論文94-31B-7-10

텍스트와 그래픽으로 구성된 혼합문서 인식에 관한 연구

(A Study on the Recognition of Mixed Documents Consisting of Texts and Graphic Images)

咸永國**, 金仁權*, 丁鴻奎*, 朴來弘**, 李昌範***, 金庠仲***, 尹炳楠***

(Young Kug Ham, In Kwon Kim, Hong Kyu Chung, Rae-Hong Park, Chang Bum Lee, Sang Joong Kim, and Byeong Nam Yoon)

要約

본 논문에서는 그래픽 이미지를 포함한 인쇄체 한글과 영숫자로 구성된 혼합문서 인식에 대한 효율적인 알고리듬을 제안하였다.

전처리 단계에서는 먼저 문서의 기운 여부에 따라, 기울어졌을 경우 Hough 변환을 이용하여 기울어진 각도를 알아내고 문서를 보정하였다. 그리고 체인코드로 표현한 연결화소를 이용하여 그래픽부분과 문자부분을 분리한 후 수직, 수평투영을 이용하여 개별문자를 추출하였다. 인식단에서는 먼저 한글과 영숫자를 분리하고 각각에 대해 효율적인 특징을 사용하여 계층적으로 인식하였다.

본 논문에서는 혼합문서 인식을 위한 효과적인 알고리듬을 제안하였으며, 컴퓨터 시뮬레이션을 통해 그 성능을 보였다.

Abstract

In this paper, an efficient algorithm is proposed which recognizes the mixed document consisting of the printed Korean/alphanumeric texts and graphic images.

In the preprocessing step, an input document is aligned, if necessary, by rotating it. We obtain the rotation angle using the Hough transform and align the input document horizontally. Then we separate graphic image parts from text parts by considering chain codes of connected components. We further separate each character using vertical and horizontal projections. In the recognition step, Korean and alphanumeric characters are classified and each of them is recognized hierarchically using several features.

In summary, an efficient recognition algorithm for mixed documents is proposed and its performance is demonstrated via computer simulations.

* 準會員, ** 正會員, 西江大學校 電子工學科
(Dept. of Electronic Eng., Sogang Univ.)
*** 正會員, 韓國電子通信研究所 通信處理研究部
(Communication Processing Dept. Elec. and Telecommunications Research Institute)

※ 본 연구는 한국전자통신연구소 연구비 지원으로 이루어졌음.
接受日字 : 1993年 2月 10日

Ⅰ. 서 론

인간의 시각 능력을 컴퓨터에 부여하려는 컴퓨터비전 분야의 발전은 컴퓨터 입력단의 사용자 편의 및 다양성 향상을 위하여 최근 많은 발전을 거듭하여 왔다. 그 분야중 하나인 패턴인식은 인간의 시각 능력 중 도형이나 기하학적인 물체의 인식, 문자인식에의 응용 등으로 발전하여, 컴퓨터와 인간의 인터페이스 기술로 발전하여 왔다.

문서의 자동 입력과 인식을 위해서 무엇보다도 먼저 문자 인식이 기본적으로 이루어져야 하는데 문자인식은 패턴인식 응용의 한 분야로 오래전부터 연구되어 왔으며, 지금까지 개개의 한글과 영문 인식에 대해서는 많은 연구가 이루어졌으나, 일반 문서나 한글, 영문 및 그래픽 등으로 이루어진 혼합문서에 대한 연구는 활발하지 못하였다. 혼합 문서는 일반적으로 그림 혹은 도표와 함께 한글, 영숫자, 기호 등이 포함된 것으로 정의할 수 있다. 이러한 문서를 인식하기 위해서는 문자영역과 그림영역을 분리하고, 분리된 문자영역으로부터 각 문자를 추출하여 인식하여야 한다.

한글 문자인식에 관한 연구는 1960년대 이후 현재까지 꾸준히 연구되어 왔으며, 최근에는 상당한 수준에까지 도달하고 있다.[1-4] 인식을 위하여 패턴을 분석하여 문자의 기본요소들을 찾아내고 이들의 구조분석에 의해 분리, 조합하는 방법으로 인식하는 구조적 문자인식 방법은 크게 나누어 세선화 과정을 거쳐 인식하는 방법과 세선화 과정을 거치지 않고 문자패턴의 형상이나 윤곽선에 의하여 인식하는 방법이 연구되었다.[4] 기존의 대부분의 문서인식은 영숫자 또는 한글로만 구성된 문서를 대상으로 하였다. 본 논문에서는 한글과 영숫자 그리고 그래픽으로 구성된 혼합문서 인식을 대상으로 하였으며, 출력단에서는 전처리 단에서 그래픽을 포함한 문서에 대해 전처리 단에서 분리된 그래픽 부분과 인식단에서 인식된 텍스트 부분을 결합하여 출력하는 알고리듬을 제안하였으며, 입력시 기울어진 문서에 대해 Hough 변환을 이용하여 기울어진 각도를 알아내고, 기울어진 문서에 대해 기존의 회전변환을 변형한 3단계 회전변환을 사용함으로써 회전에 따른 오차를 줄였다. 그리고 그래픽 추출시, 기존의 연결화소를 이용한 분류방법에 체인코드를 추가함으로써 효율적인 분리를 할 수 있었다. 인식단에서는 한글과 영숫자의 분리를 위해 특징점들의 개수와 부분투영 그리고 한글과 영숫자의 구조적 특징을 이용하였다. 특히 특징 추출시 문자를 세선화하기 전 흑백화소 변화수와 같은 특징을 추출하고, 다른 특징들은 세선화를 한 후, 추출함으로로써 세선화에 따른 특징변화를 흡수하려고 하였다. 이런 방법을 사용함으로써 기존의 문자인식이 한가지 방법을 채택함으로써 발생할 수 있는 오류를 줄일 수 있었다.

본 논문에서는 일반문서중 한글, 영숫자 그리고 그림으로 구성된 인쇄체 혼합문서의 인식을 목표로 하였다. 혼합문서가 입력되면 전처리 단계에서 먼저 입력 문서의 회전 여부를 판단하고, 만약 문서가 회전된 경우에는 Hough 변환을 이용하여 회전된 각을 추출하여, 회전각만큼 회전시켜 문서를 보정하였다. 그리고 체인코드로 표현한 연결화소를 이용하여 문자부분과 그래픽부분을 먼저 분리하고, 분리된 문자부분에서 개별문자를 추출하여 인식단으로 넘겨준다. 인식단에서는 개별문자에 대해 부분 투영과 위치 정보 및 거리 특징을 이용하여 한글과 영숫자를 분리한 후, 한글인식은 먼저 모음의 위치에 따라 수직모음과 아래 수평모음, 중간 수평모음 등 3가지로 나눈다. 그리고 각 모음 부류에 대해 자음을 인식하는데, 이때 사용된 특징은 세선화에 의한 끝점과 분기점의 개수 및 위치, 부분 투영, 거리특징이다. 영숫자 인식은 먼저 세선화 특징인 끝점과 분기점의 개수를 이용하여 대분류하고 각 부류에 대해 한글에서 사용한 특징을 이용하여 인식하였다. 이와 같이 한글 및 영숫자를 인식한 후 인식결과를 출력단으로 넘겨준다. 출력단에서는 전처리 단계에서 추출한 문자사이의 간격 등과 그래픽부분을 함께 출력하였다.

본 논문의 구성은 Ⅱ장에서 전처리 단계에 대해 서술하였으며, Ⅲ장에서는 인쇄체 혼합문서 인식 전반에 대해 서술하였다. Ⅳ장에서는 본 연구의 실험결과를 보이고 이에 대해 분석하였으며, 마지막 Ⅴ장에서 결론을 맺었다.

Ⅱ. 문서인식 시스템의 전처리 과정

문서인식 시스템에서 요구되는 전처리 과정은 입력된 문서 영상의 기울기 보정과 그림영역 분리 과정, 그리고 문자영역에서의 개별문자 추출과정으로 나눌 수 있다. 각각의 경우 인식된 문서의 출력을 위하여 단어와 단어사이의 공백, 새로운 줄의 시작 등과 같은 layout 정보도 함께 추출한다.

1. 문서 취득

본 실험에서는 IBM-PC 486의 MS-window상에서 UC-630 스캐너를 입력 장치로 하여 Photo-Styler 소프트웨어로 문서를 취득하여 TIF 화일로 저장하였으며, 문서의 전처리 과정과 인식 과정은 IBM-PC 486상에서 C언어로 알고리듬을 구현하였

다. 입력영상은 그레이레벨로 취득하였으며, 이는 인식된 문서의 출력과정에서 그림영역의 영상정보를 보존하기 위해서이다.

2. 문서 기울기 보정

스캐너를 통해 문서를 취득할 때 문서가 기울어져 취득되는 경우가 있다. 본 논문에서는 입력 문서영상으로부터 문서의 기울어진 정도를 자동적으로 추출하여 그 각도만큼 입력문서를 반대로 회전하여 기울어진 문서를 보정하는 알고리듬을 구현하였다. 이렇게 함으로써 문서의 회전으로 인한 오인식을 줄일 수 있다.

1) 기존의 방법

기존의 기울기 보정방법으로 Postl [5] 이 제안한 이차원 후리에 (Fourier) 변환을 이용한 방법이 있다. 이 방법은 후리에 변환을 통해 전력 스펙트럼 (spectrum)을 계산한 후, 임의의 방향으로의 투영값을 구한 후, 가장 큰 투영값을 갖는 각도를 기울어진 각도로 정한다. 이 방법은 후리에 변환하는 과정에서 시간이 많이 걸리는 단점이 있다. Nakano 등 [6] 은 Hough 변환 [7] 을 적용하여 문서의 기울기를 구하였다. 그러나 이 방법은 각 문자의 분리를 통해 문자에 사각형을 씌우고 사각형의 밑면을 문자의 가상 기준 라인 (virtual base line)으로 정의하고 이에 대하여 Hough 변환으로 문서의 기울어진 각도를 구하는 방법이다. 이 방법은 이를 위해서 문자분리가 선행되어야 하는 단점이 있다. Shyu 등 [8] 은 Wong 등 [9] 이 제안한 CRLA (Constrained Run Length Algorithm)을 사용하여 한자에 대해 기울기를 보정하였다. 이 방법은 문서에 해당하는 방법이 아닌 하나의 문자열에 대해서 기울기를 보정하는 방법이다. Akiyama와 Hagita [10] 는 세로로 정렬되어 있는 문서에 대해서 수평 projection profile간의 phase shift를 정의하고 이로부터 문서의 기울기를 구하였다.

2) 제안한 기울기 보정 알고리듬

본 논문에서는 도표, 그림, 문서 등이 혼합되어 있는 문서가 기울어져 입력되었을 때 기울기를 보정하는 알고리듬을 제안하였다. 제안한 보정 알고리듬은 먼저 CRLA를 적용한 후 그 영상의 에지에 대해 Hough 변환하여 기울어진 각도를 찾는 방법이다.

(1) CRLA [9]

CRLA는 같은 형태의 데이타를 포함한 패턴들을 모으기 위한 일종의 block segmentation 방법으로 다음과 같은 방법을 통해 수행된다. 백화소를 '0', 흑화소를 '1'이라고 정의하여 binary 입력 sequence를 변환한다. 이 알고리듬은 문자들 사이의 간격이 문자열 간격보다 작은 점을 이용해서 문자들을 한 부분으로 블럭화하는 효과를 갖는다. 기울기 보정을 위한 실험에서는 최적 임계치를 주어 영상을 이진화한 후 수평방향의 CRLA 결과와 수직 방향의 CRLA 결과의 OR연산으로 구하였다.

(2) 에지 처리

CRLA한 후 윤곽선을 구하기 위해 에지를 추출하였다. 에지 추출방법은 기존의 Sobel 연산자를 적용하여 추출하였다. 이미 CRLA를 거쳐 영상이 흑화소 또는 백화소로 분류되기 때문에 에지는 잘 찾아진다.

(3) Hough 변환

Hough 변환은 처리된 에지 영상의 각 점에 대해 주어진 변환식을 이용하여 파라미터 값을 계산한 후, 그 값에 대해 파라미터 영역에 대응되는 셀 (cell)들의 값을 하나씩 증가시켜, 누적영역에서 누적된 값이 가장 큰 셀의 파라미터 값을 기울기로 선택하는 기법이다.

Duda와 Hart는 직선의 파라미터 영역에서 기울기가 무한대인 직선의 표현을 위하여 Hough 변환식을 (1)식과 같이 제안하였다. [7] ρ_n은 주어진 직선과 원점 사이의 수직거리를 나타내며, θ_m은 원점으로부터 그 직선에 내린 수직선과 원점이 이루는 각도를 나타낸다.

$$\rho_n = x_i \cos\theta_m + y_i \sin\theta_m \tag{1}$$

영상 영역의 좌표 (x_i, y_i)는 파라미터 영역을 이산화시킨 좌표 (ρ_n, θ_m)으로 매핑 (mapping)된다. 본 논문에서는 문서의 회전된 정도가 -10 ~ 10도 내에서 기울어졌다고 가정하고 파라미터 영역의 범위를 제한하여 필요한 메모리를 줄였다. 이렇게 함으로써 Hough 변환시 걸리는 시간이 훨씬 단축되었다. 또한 최종 파라메타값을 Hough 셀중 가장 큰 값으로 결정하였으며 가장 큰 값이 여러 개인 경우 이들의 평균을 취하여 선택하였다.

(4) 회전 변환

최종적인 기울기 보정은 식 (2)와 같은 회전 행렬식을 통해서 변환할 수 있다. θ 는 회전할 각도이며, 원래 기울어진 입력 영상의 좌표 (X, Y)는 보정될 새로운 좌표 (X′, Y′)로 변환된다.

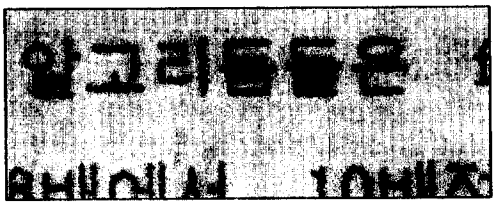
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2}$$

그러나 실제적으로 (2)식의 회전식을 이용했을 때 이산화된 영상정보가 회전시 양자화문제로 인하여 많은 화소들이 깨지는 경우를 볼 수 있었다. 이를 보완하기 위해 본 논문에서는 Tanaka 등 [11] 이 제안한

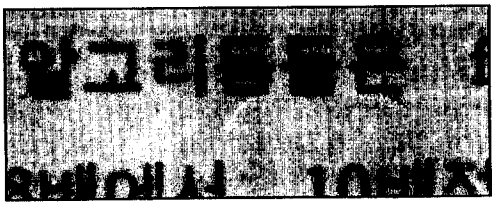
skew 변환을 이용하여 회전식을 구현하였다. 이 방법은 회전 행렬식을 분해하여 skew 변환으로만 이루어진 회전행렬을 고려한 것으로 식 (3)에서와 같이 회전 행렬식을 3단계의 skew 변환으로 분해하여 3단계 회전을 수행한다.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & -\tan\frac{\theta}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \sin\theta & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan\frac{\theta}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (3)$$

이 방법을 사용함으로써 양자화 문제가 줄어들며 계산속도도 개선된다. 특히 (2)식은 회전 행렬 매핑 과정을 거치지만 이 방법에서는 (3)식과 같이 단순히 영상을 skew변환을 통해 임의의 한 축은 고정시키고 다른 한 축만 위치 이동하기 때문에 회전시 문자의 깨짐이 적은 것이 큰 특징이다. 이 알고리듬의 우수성을 판단하기 위해 (2)식으로 구현하여 회전 보정된 영상의 일부분과 (3)식으로 구현하여 회전 보정된 영상의 일부분을 확대하여 그림 1에 나타내었다. 결과에서 보듯이 (3)식에 의한 보정이 문자 경계부분에서 양자화 오차가 적게 나타났다.



(a)

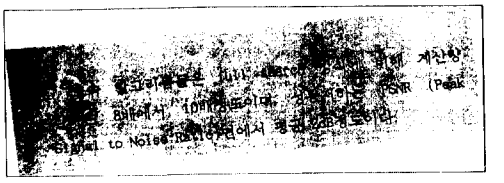


(b)

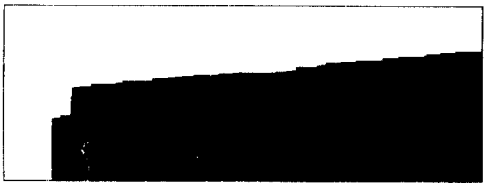
그림 1. 회전 보상한 영상
(a)(2)식에 의한 회전 (b)(3)식에 의한 회전
Fig. 1. Skew-normalized image
(a)rotation by eq. (2), (b)rotation by eq. (3).

(5) 기울기 보정 실험 결과

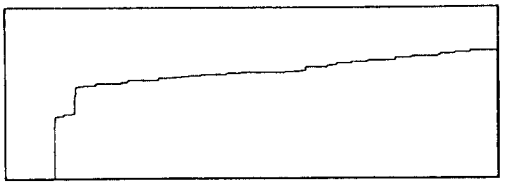
전처리 과정중 기울어진 여러가지 실험영상의 기울기 보정 결과는 표 1과 같으며 그 중 한 영상에 대해 실험한 결과를 그림 2에 보였다.



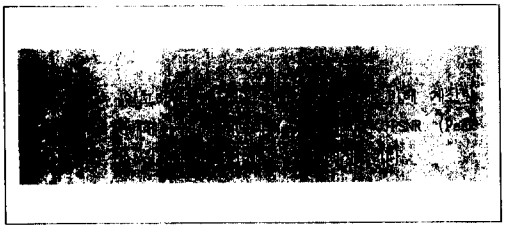
(a)



(b)



(c)



(d)

그림 2. 기울기 보정 (a)입력 영상 (b)수평, 수직 방향 CRLA 결과 (c)Sobel 연산자에 의한 edge (d)기울기 보정한 영상
Fig. 2. Skew normalization (a)input image, (b)CRLA results along the horizontal and vertical directions, (c)edge detection using the Sobel operator, (d)skew-normalized image.

보정 결과 약간의 오차가 있었으나, 인식에 영향을

주지않았다. 첫번째 과정은 수평방향으로 $Y \cdot \tan(\theta/2)$만큼 이동하는 단계이며, 두번째 과정은 수직방향으로 $X_1 \cdot \sin\theta$ 만큼 이동시키는 단계이다. 세번째 단계는 다시 두번째 단계의 결과를 수평방향으로 $Y_2 \cdot \tan(\theta/2)$만큼 이동하는 단계이다. 실험결과 기울어진 영상을 양자화 오차를 최소로 하면서 보정함을 볼 수 있었다. 여기서 (X_1, Y_1) 및 (X_2, Y_2)는 각각 첫번째 단계와 두번째 단계의 결과이다.

표 1. 기울기 보정의 결과
Table 1. Result of skew normalization.

기운 각도	보정된 각도
3	3.00
-3	-3.20
5	5.00
-5	-5.00
7	7.12

3. 그림영역 분리

문서 영상으로부터 그림영역을 분리하는 기존의 연구로는 투영값에 의한 영역 분리 방법[9][12,13], 런랭스를 이용한 블럭화 방법[14,15], 그리고 연결 화소에 의한 영역분리 방법[16] 등이 있다. 투영에 의한 방법은 간단하고 계산시간이 빠르지만 투영상에서 문자영역과 겹치는 그림영역이나, 도표내의 글자들은 분리할 수 없는 단점이 있다. 런랭스에 의한 방법은 모든 영상에 대해 블럭화하는 과정이 필요하기 때문에 많은 수행시간이 필요하며 또한 도표내의 글자를 분리하기 어려운 단점이 있다. 연결화소에 의한 방법은 도표내의 글자도 분리할 수 있지만, 연결화소를 찾기 위해 수행 시간이 오래 걸리는 단점이 있다. 본 논문에서는 그림영역을 분리하기 위해 경계선의 체인코드로 표현한 연결화소의 특징을 이용하였다. 문자영역의 경우 여러 개의 개별 문자들의 나열로 구성되어 하나의 줄을 이루고 여러 줄이 모여 하나의 문단을 이루고 있는 반면, 그림은 대부분 하나의 단일 영역으로 주어지며, 문자보다 높이나 크기, 그리고 차지하는 영역이 아주 큰 특징을 갖고 있다. 따라서 이러한 특징을 이용하여 그림 영역으로 분리한다.

1) 문서영상 이치화와 문자높이 결정

실험에 사용한 영상이 그레이레벨 밝기값을 갖는 영상이기 때문에 먼저 이치화 과정을 거쳐야 한다. 밝기값의 분포를 이용한 thresholding으로 쉽게 이치화할 수 있다. 문서를 이치화한 후에 문자의 높이를 결정하는데 문자의 높이값은 문자영역과 그림영역으로의 분리를 위한 기준값으로 사용되는데 수평방향으로의 투영값을 이용하여 문자의 높이값을 구하였다. 이 때 투영값이 존재하는 영역의 크기가 문서의 세로크기에 비해 너무 큰 영역은 계산에서 제외시켰다.

2) 특징 추출 및 그림영역 분리

그림영역 추출을 위한 특징 추출은 8방향 체인코드를 이용하였다. 영역에 대한 체인코드는 코드의 진행방향에 대해, 즉 새로운 화소로의 위치 이동에 대하여 좌측에서 시계 방향으로 다음 화소를 찾아 코드를 구한다. 따라서 흑화소 영역에 대한 경계부분의 체인코드를 구하는데 이로부터 가로크기, 세로크기, 가로크기와 세로크기의 비, 체인코드의 길이와 체인코드내의 흑백화소수의 5가지 특징을 추출하고 이 특징을 문자의 높이값과 비교하여 그림영역으로 분리한다.

그림영역 분리는 두 가지 경우에 대해 이루어진다. 먼저 체인 코드로 이루어지는 경계내에 흑화소가 대부분을 차지하면 경계내의 모든 영역을 그림영역으로 분리하며, 흑화소에 대해 백화소가 상당수 있게 되면 도표나 순서도로 간주하여 체인 코드의 시작점으로부터 연결화소만을 그림영역으로 분리한다. 따라서 도표내의 글자에 대해서도 쉽게 분리해낼 수 있다. 이러한 체인코드값에 의한 크기 비교방법으로 그림영역을 분리할 경우 투영값에 의해서는 분리할 수 없는 그림영역이나 도표부분을 분리해낼 수 있다. 또한 체인코드에 의한 방법은 연결화소에 의한 분리 방법에 비해 그림영역으로 분리하기 위한 판단을 영역의 경계에 대한 체인코드만으로 판단하기 때문에 연결 화소를 찾기 위한 시간을 줄일 수 있다.

크기값으로 그림영역을 분리할 경우 그림영역의 자그마한 부분들이나 끊어진 부분들을 분리하지 못하는 경우가 있으므로 남겨진 그림부분을 다시 그림영역으로 재분리하는 과정이 필요하다. 재분리 과정은 남겨진 부분들을 문자영역과 분리하기 위해 CRLA 블럭화 방법을 이용하였다. 불규칙적으로 남겨진 자그마한 그림영역들은 큰 블럭을 형성하지 못하며 잡음처럼 남겨지게 된다. 이러한 블럭영상에 대해 다시 체인코드를 구하여 앞서 구한 특징값들을 이용하여 이번에는 아주 작거나 코드 길이가 작은 영역을 다시 그림영역으로 재분리한다.

4. 개별문자 분리

기존의 방법중에서 개별문자의 분리는 간단하고 빠른 수행시간을 갖는 투영값에 의한 분리 알고리듬을 사용하였다. 개별 문자를 분리하기 위해서는 bimodal 히스토그램에서 두개의 peak를 효과적으로 분리할 수 있는 문턱값을 찾아 이치화하여야 한다. 실

험에서는 히스토그램 분포를 이차함수로 근사화한 후 이차함수가 갖는 최소점의 밝기값을 문턱값으로 선택하였다.[17] 즉, 히스토그램에서 빛의 밝기값을 변수 x라 하고, 각 밝기값에서의 히스토그램값을 함수 f(x)라 하면 $f(x) = ax^2 + bx + c$로 근사화한 다음 f(x)을 x에 대하여 미분하여 최소가 되는 x값 -b/2a을 문턱값으로 선택하였다. 그러나 한글은 하나의 글자가 여러 개의 자소로 구성되어 있기 때문에 투영값에 의해 분리할 경우 각각의 자소들이 개별적으로 분리되는 경우가 발생한다. 따라서 분리된 문자를 본래의 문자로 합쳐주기 위한 과정이 필요하다. 실험에서는 문자의 크기를 이용하여 분리된 글자를 합쳐주었다. 그러나 크기만으로 글자를 병합시키면 영문자의 경우 두개의 영문자가 하나의 개별문자로 분리 추출되는 경우가 있으므로 크기외에 투영상에서 두 영역을 합쳐주기 위해서는 다른 특징값이 필요하게 된다. 대부분의 분할 문자들이 'ㅏ', 'ㅑ', 'ㅓ', 'ㅕ', 'ㅣ', 'ㅐ', 'ㅔ', 'ㅒ', 'ㅖ'에 의한 경우이기 때문에 영역의 가로크기와 세로크기의 비와 세로 방향으로의 투영값을 이용하였다. 또한, 영문자에서의 i나 j, 그리고 l에 의한 특징값과 비교하기 위하여 앞서 분리된 글자와의 연관성을 사용하였다. 즉 'ㅣ'모음류에 의해 분할되는 한글의 경우에는 앞서의 분할문자 아래부분이 비어있는 대신 영문자의 경우에는 이전의 문자영역이 아래부분에서 일정한 영역을 차지하고 있다는 특성을 이용하게 된다. 병합의 기준이 되는 글자크기는 다음과 같이 구하였다. 먼저 수평방향의 투영값으로부터 문자열을 분리하고 각각의 문자열에 대해 수직방향의 투영값을 구한 후 수직방향 크기와 수평방향의 크기가 비슷한 투영영역에 대하여 평균을 취하여 문자의 크기를 결정하였다.

Ⅲ. 혼합문서 인식

본 장에서는 한글과 영숫자로 이루어진 혼합문서 인식을 위해 사용한 특징들과 한글의 구조적 특징에 대해 서술하였으며, 또한 혼합문서 인식을 위한 제안한 인식알고리듬에 대해 서술하였다.

1. 특징추출

특징추출 단계에서는 전처리 과정에서 넘겨 받은 개별문자로부터 인식에 필요한 문자의 특징을 추출한다. 본 논문에서 사용한 특징은 세선화에 의한 특징인 끝점과 분기점의 개수 및 위치와 흑백화소의 변화수, 부분적인 수직, 수평투영, 문자까지의 거리특징 등을 사용하였는데 이를 자세히 서술하면 다음과 같다.

1) 변환 특징

입력 패턴의 전체 데이타로부터 특징을 추출하는 것으로 문자의 이동이나 회전과 같은 문자 전체의 변형에 영향을 받지않는 특징을 얻으며, 특징 벡터의 차원을 줄일 수 있다. 본 논문에서는 부분투영을 사용함으로써 문자의 부류를 나눔에 있어 필요한 부분에 대해서만 투영을 하여 전체 투영을 함으로써 발생할 수 있는 불필요한 계산량을 줄였다.[18] 이 특징은 흑백화소의 변화수와 함께 모음 분류 단계에서 중요한 역할을 한다.

2) 통계적 특징

흑백 화소의 변화수, 문자의 흑화소까지의 거리 등이 이에 속하며, 이들 특징들은 왜곡에는 유연하나 문자체 변형에 대하여 영향을 받는 단점이 있다. 그러나 구현이 간단하며 계산시간이 적게 걸리는 장점이 있다.

(1) 흑백 화소의 변화수

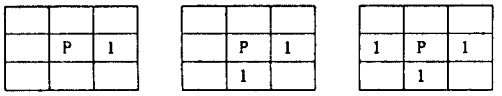
전처리 과정에서 넘겨 받은 문자 각각에 대해 여러 방향에서의 흑백 화소의 변화 수를 측정하여 문자들을 여러 부류로 나누는데 특징으로 이용하였으며, 문자의 중앙열에서 수직으로의 흑백화소의 변화에 중요성을 두었다.[19-20] 이 특징들은 특히 영숫자 인식에서 세 개의 부류로 나눌 때 효율적인 장점이 있다.

(2) 문자의 흑화소까지의 거리

문자의 흑화소까지의 거리는 흑백 화소의 변화수나 부분적인 투영과 같이 입력 문자들을 유사한 문자끼리 그룹을 만들 수 있는 특징은 되지 못하지만 각 그룹내에서 문자를 인식하는데 중요한 특징으로 작용한다.[21] 이 방법을 적용시키기 위해 입력 문자 각각에 대해 문자가 차지하는 부분에 대해 최소한의 직사각형을 만들고 직사각형을 기준으로 필요한 각 부분에 대해 직사각형으로부터 문자의 흑화소까지의 거리를 특징으로 택하였다.

(3) 세선화 특징

세선화는 입력 패턴의 골격 구조를 추출하기 위해 외곽점을 제거함으로써 수행된다. 세선화된 패턴은 연결성과 원 패턴의 형상을 유지해야 한다.



(a) (b) (c)

그림 3. 교차수에 의한 특징 (a)끝점 (b)굴곡점 (c)분기점

Fig. 3. Cross point features (a)end point, (b)curve point, (c)branch point.

세선화를 수행하는 알고리듬은 여러가지가 있으나, 본 논문에서는 수행시간이 빠르며, 비교적 좋은 성능을 나타내는 Pavlidis의 방법을 사용하였다.[22] 이진화 영상으로부터 직접 세선화하는 알고리듬 방법은 경계 화소부터 지워나가고 최종적으로 한 화소 두께의 세선화된 영상을 얻는다. 세선화된 문자로부터 자획을 추출하기 위해서는 교차수에 의한 특징점과 체인 코드에 의한 방향 변화를 고려하여 기본 성분을 찾아낸다. 특징점은 획의 상대적인 위치나 연결관계를 나타내는 점으로서 그림 3과 같이 끝점, 굴곡점, 분기점 등으로 나눌 수 있다.

2. 혼합문서 인식

전처리 과정을 통해 인식단으로 넘겨온 개별문자에는 한글과 영숫자가 함께 존재하며, 이들의 특징이 다르므로 먼저 이들을 분리한 후 각각에 대해 인식을 행한다.

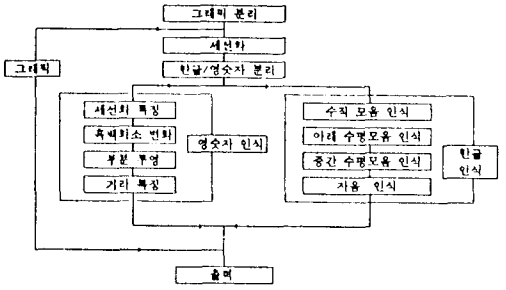


그림 4. 혼합문서 인식방법 흐름도

Fig. 4. Flowchart of a proposed hierarchical character recognition method for the mixed document.

그런데 문서에 따라 영숫자의 크기가 한글의 크기보다 작기 때문에 영숫자 두개의 문자가 함께 인식단으로 넘겨지는 경우가 발생하므로 개별문자의 특징을 추출하기 전에 한글 및 영숫자 또는 영숫자 두개가 넘겨졌는지를 먼저 검사한 후, 각각의 인식단으로 넘겨준다. 인식단에서는 먼저 한글과 영숫자를 분리하고, 한글 인식단에서는 모음의 위치에 따라 대분류를 행하고, 분류된 각 부류에 대해 끝점, 분기점, 흑백화소의 변화수, 흑화소까지의 거리와 같은 특징을 이용하여 계층적으로 인식하였다. 영숫자의 경우도 한글에서 사용한 특징을 이용하여 인식한 후 출력단에서는 인식된 한글과 영숫자를 출력하였다. 출력부분에서는 전처리 단계에서 추출한 그래픽부분과 인식결과를 결합하여 출력하였다. 그림 4에 혼합문서 인식을 위한 전체 흐름도를 보였다.

1) 한글과 영숫자 분리

본 논문에서 한글 문서는 도표와 함께 한글, 영숫자, 기호 등이 포함되어 있는 것으로 정의한다. 그러므로 그림부분 또는 도표부분을 먼저 추출하고, 문자부분에 대해서도 한글과 영숫자의 구조나 특징이 다르므로 분리해야 한다. 본 논문에서는 전처리 단계에서 인식단으로 넘겨진 개별문자에 대해 그림 5와 같은 수직투영과 거리특징을 이용하여 한글과 영숫자를 분리하여 각각의 인식단으로 넘겨주었다. 즉 개별문자의 중간부분에서 수직으로 투영하여 두개의 문자부분으로 분리된 문자에 대해 수평으로의 투영값이 두 부분으로 분리되지 않고 밑부분에서의 거리 특징비가 작으면 영숫자가 입력된 것으로 판단하고 두 부분에 대해 각각을 영숫자 인식단으로 넘겨준다.

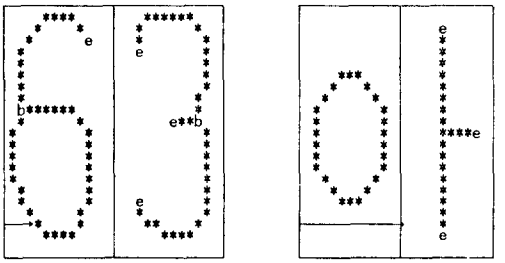


그림 5. 문자 분리

Fig. 5. Character separation.

그림 5에서 'e'는 세선화에 의한 특징인 끝점이며, 'b'는 분기점을 나타낸다. 이는 한글의 구조적 특성상 자음은 모음의 중앙에 위치한다는 것과 영숫자의 경우는 밑 라인을 기준으로 위치한다는 특징을 이용하였다. 그리고 영숫자의 크기와 한글의 크기가 같은 문자인 경우는 수직모음의 존재를 확인한 후, 첫번째로 세선화에 따른 특징점의 개수를 측정하고 개수가 2개 이하인 경우는 영숫자로 분리하고, 다음 단계로 문자의 양끝에서 수직으로 부분투영값을 구한 후, 투영값이 클 경우는 영숫자로 분리하였다. 마지막 단계에서는 자음의 위치가 모음의 중앙에 위치 한다는 한글의 특성을 이용하여 한글과 영숫자를 분리하였다. 수평모음이 존재하는 경우는 한글과 영숫자의 특성이 다르기 때문에 쉽게 분리할 수 있다.[23]

2) 한글의 구조

한글은 자음과 모음에 의해 각 자소 (초성, 중성, 종성)들의 조합에 의해서 이루어지며 초성 19자, 중성 21자, 종성 27자로 생성되는 받침이 없는 399자와 받침이 있는 글자 10,773자를 합하여 11,172자에 이른다.[24] 이와 같이 한글은 방대한 문자수와 형태

에 있어서 자소들의 배열이 2차원적으로 모아쓰기로 되어 있어 유사문자가 많고 자소접촉에 의해 분리가 쉽지 않아 인식에 있어서 문제가 되고 있다. 또한 하나의 문자는 2 ~ 7개의 자소로 이루어지며 각 자소는 1 ~ 4개의 획으로 이루어진다. 그러므로 획의 조합에 의하여 자소를 분리하여 문자를 인식할 수 있다. 한글의 자소의 조합에 의해 여러가지 방법으로 분류할 수 있으나 일반적으로 그림 6과 같이 크게 6가지 형태로 분류한다.

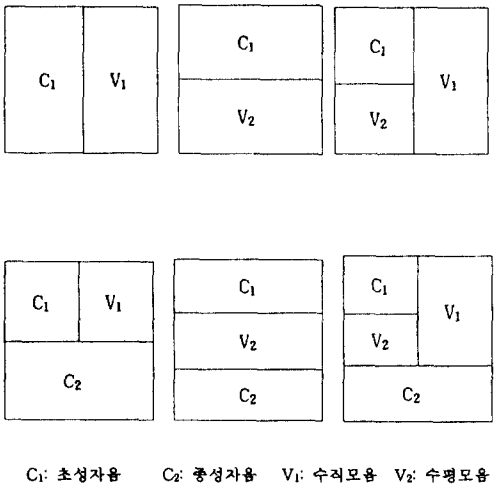


그림 6. 한글의 6가지 유형
Fig. 6. 6 Korean character types.

3) 한글의 모음 분류 및 인식

문서에 나타난 한글의 특징은 다음과 같이 요약할 수 있다.[3] 한글에서 종횡의 직선 선분의 빈도가 대단히 크며, 직선에는 긴 직선과 짧은 연결선으로 구분할 수 있고, 짧은 연결선은 모음의 경우 대개 직각으로 연결된다. 그리고 사선 방향의 빈도는 비교적 낮고 곡선은 ㅇ형에 국한된다. 낱자의 경우 4개 변두리에 모든 자모가 걸쳐있으며, 획의 굵기는 비교적 평탄하다. 그런데 획의 끝부분이 약간 굽은 경우가 명조체의 경우에 빈번히 나타난다.

한글은 글자 구성이 자모의 결합이며, 그 결합은 자모 2개부터 자모 7개까지 사용되어 결합관계가 비교적 복잡하다. 더우기 조합된 자소가 때로는 밀착하거나 결합되므로 인식측면에서 보면 자소 분리가 상당히 중요하다. 더우기 한자, 영숫자가 추가되면 인식 알고리듬은 더욱 복잡하게 된다.

한글의 모음은 수평모음과 수직모음으로 나눌 수 있는데, 수평모음 중에서 긴 수평모음은 크기 변화는 심하지 않으나 위치 변화가 있으며, 수직모음은 반대로 위치 변화는 심하지 않으나 크기 변화가 종성의 존재 여부에 따라 달라진다. 본 논문에서는 세선화를 행한 후 끝점과 분기점을 기준으로 문자를 인식하기 때문에 한글을 6가지로 분류하기보다는 부분적인 투영에 의하여 그림 7과 같이 수직모음, 밑부분의 수평모음, 중간부분의 수평모음 등 3가지의 부류로 나누어 각각을 인식하였다.

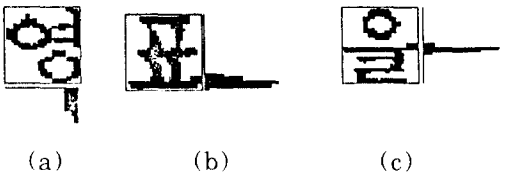


그림 7. 3가지 모음 예 (a)수직모음 (b)수평모음 (밑부분) (c)수평모음 (중간부분)
Fig. 7. 3 types of vowels (a)vertical vowel, (b)horizontal vowel (bottom), (c) horizontal vowel (center).

(1) 수직모음 인식

긴 수직모음은 모두 문자의 오른쪽 부분에 존재하므로 수직모음 추출시, 오른쪽의 부분 투영을 이용하여 수직모음의 존재여부를 추출하였다.

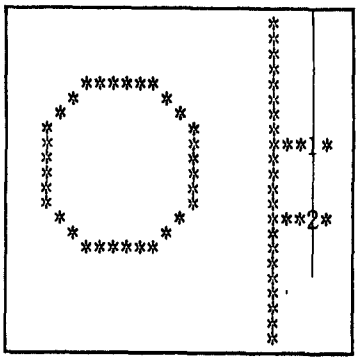


그림 8. 'ㅏ' 와 'ㅑ'모음 인식을 위한 흑백화소 변화수
Fig. 8. Cross point example for the 'ㅏ' and 'ㅑ' vowel recognition.

이때 종성 자음의 영향을 없애기 위해 종성 자음의 위치부분을 제외한 부분에 대해 수직 투영을 하였다. 이때 추출될 수 있는 수직모음은 'ㅏ', 'ㅑ', 'ㅓ', 'ㅕ', 'ㅐ', 'ㅒ', 'ㅔ', 'ㅖ' 'ㅘ', 'ㅙ', 'ㅚ',

'ㅝ', 'ㅞ', 'ㅟ', 'ㅢ' 등이다. 다음 단계에서는 문자의 우측 가장자리에서 수직방향으로의 거리특징과 흑백화소 변화수를 이용하여 'ㅏ' 모음의 부류와 'ㅣ' 모음의 부류로 나눈다. 'ㅏ' 모음의 부류에는 'ㅘ' 모음과 같은 복모음도 함께 추출되며, 이와 같은 복모음은 자음의 인식 과정에서 인식하였다. 또한 'ㅣ' 모음 부류에서도 복모음중 'ㅐ', 'ㅚ', 'ㅝ', 'ㅞ', 'ㅟ', 'ㅢ' 등의 모음은 자음 인식시에 함께 인식하였다. 그림 7(a)는 수직모음 인식을 위한 부분투영의 예를 보여주고 있다.

그림 8은 수직모음 중 'ㅏ' 모음과 'ㅑ' 모음을 인식하기 위한 흑백화소의 변화수를 보여주는데 문자의 왼쪽 끝부분에서 수직으로 문자의 그레이 레벨이 변하는 구간의 개수를 측정하여 'ㅑ' 모음을 인식하였다. 'ㅓ' 모음과 'ㅕ' 모음에 대해서도 같은 방법으로 인식하였다.

(2) 수평모음 인식

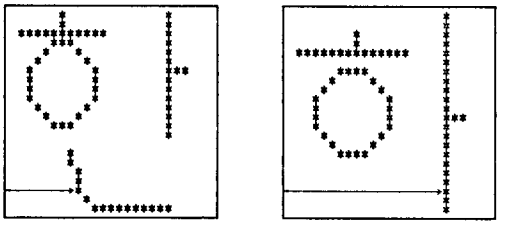
긴 수평모음의 추출은 두 단계로 나뉘어진다. 먼저 문자의 밑부분에 대한 수평모음을 추출하는데 그림 7(b)와 같이 '도'나 '표'와 같은 문자는 자음부분이 긴 수평모음으로 오인식되는 경우가 종종 발생하기 때문이다. 따라서 밑부분의 수평모음을 먼저 추출함으로써 이와 같은 모음인식의 오류를 막을 수 있었다. 밑부분의 수평모음으로는 'ㅗ', 'ㅛ', 'ㅡ' 등 세 가지의 수평모음만이 존재한다.

다음으로 문자의 중간부분에서 긴 수평모음을 추출하는데 앞부분에서 수직모음과 밑부분의 수평모음이 존재하지 않은 경우는 반드시 중간부분의 수평모음이 존재하기 때문에 마지막에 중간부분의 수평모음을 인식하였다. 그러나 중간부분의 수평모음 인식시에 초성이나 종성과의 접촉때문에 모음의 짧은 수직획 인식에 오류가 생기는 경우는 모음의 구조적 특징을 사용하여 오류를 없앴다. 중간부분의 수평모음으로는 'ㅗ', 'ㅛ', 'ㅡ', 'ㅜ', 'ㅠ' 등이 존재한다. 그림 7(c)는 중간 수평모음이 존재하는 문자를 인식하기 위한 수평투영값을 보여주고 있다.

입력문자에 대해 수평모음의 존재를 인지한 후 분기점의 위치 및 개수를 구한 후, 수평으로의 흑백화소 변화수를 구하여 수평모음을 인식한다. 분기점의 개수가 하나일 때는 'ㅗ' 또는 'ㅜ'모음으로 분리하고, 분기점 위치의 밑에서 수평으로 흑백화소의 변화수를 측정하여 흑백화소 변화수가 존재할 때에는 'ㅜ'모음으로, 변화수가 존재하지 않을 때에는 'ㅗ' 모음으로 인식한다. 이와 같은 방법으로 모음을 인식할 때 중성과 종성이 접촉되었을 경우 모음인식에 오류가 생길 수 있다.

4) 한글자음 인식

한글의 자음은 초성과 종성으로 나뉘어지는데 초성 19자, 종성 27자가 존재한다. 본 논문에서는 초성과 종성을 따로 인식하였으며, 초성은 모음의 부류에 따라 각각을 인식하였다. 자음인식에 있어 먼저 종성의 존재여부를 인지하였다. 이때는 한글과 영문 분리시에 사용하였던 거리특징을 이용하여 종성여부를 판단한 뒤, 각 모음 부류에 대해 종성이 존재할 때와 존재하지 않을 때를 분리하여 인식하였다. 이는 종성자음의 존재여부에 따라 자음의 크기 및 모양이 다르기 때문에 특징추출에 있어 약간의 오류가 발생할 수 있기 때문에 이를 줄이기 위함이다. 그림 9는 종성이 존재하는 문자와 존재하지 않는 문자에 대해 밑부분에서의 거리특징을 보여주고 있다.



(a) (b)

그림 9. 종성의 유뮤에 따른 거리특징 (a)종성이 있는 문자 (b)종성이 없는 문자

Fig. 9. Distance feature to test the existence of a last consonant (a)character with a last consonant, (b) character without a last consonant.

그림 9(a)는 종성이 존재하는 문자로 밑부분에서의 거리특징과 문자의 너비와의 비가 작으나, 그림 9(b)와 같이 종성이 존재하지 않는 문자는 이 값이 크다. 따라서 수직모음이 존재하는 문자에 대해 밑부분에서 문자의 너비와 거리특징과의 비를 이용하여 종성유무를 판단하였다.

자음인식시에 사용한 특징은 세선화함으로써 생기는 끝점과 분기점 등의 특징과 문자의 변형으로 인해 세선화에 따른 특징값이 달라질 수 있기 때문에 이를 보완하기 위해 부분투영과 흑백화소의 변화수 그리고 거리특징을 이용하여 인식하였다.

5) 영숫자 인식

전처리 단계에서 추출된 개별문자에 대해 인식단의 첫단계로 한글과 영숫자를 분리한 후, 영숫자로 분리된 문자에 대해 끝점, 분기점의 개수 및 위치를 이용

하여 인식하였다.[23] 끝점 및 분기점의 개수를 이용하여 5개의 부류로 나누면 그림 10과 같이 나눌 수 있다. 그림 10과 같이 끝점과 분기점의 개수는 문서의 상태에 따라 변할 수 있다. 이와 같은 변화를 흡수하기 위해 본 논문에서는 수직, 수평투영 및 흑백화소 변화수를 이용하였다.

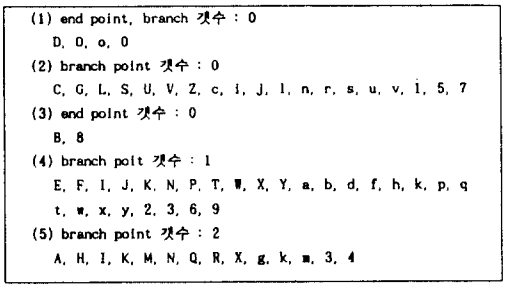


그림 10. 끝점 및 분기점에 의한 분류
Fig. 10. Classification by end points and branch points.

6) 그래픽과 인식결과 출력

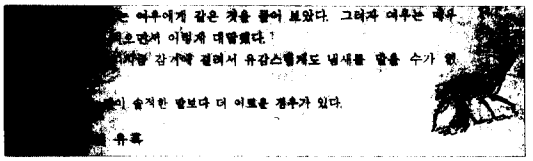
인식단에서 인식한 결과는 아스키코드와 상용조합형 코드로 만들어진다. 그 결과를 그래픽으로 나타내기 위해 문자코드를 화소패턴으로 변환하는 과정을 거친다.[20] 이는 32×24 크기의 한글 폰트와 16×24 크기의 영문 폰트로 입력되는 코드에 해당되는 문자를 생성하여 그래픽 디스플레이 장치에 그려주는 것이다. 본 연구에서는 전처리에서 제거된 그래픽 부분과 인식한 결과를 동시에 그래픽으로 보이기 위해 전처리 단계에서 대략적인 글자의 간격과 글자의 시작위치의 정보를 저장해두었다가 사용하였다.

Ⅳ. 실험결과 및 분석

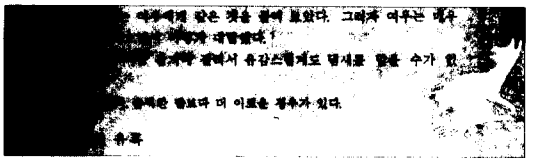
본 논문에서는 그래픽을 포함한 한글과 영숫자로 구성된 혼합문서 인식 알고리듬에 대해 연구하였다. 전처리과정에서는 Hough 변환을 이용하여 회전된 문서를 보정하였으며, 체인코드로 표현한 연결화소를 이용하여 혼합문서에서 그래픽 부분을 분리하였다. 그리고 분리된 문자열에 대해 수직, 수평투영을 이용하여 개별문자를 추출하였다. 인식과정에서는 한글과 영숫자의 특징이 다르기 때문에 개별문자에 대해 수직투영과 거리특징을 이용하여 한글과 영숫자를 분리한 후 따로 인식하였다. 실험은 workstation (40Mips)상에서 C언어로 구현하였다.

그림 11에 300 dpi (dot per inch) 해상도로 취득한 한글과 한자로 구성된 그레이레벨 혼합문서에 대해 본 논문에서 제안한 개별문자 추출 알고리듬을 적용한 결과를 보였다. 300 dpi로 문서영상을 취득할 경우 각 문자의 크기가 200 dpi에 비해 커져 한 문자의 크기가 40 40였다. 그림에서 보듯이 본 논문에서 제안한 개별문자 알고리듬을 한글과 한자로 구성된 혼합문서에 적용한 결과 정확하게 분리하였다.

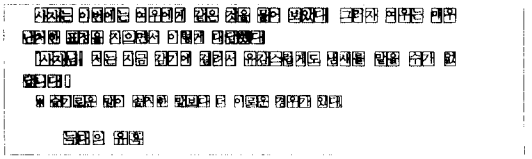
그림 11. 한글, 한자 혼합문서의 개별문자 분리
Fig. 11. Individual character separation of a mixed document consisting of the Korean and Chinese characters.



(a)



(b)

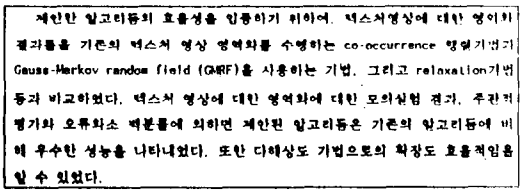


(c)

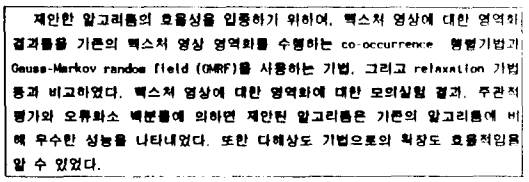
그림 12. 그림을 포함한 문서의 개별문자 추출 (a)그림을 포함한 혼합문서 (b)그림부분 추출 (c)개별문자 분리
Fig. 12. Individual character separation of a mixed document containing a pic-ture (a)document containing a pic-ture, (b)extraction of the picture part, (c)individual character separation.

그림 12(a)와 12(b)에 투영상에서 문자열과 그림이 겹치는 문서에 대해 본 논문에서 제안한 알고리듬을 이용하여 그림부분만을 정확히 추출한 결과를 보였다. 기존의 투영방법으로는 그림 12에서 그림부분만을 추출하기 어렵기 때문에 본 논문에서는 체인코드로 표현한 연결화소를 이용하여 그림부분을 추출하고 남는 문자열에 대해 그림 12(c)와 같이 개별문자를 추출하였다.

그림 13(a)는 문자열만으로 이루어진 혼합문서로 300 dpi의 해상도로 취득되었다. 문서의 크기는 600×1,284정도이며, 256 그레이레벨의 영상이다. 그림 13(b)는 혼합문서 1에 대한 인식결과를 보여주고 있다. 그림 13(a)에서 보듯이 한글에서 중성과 종성, 또는 초성, 중성 및 종성이 접촉되어 끝점과 분기점만을 이용하여 모음을 인식할 경우에는 오류를 보일 수 있다. 본 논문에서는 이와 같이 오류를 보일 수 있는 접촉된 모음의 경우, 세선화하기 전의 영상에서 투영과 거리특징을 이용하여 모음을 인식함으로써 오류를 없앨 수 있었다.


제안한 알고리듬의 효율성을 입증하기 위하여, 텍스처영상에 대한 영역화 결과를 기존의 텍스처 영상 영역화를 수행하는 co-occurrence 행렬기법과 Gauss-Markov random field (GMRF)를 사용하는 기법, 그리고 relaxation기법 등과 비교하였다. 텍스처 영상에 대한 영역화에 대한 모의실험 결과, 주관적 평가와 오류화소 백분률에 의하면 제안된 알고리듬은 기존의 알고리듬에 비해 우수한 성능을 나타내었다. 또한 다해상도 기법으로의 확장도 효율적임을 알 수 있었다.

(a)


제안한 알고리듬의 효율성을 입증하기 위하여, 텍스처 영상에 대한 영역화 결과를 기존의 텍스처 영상 영역화를 수행하는 co-occurrence 행렬기법과 Gauss-Markov random field (GMRF)를 사용하는 기법, 그리고 relaxation 기법 등과 비교하였다. 텍스처 영상에 대한 영역화에 대한 모의실험 결과, 주관적 평가와 오류화소 백분률에 의하면 제안된 알고리듬은 기존의 알고리듬에 비해 우수한 성능을 나타내었다. 또한 다해상도 기법으로의 확장도 효율적임을 알 수 있었다.

(b)

그림 13. 문서 인식 (a)입력문서 1 (b)인식결과
Fig. 13. Text recognition (a)input document 1, (b)recognition result.

그림 14는 그림 2의 보정한 문서에 대한 인식결과를 보여주고 있다. 본 혼합문서는 약 7도 정도 기울어져 입력되었다. 그리고 문서의 구성은 명조체의 한글, 영숫자로 이루어졌으며, 해상도는 300 dpi이다. 전처리 단계에서 기울어진 입력영상을 양자화 오차를 최소한으로 줄이면서 보정하였기 때문에 문자의 파손이나 기울어짐이 적었다. 따라서 인식단에서도 오인식이 없었다.

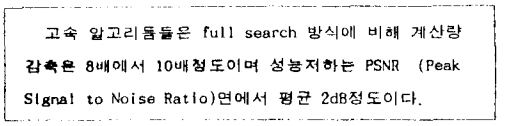
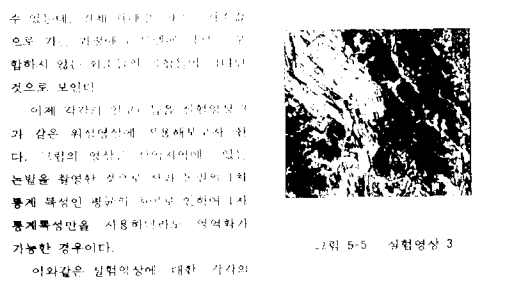
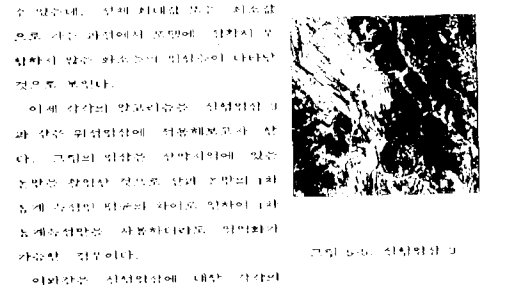

고속 알고리듬들은 full search 방식에 비해 계산량 감축은 8배에서 10배정도이며 성능저하는 PSNR (Peak Signal to Noise Ratio)면에서 평균 2dB정도이다.

그림 14. 그림 2(a) 입력영상에 대한 인식결과
Fig. 14. Recognition result of an input image in Fig. 2(a).

그림 15(a)는 그림을 포함한 256 그레이레벨의 200 dpi로 취득한 혼합문서이다. 문서의 크기는 644 1,067이며 한글과 숫자로 구성된 혼합문서이다. 그림 15(b)는 혼합문서 2에 대한 인식결과와 전처리 단계에서 분리한 그림부분을 결합하여 출력한 결과를 보여주고 있다.



(a)



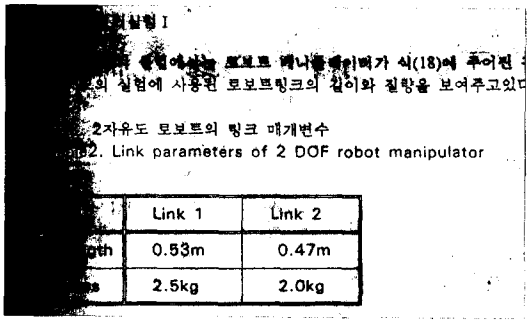
(b)

그림 15. 그림을 포함한 혼합문서 인식 (a)입력문서 2 (b)인식결과
Fig. 15. Recognition of a mixed document containing a picture (a)input docu-ment 2, (b)recognition result.

출력결과에서 문자 폰트는 32×24 크기의 한글 폰트와 16×24 크기의 영문 폰트를 사용하여 그래픽 디스

플레이 장치에 그렸다. 결과에서 보듯이 오인식이 없고 정확히 인식되었다. 이와 같이 200 dpi 해상도의 혼합문서일지라도 문서의 상태가 양호하여 파손된 문자가 없는 경우는 인식하기 쉽다. 그러나 200 dpi의 해상도로 실험영상을 취득할 경우에는 가끔 문서가 자소간의 접촉이 발생하거나 문자의 파손이 심하여 인식에 어려움이 있었다.

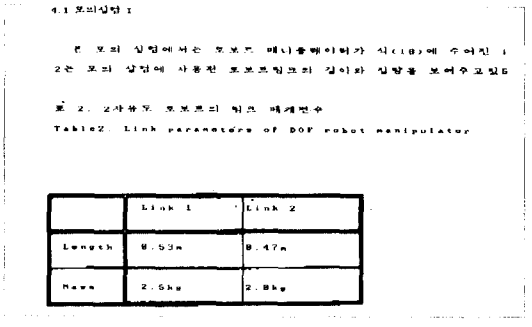
그림 16(a)는 표를 포함한 혼합문서로 폰트는 고딕체와 명조체로 이루어져 있으며, 300 dpi로 취득된 문서이다. 문서의 크기는 866 1,431이다. 이 문서의 특징은 영숫자의 크기가 한글의 크기와 유사하기 때문에 크기를 이용한 한글과 영숫자 분리는 어렵다. 이와 같이 영숫자의 크기가 한글과 유사한 경우에는 부분투영을 이용하여, 한글과 영숫자의 특징에 의하여 분리하였다. 그림 16(b)는 입력문서 3의 인식결과와 전처리 단계에서 추출한 표를 결합한 후, 본 연구에서 구현한 글자체를 이용하여 출력한 결과를 보여주고 있다.



2자유도 모보트의 링크 매개변수
2. Link parameters of 2 DOF robot manipulator

	Link 1	Link 2
th	0.53m	0.47m
s	2.5kg	2.0kg

(a)



(b)

그림 16. 표를 포함한 혼합문서의 인식
(a)입력문서 3 (b)인식결과

Fig. 16. Recognition of a mixed document containing a table (a)input document 3, (b)recognition result.

결과에서 보듯이 잡음의 영향으로 표부분을 정확히 추출하지는 못하였다. 본 문서는 영숫자의 크기가 한글의 크기와 비슷하기 때문에 한글과 영숫자 분리 단계에서 문자의 크기를 이용하여 분리하기 어렵다. 이와 같은 문서는 한글과 영숫자의 문자특징을 이용하여 분리하여야 가능한 문서이다. 개별문자 추출 결과에서 보듯이 영문자 'g'는 숫자 '9'와 비슷한 형태이며 이와 같이 유사한 특징을 가진 문자는 주변 상황을 고려하여 모두 옳게 인식하였다.

본 연구에서 문서의 대상은 우리가 주로 사용하는 명조체와 고딕체로 하였으며, 한글 및 영숫자 혼합문서의 경우, 한글은 문자간의 접촉이 없다는 전제하에서 실험을 하였으며, 영숫자의 경우 두개의 문자가 접촉된 경우에는 문자의 중앙 열을 기준으로 두 문자를 따로 인식하였다. 또한 전처리 단계에서 기울어짐 보정은 입력시 문서자체의 기울어짐을 대상으로 하였다. 본 연구의 인식률은 영숫자의 경우 약 98% 정도이며, 한글의 경우는 약 95% 정도이다. 본 알고리듬의 처리속도는 전처리 단계에서 2초가 소요되고, 인식단계에서 한글은 초당 10자, 영숫자는 초당 25자의 인식속도를 보였다.

V. 결 론

본 논문에서는 그래픽 이미지를 포함한 인쇄체 한글과 영숫자로 구성된 혼합문서에 대한 효율적인 전처리 과정 및 문서인식 알고리듬을 구현하였다.

전처리 단계에서는 입력문서가 기울어져 있을 경우 Hough 변환을 이용하여 이를 보정하여 문자가 똑바로 놓이게 하였으며, 문서를 그레이레벨로 취득하여 입력시의 상황 변화를 흡수하도록 하였다. 그리고 그림부분과 문자부분의 분리를 체인코드로 표현된 연결화소를 이용하여 분리함으로써 문자열과 그림이 투영상에서 겹쳐진 문서도 올바르게 분리할 수 있었다. 그림과 분리된 문자열에 대해 수직, 수평투영을 이용하여 개별문자를 추출한 후 인식단으로 넘겨주었다.

인식단에서는 전처리 단계에서 넘겨온 개별문자에 대해 수직투영과 거리특징을 이용하여 먼저 한글과 영숫자를 분리하고 각각 세선화에 의해 구해진 끝점, 분기점과 흑백화소의 변화수, 부분투영, 거리특징을 사용하여 인식하였다. 한글인식에서는 부분 투영을 이용하여 모음부분을 수직모음과 수평모음으로 나누고 각 그룹에 대해 초성과 종성을 인식하는 계층적인 방법을 사용하였다. 영숫자 인식에서는 세선화에 의한 끝점, 분기점의 개수 및 위치와 세선화 특징의 단점을 보완해줄 수 있는 흑백화소의 변화수와 흑화소

까지의 거리특징을 이용하여 인식하였다.

본 논문에서 사용한 문체는 명조체와 고딕체 두가지를 사용하였다. 그리고 문서는 그림을 포함한 문서와 표를 포함한 문서, 문자열만으로 이루어진 문서에 대해 실험하였다. 컴퓨터 시뮬레이션 결과 한글과 영숫자 분리과정에서 영숫자가 접촉되어 있을 경우 분리과정에서 중간부분에서의 수직투영과 거리특징으로 분리되지 않아 한글과 영숫자의 특성을 이용한 부분투영을 이용하여 분리하였다. 실험결과 양호한 문서에 대해서는 효율적으로 인식하였다. 앞으로의 과제는 본 논문에서 사용한 방법을 바탕으로 여러가지 문자 및 필기체 문서인식 알고리듬을 개발하는 것이다.

參 考 文 獻

[1] T. Agui, M. Nakajima, T. K. Kim, and E. T. Takahashi, "A method of recognition and representation of Korean characters by tree grammars," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 3, pp. 245-251, July 1979.

[2] 이 주근, 남궁 재찬, 김 영건, "한글 pattern에서 subpattern 분리와 인식에 관한 연구," 대한전자공학회 논문지, 제 18권, 제 3호, pp. 1-9, 1981년 6월.

[3] 이 행세, 최 태영, 김 영길, 김 정우, "인공지능 기법을 이용한 텍스트 인식에 관한 연 구," 대한전자공학회 논문지, 제 26권, 제 11호, pp. 153-164, 1989년 11월.

[4] 김 정욱, 최 동혁, 강 석건, 박 건작, 박 규태, "계층구조 획 추출에 의한 한글 인식," 대한전자공학회 논문지, 제 28권 B편, 제 8호, pp. 1-9, 1991년 8월.

[5] W. Postl, "Detecting of linear oblique structures and skew scan in digitized documents," in *Proc. 8th Int. Conf. Pattern Recognition*, pp. 687-689, Paris, France, Oct. 1986.

[6] Y. Nakano, Y. Shima, H. Fujisawa, J. Higashino, and M. Fujinawa, "An algorithm for the skew normalization of document image," in *Proc. 10th Int. Conf. Pattern Recognition*, vol. 2, pp. 8-11, Atlantic City, NJ, June 1990.

[7] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Commun. ACM*, vol. 15, no. 1, pp. 11-15, Jan. 1975.

[8] K. H. Shyu, B. S. Jeng, I. C. Jou, and P. Y. Ting, "Image rotation correction with CORDIC array processing," in *Proc. SPIE Conf. Visual Communications and Image Proc. '88*, vol. 1001, pp. 484-490, Cambridge, MA, Nov. 1988.

[9] K. Y. Wong, R. G. Casey, and F. M. Wahl, "Block segmentation and text extraction in mixed text/image documents," *Computer Vision, Graphics, Image Processing*, vol. 20, no. 4, pp. 375-390, Dec. 1982.

[10] T. Akiyama and N. Hagita, "Automated entry system for printed documents," *Pattern Recognition*, vol. 23, no. 11, pp. 1141-1154, Nov. 1990.

[11] A. Tanaka, M. Kameyama, and O. Watanabe, "A rotation method for raster image using skew transformation," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pp. 272-277, Miami Beach, Florida, June 1986.

[12] 이 인동, 권 오석, 김 태균, "문서인식을 위한 전처리 기술의 소개," 정보과학회지, 제 9권, 제 1호, pp. 14-21, 1991년 2월.

[13] 이 인동, 권 오석, 김 태균, "문서영상에서 문자와 비문자의 분리 추출방법," 한국정보 과학회 논문지, 제 17권, 제 3호, pp. 247-258, 1990년 5월.

[14] 이 용일, 조 용주, 남궁 재찬, "신문 자동인식을 위한 신문표제에서의 문자추출에 관한 연구," 영상처리 및 이해에 관한 워크숍 발표 논문집, pp. 117-126, 1989년 1월.

[15] D. Wang and S. N. Srihari, "Classification of newspaper image blocks using texture analysis," *Computer Vision, Graphics, Image Processing*, vol. 47, no. 3, pp. 327-352, Sept. 1989.

[16] L. A. Fletcher and R. Kasturi, "A robust algorithm for text string

separation from mixed text/graphics images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-10, no. 11, pp. 910-918, Nov. 1988.

[17] W. K. Pratt, *Digtal Image Processing*, 2nd. ed., pp. 597-600, John Wiley & Sons, Inc., 1991.

[18] 함 영국, 도 상윤, 김 우성, 박 래홍, 이 창범, 김 상중, "저해상도 영·숫자 데이타의 고속인식," 춘계종합학술발표회 논문집, 한국통신학회, 제 10권, 제 1호, pp. 106-109, 1991년 5월.

[19] Y. K. Ham, C. B. Lee, W. S. Kim, S. Y. Doh, R.-H. Park, and S. J. Kim, "A simple sequentially designed rule-based alphanumerics recognition algorithm for OCR document processing using a thinning process," in *SPIE Proc. Intelligent Robots and Computer Vision X: Algorithms and Techniques*, vol. 1607, pp. 146-157, Boston, MA, Nov. 1991.

[20] 함 영국, 도 상윤, 정 홍규, 김 우성, 박 래홍, 이 창범, 김 상중, "문서 입출력 시스템 구성에 관한 연구," 대한전자공학회 논문지, 제 29권 B편, 제 10호, pp. 100-112, 1992년 10월.

[21] Y. K. Ham, H. K. Chung, I. K. Kim, R.-H. Park, C. B. Lee, S. J. Kim, and B. N. Yoon, "Hierarchical recognition of mixed documents of the Korean/Alphanumeric texts and graphic images," in *Proc. MVA'92 IAPR Workshop*, pp. 287-290, Tokyo, Japan, Dec. 1992.

[22] T. Pavlidis, "A thinning algorithm for discrete binary images," *Comput. Vision, Graphics, Image Processing*, vol. 13, no. 2, pp. 142-157, June 1980.

[23] 함 영국, 김 인권, 정 홍규, 박 래홍, "한글/영숫자 및 그래픽으로 구성된 혼합문서의 계층적 인식," 제 5회 신호처리 합동학술대회 논문집, 제 5권, 제 1호, pp. 295-298, 1992년 9월.

[24] 한글 기계화 연구소, 한글 기계화 연구, 1975년.

著 者 紹 介

咸 永 國(正會員) 第 29卷 B編 第 10號 參照
현재 동대학원 박사과정

•

丁 鴻 奎(準會員) 第 29卷 B編 第 10號 參照
현재 삼성전자 근무중

•

李 昌 範(正會員) 第 29卷 B編 第 10號 參照
현재 한국전자통신연구소 통신접속 연구실 근무

•

尹 炳 楠(正會員)
1949年 11月 15日生. 1975年 2月 한양대학교 전자공학과 졸업(공학사). 1989年 8月 청주대학교 전자공학과 졸업(공학석사). 현재 한국전자통신연구소 통신처리 연구부 근무

金 仁 權(準會員)
1970年 2月 16日生. 1992年 2月 서강대학교 물리학과 졸업(이학사). 1994年 2月 서강대학교 대학원 전자공학과 졸업(공학석사). 현재 동대학원 박사과정. 주관심 분야는 영상처리 등임.

•

朴 來 弘(正會員) 第 23卷 第 6號 參照
현재 서강대학교 전자공학과 교수

•

金 庠 仲(正會員) 第 29卷 B編 第 10號 參照
현재 한국전자통신연구소 통신접속 연구실 근무