

Japanese Puzzle을 이용한 인쇄체 영문자, 숫자, 기호의 인식

A Recognition of the Printed Alphabet, the Number and the Symbols by Using Japanese Puzzle

손 영선, 김 보성
Young-Sun Sohn, Bo-Sung Kim

동명대학교 정보통신공학과
E-mail: yssohn@tu.ac.kr

Abstract

지금까지의 연구에서 Japanese Puzzle을 이용한 인쇄체 영문자 인식을 구현하여 좋은 결과를 얻었고, 본 논문에서는 인쇄체 영문자(바탕, 돋움) 인식을 확장시켜 영문장을 인식하기 위해 키보드에서 입력 가능한 숫자 및 기호를 포함하여 인식하는 시스템을 구현하였다. 이미지를 입력 받아 이진화 처리, 히스토그램 투영을 이용한 문자 분리는 영문자 인식에서와 동일한 처리를 한다. 기호 중에서 세로 길이보다 가로 길이가 긴 기호인 -, = 만 가로를 정규화 하였고, 나머지는 세로를 정규화 하였다. 정규화 된 문자에 Japanese Puzzle을 역으로 적용하여 구하여진 수치 정보로부터 영문자, 숫자, 기호를 분류 및 인식하여 좋은 결과를 얻었다.

Key Words : 퍼즐, 이미지, 문자인식, 영어 문자, 기호 인식

1. 서론

기존의 컴퓨터 입력 장치로서 키보드, 마우스, 테블릿, 터치 스크린 등은 컴퓨터를 좀 더 쉽게 사용할 수 있도록 사용자에게 편리함을 가져다주었다. 그러나 컴퓨터의 발전에 따라 고속성, 대용량성, 정확성이 요구되고 이에 따라 대량의 문서를 빠르고 정확하게 처리하기 위해 많은 문자 인식 연구가 진행되어 왔다. 문자 인식을 위한 대표적인 알고리즘으로 원형 비교 방법[1], 구조적 방법[2], 통계적 방법[3], 신경망 방법[4] 등이 있고 명함 인식, 실시간 영어 사전, 자동차 번호판 인식, 우편물 인식, 여권 인식 등 여러 분야에 적용되고 있다.

본 논문에서는 영어 문자 인식을 영어 문장 인식으로 발전시키기 위하여 Japanese Puzzle을 이용하여 인쇄체 영문자(바탕, 돋움) 외에 0-9 숫자와 키보드에서 주로 사용되는 28개의 기호를 포함하는 문자 인식 알고리즘을 제안한다.

2. Japanese Puzzle

Japanese Puzzle은 1988년 테츠야 니시오가 개발한 논리 퍼즐로서 검정색과 흰색으로 그림을 표현하는 사각형 그리드를 기반으로 한다. 각 행과 열 옆에는 검은색 사각형들을 표현하는 일련의 수가 있고, 각 수는 연속적인 검은 사각형의 개수를 의미한다.

예를 들면, 수치 정보 "4 11"은 4개의 연속적인 검은색 사각형과 11개의 연속적 검은색 사각형 사이에 하나 이상의 흰색 사각형이 있다는 의미이다.[5]

그림 1은 주어진 수치 정보를 활용하여 "뽀빠이" 이미지를 나타낸 예이다.

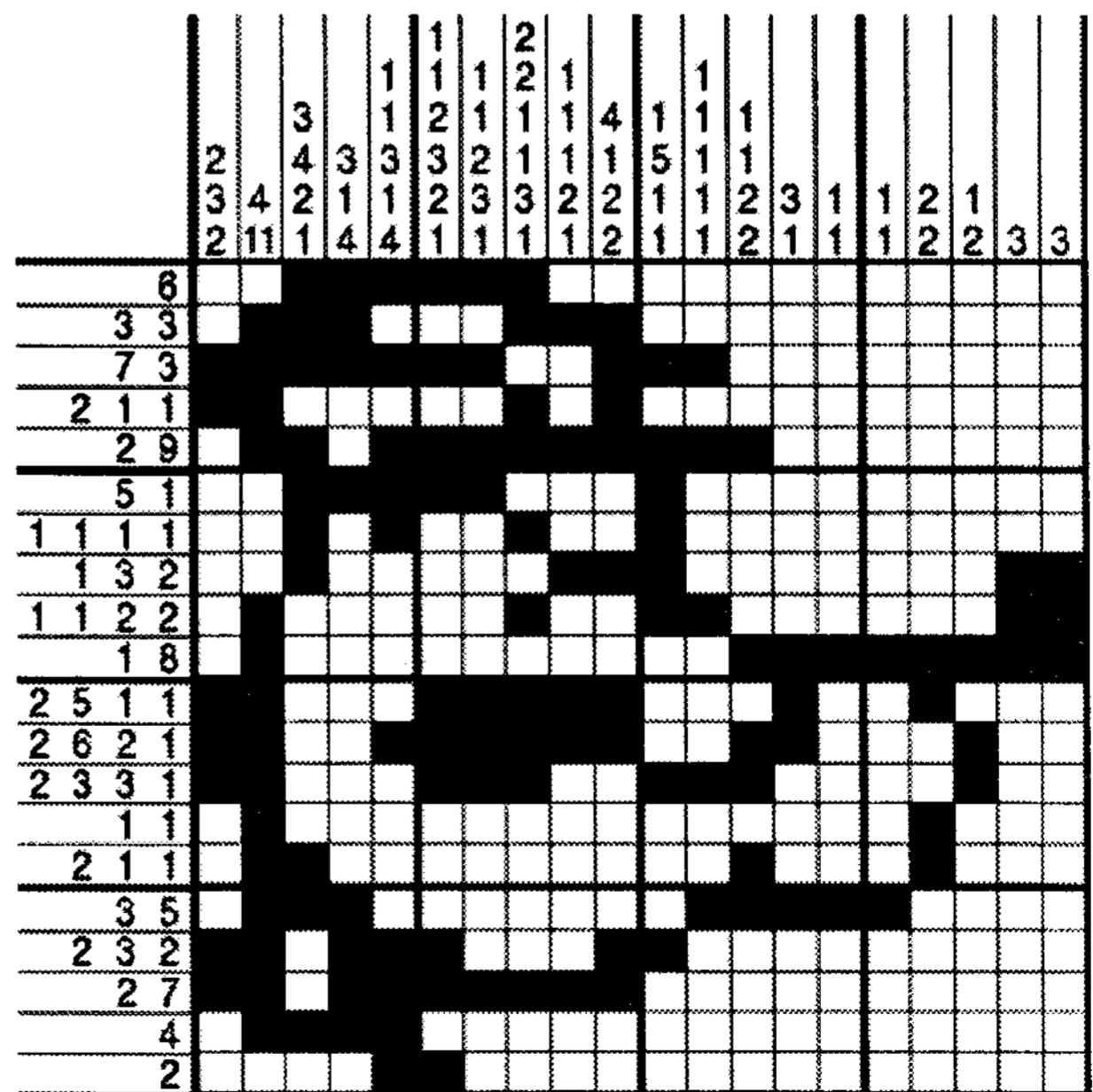


그림 1. Japanese Puzzle의 예

접수일자 :

최종완료일자 :

3. 시스템 알고리즘

그림 2의 전체 시스템 흐름도에서 알 수 있듯이 문서를 촬영한 디지털 영상을 입력 받아 이진화, 개별 문자 영역 추출, 개별 문자의 정규화 후 제안한 알고리즘을 적용하여 인식할 문자 이미지의 수치 및 위치 정보로부터 문자의 패턴을 구하여 표준 패턴과 비교하여 인식한다. 모든 문자 이미지의 비교·인식이 끝나면 인식된 문자들을 문서 편집기에 출력시킨다.

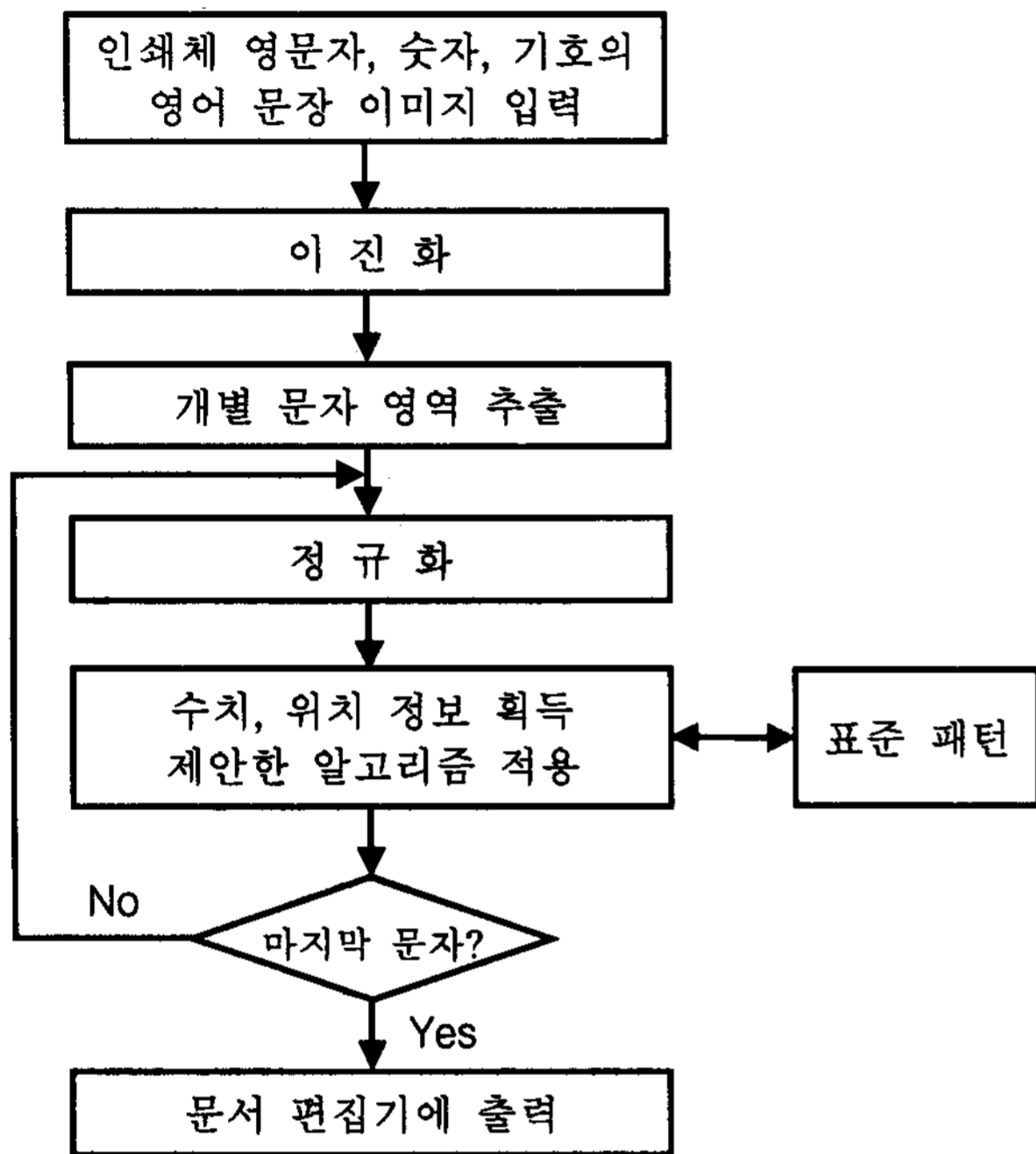


그림 2. 전체 시스템 흐름도

3.1 영상 이진화 및 개별 문자 영역 추출

흑백 CCD 카메라를 사용하여 문서 영상을 획득한 후 문자색과 배경색을 각각 0과 1로 이진화 처리한다. 전체 영상에서 각 문자 행을 분리하기 위해 수평으로 히스토그램을 적용하였고, 각 문장 행에서 개별 문자를 분리하기 위해 수직으로 히스토그램을 적용하였다. 그림 3 (a), (b)에 이진화 된 영상에서 히스토그램을 적용하여 문장 행과 개별 이미지를 분리하는 예를 나타내었다.

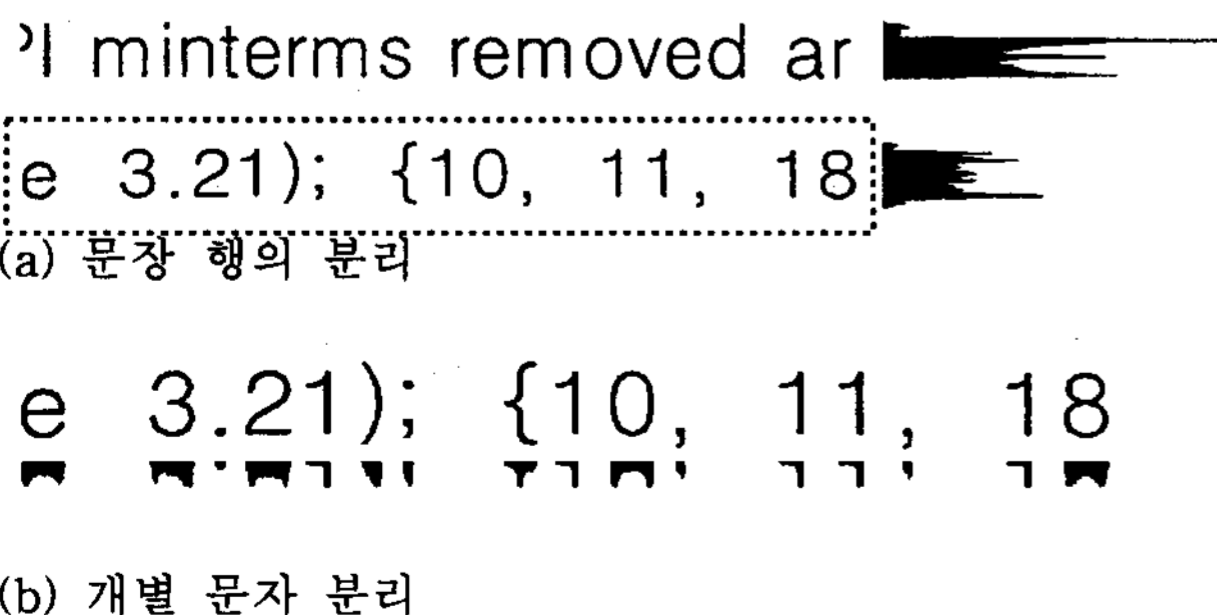


그림 3. 히스토그램 적용 예

분리된 개별 문자(영어 문자, 숫자, 기호)들의 높이가 동일하지 않으므로 각 이미지들의 높이를 구하기 위하

여, 개별 문자의 영역에서 다시 수평으로 히스토그램을 적용하여 정확한 문자 영역을 획득한다. 그림 4 (a)에서 문자 {10, 11, 18}의 폭은 영역에 맞게 분리가 되었지만 높이는 문장 행의 영역이 적용되어 있음을 알 수 있고, 그림 4 (b)에서는 정확한 영역이 획득된 문자의 예가 보여진다.



(a) 문장 행 분리에 의해 얻어진 개별 문자의 높이



(b) (a)의 문자에 수평 히스토그램을 적용하여 구한 실제 문자 높이

그림 4. 개별 문자 영역 추출 예

그림 4 (a)와 같이 개별 문자의 영역을 분리한 상태에서 각 개별 문자의 중앙을 수평 투영하여 수치 정보가 나타나지 않을 때 수평 투영한 중앙선을 기준으로 상·하 위치 정보를 이용하여 상단에 위치하는 기호들(' ' " " ' ^), 하단에 위치하는 기호들(.,), 양단에 위치하는 기호들(; :) 등의 세 영역으로 분류가 된다.

3.2 크기 정규화

입력된 개별 문자 이미지와 표준 패턴과 비교하기 위해서는 인식 대상의 수치 정보가 동일한 조건에서 획득·비교되어야 하므로 모든 입력 개별 문자들은 정규화 처리한다. 대부분의 문자, 숫자, 기호들은 세로 픽셀 수를 48픽셀로 정규화 하였으나, 마이너스 기호(-), 등호 기호(=), 틸트 기호(~) 등은 세로의 길이가 가로 길이보다 짧기 때문에 가로 픽셀 수를 48픽셀로 정규화 함과 동시에 분류 조건으로 사용하였다.

예를 들어 22×3(너비 22픽셀, 높이 3픽셀)인 마이너스 기호(-)의 정규화 된 이미지 크기는 48×6 이 되고, 40×32(너비 40픽셀, 높이 32픽셀)인 대문자 W의 정규화 된 이미지 크기는 60×48 이 된다. 다양한 크기의 문자, 숫자, 기호를 정규화는 선형 보간법[6]을 사용하였다.

3.3 Japanese Puzzle을 이용한 이미지 및 위치 정보 수치에 의한 문자 인식

정규화 된 개별 문자 영역에서 Japanese Puzzle 을 적용하여 획득된 수치 및 위치 정보를 이용하여 입력 패턴과 표준 패턴을 비교·인식한다. 각 노드의 분류 조건은 특정 영역에서 임의의 수치 정보나 위치 정보의 유무로 판단한다. 특성 영역은 영문자, 숫자, 기호의 특성을 나타내는 영역으로 그림 5와 같이 세로 영역을 R₁-R₅, 가로 영역을 C₁-C₃로 구분하여 표현한다.

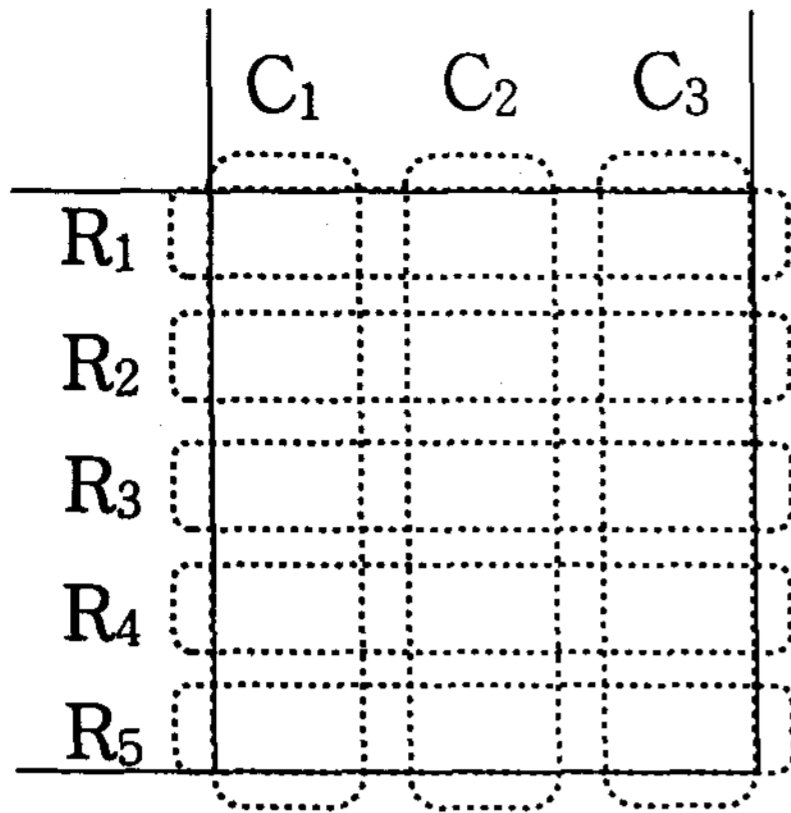


그림 5. 8가지 특성 영역

수치 정보는 임의의 문자 영역의 세로 또는 가로로 일직선 투영하여 연속된 검은색 픽셀이 나타나는 횟수를 P_n 으로 표현하고, 위치 정보는 문자의 가로와 세로 영역에서 특정 검은색 픽셀의 위치를 나타내는 좌표 (x, y)를 나타낸다. 투영 좌표에서 검은색 픽셀이 1회 나타날 경우에 (P_1) , 2회 나타날 경우 (P_1, P_2) 등과 같이 검은색 픽셀이 나타나는 횟수에 따라 수치 정보가 표현되는데 영문자, 숫자, 기호의 경우에는 (P_1) , (P_1, P_2) , (P_1, P_2, P_3) , (P_1, P_2, P_3, P_4) 등 4가지 패턴으로 표현 가능하다.

그림 6에 보여지듯이 문자 인식을 위해 개별 문자들은 초기에 개별 문자 영역 추출과정에서 분류된 3영역, 정규화 과정에서 분류된 가로 48픽셀 영역, 일반적인 문자 이미지가 표현되는 세로 48픽셀 영역 등 5영역으로 분류되어진다.

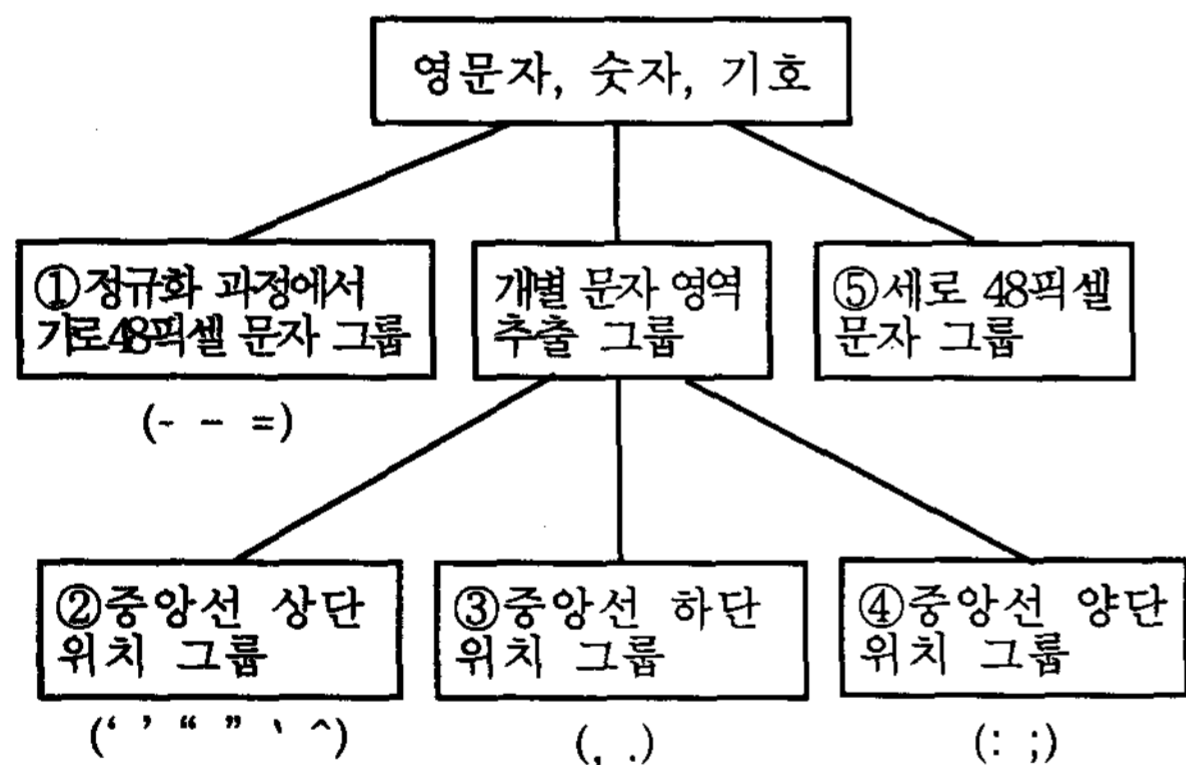


그림 6. 문자들의 초기 5분류

먼저 정규화 과정에서 분류된 가로 48픽셀 영역의 세 가지 기호 문자에서 모든 세로 영역을 수평 투영하여 직선이 검출이 되지 않은 기호 문자 틸트(-)를 분류하고 남아 있는 두 개의 기호문자의 C_2 구간을 수직 투영하여 (P_1, P_2) 가 (3, 3)이상인 픽셀 정보가 검출되면 등호 문자(=)로 인식하고, 검출되지 않으면 마이너스 문자(-)로 인식한다.

다음으로 개별 문자 영역 추출 과정에서 분류된 세 영역 중에서 중앙선을 기준으로 상·하 양단에 위치하는 기호들은 가로를 수직 투영하여 (P_1, P_2, P_3) 가 (5, 5, 3) 이상인 픽셀 정보가 검출되면 세미콜론 문자(;)로 인식하고, 검출되지 않으면 콜론 문자(:)로 인식한다.

중앙선을 기준으로 상단에 위치하는 기호들은 기호 문자 이미지의 C_2 영역에서 수직 투영하여 (P_1, P_2) 가 (10, 3) 또는 (3, 10) 이상인 픽셀 정보가 검출되면 단일 따옴표

표쌍(' ') 그룹으로 분류되고, 여기에서 R_2 영역에 수평 투영하여 (P_1) 이 (15)이상인 픽셀 정보가 검출되면 우측의 단일 따옴표(')로 인식하고, 검출되지 않으면 좌측의 단일 따옴표(')로 인식한다. 이중 따옴표(" ")는 개별 문자 분류 과정에서 하나의 문자 영역으로 분리할 수가 없다. 따라서 개별 문자 영역을 추출하는 과정에서 중앙선을 기준으로 상단에 위치해 있으면서 기호에 해당되는 크기를 가질 때 C_2 구간을 투영하여 (P_1, P_2) 가 두 번 연속으로 나타나면 개별 문자의 인식 순서를 저장한다. 인식 과정에서 해당되는 인식 순서의 문자가 인식될 때 첫 번째 따옴표는 무시하고 두 번째 따옴표를 이중 따옴표로 인식한다. C_2 영역에서 수직 투영하여 (P_1, P_2) 가 (10, 3) 또는 (3, 10) 이상인 픽셀 정보가 검출되지 않는 그룹에서는 R_3 영역에 수평 투영하여 (P_1, P_2) 가 (3, 3)이상인 픽셀 정보가 검출되면 서컴플렉스 기호문자(^)로 인식하고 검출되지 않으면 그레이브 기호문자(`)로 인식한다.

중앙선을 기준으로 하단에 위치하는 기호들은 C_2 에서 수직 투영하여 (P_1, P_2) 가 (10, 3) 이상인 픽셀 정보가 검출되면 콤마 기호문자(,)로 인식하고, 검출되지 않으면 마침표 기호문자(.)로 인식한다.

마지막으로, 세로 48픽셀 영역의 개별 문자 인식 예로 영문자 'Z'의 인식 과정을 설명한다. 표 1에 영문자 'Z'의 가로·세로 전체 구간을 일직선 투영하여 획득한 수치 정보가 보여지고, 그림 7에 영문자 'Z'에서 임의의 특정 픽셀의 좌표를 나타낸 예가 보여진다.

표 1. 영문자 'Z'의 수치 정보

수치 정보: $(P_1) \sim (P_1, P_2, P_3, P_4)$

수평 투영		수직 투영	
[1]-(33)	[25]-(6)	[1]-(3)	[25]-(4, 8, 4)
[2]-(33)	[26]-(5)	[2]-(3)	[26]-(4, 7, 3)
[3]-(8, 7)	[27]-(5)	[3]-(4, 4)	[27]-(5, 7, 3)
[4]-(8, 7)	[28]-(5)	[4]-(4, 4)	[28]-(5, 8, 3)
[5]-(6, 5)	[29]-(5)	[5]-(6, 7)	[29]-(5, 7, 3)
[6]-(3, 5)	[30]-(5)	[6]-(8, 9)	[30]-(3, 7, 3)
[7]-(3, 5)	[31]-(7)	[7]-(8, 9)	[31]-(3, 6, 4)
[8]-(5, 5)	[32]-(5)	[8]-(5, 6, 4)	[32]-(10, 6)
[9]-(3, 6)	[33]-(5)	[9]-(5, 7, 4)	[33]-(10, 6)
[10]-(3, 6)	[34]-(4)	[10]-(5, 7, 4)	[34]-(7, 6)
[11]-(2, 4)	[35]-(4)	[11]-(4, 6, 4)	[35]-(7, 6)
[12]-(5)	[36]-(6)	[12]-(4, 8, 4)	[36]-(5, 4)
[13]-(5)	[37]-(5)	[13]-(4, 8, 4)	[37]-(4, 3)
[14]-(6)	[38]-(5)	[14]-(4, 8, 4)	[38]-(4, 1)
[15]-(4)	[39]-(4, 3)	[15]-(5, 6, 4)	
[16]-(4)	[40]-(5, 3)	[16]-(4, 6, 3)	
[17]-(5)	[41]-(5, 3)	[17]-(4, 7, 3)	
[18]-(5)	[42]-(6, 3)	[18]-(5, 7, 4)	
[19]-(4)	[43]-(4, 4)	[19]-(5, 6, 3)	
[20]-(6)	[44]-(4, 4)	[20]-(5, 8, 4)	
[21]-(6)	[45]-(5, 5)	[21]-(5, 8, 4)	
[22]-(5)	[46]-(7, 8)	[22]-(5, 7, 5)	
[23]-(4)	[47]-(7, 8)	[23]-(4, 6, 4)	
[24]-(4)	[48]-(33)	[24]-(4, 6, 4)	

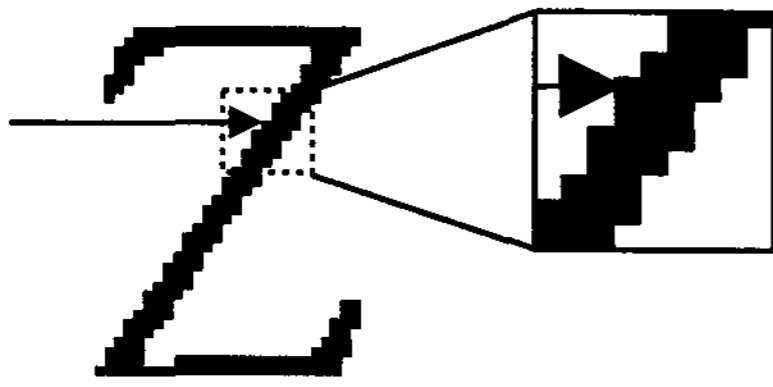


그림 7. 영문자 'Z'에서 위치 정보 예

영문자 'Z'는 그림 6에서 보여진 5번째 그룹인 세로 48픽셀 문자 그룹으로부터 그림 8에서 보여지듯이 $X_1, X_3, X_6, X_{13}, X_{26}, X_{42}, X_{55}, X_{65}$ 등의 분류 조건에 의해 인식되어진다.

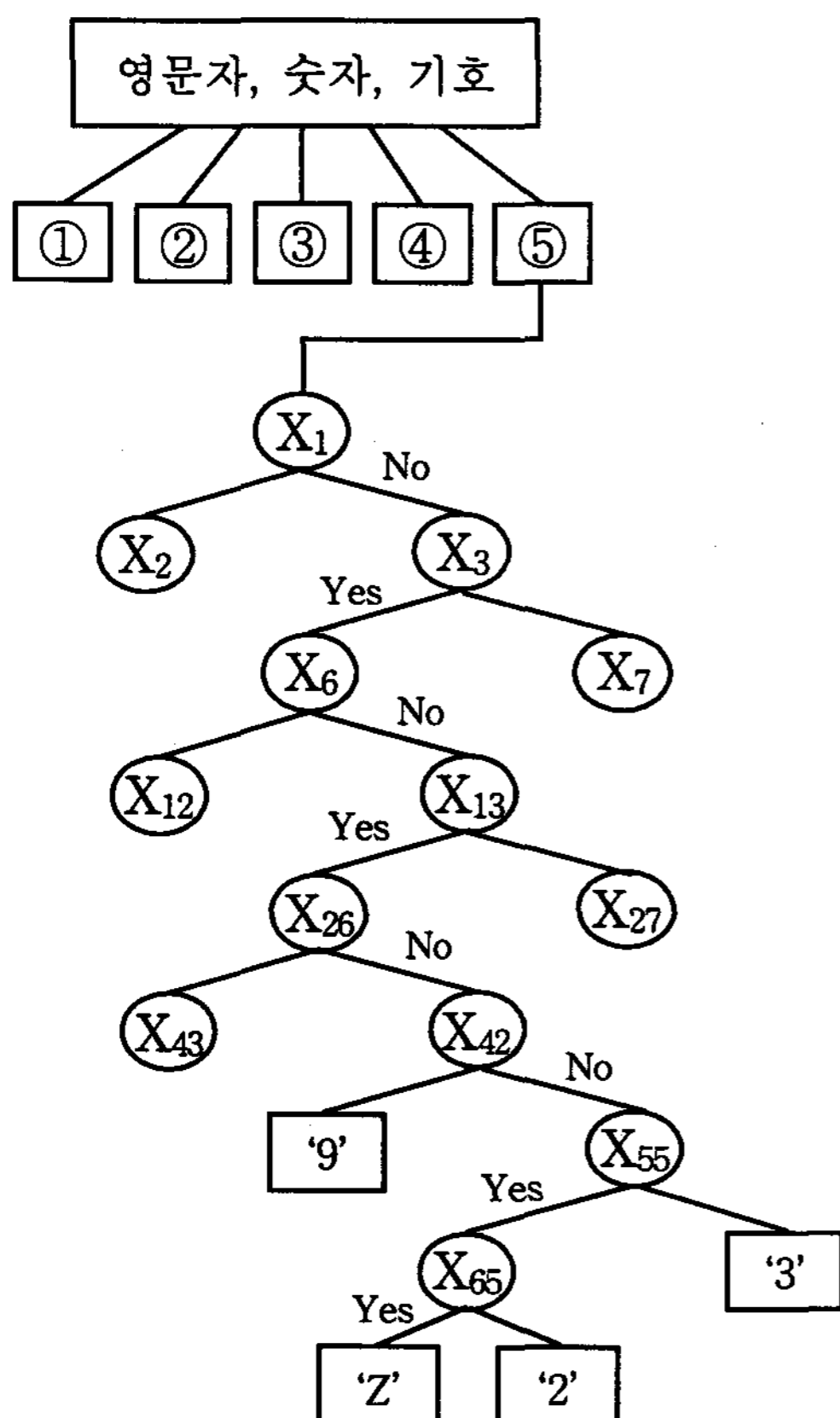


그림 8. 영문자 'Z'의 인식을 위한 분류도

여기서 X_1 의 분리 조건은 너비의 전체 영역을 수직 투영하여 직선에 해당하는 (P_1) 이 (40) 이상인 픽셀 정보의 검출 유무이다. X_3 은 R_3 구간에 수평 투영하여 (P_1) 이 (3) 이상이고 (20) 이하인 픽셀 정보의 검출 유무, X_6 은 R_1, R_5 구간에 수평 투영하여 (P_1) 이 0 인 픽셀 정보의 검출 유무, X_{13} 은 C_2 구간에 수직 투영하여 (P_1, P_2, P_3) 가 (3, 3, 3) 이상인 픽셀 정보가 검출된 횟수를 카운트하여, 8회 이상일 때와 8회 미만일 때를 구분하여 분류한다.

X_{26} 분류 조건을 적용할 문자 그룹 S Z e s z 2 3 5 6 8 9 중에서 바탕체 S Z 2 같은 경우 비슷한 수치 정보를 가지고 있기 때문에 특정 검은색 픽셀이 나타내는 좌표 값을 이용하여 분류를 하였다. R_2, R_3 구간을 수평 투영하여 처음 검출되는 (P_1) 의 픽셀 값이 (3) 이상 (12) 이하이면서 (x, y) 좌표 값 중 x 좌표의 값이 문자 너비의 좌측 또는 우측에 존재하는가의 여부가 분류 조건이다. 'Z' 문자는 R_2 구간에서 처

음 검출되는 연속된 검은색 픽셀 정보 (P_1)에서 가장 좌측에 위치한 특정 픽셀의 (x, y) 중 x 값이 너비 절반의 오른쪽에 위치해 있기 때문에 No로 판단한다.

X_{42} 는 R_3 구간에서 (P_1, P_2) 가 (3, 3) 이상인 픽셀 정보의 검출 유무, X_{55} 는 C_1 구간에서 (P_1, P_2, P_3) 가 (3, 3, 3) 이상인 픽셀 정보의 검출 유무로 판단한다. 마지막으로 X_{65} 에서는 R_1 구간을 수평 투영하여 (P_1) 이 (25) 이상인 픽셀 정보가 검출되어 Yes로 판단하고 문자 'Z'를 인식한다.

영문자 Z, z 와 같이 형태는 같고 크기가 다른 대·소문자는 문장의 높이와 개별 문자의 높이 비율을 적용하여 분류하였다.

4. 실험

안정된 인식을 위해 인식 대상 문자의 높이는 30픽셀 이상 유지시켰으며, 영문자 인식을 위해 모든 알파벳이 포함되도록 하여 4312자(바탕체 2156자, 돋움체 2156자)를 실험하였다. 영어 문장에서 숫자와 기호는 출현 빈도가 낮기 때문에 숫자 400자(바탕 200자, 돋움 200자), 기호 1120자(바탕 560자, 돋움 560자)를 포함하는 문장을 만들어서 별도로 인식 실험을 하였다.

그림 9은 입력 이미지의 예이고, 그림 10는 제안한 알고리즘에 의해 문서 편집기에 출력된 예이다.

5. 결과 및 향후 연구 과제

본 논문에서는 인쇄체 영문자 인식을 영어 문장 인식으로 발전시키기 위해 숫자와 기호를 추가하여 시스템을 구현하였다. 두 가지 문자체 4312 영문자와 400개의 숫자 문자, 1120개의 기호 문자 등을 실험한 결과 100%의 인식률을 얻었다.

향후 연구로는 책 영상을 카메라로 입력받아 문자 부분과 그림, 도형 등의 부분을 구분하여 문자 인식이 가능한 시스템 구현도 고려할 수 있다.

참고 문헌

- [1] 백승복, 손영선, 손가락 이동에 의해 선택된 영역의 인쇄체 한글 영상 문서화, 한국 퍼지 및 지능시스템 학회 논문지, 제 12권, 제 4호, pp. 306-310, 2002.
- [2] Z. Y. Lin, and P. Liu, Structural Attribute Feature Code Representation and Recognition of Multi font Printed Chinese Characters, International journal of pattern recognition and artificial intelligence, vol. 15, no. 2, pp. 218-310, 2001.
- [3] K. C. Jung and H. J. Kim, Korean character recognition using a TDNN and an HMM, Pattern recognition letters, vol. 20, no. 6, pp. 551-563, 1999.
- [4] J. H. Kim, K. K. Kim and S. I. Chien, Korean and English Character Recognition System Using Hierarchical Classification Neural Network, IEEE International Conference on System, Man and Cybernetics, vol. 1, pp. 759-764, 1995.
- [5] <http://www.nonosweeper.com/aboutnonograms.html>
- [6] 강동중, 하중은, Visual C++ 을 이용한 디지털 영상처리, SciTech 미디어, pp. 184-194, 2003.
- [7] Y. S. Sohn, K. S. Oh, B. S. Kim, A Recognition method of the Printed Alphabet by using Nonogram Puzzle, Proceedings of The 8th International Symposium on Advanced Intelligent Systems, pp. 232-236, 2007.