

노노그램 퍼즐을 이용한 인쇄체 영문자 인식

A Recognition of the Printed Alphabet by Using Nonogram Puzzle

손영선 · 김보성

Young-Sun Sohn and Bo-Sung Kim

동명대학교 정보통신공학과

요약

본 논문에서는 흑백 CCD 카메라로부터 입력되는 2가지 인쇄체(바탕, 돋움) 영문자를 인식하여 편집 가능한 텍스트 형식으로 변환하는 시스템을 구현하였다. 입력된 인쇄체 영어 문장 영상을 이진화 처리 후, 히스토그램 기법을 적용하여 수평 투영으로 각 문장의 행을 분리하고 수직 투영으로 개별 문자를 분리하였으며, 문자의 높이를 48픽셀로 변환하여 정규화 하였다. 정규화 된 개별 문자에 노노그램 퍼즐 원리를 역으로 이용하여, 픽셀을 단위로 하는 작은 사각형들로 구성된 사각형으로 문자를 덮은 후 문자의 특성을 노노그램 퍼즐의 수치 정보로 나타내어 표준 패턴 정보와 비교하여 인식하게 하였다. 바탕체 2609개, 돋움체 1475개의 문자를 대상으로 실험하여 100% 인식률을 얻었다.

Abstract

In this paper we embody a system that recognizes the printed alphabet of two font types (Batang, Dodum) inputted by a black-and-white CCD camera and converts it into an editable text form. The image of the inputted printed sentences is binarized, then the rows of each sentence are separated through the vertical projection using the Histogram method, and the height of the characters are normalized to 48 pixels. With the reverse application of the basic principle of the Nonogram puzzle to the individual normalized character, the character is covered with the pixel-based squares, representing the characteristics of the character as the numerical information of the Nonogram puzzle in order to recognize the character through the comparison with the standard pattern information. The test of 2609 characters of font type Batang and 1475 characters of font type Dodum yielded a 100% recognition rate.

Key Words : Recognition, Image, Histogram, Nonogram, Character

1. 서 론

컴퓨터의 성능 발전과 하드디스크의 대용량화에 더불어, 인간은 많은 입력 장치들을 개발하여 컴퓨터를 쉽게 사용하고자 노력하였다. 종래의 입력 장치인 키보드, 마우스, 테블릿 등은 대량의 문서 정보를 처리하기 어렵다. 이러한 문제를 해결하기 위하여 많은 자동문서 인식 시스템을 개발하여 왔다.

대표적인 알고리즘으로는 문자의 무게 중심을 중심으로 그려진 원들과 문자가 교차하는 부분을 문자의 특징 정보로 사용하는 원형 패턴 벡터 방법[1], 패턴을 구성하고 있는 기본 원소들의 관계를 tree 나 graph 형태로 나타내어 인식하는 구조적 방법[2], n차원의 특정 벡터로 표현된 인식 패턴들의 통계적분포를 이용하여 인식하는 통계적 방법[3], 신경망에서 학습된 문자의 학습 유사도를 사용하여 인식하는 신경망 방법[4] 등이 있다.

본 논문에서는 인쇄체 영문자 인식을 위해 노노그램 퍼즐을 이용하여 문자 특징을 분류하는 문자 인식 알고리즘을 제안한다.

2. 노노그램

노노그램은 일본에서 보급된 숨은 그림 로직 퍼즐로서, 2 가지 색으로 그림을 표현하는 사각형 그리드를 기반으로 한다. 퍼즐 게임의 목표는 제시된 수치 정보로부터 숨겨진 이미지를 찾는 것이다. 각 행과 열 옆에는 해당 줄의 검은 사각형들을 표현하는 일련의 수가 있다. 각 수는 연속적인 검은 사각형의 개수를 의미한다. 예를 들면, 수치 정보 “3, 2”는 3개의 연속적 검은색 사각형과 2개의 연속적 검은색 사각형 사이에 하나 이상의 흰색 사각형이 있다는 의미이다 [5].

기본적으로, 이 게임을 풀기 위해서는 가로와 세로의 수치 정보를 활용하여 유용한 정보를 찾아야 한다. 그림 1에 인쇄체 영문자 y에 대한 노노그램 퍼즐의 예를 나타낸다.

3. 시스템 알고리즘

그림 2의 전체 시스템 흐름도에서 알 수 있듯이, 인쇄체 영문자 영상을 입력 받아 이진화를 하고 히스토그램 기법을 적용하여 개별 문자를 분할한다. 분할된 개별 문자를 정규화 한 후, 제안한 알고리즘을 적용하여 인식할 문자 이미지의 수치 정보를 획득한다. 획득한 수치 정보로부터 특징 패턴을 추출하여 표준 패턴과 비교하여 문자를 인식한다. 최종적으로 인식된 문자들은 문서 편집기에 출력시킨다.

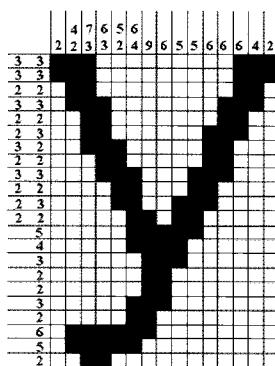


그림 1. 노노그램 퍼즐의 예

Fig 1. An example of the Nonogram puzzle.

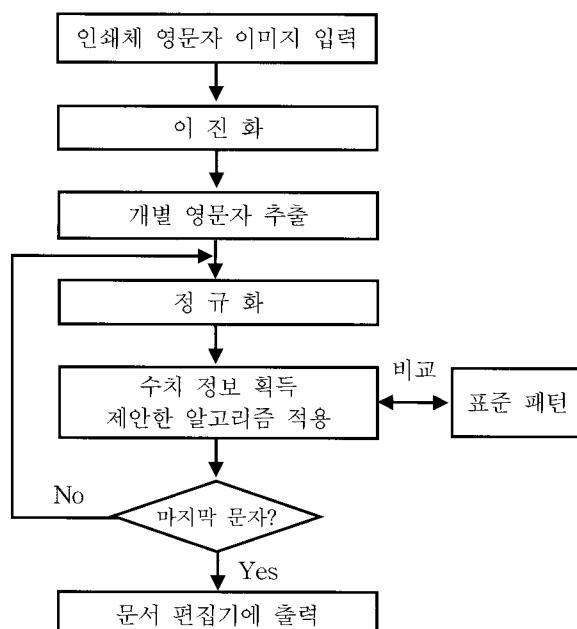


그림 2. 전체 시스템 흐름도

Fig 2. Whole system flow-chart.

3.1 이미지 이진화 및 문자 영역 선택

PC에 장착된 카메라 보드에 연결된 흑백 CCD 카메라로 텍스트 문서를 촬영하여 저장된 영상을 이진화 처리한다. 문자 행간의 분리를 위해 수평으로 히스토그램을 투영하고, 수직 투영을 통하여 개별 문자들을 분리한다. 그림 3에 이진화 된 이미지에 히스토그램 방법을 적용한 예가 보여진다.

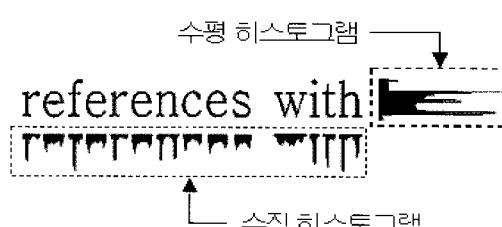


그림 3. 히스토그램 기법의 예

Fig 3. An example of the Histogram method.

그림 4에 보이는 단어 'with'의 경우에 각 문자 'w', 'i', 't', 'h'의 높이가 다르므로, 빈 공간에 의하여 정규화 처리에 문제가 된다. 이를 해결하기 위하여, 분리된 개별 문자에 수평 히스토그램을 적용하여 문자 영역만을 획득한다.



그림 4. 문자 영역 결정 예

Fig 4. An example of the decision of the character area.

3.2 크기 정규화

히스토그램에 의해 분리된 문자의 크기가 다양하기 때문에 정규화하여 모든 문자에 대하여 동일한 크기로 처리하였다. 문자들의 세로 픽셀 수를 48픽셀로 정규화함으로써 동일 좌표상에서 데이터 접근을 가능하게 하였다. 40×32(폭 40픽셀, 높이 32픽셀)인 대문자 W에 대한 정규화의 경우 세로가 48픽셀이 되기 위한 비율이 1.5이므로 정규화 된 이미지 크기는 60×48이 된다. 맵핑 과정에서 선택되지 않는 픽셀들은 선형 보간법[6]에 의해 보완한다.

3.3 노노그램 수치 정보에 의한 인식 방법

정규화된 문자 이미지에 노노그램 퍼즐을 역으로 적용하여 획득한 수치 정보를 이용하여 입력 패턴과 표준 패턴을 비교하고 문자를 인식한다. 이미지 정보를 수치 정보로 나타낸 노노그램의 적용 예로서 정규화된 인쇄체 대문자 'B'의 수평 및 수직 수치 정보를 그림 5에 나타내었다.

정규화된 인쇄체 문자의 가로와 세로 구간에 픽셀 단위로 일직선 투영하여 노노그램 수치 정보를 획득한다. 획득된 수치 정보에서 문자의 이미지 정보를 나타내는 연속된 검은색 픽셀이 나타나는 횟수를 P_n 으로 표현한다. 예를 들어 1회 나타날 경우에 (P_1) , 2회 나타날 경우 (P_1, P_2) 등과 같이 검은색 픽셀이 나타나는 횟수에 따라 (P_1, P_2, P_3) , (P_1, P_2, P_3, P_4) , $(P_1, P_2, P_3, P_4, P_5)$...의 형태로 표현하고, 영문자의 경우 문자의 형태 특성상 (P_1) , (P_1, P_2) , (P_1, P_2, P_3) , (P_1, P_2, P_3, P_4) 등 4가지 패턴으로 표현된다.

개별 영문자의 인식을 위하여 특정 영역 데이터의 존재 유무에 따라 패턴을 분류한다. 그림 6은 영문자 인식에 유효한 정보를 나타내는 8가지 특정 영역을 보여준다. 한 문자가 인식되기까지 탐색 횟수가 평균적인 수치가 나오려면 좌우 부트리의 개수가 1:1의 비율을 가지면 좋다.

입력 문자의 수치 정보로부터 특정 구간 내에 조건에 맞는 픽셀 값이 있는지 없는지 판단하는 이진 트리 구조를 이용하여 최종 문자를 인식하기 위한 42가지 분리 조건을 사용하였다.

그림 6에서 제시된 문자 'B'의 탐색 과정은 그림 7에서 알 수 있듯이 $X_1, X_2, X_5, X_{11}, X_{22}$ 의 분리 조건을 거쳐 B를 인식한다.

그림 7의 첫번째 분류 조건인 X_1 은 긴 수직선을 가진 문자를 분리하기 위하여 그림 6의 C_1, C_2, C_3 구간에서 수직 투영시 전체 48픽셀 중 40픽셀이 넘는 값의 유무를 분리 조건으로 결정했다. 분리 조건 X_1 으로 전체 영문자를 두 그룹으로 분리한 결과가 그림 8에 보여진다.

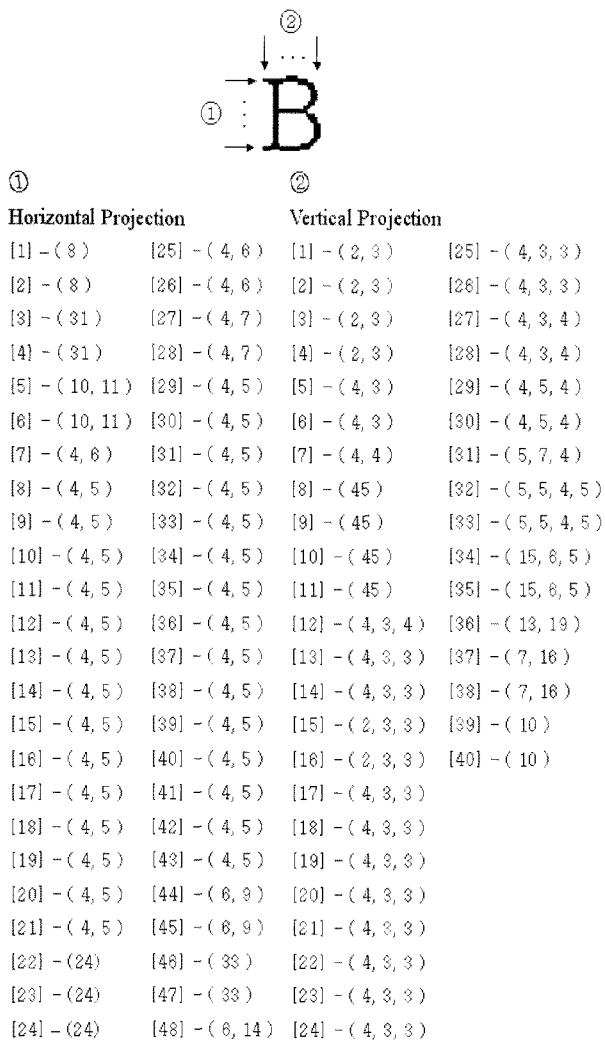


그림 5. 영문자 'B'에 대한 수치 정보

Fig 5. The numerical information for the character 'B'.

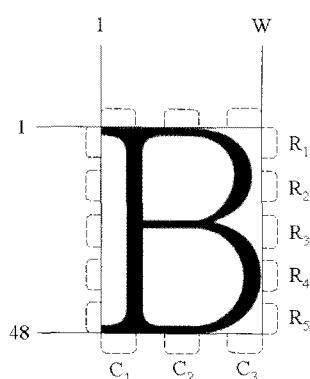


그림 6. 8가지 특징 영역

Fig 6. 8 kinds of area of feature

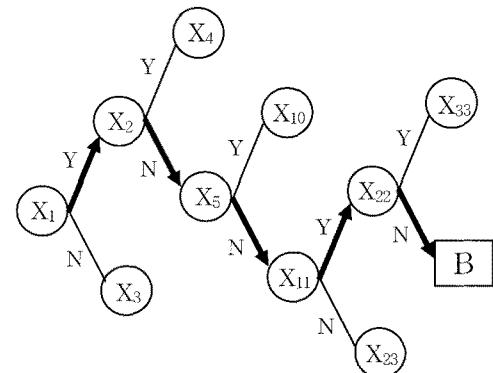


그림 7. 영문자 'B'의 이진 트리 경로

Fig 7. The paths of the binary tree for the character 'B'.

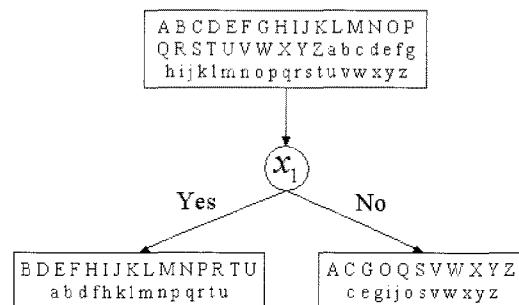


그림 8. X1 분리 조건 결과

Fig 8. The result of the separation condition X1

분리 조건 X_2 는 수평 투영하여 연속된 검은색 픽셀이 나타나는 횟수가 1회인 문자와 그 이외의 문자를 분리하기 위해 R_2 구간에서 3픽셀 이상의 연속된 검은색이 (P_1)인지 판단하였고, 그 결과는 그림 9와 같다.

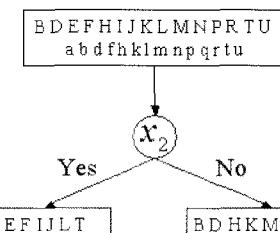


그림 9. X2 분리 조건 결과

Fig 9. The result of the separation condition X2

분리 조건 X_5 는 수직 투영하여 연속된 검은색 픽셀이 나타나는 횟수가 1회인 문자와 그 이외의 문자를 분리하기 위해, C_2 구간에서 3픽셀 이상의 연속된 검은색이 (P_1)인지 판단하였고, 그 결과는 그림 10에 보여 진다.

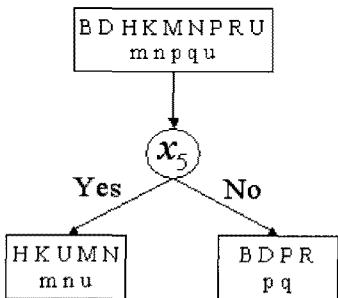


그림 10. X_5 분리 조건 결과

Fig 10. The result of the separation condition X_5

분리 조건 X_{11} 은 수평 투영하여 연속된 검은색 픽셀이 나타나는 횟수가 1회이면서 약간 긴 수평선을 가진 문자와 그 이외의 문자를 분리하기 위해 R_3 구간에서 15픽셀 이상의 연속된 검은색이 (P_1)인지 판단하였고, 그 결과는 그림 11과 같다.

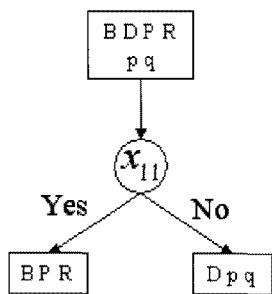


그림 11. X_{11} 분리 조건 결과

Fig 11. The result of the separation condition X_{11}

분리 조건 X_{22} 는 수직 투영하여 연속된 검은색 픽셀이 나타나는 횟수가 2회인 문자와 그 이외의 문자를 분리하기 위해 C_2 구간에서 3픽셀 이상의 연속된 검은색이 (P_1, P_2)인지 판단하였으며, 그림 12와 같다.

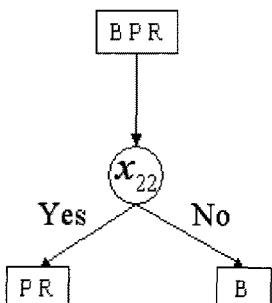


그림 12. X_{22} 분리 조건 결과

Fig 12. The result of the separation condition X_{22}

개별 문자를 인식하기 위한 각 노드의 분리 조건을 정리하면 표 1과 같다.

표 1. 문자의 분리 조건

Table 1. The separation condition of the characters

노드	구간	조건
X_1	C_1, C_2, C_3	연속 40픽셀 이상이고 (P_1)
X_2	R_2	연속 3픽셀 이상이고 (P_1)
X_3	R_3	연속 3픽셀 이상이고 (P_1)
X_4	R_4	연속 3픽셀 이상이고 (P_1)
X_5	C_2	연속 3픽셀 이상이고 (P_1)
X_6	R_2	연속된 픽셀 값이 없다.
X_7	C_2	연속 3픽셀 이상이고 (P_1)
X_8	R_2, R_3	연속 12픽셀 이상이고 (P_1)
X_9	C_1	연속 45픽셀 이상이고 (P_1)
X_{10}	R_2	연속 3픽셀 이상이고 (P_1, P_2)
X_{11}	R_3	연속 15픽셀 이상이고 (P_1)
X_{12}	C_1	연속 3픽셀 이상인 (P_1)이
X_{13}	R_3	연속 10픽셀 이상이고 (P_1)
X_{14}	R_3	연속 3픽셀 이상이고 (P_1, P_2)
X_{15}	R_4	연속 25픽셀 이상이고 (P_1)
X_{16}	C_2	연속 40픽셀 이상이고 (P_1)
X_{17}	R_1, R_5	연속 25픽셀 이상이고 (P_1)
X_{18}	C_2	연속 3픽셀 이상이고 (P_1)
X_{19}	C_2	연속 3픽셀 이상이고 (P_1, P_2, P_3)
X_{20}	R_3	연속 35픽셀 이상이고 (P_1)
X_{21}	R_3	연속 3픽셀 이상이고 (P_1)
X_{22}	C_2	연속 3픽셀 이상이고 (P_1, P_2)
X_{23}	C_1	연속 45픽셀 이상이고 (P_1)
X_{24}	R_3	연속 12픽셀 이상이고 (P_1)
X_{25}	C_2	연속 3픽셀 이상이고 (P_1)
X_{26}	C_2	연속 3픽셀 이상이고 (P_1, P_2, P_3)
X_{27}	* R_5 , ** C_1	*연속 3픽셀 이상, 8픽셀 이상이고 (P_1, P_2) and **연속 3픽셀 이상이고 (P_1)
X_{28}	C_2	연속 3픽셀 이상이고 (P_1, P_2, P_3)
X_{29}	C_2	연속 45픽셀 이상이고 (P_1)
X_{30}	R_1	연속 3픽셀 이상, 8픽셀 이상이고 (P_1, P_2)
X_{31}	C_3	연속 25픽셀 이상이고 (P_1)
X_{32}	C_3	연속 40픽셀 이상이고 (P_1)
X_{33}	R_3	연속 3픽셀 이상이고 (P_1, P_2, P_3)
X_{34}	R_4	연속 3픽셀 이상이고 (P_1)
X_{35}	R_4	연속 3픽셀 이상이고 (P_1)
X_{36}	R_3	연속 30픽셀 이상이고 (P_1)
X_{37}	* C_1 , ** C_2	*연속 3픽셀 이상이고 (P_1) or **연속 20픽셀 이상이고 (P_1)
X_{38}	C_2	연속 3픽셀 이상이고 (P_1, P_2, P_3, P_4)
X_{39}	C_1	연속 3픽셀 이상인 (P_1)이 3번 겹출
X_{40}	R_1	연속 25픽셀 이상이고 (P_1) or 연속 3픽셀 이상, 20픽셀 이상이고 (P_1, P_2)
X_{41}	C_3	연속 3픽셀 이상이고 (P_1, P_2)
X_{42}	R_3	연속 3픽셀 이상, 10픽셀 이상이고 (P_1, P_2)

4. 실험

실험에 사용된 컴퓨터는 CPU가 펜티엄4-3.0GHz, OS는 Window XP, RAM은 512MB의 사양을 가지며, 문자 1개를 인식하는데 소요된 평균 시간은 5.38msec이며, A4 문서에 있는 175개의 문자를 인식하는데 소요되는 시간은 0.28sec였다. 본 논문에서는 안정된 인식을 위해 입력 이미지 문자의 높이를 30픽셀 이상 유지하여, 문서 작성에 많이 사용되는 2가지 문자체인 바탕체와 돋움체에 대하여 알파벳의 모든 대·소문자가 포함되도록 A4크기 문서 25장을 입력 받아 바탕체 2609자, 돋움체 1475자를 제안한 알고리즘에 적용하여 실험하였다. 바탕체는 획의 끝에 serif 장식을 한 글자체이고, 돋움체는 sans serif 장식을 한 글자체이다. 입력 이미지의 예가 그림 13에 보이고, 제안된 알고리즘을 적용하여 인식된 문자를 문서 편집기에 출력한 예가 그림 14에 보인다.

Macro parameters are substitution symbols that represent a character string. These symbols can also be used outside of macro to equate character string to a symbol name.

그림 13. 문자 이미지 예

Fig 13. An example of the image of the characters

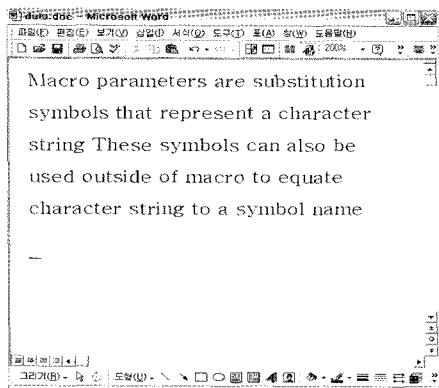


그림 14. 문서 편집기 출력 예

Fig 14. An example of the output of the word editor

5. 결과 및 향후 연구 과제

본 논문에서는 흑백 CCD 카메라로 영어 문서 영상을 입력 받아 제안된 알고리즘을 적용하여 영문자 인식 system을 구현하였다. 한 문자의 인식을 위해 6~8회의 탐색이 필요하였고, 실험 결과 100%의 인식율을 얻었다.

향후 연구로는 영문자뿐만 아니라 기호와 숫자도 인식하여 일반적인 문서를 인식케 함으로서 대용량의 문서 정보를 처리할 수 있는 시스템 구현도 고려할 수 있다.

참 고 문 헌

- [1] 백승복, 손영선, “손가락 이동에 의해 선택된 영역

- 의 인쇄체 한글 영상 문서화”, 퍼지 및 지능시스템 학회 논문지, 제12권 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, 하종운, “Visual C++을 이용한 디지털 영상처리”, 사이텍 미디어, pp. 184-194, 2003

저 자 소 개



손영선(Young-Sun Sohn)
1981년: 동아대학교 전자공학과 졸업
(공학사)
1983년: 동대학원 졸업(공학석사)
1990~1998년: 한국전자통신연구소
선임 연구원
1998년: 쯔쿠바대학 졸업.(공학박사)
1998년~현재: 동명대학교 정보통신대학
부교수

관심분야 : 휴먼인터페이스, 퍼지 측도·적분, 평가
Phone : 051-610-8374
Fax : 051-610-8846
E-mail : yssohn@tu.ac.kr



김보성(Bo-Sung Kim)
2008년: 동명대학교 정보통신공학과
졸업예정(공학사)

관심분야 : 휴먼인터페이스, 퍼지이론
Phone : 010-2041-8580
Fax : 051-759-5448
E-mail : kbs8580@gmail.com