 약 14.6ms가 소요되었고, 구현된 하드웨어에 의한 처리 시간은 약 2.2ms로 약 7배의 처리 속도가 개선되었다.

4. 결 론

화상처리 분야에 있어서 입력화상을 임의의 크기로 축소 및 확대할 수 있는 정규화시스템을 구현하였다. 특히 중형변환회로의 구조를 제안하고 구현하므로써 정규화 알고리즘을 고속화시킬 수 있었다.

문자인식 알고리즘의 개발에 있어서 각 처리단계를 파이프라인 구조로 구성함으로써 시간적 병렬성을

갖는 고속 문자인식기를 구현할 경우, 각 처리단계의 단위시간이 균일하여야 한다. 문자인식과정에 있어서 정규화 과정의 처리시간이 호스트 컴퓨터에서의 개별 문자화상의 추출시간보다 작거나 같다면 호스트 컴퓨터 개별문자 추출과정과 정규화과정을 파이프라인화 할 수 있다. 본 논문에서 구현된 하드웨어는 정규화 크기 42×42 화소로 할 때, 처리시간이 2msec 이하로 IBM-PC 486(33MHz)에서 개별 문자추출 시간과 정규화 과정의 파이프라인화 가능성을 확인할 수 있었다.

5. 참고문헌

- 1) "OCR의 기술동향 및 인식기술", 산업연구원, 1989.
- 2) 김진형 외, "문서 인식 및 처리기의 개발에 관한 연구", 연구보고서, 한국 과학 기술원, 1989. 4.
- 3) 김덕진 외, "병렬처리 기술을 이용한 인쇄문자 인식기의 구현에 관한 연구", 연구보고서, 고려대학교 정보통신 기술공동연구소, 1992. 6.
- 4) Michio Umeda, "Recognition of Multi-font Printed Chinese Characters", CH1801-0/82/0000/0793 \$00.75 IEEE, pp.793-796, 1982.
- 5) H.Yamada, T.Saito, K.Yamamoto, "Line Density Equalization - A Nonlinear Normalization for Correlation Method", *Trans. IECE of JAPAN*, Vol.J67-D, No.11, pp.1379-1383, 1984.11.
- 6) 황승욱, "Pipeline 구조를 이용한 인쇄체 한글문자 인식시스템의 설계및 구현", 1992
- 7) 酒卷久 et. al., "認識速度 70字/秒の日本語 OCR,専用 LSIと DSPのパイプライン處理で高速化", キヤノン ソフトウェア戰略本部, NIKKEI ELECTRONICS, pp.195-201,1990.7.
- 8) 弘具, 大町眞一郎, 木村 正行, 勝山 裕, "高速高精度知的認識システム SEIUN", 電子情報通信學會論文誌 Vol.J76-D-II No.3 pp.474-484, 1993. 3.
- 9) Shin'ichiro OHMACHI, Hiroto ASO and Masayuki KIMURA, "A Systolic Array for Preprocessings for Fast Characrer Recognition", 電子情報通信學會論文誌 Vol.J 73-D-II No.2 Februar 1990.
- 10) MOTOROLA, "MC68000 COURSE NOTES".
- 11) ALTERA, "DATA BOOK", 1992.

