

인공지능 기법을 이용한 텍스트 인식에 관한 연구

(A Study On The Text Recognition Using Artificial Intelligence Technique)

李 幸 世*, 崔 太 永*, 金 榮 吉*, 金 晶 雨*

(Haing Sei Lee, Tae Young Choi, Young Kil Kim, and Jeong Woo Kim)

要 約

한글 텍스트 인식 알고리즘으로서 획교차수, 구문적 패턴인식 과정, 하향식 인식구조와 heuristic 접근법을 검토하였다. 1) 한글문자에서 제한 스캐닝법을 이용한 모음분리, 2) 획 두께를 이용한 획 추출법, 3) 획교차수와 그의 성질, 4) 획교차수의 평균치, 표준편차 및 최빈수의 활용, 5) 제한 한자의 획교차수를 이용한 분류 및 인식법을 제안하였고, 컴퓨터를 이용하여 이들에 관한 시뮬레이션과 실험을 행하였다.

Abstract

Stroke crossing number, syntactic pattern recognition procedure, top down recognition structure, and heuristic approach are studied for the korean text recognition. We propose new algorithms: 1) korean vowel seperation using limited scanning method in the korean characters, 2) extracting strokes using stroke width method, 3) stroke crossing number and its properties, 4) average, standard deviation, and mode of stroke crossing number, and 5) classification and recognition methods of limited chinese character. These are studied with computer simulations and experiments.

I. 서 론

한글의 창제이후 우리 민족은 고유 문자 문화의 혜택을 누리게 되었다. 그러나 한자 문화와의 교류를 통한 발전과 아울러 반문화적 요소가 동시에 개입되어 한글문자문화의 독창성이 한자혼용으로 손상되어 있으며, 최근 영어문화와의 잦은 교류로 인해서 영문자 혼용이 증대되고 있다. 기계화 작업에서 고유한 문자문화의 이점을 충분히 발휘하지 못하는 사례

가 나타나기도 한다. 문자문화는 우리 역사적 사실 속에서 성장해 오고 또 우리들이 가꾸어온 소중한 문화 유산이다. 서로 다른 언어학적 상황을 고려하지 않고서 접근하였을 경우 우리의 소중한 유산을 파괴할 가능성이 높아진다. 한글의 문자 문화를 가꾸어야 하는 막중한 소임을 의식하고, 기계화 측면, 문화적 측면, 예술적 측면을 총합한 의미의 발전에 기여하여야 할 것이다.

한글 기계화 분야는 끊임없이 개혁이 요구되는 분야이므로 상당한 수준으로 실용화되기 까지에는 앞으로 많은 도전이 있을 것이 기대된다. 강 인규, 이 행세는 1969년에 한글의 특징추출에 관해서 8개 방향성분의 획으로써 표시할 것을 제안하였다. 그 이후 이 주근을 비롯한 많은 연구가에 의해서 한글 문

*正會員, 亞洲大學校 電子工學科

(Dept. of Elec. Eng., Ajou Univ.)

接受日字: 1989年 6月 29日

(※ 본 논문은 1987년도 학술진흥재단 자유공모과제 연구비에 의하여 수행되었음.)

자 인식에 관한 연구가 진행되었다.^{1)~3)} 인식은 주로 획과 이들 상호관계로서 특징점이 사용되었다. 또한 인쇄체 한글과 인쇄체에 준하는 필기체 한글에 관한 알고리즘들이 개발되었다. 또한 문법적 인식이 수행되었고, 문법은 트리문법(tree grammar), 정규문법(regular grammar), 문맥자유문법(context free grammar)이 획과 특징점을 수용하도록 변형되어 사용되었다.

여기서는 한글 텍스트가 문자열, 표, 화상으로 구성된 것으로 보고 이것을 추적하는 방안에 대해서 검토하였고, 문자에서도 한글, 한자, 숫자, 영문자, 특수기호로서 구성된 것으로 규정하였다. 기존 개발된 인식 알고리즘을 수용하고 발전시키기 위해서 획 추출에 관한 고찰이 수행되었고, 제한 스캐닝법을 이용한 모음분리 및 인식과 아울러 획과는 전혀 성질을 달리하는 획교차수를 도입하여, 문자별 분류 및 인식에 활용토록 하였다. 컴퓨터 출력으로 사용하는 자형(font)과 이미지 스캐너로 입력한 자료를 처리하여 시뮬레이션과 실험을 수행하였다.

II. 한글 텍스트의 특징

한글은 우리의 고유한 음운을 정확히 표현하면서도 아울러 기존 한자 문화를 수용할 수 있도록 창제되었다. 한글은 한자를 한글문자와 문자 대 문자의 대응이 가능하게 하였고, 한 문자가 한 음절을 표시하여 읽고 쓰기에 편리하다. 또한 한글은 자모가 서로 명확히 구분되어 혼동의 소지가 거의 없다. 한글은 음운의 경우 표기된 자모는 거의 그대로 발음되므로 발음기호를 사전으로 만들 필요가 없으므로 음성합성이나 음성인식에 탁월한 장점이 나타나게 된다. 따라서 한글은 자모가 발음기호를 겸하는 언어로 볼 수 있다. 중국어나 영어의 경우 별도의 발음기호가 필요하고, 일본어의 경우에도 발음기호를 겸할 수도 있으나 자모가 분리가 안되어 있으므로 자모가 분리된 발음기호를 빌려 다시 표기해야 할 것이다.

문자가 음절단위로 조합되며 각 자소의 음가가 거의 유일하고 명확하므로 인간이 읽고 쓰기에는 대단히 편리하지만, 기계화에서는 조합된 자모를 분해하여 인식하고 표기해야 하는 어려움이 있다. 풀어쓰기를 할 경우에 기계적인 인식속도는 빠르게 되나, 인간의 인식속도는 느리게 된다. 물론 풀어쓰기에 적합한 문자를 다시 창안하기까지는 그렇다. 그리고 이것은 새로운 문자문화를 의미하는 것이다.

문화적 특징으로써 한글 텍스트에는 한자와 영자, 숫자 및 특수기호를 포함한다. 한글, 숫자, 한자, 영

문자, 특수기호는 세 부분으로 구분하면, 한글, 한자, 영숫자(숫자, 영문자, 특수기호)로 구분한 다음, 영숫자를 숫자, 영문자, 특수기호로 다시 구분하면, 문자 인식에 도움이 될 수도 있다. 개별 한글 자모가 기호로 이용될 때는 특수기호에 포함시킬 수 있다. 한글 텍스트에서 한글의 날자수는 대략 11,000 자이며, 일상생활에 흔히 사용되는 날자수도 1,500자 정도이며, 상용한자 2,000자와 영어 대문자와 소문자 각 26자와 숫자 10자 및 기호 몇십자 정도가 이용된다. 실제 텍스트의 특징은 문화적 성격이 포함되므로 한마디로 표현하기는 곤란하며 전체적 성격을 파악하고 필요에 따른 제한 설정을 필요하다고 본다.

한글 텍스트의 특징을 규정하기 위하여 다음이 정의되어야 한다.

- 1) 문자; 한글, 숫자, 한자, 영문자, 특수기호.
- 2) 자료; 설명문, 도표, 사진.
- 3) 연계; 설명문의 연계구조, 도표 사진의 설명문.
- 4) 계층구조; 표제어, 문단.

1)의 경우 문자는 한글 외에 어떤 것이 포함되어 있고, 그 비율이 어느 정도인지 추정되어야 한다. 2)의 경우 텍스트에 설명문 외에 도표나 사진이 포함되어 있는지, 있으면 어떤 특징을 가지는지가 구분되어야 한다. 3)의 경우 가로쓰기와 세로쓰기 또는 신문처럼 혼합구조인지가 추정되어야 하고 신문과 같은 내용의 연결이 어떻게 이어지는지 정의되어야 한다. 또한 도표나 사진의 설명문을 어떻게 연결시켜 줄 것인지 정의되어야 한다. 4)에서 책과 같은 경우 편, 장, 절 등의 기호와 표제어 등이 정의되어야 한다. 또는 문단의 시작은 글자의 위치에 의존하는 것인지, 라인과 라인 사이의 간격에 의존하는 것인지가 정의되어야 한다. 이들 외에도 인식을 위한 각종 자료가 정의 또는 추정되어지고 기초자료로서 입력되어야 한다. 2)항에서 4)항까지는 데이터 베이스에 관련된 것이며, 1)항은 입력문자로서 인식대상이다. 2)항을 처리하기 위해서는 구분 알고리즘이 제시되어야 하는데 인공지능기법의 이용이 요청된다.

하향식(Top-down) 방식으로 추적하는 텍스트 특징 추출 나무를 그림 1에 표시하였다. 한글 텍스트는 크게 문장 영역, 도표, 화상 영역으로 구분되는데 이 부분은 비교적 기계화하여 자동으로 처리하기에는 어려운 요소가 많이 포함되어 있다. 텍스트에서 문장은 문자열이 질서 정연하고 문자 간격도 비교적 모듈화되어 있어서 이것을 기준으로 사용하면 분리가 가능하다. 또한 글자는 활자 크기로써 제한되므로 구역을 분할한 후 스캐닝하여 활자외곽 형태의 직사각형

을 추출할 수 있다. 대개 화상은 이와 같은 규칙에 어긋나므로 쉽게 검출된다. 그러나 이와 같은 규칙을 가진 화상의 경우에는 문자에 속한 분류로 끼어들게되나 인식 결과로서 검출되게 된다.

문자 인식에서 텍스트에 따라서 자주 쓰이는 한자는 제한되어 있으므로 매칭 알고리즘을 별도로 구성하거나, 인식에서 제외하고 한자 위치 표시와 피인식 문자데이터를 화상 데이터로 취급하여 각주처럼 별도로 공급하는 방식이 고려 대상이 될 수 있다. 인식의 경우에 한자는 결합 특징과 결합 규칙이 복잡하여 한글처럼 취급하기에는 곤란한 점이 많다. 회교차수를 이용한 분류 및 인식이 다음 장에서 제시된다. 한글인식이나, 영숫자 인식방법은 상당히 발전되어 있어 이들을 활용하면 편리하다.²⁻¹¹⁾ 문자 특징 추출나무가 그림 2에 표시되었다. 검토한 바와 같이 한글 텍스트의 가장 큰 애로점은 한글인식, 한자인식, 영숫자 인식에 관한 알고리즘이 서로 크게 다르게 되는 점이다. 따라서 개별문자 식별에 앞서 먼저 한글인지 한자인지 영숫자인지 구분이 앞서야 하는데 문제가 있다. 이 문제를 해결하려면 종래의 인식 알고리즘에는 포함되지 않았던 그림 2와 같은 하향식 추적이 필요하다. 이를 실현하기 위해서 인공지능 기법 활용이 요청된다.

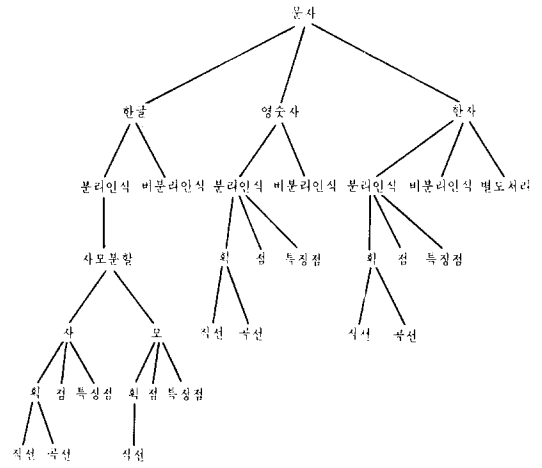


그림 2. 문자특징 추출나무

Fig. 2. The characteristic extraction tree of character.

III. 획 추적 및 한글의 자모 분할

1. 전처리 및 회추적

본 실험을 위해서 마이크로 컴퓨터 IBM PC-AT에 준하는 마이크로 컴퓨터와 이미지 스캐너 MICR-OTEK MSF300C와 인터페이스를 이용하여 데이터를 수집하여 실험하였다. 이미지 스캐너의 해상도는

300dpi이다. 한글의 경우 한 글자의 이미지는 (40~50) × (40~50), 한자의 경우 51 × 49를 사용하였다. 아울러 컴퓨터 인쇄체 폰트로서 완성형 한글과 한자 16 × 16을 사용하였다. 2진 데이터는 회 또는 흑색을 1로, 공백 또는 백색을 0으로 변환하여 취급하였다. 2진 데이터를 이용하여 이미지 단위 페이지를 구성하였다. 이를 종방향 및 횡방향으로 스캐닝함으로써 1의 갯수를 계수하여 이를 이용하여 0으로 둘러싸인 라인을 추적하였고, 화상을 분리할 수 있었다. 그림 3은 이렇게하여 얻은 문자열을 표시한다.

한글은 결합구조에 따라 자모가 적응 수축 확장되어 형태가 달라지며 골격과 형태에 따른 미적감각의 부여가 용이하다. 그러나 분할 인식의 경우 이것이 약간의 장애 요인으로 되어 영역 분할에 제약점으로 나타난다. 또한 인간에게는 획이 굵을수록 중요하게 느껴지고, 미적 감각이 많이 포함되게 되며 잡음이 적어서 확실하게 느껴진다. 그러나 종래 사용해오던 세선화 알고리즘은 한점 한점 판정해서 소거하므로

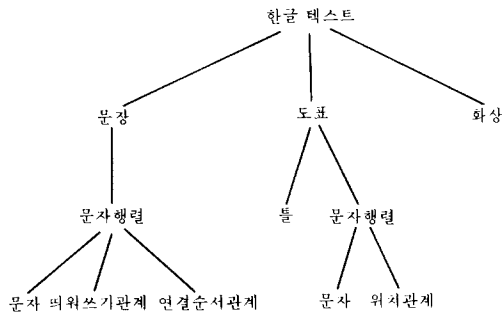


그림 1. 텍스트특징 추출나무

Fig. 1. The characteristic extraction tree of korean text.

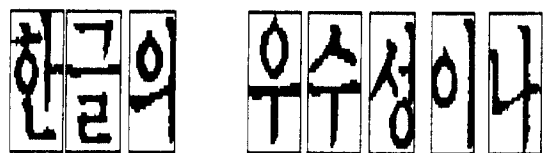


그림 3. 문자열 digitizing 된 것

Fig. 3. Character Digitizing.

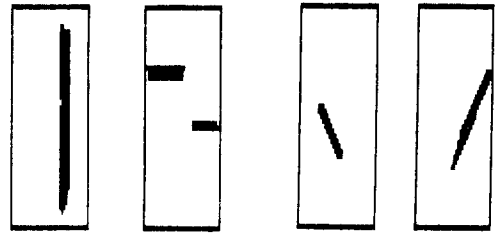
시간이 걸리고 세션화 된 결과만 사용할 경우는 회에 관한 정보이외의 중요한 정보를 소실하게 된다. 또한 세션화 작업 후에 다시 회이 어떻게 되는지 추적하여 판정하는 과정을 거치게 된다. 반면에 회이 흑점 1~2개 정도로 가늘어지면 보다 효과적으로 처리되나, 양자화 잡음 등의 개입되는 잡음의 영향이 커지고 신뢰성이 낮아진다. 현재로는 200~300dpi 짜리 이미지 스캐너로서는 흔히 쓰는 텍스트가 상당히 굵게 회이 나타나고, 세션화에 드는 시간이 일반적으로 크게 소요된다.

한글의 경우는 수평선과 수직선이 주종을 이루고 있으며 사선 성분의 빈도는 비교적 낮고 곡선은 0에 한정되어 있다. 인쇄체 한글의 경우 회이 굵기가 비교적 균등하다고 가정하면 복잡한 세션화 알고리즘들이 사용하는 대신에 간단한 회을 찾아내는 방법이 가능하게 된다. 세션화 알고리즘을 대신하여 회을 추적하는 방법이 Y. Yamashita [12]에 의해 제안되었으나 회 두께를 구하는 식이 다른 것으로서 회 두께(stroke width)에 의한 회 추적 알고리즘을 제안한다. 횡방향과 종방향으로 스캐닝하여 얻은 연속된 1의 갯수를 계수하여 회성분의 두께의 히스토그램을 구하고, 피크치를 중심으로 하여 두께의 상한과 하한값을 택하였다. 이 두께에 해당하는 회을 추적한 결과가 그림 4-a에 나타나있다. 추적한 회을 제거한후 같은 기법으로 45° 및 135° 방향으로 스캐닝하여 회을 추적한 결과가 그림 4-b에 나타나 있다. 이 방법은 블럭 단위로 추적하여 가므로 회이 두터울 때 세션화 알고리즘보다 효과적이라고 생각한다. 또한 회이 추적되면서 아울러 회의 두께, 길이, 방향 및 위치에 대한 정보가 얻어진다. 나머지 잔류 이미지는 이미 얻어진 회을 기초로 하여 분석한 다음에 추가적 정보로 취급하여 사용한다.

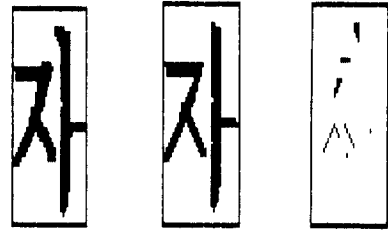
2. 한글의 모음 추출과 그 응용

텍스트에 나타난 한글의 특징은 다음과 같이 요약할 수 있다.

- 1) 종횡의 직선 선분의 빈도가 대단히 크다.
- 2) 직선에는 긴 직선과 짧은 연결선으로 구분할 수 있다.
- 3) 짧은 연결선은 모음의 경우 대개 직각으로 연결된다.
- 4) 사선 빈도는 비교적 낮고 곡선은 O 형에 국한된다.
- 5) 낱자의 경우 4개의 변두리에 모든 자모가 걸쳐있다.
- 6) 회이 굵기는 비교적 평탄하다.



(a) 수직, 수평 추적 결과 (b) 45°, 135° 추적 결과



(c) 원 이미지, 합성 이미지, 잔류 이미지

그림 4. 회 두께에 의한 추적 결과

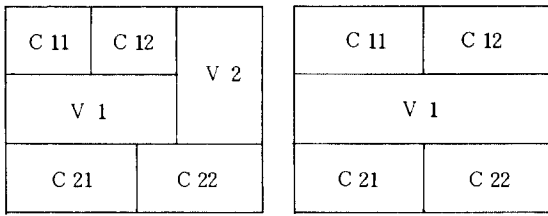
Fig. 4. Stroke extraction using stroke width method.

7) 회이 끝부분이 약간 굵은 경우가 명조체의 경우에 빈번히 나타난다.

8) 비교적 자모는 분할되어 있다. 그러나 대부분 텍스트에서는 결합된 형태가 나타난다.

한글은 글자 구성이 자모의 결합이며, 그 결합은 자모 2개부터 자모 7개까지 사용되어 결합관계가 비교적 복잡하다. 더우기 조합된 자소가 때로는 밀착하거나 결합되므로 인식의 입장에서 보면 분리 문제가 상당한 비중을 가지게 된다. 더우기 한자, 영숫자가 추가되면 인식의 문제는 복합된 알고리즘을 추구할 수 밖에 없다. 제한된 한자의 경우는 사전식 템플리트를 작성하여 매칭하는 방법이 바람직하기 때문이다. 여기에 대해서 한글은 자모 분리하여 자모를 인식하여 문자를 인식하는 방법이 유리하고, 영숫자는 전체가 수십자에 불과하고 지극히 간단하므로 자소분할 과정을 거치지 않고 특징추출하여 인식하거나 템플리트 인식하는 방식들이 모두 허용될 수 있다.

한글의 자모의 위치를 표시하는 형태는 그림 5-a와 같이 표시된다. 그림에서 C11과 (V1 또는 V2)가 기본적 요소이고 기타는 부차적 요소이다. 사용되지 않은 경우 면적이 영이된다. 그리고 형태적으로 각 부분은 직사각형을 이루며 전체 틀인 직사각형을 완

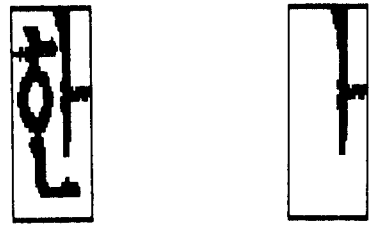


(a) 자모 구성 형태 (b) 수직모음 제거시 자모 구성 형태

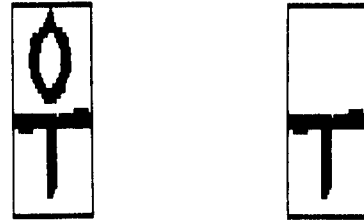
그림 5. 자모 구성 형태
Fig. 5. The construction structure of Korean character.

전히 채우는 위상 기하학적 표시를 가정하면 모든 문자의 표현이 가능하다. 현대 한글에서는 C 11과 C21에는 기본자음 14자가 사용될 수 있으며, C12에는 ㄱ, ㄷ, ㅂ, ㅅ, ㅈ의 5자가 사용되고 쌍을 이룰 때 (중자음)에만 사용된다. V1은 수평모음이라 부를 수 있고, 수평모음은 ㅡ·ㅓ·ㅕ·ㅗ·ㅛ의 5자가 사용되고, V2는 수직모음이라 부를 수 있고, 수직모음은 ㅣ, ㅜ, ㅠ, ㅡ, ㅝ, ㅟ, ㅞ, ㅟ의 9자가 사용된다. V2의 마지막 4자를 보면 창제 당시에는 ㅟ가 ㅜ와 ㅣ로 구별되는 식으로 두개의 모음으로 구분되었으나 현대 한글에서는 거의 구분되지 않으므로 문자형태(morphology)적으로 한개의 모음으로 취급하는 것이 적합하다고 본다. C 22는 중자음이나 복자음을 이루는 것이며 14개의 자음 중에서도 한정된 결합에 의한 것이다.

한글의 자모에서는 수직, 수평의 형태를 취하는 것이 모음이며, 수직모음의 경우 빈도는 대단히 높고, 있다면 반드시 우측에 나타나므로 추적이 용이하다. 이를 추적하기 위해서 제한 스캐닝법(limited scanning method)을 제안한다. 이것을 설명한 것이 그림 6에 나타나있다. 즉 문자의 우측상단에서부터 횡방향으로 우측에서 좌측으로 스캐닝하되 계속되는 흑점만을 추적하고 백점이 나타나면 스캐닝을 끝낸다. 이것을 순서적으로 위에서 아래로 계속하면 쉽게 수직모음의 획을 추적하게 된다. 이렇게하여 수직모음을 자연 분리 또는 강제분리할 수 있다. ㅣ, ㅜ, ㅠ가 추적되면 계속 추적하여 ㅟ, ㅞ, ㅟ, ㅟ 인지를 추적한다. 수직모음을 제거하면 나머지는 그림 5-b와 같은 적층형이 된다. 이때 최상부에는 반드시 자음이 있음을 알 수 있다. 수직모음이 인식되면 일단 한글인식 과정에 진입하고 실패한 경우 같은 방법으로 수평모음을 추적한다. 이렇게 하여 둘다 실패한 경우



문자 수직모음 추적



문자 수평모음 추적

그림 6. 제한 스캐닝 추적법
Fig. 6. Limited stroke scanning method.

는 한글이 아님을 알 수 있다. 성공한 경우는 한글 인식 알고리즘에 모음 종류와 글자 형태에 관한 사전 지식을 가진 접근(heuristic approach)이 가능하다. 이렇게 하여 그림 2의 하향식 추적에서 중요한 정보가 획득된다. 회교차수에 의한 정보는 다음 장에서 설명하기로 한다.

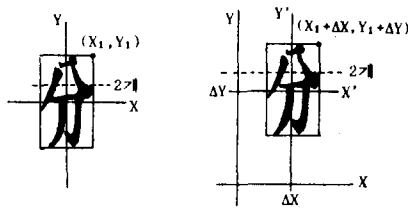
IV. 회교차수를 이용한 문자 분류 및 인식

1. 회교차수의 성질

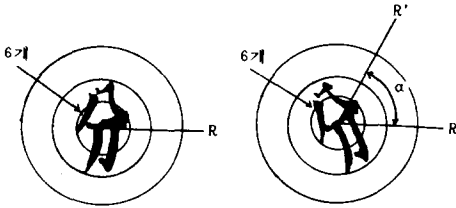
인쇄체 단일 자형(font)을 가진 텍스트에서 회교차수는 임의의 선분이 문자와 교차하는 수를 나타내며 이것은 그래프의 위상기하학적 성질을 가진다. 따라서 문자의 이동, 회전, 스케일링(scaling)에 대하여 무관한(invariant)한 성질을 나타내는 처리가 가능하다.

1) 평행이동에 대한 성질

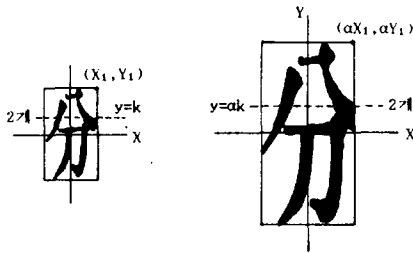
문자가 평행 이동하였을 경우 두 문자의 무게 중심을 원점으로하여 각각에 직교 좌표축 XY를 설정할 수 있고, 문자를 포함하는 XY축에 평행한 경계면으로써 최소 직사각형을 설정할 수 있다. 이때 두 최소 직사각형은 합동이되며 또한 직사각형 안의 두 문자도 합치된다. 그림 7 (a)와 같이 임의의 합치되는 선분을 겹쳐 그을 경우 회교차수는 각각 동일하다.



(a) 평행 이동



(b) 회 전



(c) 스케일링

그림 7. 회교차수의 성질
Fig. 7. The characteristic of stroke crossing number.

2) 회전에 관한 성질

변형없이 회전한 두 문자에 대해서 각각의 무게 중심에 극좌표 r, θ 를 설정하고, 중심에서 방사선 방향으로 회교차수가 최대가 되는 구간을 설정하고 이를 겹치도록 회전하면 두 문자가 합치될 수 있다. 그림 7 (b)와 같이 좌표축의 회전 차이값을 편차로 주면 합치되는 좌표축을 얻고, 이 때 문자도 합치된다.

3) 스케일에 관한 성질

문자의 형태는 유지하면서 배율을 일정하게하여 바꾸었을 경우 각 문자의 무게 중심에 좌표의 원점을 설정하고 직교좌표 XY 축을 설정하면서 문자를 포함하면서 XY 축에 평행한 경계면으로써 최소 직사각형을 설정할 수 있다. 그림 7 (c)와 같이 임의 선분과 만나는 한 문자의 회교차수는 다른 문자와의 배율에 해당하는 배율로 선분을 변환하였을 때 같은 회교차

수를 항상 나타낸다. 따라서 문자의 형태가 유지되면 회교차수는 이동, 회전 및 스케일링에 무관하다.

실제로 이미지스캐너로 문자를 입력할 때 두 가지 형태의 양자화가 이루어진다. 첫째 평면을 픽셀(pixel)로 구분하는 샘플링(sampling)이 이루어진다. 둘째 회색준위(gray level)의 임계값(threshold)의 선택에 따른 이진 양자화가 이루어진다. 따라서 양자화에 따른 오차와 아울러 평행이동, 회전, 스케일에 따른 추가적인 오차 개입이 따른다. 여기서는 하나의 문자체(font)를 가지는 텍스트에 대해서 국한하였으나, 다중문자체(multi-font)에 대해서도 확대 적용할 수 있다.

2. 분류법

회교차를 이용하여 다음과 같이 분류 방법을 제안한다.

임의의 방향으로 회교차수를 연속적으로 구해서 얻은 벡터를 회교차수 벡터로 정의한다. 또한 회교차수 벡터로부터 요소의 값에 관한 빈도수를 계산하고 빈도수가 1 이상인 경우 최대빈도 교차수부터 나열하여 회교차수 빈도순벡터를 얻는다. 만일 빈도수가 같은 것이 2개 이상일 경우 교차수가 큰것을 우선하도록 한다.

1) 평균치에 의한 방법

직교좌표계 X, Y 축으로 회교차수를 구하고 각각의 평균치를 얻는다. 사용하고자 하는 모든 문자에 대해서 구한 결과를 이용하여 최대 평균치값을 택한다. 분류하고자 하는 최대구간값을 다음과 같이 구한다.

$$\text{최대구간} = [\text{회교차수 평균치의 최대값} \times \alpha] + 1$$

단 []는 실수의 정수부분을 의미한다.

이렇게 하여 얻은 구간을 이용하여, 두 방향으로 분류하면 매트릭스 형태의 분류가 가능하다. 표 1은 컴퓨터 출력용 완성형 한글 폰트 1,800자와 한자 폰트 5,000자를 회교차수 빈도순 벡터의 평균치값을 계산하여 해당하는 구간에 계수하여 분류한 것이다.

2) 표준편차를 이용하는 방법

1)과 동일하나 평균치 대신에 각 문자의 표준편차를 구하여 표준편차의 값으로 분류하는 방법이다.

$$\text{최대구간} = [\text{회교차수 표준편차의 최대값} \times \beta] + 1$$

이렇게 얻은 구간을 이용하여 매트릭스 분류를 행한다. 표 2는 표 1 과 동일한 대상을 표준편차로 분류한 것이다.

3) 최빈회교차수를 이용하는 방법

각 문자에 대해서 X, Y 방향으로 회교차수를 구하

표 1. 획교차수 평균치를 이용한 문자 분류

Table 1. The character classification using the average of stroke crossing number.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	1	0	1	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	4	7	0	9	4	10	6	1	4	4	0	0	1	2	0	0	0	0	52
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	2	0	10	29	3	48	5	42	22	9	22	19	6	9	0	2	1	0	0	1	230
6	0	0	0	2	0	2	0	4	1	0	4	7	0	3	0	0	0	0	0	0	23
7	2	0	7	35	3	80	21	117	110	33	92	63	14	39	12	23	2	0	0	0	653
8	0	0	0	0	0	2	0	3	2	2	4	1	1	3	1	0	0	0	0	0	19
9	1	0	13	45	3	129	26	198	181	66	180	133	31	63	16	26	11	0	0	0	1122
10	0	0	0	0	0	4	0	2	8	2	3	1	1	3	0	2	0	0	0	0	26
11	0	0	0	1	1	3	1	10	14	12	15	12	4	11	2	7	1	0	0	0	94
12	0	0	7	33	8	150	19	219	243	63	213	157	48	56	13	16	7	0	0	0	1252
13	0	0	0	4	0	11	1	17	6	2	8	10	3	6	1	2	0	0	0	0	71
14	1	0	3	31	1	102	16	165	196	70	134	91	21	21	4	3	1	1	0	0	861
15	0	0	1	1	0	10	1	14	16	2	16	10	2	3	0	0	0	0	0	0	76
16	0	0	1	13	0	31	11	76	83	23	65	41	4	13	0	2	1	0	0	0	364
17	0	0	1	2	1	7	0	10	10	3	10	4	2	4	0	0	0	0	0	0	54
18	0	0	0	4	0	11	2	11	18	5	16	6	2	3	0	0	0	0	0	0	78
19	0	0	0	0	1	2	1	0	2	0	0	1	0	0	0	0	0	0	0	0	7
20	0	0	0	0	0	0	1	4	2	2	4	0	0	0	0	0	0	0	0	0	13
	7	0	48	209	21	602	109	902	920	295	790	560	139	237	50	85	24	1	0	1	5000

(a) 한자 폰트(5,000자)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	1	0	0	3	0	2	9	2	0	10	3	6	7	0	0	4	0	1	4	0	52
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	13	0	0	33	1	0	100	6	5	95	2	5	47	23	4	10	1	345
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	8	0	1	56	8	0	103	11	0	71	5	5	16	4	5	0	1	294
10	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	2
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	2	0	0	40	0	1	216	11	0	275	18	2	109	8	12	28	3	1	3	0	729
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	14	0	1	101	2	0	145	12	0	67	4	4	9	1	1	0	0	361
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	1	12	2	0	10	0	0	0	0	0	2	0	0	0	0	27
	3	0	0	78	0	6	429	26	0	643	50	13	349	19	26	106	31	12	17	2	1810

(b) 완성형 한글 폰트(1,810자)

표 2. 회교차수 표준편차를 이용한 문자 분류

Table 2. The character classification using the standard deviation of stroke crossing number.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20		
1	1	0	0	6	0	5	1	2	1	0	0	0	0	0	0	0	0	0	0	0	0	16
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	21	0	61	1	58	10	34	18	16	6	4	5	1	0	1	0	0	237	
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	17	0	133	0	192	20	182	60	116	45	58	47	20	10	3	5	0	909	
8	0	0	0	6	0	11	0	7	3	3	6	3	4	0	0	0	0	0	0	0	0	43
9	1	0	0	13	0	150	0	288	25	321	69	268	82	133	86	45	15	12	3	1	1512	
10	0	0	0	4	0	16	0	20	1	17	15	17	7	9	9	3	1	2	0	0	121	
11	0	0	0	4	0	60	0	203	13	265	41	236	52	143	73	42	16	9	1	1	1159	
12	0	0	0	1	0	15	0	24	1	28	10	33	16	21	17	11	3	1	1	0	182	
13	0	0	0	0	0	4	0	11	0	9	6	9	1	2	3	0	1	0	0	0	46	
14	0	0	0	4	0	28	1	61	6	127	21	136	27	80	26	20	4	3	1	0	545	
15	0	0	0	0	0	2	0	10	2	12	4	12	5	7	3	4	0	2	0	0	63	
16	0	0	0	0	0	1	0	14	2	26	3	33	5	16	8	1	2	0	0	0	111	
17	0	0	0	0	0	2	0	2	0	6	3	7	0	4	0	1	1	0	0	0	26	
18	0	0	0	0	0	1	0	2	1	5	0	6	0	0	1	0	0	0	0	0	16	
19	0	0	0	0	0	1	0	2	0	0	1	3	0	0	0	1	0	0	0	0	8	
20	0	0	0	0	0	0	0	1	1	3	0	1	0	0	0	0	0	0	0	0	6	
	4	0	0	76	0	490	3	897	86	1038	257	896	250	477	278	149	53	33	11	2	5000	

(a) 한자 폰트 (5,000자)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	10	0	0	58	5	87	12	3	58	6	19	1	2	0	0	3	3	268
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	2	0	0	54	0	0	246	8	338	10	32	203	35	78	21	14	16	7	3	0	1067
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
15	0	0	0	23	0	0	117	4	160	8	4	94	6	17	4	0	0	2	0	0	439
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	3	0	0	14	1	12	2	0	0	0	0	0	0	0	0	0	0	32
20	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	2
	3	0	0	91	0	0	438	18	597	32	39	355	47	114	26	16	16	9	6	3	1810

(b) 완성형 한글 폰트 (1,810자)

교 단일방향에 대해서 가장 많은 빈도를 나타내는 회교차수를 택한다. 이때 같은 빈도의 회교차수가 다수일 경우 제일 큰 회교차수를 택하여 최빈 회교차수라 부른다.

$$\text{최대구간} = \text{최빈 회교차수의 최대치}$$

이렇게 얻은 구간을 이용하여 매트릭스 분류를 행한다. 표 3 은 표 1 과 동일한 대상의 한글, 한자와 컴퓨터 출력용 영어 대문자와 소문자, 그리고 특수기호를 최빈 회교차수로 분류한 것이다. 각 문자의 최빈 회교차수를 구하고, 해당하는 구간에 해당 문자의 갯수를 계수하여 분류한다.

4) 회교차수 벡터

각 문자에 대해서 X, Y 방향으로 회교차수를 구하고 회교차수를 나열하여 두 개의 벡터를 얻는다. 활용상 벡터의 차수를 통일코저 할 경우 0 를 양 끝에 첨가할 수가 있다.

3. 회교차수를 이용한 문자 인식

회교차수 벡터를 이용한 문자 인식 알고리즘을 정의하면 다음과 같다.

- 1) 회교차수 평균치에 의한 문자 분류를 행한다.
- 2) 각 구간마다 해당하는 문자의 부호와 회교차수 벡터를 저장한다.
- 3) 문자를 입력하여 회교차수 벡터를 구하고 회교차수 평균치에 의한 분류를 행한다.
- 4) 입력 문자에 해당하는 구간의 문자를 택하여 회교차수 벡터를 읽어낸다.
- 5) 매칭도를 계산한다. 매칭도는 X, Y 방향으로 얻은 회교차수 벡터를 순서대로 비교하여 같으면 1 을 더하고 다르면 차이를 뺀다.
- 6) 구간에 있는 모든 문자를 전부 계산했으면 7) 로 가고 아니면 4) 로 간다.
- 7) 매칭도가 문턱값을 넘은 것이 하나 있으면 문자를 인식한 것으로 한다.

표 1 에서와 같이 회교차수 평균치를 이용하여 문자를 분류하면, 20×20으로 분류한 한자의 경우 5,000자의 대상이 최대구간에서 1/20 이하로 분류되었다. 회교차수를 이용한 여러가지 분류법은 활용하면 문자를 그림 2 와 같이 대별하는데 사용될 수 있다. 한자 250자를 이미지 스캐너로 입력하여 회교차수를 이용한 한자 인식으로 회교차수 평균치와 표준편차를 이용한 20×20 분류의 각 표에서 한구간에 포함되는 20자를 선택하였다. 이들은 표 4 와 같다. 선택된 문자를 이미지 스캐너로 다시 입력하여 인식한 결과 회교차수 평균치에 의한 재분류는 17개가 같은

표 3. 최빈 회교차수를 이용한 문자 분류
Table 3. The character classification using the mode of stroke crossing number.

	0	1	2	3	4	5	6	7	8	9	
0	0	1	1	2	0	0	0	0	0	0	4
1	0	12	43	91	106	119	105	76	26	0	578
2	1	83	281	336	270	248	178	77	15	0	1489
3	4	123	338	442	308	235	104	36	2	0	1592
4	4	111	239	274	192	111	25	9	0	0	965
5	0	59	84	86	50	29	6	0	1	0	315
6	0	9	16	11	5	4	3	1	0	0	49
7	0	2	2	2	1	0	0	0	0	0	7
8	0	1	0	0	0	0	0	0	0	0	1
9	0	0	0	0	0	0	0	0	0	0	0
9	401	1004	1244	932	746	421	199	44	0	0	5000

(a) 한자 폰트 (5,000자)

	0	1	2	3	4	5	6	7	8	9	
0	1	4	13	5	2	1	0	0	0	0	26
1	25	198	279	222	171	70	11	2	0	0	978
2	17	104	181	236	81	11	0	0	0	0	630
3	1	47	52	44	7	0	0	0	0	0	151
4	0	11	5	9	0	0	0	0	0	0	25
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
44	364	530	516	261	82	11	2	0	0	0	1810

(b) 완성형 한글 폰트 (1,810자)

	0	1	2	3	4	5	6	7	8	9	
0	6	24	10	1	0	0	0	0	0	0	41
1	9	12	8	2	0	0	0	0	0	0	31
2	0	15	6	1	0	0	0	0	0	0	22
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
15	51	24	4	0	0	0	0	0	0	0	94

(c) 영숫자, 특수기호 폰트 (94자)

표 4. 입력된 250자 한자중 인식용 한자의 실제 자형 (20자)

Table 4. Input hinese character for recognition.

馬	祭	旗	緊	幅
築	張	競	題	習
接	絡	區	鹽	遂
謀	親	賤	前	耕

구간값을 가졌고, 회교차수 벡터를 비교하여 매칭도로 인식한 결과 17개 모두 인식하였다.

V. 인공지능 기법을 이용한 텍스트 인식 방법

인공지능은 쉽게 말해서 컴퓨터보다 생물체가 잘 해내는 것을 컴퓨터로 하여금 추적하게 하는 기술이다. 컴퓨터는 순서적으로 처리하나, 인간에게는 평면적 때로는 입체적 인식이 가능하다. 텍스트에서도 면을 파악하고, 여러가지 종류의 문자도 하나씩 분리 인식이 가능하다. 그림 2 에서 특징추출의 방안으로써 깊이 우선(depth first) 경우 영자가 인식되었음에도 불구하고 다음 글자를 한글 인식 과정부터 시작하게 되어 시간 낭비가 크게된다. 여기서 heuristic 접근이 요청된다. 또 한편 이미 인식된 글자들중 빈도가 높은 글자는 이 자료를 이용하여 학습된 인식으로 전환할 수 있게 된다. 이 방법은 선형적 지식을 활용하여 추적하는 길을 제한하고 우선적으로 처리할 길을 선정하는 것이다. 여기서 이 과정을 수립하기 위한 알고리즘과 평가함수의 복잡성 등이 문제가 된다. 그러나 숫자 하나를 인식하기 위하여 한글 인식의 모든 과정을 거친다는 것은 무리한 요구이다. 다른 한편 숫자를 우선적으로 고려하여 모든 한글을 인식할 때 숫자로 부터 시작하는 것은 더더욱 큰 문제이다. 그럼에도 즉각적으로 한글, 한자, 영숫자를 추정하는 것은 용이하지 않다. 텍스트 인식은 그림 1 과 같이 문장, 도표, 화상으로 구성되며 다시 문장은 한글, 영숫자, 한자로 나누어진다. 영숫자 및 한글은 문맥자유문법(Context Free Grammar)를 이용하여 세선화된 데이터와 회과 특징점을 추출하

여 인식하였다. 이 내용은 류승필, 김태균의 결과[11]와 비슷하므로 생략한다. 표 3 에서 볼 수 있는 것처럼 한자와 영숫자는 최빈 회교차수 분포에서 대부분 구별되고 공통부분에 비교적 적은수가 있으므로 전후 문자의 최빈 회교차수에 의해서 구분류가 가능하다고 생각된다. 한자의 경우 전술한 것 처럼 회교차수를 이용하여 인식이 가능하다.

화상의 경우 전술한 것처럼 문자 크기 형태의 모듈화를 제시하여 구분할 수 있다. 무늬처럼 문자 형태와 같이 분류되는 경우는 문자와 같은 차원에서 분리되게 된다. 이때도 회교차수는 무늬 추정에 이용될 수 있다. 이렇게 서로 다른 알고리즘을 조화있게 다루기 위해서는 탐색(search)을 비롯한 각종 인공지능 기법이 활용된다. 텍스트 인식에서 4개의 계층적 구조를 표시하면 다음과 같다.

- (1) 문장, 도표, 화상 분리
- (2) 한글, 한자, 영숫자(영문자, 숫자, 특수기호) 분리
- (3) 문자 인식
- (4) 전처리 및 특징 추출

문자인식에는 문법 구조, 회 교차수, 신경 회로망을 이용한 패턴 인식 기법이 크게 요청되나 텍스트 인식에는 인공지능 기법이 활용되고, 데이터의 구조적 측면에서 지식 데이터 베이스가 구축되어야 할 것으로 본다. 세선화에 대해서 효율적 방법으로 블럭 분할법, 제한 스캐닝법, 회교차수에 관한 연구가 계속 되어야 한다고 생각된다.

인공지능 기법을 이용한 텍스트 인식은 계층을 구성하고 각 계층내에서 탐색 알고리즘을 수립하며, 계층간의 연계를 원활하게 하는 전문가 시스템 구축을 목표로 한다. 본 논문에서 이 부분이 개발적 검토로서 제시되었으며, 앞으로 폭넓은 검토와 실험을 거쳐 지식베이스가 구축될 것이 기대된다. 제한 스캐닝을 이용한 한글 모음 분리인식과 회교차수는 탐색 알고리즘에 중요한 정보를 제공하는 요소가 되고 있다. 이상의 알고리즘이 문자인식 전문가 시스템 구축에 도움이 되기를 희망한다.

VI. 결 론

한글 텍스트의 특징을 검토하여 계층구조를 얻었고, 계층별로 탐색트리를 구성하였다. 회교차수를 도입하여, 한글, 한자, 영숫자의 분류법 및 제한 한자인식법을 제안하고, 단일 자체의 컴퓨터 출력 문자체(font)와 이미지 스캐너로 포착한 인쇄체 문자를 사용한 실험을 통하여 알고리즘의 타당성을 얻었다.

획두께에 의한 회추적법으로 세선화 하였고, 제한적 스캐닝법으로 모음 인식법을 제안하고 실험하였다.

한글 인식에 대해서는 문맥자유문법을 이용하여 세선화된 획과 획사이의 관계로서 특징점을 사용하여 인식 알고리즘을 구축하고 실험하여 단일 자체에 대해서 타당성을 얻었으나, 이 부분은 계속 보완하고 있다.

Ⅶ. 감 사

본 논문은 1987년도 문교부 학술진흥재단 (자유공모과제)의 연구비에 의해서 수행되었다. 또한 논문심사위원의 세밀한 지적으로 논문 수정이 충실한 방향으로 이루어졌다. 본 논문의 상당한 부분이 연구실 박사과정 김석동, 송도선과 석사과정 김태호, 고연순의 협조로 이루어졌다.

參 考 文 獻

- [1] 강인구, 이행세, "한글 자체의 특징 추출의 한 방식", 대한 전자공학회 논문집 제 6 권 2 호, pp. 1~5, 1969. 9.
- [2] 이주근, "한글 문자의 인식에 관한 연구", 대한 전자공학회지 논문집, 제 9 권 4호, 1972. 9.
- [3] T. Agui et al., "A method of recognition and representation of korean character by tree grammer," IEEE Trans. Pattern Anal. Mach. Intell. vol. PAM-1, pp. 245-251, July 1979.
- [4] 이균하, "속성에 구속을 받는 문법을 이용한 문자 패턴 인식", 인하대학교 박사학위 논문, 1981. 2.
- [5] 최병욱, T. Ichikawa, H. Fujita, "한글 인식에 있어서의 자소 추출", 대한 전자공학회 논문집, 제 18 권 2 호, pp. 36~43, 1981. 4.
- [6] 이주근, 남궁 재찬, 김영건, "한글 Pattern 에서 Subpattern분리와 인식에 관한 연구", 대한 전자공학회 논문집, 제 8 권 3호, pp. 1~8, 1981. 6.
- [7] 남궁 재찬, "Index-Window 알고리즘에 의한 한글 Pattern의 부분분리와 인식에 관한 연구", 인하대학교 박사학위 논문, 1982. 2.
- [8] Soo Dong Lee, "An artificially intelligent pattern recognition system for Korean characters recognition," 울산공대 1984. 6.
- [9] 남궁 재찬, 류형빈, 남궁 윤, "한국어 문서로부터 문자분리 및 도형추출에 관한 연구", 대한 전자공학회 논문집. 제 25 권 9호, pp. 73~83. 1988. 9.
- [10] 이희동, 김태균, T. Agui, M. Nakajima, "확장된 DP매칭법에 의한 홀림체 한글 온라인 인식", 대한 전자공학회 논문집. 제 26 권 1호, pp. 29~37. 1989. 1.
- [11] 류승필, 김태균, "속성문법을 이용한 필기체 한글 문서내의 자모인식", 대한 전자공학회 논문집. 제 26 권 3호, pp. 85~93. 1989. 3.
- [12] Y. Yamashita, K. Higuchi, Y. Yamada, and Y. Haga, "Classification of handprinted Kanji characters by the structured segment matching method," Pattern Recognition Letters 1 (1983) pp. 475-479. July 1983.

著者紹介



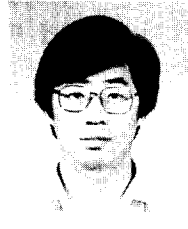
李幸世(正會員)

1943年 8月 29日生. 1966年 전북대학교 전기공학과 공학사. 1972년 서울대학교 전자공학과 공학석사. 1984년 고려대학교 전자공학과 공학박사. 1973년~현재 亞洲大學校 教授. 1982년~1983년 미국 Columbia Univ. (N. Y.) 객원교수. 1987년~1988년 프랑스 INRIA (Paris) 객원교수. 1968년~1970년 해군사관학교 전자공학 교관. 1973년~현재 IEEE 회원. 주관심분야는 문자 및 음성 인식, 인간기계 인터페이스, 인공 지능, 신경회로망, VLSI 회로, 디지털 신호처리 등임.



崔太永(正會員)

1950年 10月 24日生. 1970年 2月~1974年 2月 서울大學校 工科大学 電子工學科卒業(工學士). 1976年 9月~1978年 8月 서울大學校 大學院 電子工學科(工學碩士). 1979年 10月~1982年 12月 프랑스 Aix-Marseill 3大學(工學博士). 1978年 9月~現在 亞洲大學校 工科大学 電子工學科 副教授.



金榮吉(正會員)

1954年 10月 3日生. 1978年 고려대학교 전자공학과 졸업. 1979년 체신부 전자교환기 전담반 근무(통신기좌). 1980년 한국과학기술원 전자과 석사학위 취득. 1981년 블란서 ENST의 CES 졸업. 1984년 블란서 ENST 공학박사학위 취득. 1984년 9월~현재 아주대학교 전자공학과 부교수. 주관심분야는 초음파의 의학적 응용, Digital Signal Processor의 응용 및 Processor 간 통신, 디지털 신호처리 등임.



金晶雨(正會員)

1960年 10月 5日生. 1983年 2月 광운대학교 전자공학과 졸업 공학사학위 취득. 1985년 2월 아주대학교 대학원 전자공학과 졸업 공학석사학위 취득. 1985년 3월~현재 아주대학교 대학원 전자공학과 박사과정 재학중. 1987년 9월~현재 아주대학교 강사. 주관심 분야는 Computer Vision, 패턴인식, 인공지능 등임.