

論文96-33B-9-13

자소 단위의 온라인 흘림체 한글 인식

(Grapheme-based On-line Recognition of Cursive Korean Characters)

鄭基哲*, 金相均*, 李鍾國**, 金恒俊*

(Kee Chul Jung, Sang Kyoon Kim, Jong Kook Lee, and Hang Joon Kim)

요 약

본 논문에서는 자소 단위의 온라인 흘림체 한글 인식 방법을 제안한다. 한글은 방대한 문자 집합을 가지며, 각 글자는 이차원 상에서 기본 자소의 결합으로 표현된다. 필기시 기본 자소들은 대부분 흘림획으로 구성되기 때문에, 자소내에서 뿐만 아니라 자소간에도 많은 변형이 존재한다. 이러한 문자 집합을 부가적인 기억 공간 없이 조합 규칙만으로 표현하고 흘림획의 변형들을 흡수하기 위해서는, 한글을 이차원 구조를 수용하는 자소 단위의 인식 방법이 효과적이다. 제안한 인식 방법은 시간지연신경망(Time-Delay Neural Network: TDNN)을 이용하여 자소 단위의 인식을 수행하고, 인식 결과 그래프상에서 비터비 알고리즘을 이용하여 한글 문자조합 규칙에 맞는 최적의 인식 경로를 탐색함으로써 최종 문자 인식이 이루어진다. 실험 결과, 흘림체 한글에 대하여 94.5%의 인식율을 보였으며, 이는 제안한 인식 방법이 흘림체 한글 인식에 효율적임을 제시한다.

Abstract

Korean has a large set of characters, and has a two-dimensional formation: each character is composed of graphemes in two dimensions. Whereas connections between characters are rare, connections inside a grapheme and between graphemes happen frequently and these connections generate many cursive strokes. To deal with the large character set and the cursive strokes, using the graphemes as a recognition unit is an effective approach, because it naturally accommodates the structural characteristic of the characters.

In this paper, we propose a grapheme-based on-line recognition method for cursive Korean characters. Our method uses a TDNN recognition engine to segment cursive strokes into graphemes and a graph-algorithmic postprocessor based on Korean grapheme composition rule and Viterbi search algorithm to find the best recognition score path. We experimented the method on freely hand-written characters and obtained a recognition rate of 94.5%.

I. 서 론

컴퓨터 사용이 증가되고 정보화 사회로 변화해 가면서, 온라인 문자 입력을 통한 인터페이스와 같이 간편

하고 자연스러운 한글 데이터의 입력 방법이 많은 주목을 받고 있다.

한글은 그 구성이 매우 독특하고 훌륭한 언어적 기능을 갖고 있지만, 기계적 문자 인식의 대상으로서는 서양의 문자 체계가 갖지 않는 다음과 같은 특징들을 가지고 있다. 첫째, 양적인 문제로서, 한글은 기본 자소의 이차원적 결합에 의하여 구성되는 11,172자의 방대한 문자 집합을 갖는다. 둘째, 질적인 문제로서, 자소의 변형이 매우 심하다. 이차원 상에서의 자소의 결합은 자소내에서 뿐만이 아니라 자소간의 많은 흘림을 야기

* 正會員, 慶北大學敎 컴퓨터工學科
(Dept. of Computer Eng., Kyungpook Univ.)

** 正會員, 安東大學校 컴퓨터工學科
(Dept. of Computer Eng., Andong Univ.)

接受日字: 1996년2월24일, 수정완료일: 1996년7월23일

한다. 따라서 실용적인 온라인 한글 인식 시스템을 구현하기 위해서는 방대한 문자 집합과 홀림획의 표현 방법, 실시간 자동 학습을 통한 새로운 필기 형태에 대한 적응에 관한 연구가 필수적이다.

온라인 필기 한글의 인식 방법은 문자, 획, 획 세그먼트, 그리고 자소와 같이 인식 단위에 의해서 구분할 수 있다. 이러한 인식 단위를 기준으로 필기 한글 인식에 사용된 기존의 방법들을 분류하면 다음과 같다.

문자 단위의 인식¹¹⁾은 주로 통계적 기법에서 사용되며 비교적 간단한 방법으로, 기본적으로 모든 문자 템플릿과의 비교를 필요로 하므로 필기 변형이 큰 문자의 인식일수록 인식 속도가 느리고 기억 공간을 많이 요구한다는 단점이 있다. 따라서 방대한 문자 집합을 표현하기에는 비효율적이다.

획 단위의 인식은 필기자가 한번에 쓰는 획 전체를 기본획으로 정의하여 인식 단위로 사용하는 방법이다¹²⁾. 이 방법에서는 기본획이 적절히 정의되면 유사한 획들을 구분하기 쉽고 인식 시간이 적게 걸린다. 또한 문자의 표현을 획사이의 위치 관계의 열로 나타낼 수 있다. 그러나 기본획을 정하는 일정한 방법이 없고 기본획의 수가 많아질 수 있다는 단점이 있다.

획 세그먼트를 인식 단위로 하는 방법은 문자내 획들이 직선획들로 구성되어 획으로부터 획 세그먼트를 분리하기 쉬운 경우 주로 사용된다¹³⁾. 그러나 홀림체 한글은 직선획보다 곡선획이 많이 존재하기 때문에 일정하게 획 세그먼트를 구하는 것이 쉽지 않다. 따라서 문자에 대한 표현 수가 커진다.

자소 단위의 인식은 자소의 모양이 비교적 일정한 정서체 문자에 적용되어 좋은 결과를 보였다¹⁴⁾. 한글은 기본 자소의 결합으로 하나의 글자를 표현하는 이차원 문자이므로 자소 단위의 인식은 문자 인식의 문제를 자소 인식 문제로 축소시켜 주는 장점을 가진다. 단점으로는 자소내의 변형과 자소간의 접촉이 빈번한 홀림체 한글 인식 경우에는 자소 분리 자체가 매우 어렵다는 것이다. 그러나 획 분리를 위한 정보를 얻기 위하여 인식을 병행한다면, 홀림획을 의미 있는 자소 단위로 분리할 수 있다¹⁵⁻¹⁷⁾. 이러한 기법을 이용하여 입력 문자를 자소 단위로 분리하면, 자소나 몇 개의 기본획만을 인식함으로써 문자 인식을 수행할 수 있을 뿐만 아니라 한글의 구조적 특징을 수용하여 보다 효율적인 인식 시스템을 구성할 수 있다.

본 논문에서는 인식을 병행한 문자의 자소분리 기법

을 이용한, 자소 단위의 온라인 홀림체 한글 인식 방법을 제안한다. 자소 분리의 기본적인 개념은 “가자소”(tentative grapheme/假字素)들을 학습한 자소 인식기를 이용하여 입력획들을 인식하고, 이러한 결과들 중 최적의 인식 결과를 자소 분리 정보로 이용하여 입력 획을 자소 단위로 분리하는 것이다.

본 연구에서 자소 인식기로는 순차적인 신호 처리에 적합한 것으로 알려져 음성 인식에 많이 사용되는 시간지연신경망을 사용하였다^{17,18)}. 각 가자소들의 TDNN 자소인식기를 이용한 인식 결과 그래프는 입력 획을 자소 단위로 분리하기 위한 정보를 제공한다. 이러한 그래프를 비터비 알고리즘¹⁹⁾으로 탐색하여 최적의 경로를 구함으로써 입력획들에 해당하는 문자를 인식한다. 제안한 방법은 자소 단위 인식의 장점을 수용할 뿐만 아니라, 자소 내와 자소간의 홀림도 허용하여 홀림체 한글의 인식에 적합하다. 또한 새로운 필기 형태에 대하여 실시간 학습을 통하여 적응할 수 있다. 본 논문의 구성은 다음과 같다. 제2장에서는 온라인 홀림체 한글 인식 시스템의 전체적인 구조, 전처리와 특징 추출, TDNN에 대한 소개와 이를 이용한 자소 단위의 분리, 그리고 최적의 경로 탐색을 통한 문자 인식에 대하여 각각 서술한다. 제3장에서는 실험 및 결과에 대하여 서술하고, 제4장에서 결론을 맺는다.

II. 온라인 홀림체 한글 인식

본 장에서는 제안한 온라인 홀림체 한글 인식 시스템의 방법론을 설명한다. 전체 인식 시스템을 개괄적으로 기술한 다음, 각 단계에 대하여 구체적으로 설명한다.

1. 인식 시스템의 개괄적 구조

제안한 온라인 홀림체 인식 시스템은 그림 1과 같이 여섯 단계-문자 입력, 전처리, 코너점 추출, 특징 벡터 추출, TDNN을 이용한 자소 인식, 최적 경로 탐색-로 이루어진다. 그림 1에서 음영처리된 부분은 주요 단계의 결과물을 나타낸다.

문자 입력 단계에서는 전자 테블릿이나 마우스 등을 이용하여 획을 구성하는 좌표 열을 얻고, 전처리 단계에서는 인식을 향상을 위하여 삐침획의 제거, 평활화, 정규화, 재생플링 과정 등을 거친다. 코너점 추출 단계에서 입력획을 가자소 단위로 분리하고, 특징 벡터 추출 단계에서는 분리된 각각의 가자소들에 대해서 특징

벡터를 추출한다. 그리고 TDNN 자소인식기를 이용하여 이전 단계에서 추출된 특징 벡터들에 대한 자소 단위 인식을 수행한다. 마지막으로, 최적 경로 탐색 단계에서 자소 인식 결과 그래프상에서의 최적의 문자 인식 경로를 찾음으로써 완전한 문자 인식 결과를 얻는다.

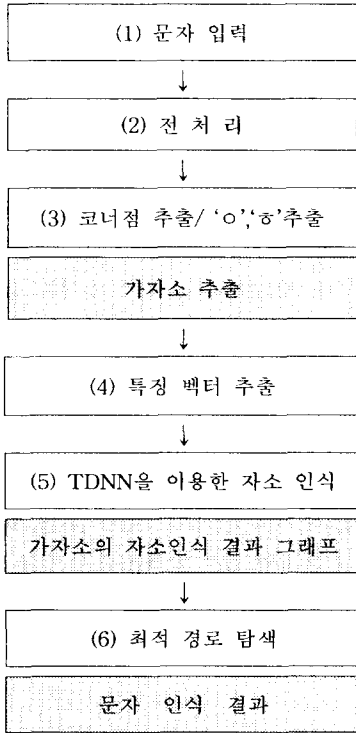


그림 1. 제안한 인식시스템의 전체적인 흐름도
Fig. 1. The overall structure of the system.

2. 문자 입력과 전처리 단계

필기자가 태블릿 상에서 문자를 입력할 때, 입력 문자들은 (x, y)좌표들의 열로써 표현되며, 많은 왜곡이 생길 수 있다. 왜곡 현상의 첫 번째 이유로는 흘림체의 특성을 들 수 있다. 흘림체는 정서체에 획간의 불필요한 획이 추가되거나, 각도의 변화가 급격하거나, 급격히 변화되어야 할 곳에서 완만히 변화하는 특성을 가지고 있다. 두 번째 이유로는 필기구의 특성을 들 수 있는데, 이에 의해 부분적으로 생긴 왜곡은 각도 면에서 전체적 획의 방향과 크게 다를 수 있다. 이러한 작은 부분에서 심한 각도의 변화를 일으킨 부분을 제거하는 것은 인식율 향상과 관계된다. 이러한 왜곡을 줄이기 위해 빠침획 제거^[2], 평활화, 정규화, 재 샘플링을 수행한다.

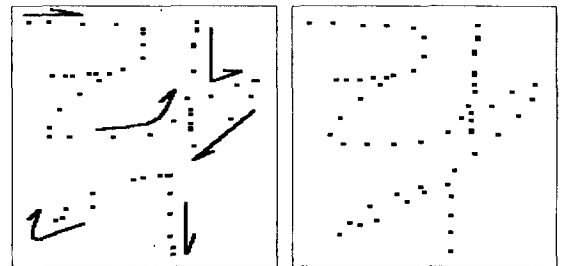
평활화는 손의 떨림 등의 이유로 인한 거칠어진 획을 부드럽게 해준다. 한 점을 이웃 점과 평균함으로써 평활화 할 수 있는데, 다음의 식(1)을 이용하였다^[10]. 식 (1)의 x_i 는 입력획의 i 번째 x 좌표를 의미한다. y 축 좌표에 대해서도 또한 같은 모양의 식을 사용한다.

$$x_i = \frac{(-3x_{i-2} + 12x_{i-1} + 17x_i + 12x_{i+1} - 3x_{i+2})}{35} \quad (1)$$

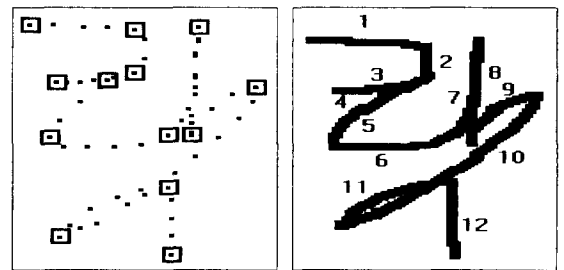
정규화와 재 샘플링 단계는 속도와 문자의 크기 변화에 따른 획의 변형을 줄여 준다. 전처리의 예는 그림 2(b)에 나타나 있다.

3. 코너점 추출

이 단계에서는 전처리 과정을 거친 입력 문자에서 코너점을 추출한다. 본 연구에서는 참조 논문 14에서의 코너점 추출 방법을 사용하였다. 추출된 코너점들은 자소 인식 단계에서 자소 분리 가능점/예측점으로 사용한다. 이렇게 코너점을 기준으로 분리된 획 세그먼트들을 본 논문에서는 가자소라고 부른다. 그림 2(d)는 입력 문자를 획 세그먼트들로 분리한 예를 보여 준다.



(a) 입력 문자 (a) input character
(b) 전처리 (b) preprocessing



(c) 코너점 추출 (c) corner point
(d) 획 세그먼트 (d) stroke segment

그림 2. 인식 시스템의 단계 (1)에서 (3)까지의 처리 결과

Fig. 2. A processing result of step (1) to (3) of the system.

정서체 한글에서는 자소 'ㅇ'과 'ㅎ'을 제외한 나머지 대부분의 획이 직선으로 이루어져 있다. 그러나 흘림체에서는 대부분의 획이 곡선으로 구성되므로, 본 연구에서는 코너점을 자소 분리 가능점으로 선택하였다. 그런데 자소 'ㅇ'과 'ㅎ'에는 다른 획 보다 많은 코너점이 존재하기 때문에, 이들은 코너점을 추출하기 전에 동적 프로그래밍(Dynamic Programming)을 사용하여 미리 분리한다.

$$D(i, j; k) = \min \begin{cases} D(i-1, j) \\ D(i-1, j-1) \\ D(i, j-1) \end{cases} + d(i, j; k), \quad (2)$$

$i=1, \dots, n, \quad j=1, \dots, M_k$

위의 식은 동적 프로그래밍의 비교 루틴이며 그림 3은 'ㅇ'과 'ㅎ'에 대한 표준 패턴들이다.

식 (2)에서 $D(i, j; k)$ 는 (1,1)에서 (i,j)까지의 거리이고 $d(i, j; k)$ 는 입력 패턴 i번째 점과 표준 패턴 k의 j번째 점간의 거리이다. 입력 패턴의 특징점 수는 n이며, M_k 는 표준 패턴 k의 특징점 수이다. 입력 문자들은 코너점 추출 시에 생성된 방향 코드를 사용하여 표현되며, 두 방향 코드가 인접한 정도에 따라 0, 1, 2, 4, 8을 사용하였다.

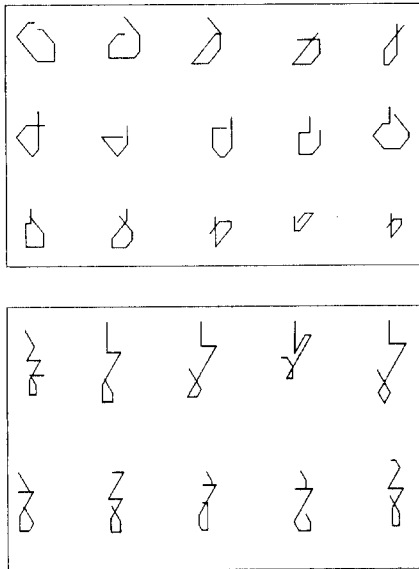


그림 3. 가자소 분리를 위한 'ㅇ'과 'ㅎ'의 표준 패턴
Fig. 3. Templates for 'o' and 'h'.

그림 4는 그림 2의 입력 문자에 대한 가자소들이다.

각각은 "i-j grapheme"의 형식을 가지며, i번째 획 세그먼트에서 j번째 획 세그먼트까지를 결합시킨 가자소임을 의미한다. 분리된 각 가자소들은 특징 벡터 추출 후 TDNN의 입력으로 사용된다. 그림 4의 가자소들의 예에서, 각 숫자들은 가자소들에게 매겨진 번호이며, 각각을 획 세그먼트라고 한다. 이러한 획 세그먼트들을 하나씩, 혹은 여러 개를 연결해서 인식한다.

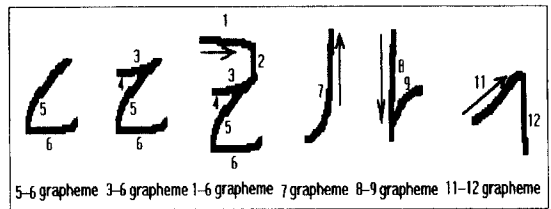


그림 4. 가자소들의 예
Fig. 4. Examples of tentative graphemes.

4. 특징 벡터 추출

이 단계에서는 TDNN의 입력으로 사용할 54개의 점으로 정규화된 각 가자소들의 특징 벡터들을 생성한다. 특징 벡터들은 펜의 업/다운, 펜 케지의 (x,y)좌표, 획의 방향과 곡률들^[7]을 이용하여 추출한다. 다음의 식들은 7가지의 특징 벡터 추출 함수들로서 모두 0에서 1까지의 값을 출력한다. 그림 5는 입력된 문자의 방향성과 곡률을 나타내기 위해서 사용된 θ 와 ϕ 의 의미를 나타낸다.

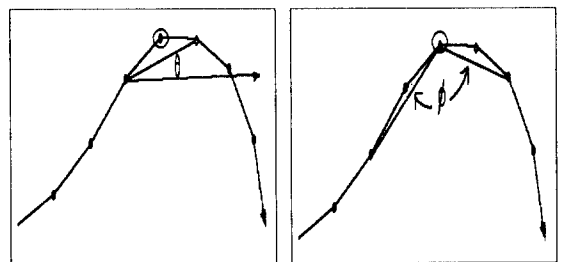


그림 5. 방향(θ)과 곡률(ϕ)의 추정
Fig. 5. The slope(θ) and curvature(ϕ).

$$f_0 = \text{penup/down}, \quad f_1 = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \quad f_2 = \frac{y - y_{\min}}{y_{\max} - y_{\min}},$$

$$f_3 = (\cos \theta + 1)/2, \quad f_4 = (\sin \theta + 1)/2, \quad f_5 = (\cos \phi + 1)/2,$$

$$f_6 = (\sin \phi + 1)/2.$$

그림 6에 입력 문자 '략'으로부터 추출된 특징 벡터의 예를 보였다. 각 행은 시간축 상의 7가지의 특징 벡

터들을 나타낸다(색깔이 검을수록 1에 가까운 값이다).

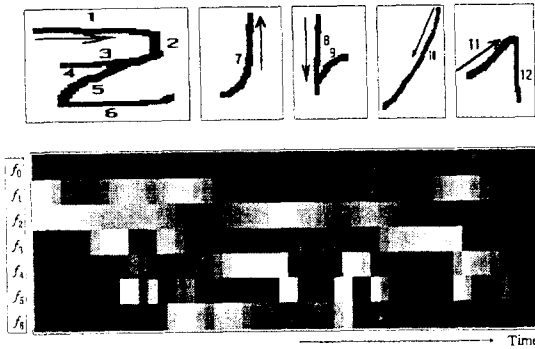


그림 6. 그림 2의 입력 문자 '락'에 대한 특징 벡터의 예

Fig. 6. An example of feature vector for the character '락' of Fig. 2.

5. TDNN을 이용한 자소 단위의 인식

TDNN은 다중 퍼셉트론 네트워크에 지연 요소를 첨가하여 입력 패턴의 동적인 특성을 위치에 무관하게 감지하도록 구성된 시스템이다.

TDNN의 기본 처리 요소는 그림 9과 같이 지연 요소 D_1 에서 D_n 으로 하나의 입력을 시간상으로 지연시키고, 지연된 입력과 가중치(weight)가 곱해진 합을 활성화 함수(activation function) F 를 통해 다음 계층에 인가하는 구조로 되어 있다. 따라서 처리 요소들은 시간축 상에서 N 개의 서로 다른 위치에 배열되어 있는 입력들을 검출한다. 이렇게 함으로써 입력 신호의 현재의 입력에 대해 앞 뒤 사이의 관계를 감지할 수 있고, 입력 패턴의 시간적인 특징을 추출할 수 있다. 이러한 네트워크의 구성 때문에 입력 신호의 지연(delay)현상을 극복할 수 있는 특성(shift invariant)을 검출하고, 입력 신호의 가장 큰 특징인 시간 굴곡(time warping)을 학습을 통하여 배운다. 따라서 분할과 시간 굴곡 문제를 자연스럽게 해결할 수 있다. 그림 9의 U_i 와 U_j 는 시간축상으로 각각의 특징벡터를 의미한다. W 는 신경망의 연결강도를 의미한다. 자소 인식을 위한 전체적인 TDNN의 구조와 패러미터들은 그림 7과 8에 나타내었다. TDNN자소인식기는 세계층의 피드포워드(three layer feed-forward)네트워크이다. 최하위층인 입력 노드에서는 이전 단계에서 추출된 각 가소에 대한 54개의 정규화된 특징 벡터열을 입력으로 받아들인다. 각 뉴런들은 시간 축 상에서 일정한 간격

으로 제한되어 입력을 받아들이며 출력층으로 전파되어 가면서 점점 전역적인 특징 벡터들을 추출한다. 최종적으로, 마지막 층인 출력 층에서 입력 벡터들에 대한 최종의 자소 인식 결과를 그래프 형식으로 출력한다.

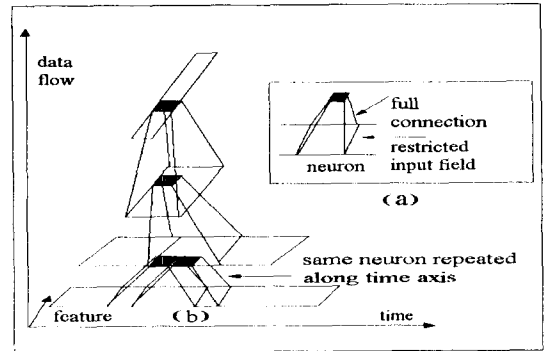


그림 7. 사용된 TDNN (a) 뉴런의 모양 (b) 전체적인 구조

Fig. 7. Time delay neural network (a) shape of neuron (b) architecture of the TDNN.

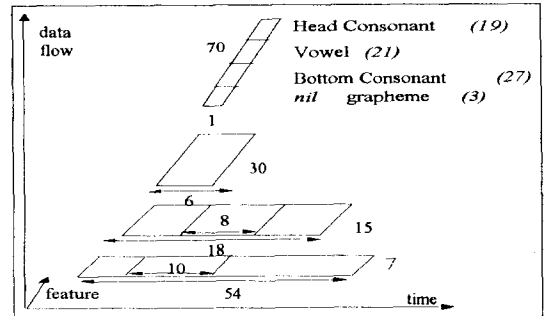


그림 8. TDNN의 패러미터

Fig. 8. Parameters of the TDNN.

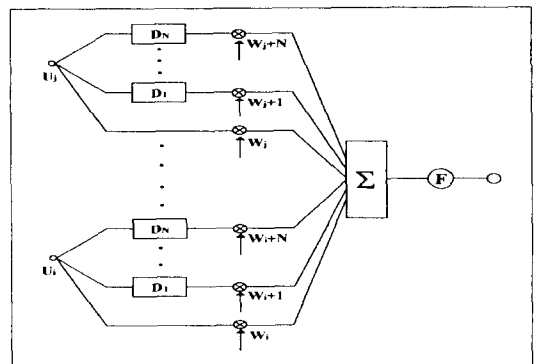


그림 9. TDNN의 기본 구성 단위

Fig. 9. A Time-Delay Neural Network unit.

최종 출력층에는 한글의 각 자소를 인식하기 위한 하나씩의 출력 노드—초성 19개, 중성 21개, 종성 27개로써 총 자소의 개수가 67—를 할당하고, 부가적으로 연결획을 위한 세 개의 노드를 할당하였다.

이러한 자소인식기는 한글의 모든 자소들, 즉 67개의 초, 중, 종성에 대하여 학습할 뿐만 아니라, 자소들에 대한 반례로써 무효(nil)자소, 즉 자소 내부의 연결획이나, 자소간의 연결획들에 대해서도 학습한다. 이들 연결획들은 특징 벡터 공간상에서 다양한 모양을 가지기 때문에 세 개의 출력 노드를 할당하였다. 그림 10의 점선은 연결획의 예를 나타내는 것으로, 다양한 모양을 보이고 있다. 학습 알고리즘으로는 다층 퍼셉트론의 대표적인 학습 방법인 오류 역전파 알고리즘(Error Backpropagation Algorithm)을 사용하였다. 일반적인 오류 역전파 알고리즘과 달리 모든 시간적으로 반복 사용되는 뉴런들 각각에 대해서 별개로 포워드/백워드 단계를 거친다. 다음으로 시간적으로 반복 사용된 연결 강도들을 각각 갱신한 후에, 이들의 평균값을 취하게 된다. 학습 동안 입력시에 주어진 출력 기대값과 TDNN의 출력값의 차의 제곱근 값이 줄어들게 뉴런간의 연결 웨이트들을 조정한다.

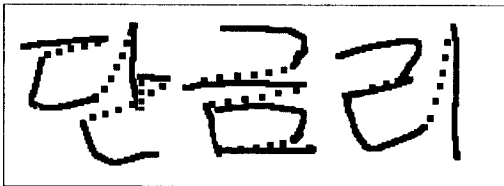
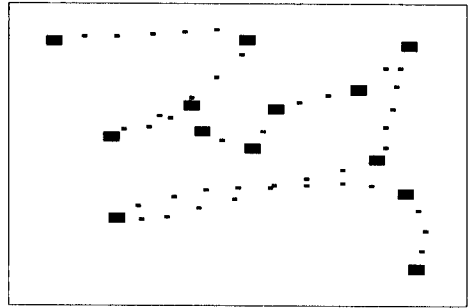


그림 10. 연결획(점선)
Fig. 10. Ligatures(dotted line).

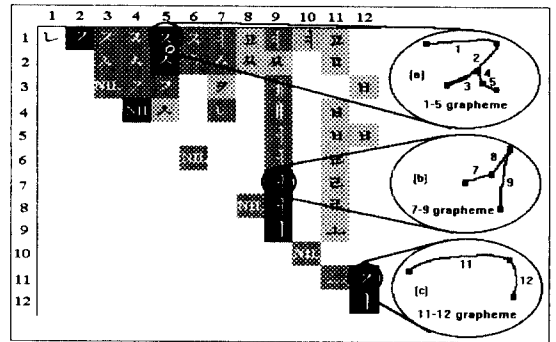
6. 최적 경로 탐색

입력획들을 하나의 문자로 인식하기 위해 TDNN의 자소 인식 결과상에서 최적의 인식 경로를 탐색한다¹²⁾. 그림 11은 입력 문자 '적'에 대한 제안한 시스템의 최종 출력을 보여 준다. 그림 11에서 왼쪽의 각 격자 무늬는 각 가자소의 인식 결과를 나타낸다(색깔이 진할 수록 그 자소에 대한 높은 인식율을 나타낸다). 각 가자소들은 그림 11(b)-(a)의 "i-j grapheme"과 같이 두 개의 인덱스가 있는데 첫 번째 인덱스 i는 전체 문자에서 가자소의 시작 세그먼트 번호를 의미하고, 두 번째 인덱스는 끝 세그먼트를 가리킨다. 이들 i에서 j까지의 획 세그먼트들을 결합시킨 것이 그림 11에서 i행

j열에 위치한 격자 무늬이다. 그림 11(a)는 첫 번째 세그먼트에서 다섯 번째 세그먼트까지 결합된 가자소이다.



(a) 전처리, 코너점 추출을 거친 입력 문자
(a) An input character after preprocessing and corner point extraction



(b) 인식 결과
(b) recognition result

그림 11. 입력 문자 '적'에 대한 인식 결과 (적/ 격/ 잭/ 벡/ 픽 순서의 후보)

Fig. 11. Recognition result for the character '적' (candidates are 적/ 격/ 잭/ 벡/ 픽).

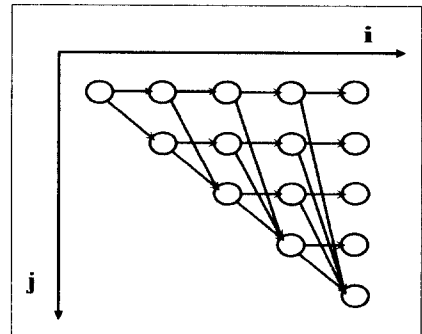


그림 12. 문자 인식 네트워크
Fig. 12. Character recognition network.

최적 경로 탐색을 위한 비터비 알고리즘

- 1) Initialization
 $\delta_i(1) = p_{11}$
 $\phi_i(1) = 0$
- 2) Recursion

$$\delta_i(j) = \max_{1 \leq s \leq i-1} [\delta_{i-1}(s)p_{si}, \delta_{i-1}(j)p_{jj}] \quad 2 \leq i \leq T$$

$$\phi_i(j) = \arg \max_{1 \leq s \leq i-1} [\delta_{i-1}(s)p_{si}, \delta_{i-1}(j)p_{jj}] \quad 2 \leq i \leq T$$
- 3) Termination

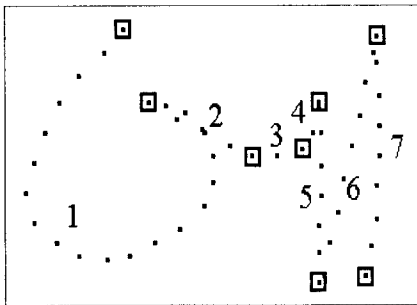
$$P = \max_{1 \leq j \leq T} [\delta_T(j)]$$

$$q_T = \arg \max_{1 \leq j \leq T} [\delta_T(j)]$$
- 4) Backtracking

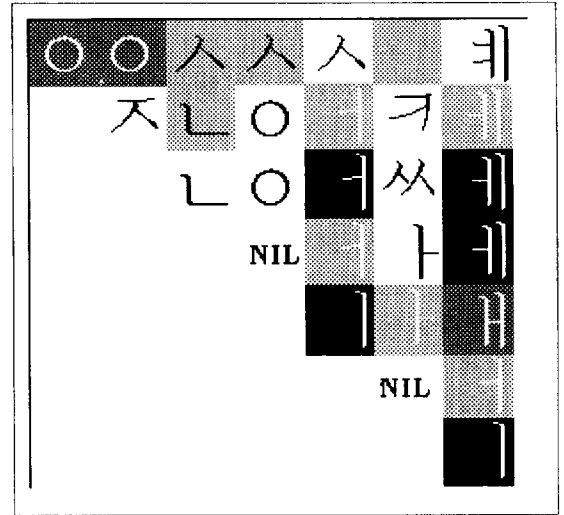
$$q_i = \phi_{i+1}(q_{i+1}) \quad i = T-1, T-2, \dots, 1$$

그림 12는 최적의 인식 경로를 탐색하기 위한 네트워크의 구조를 나타낸다. 각각의 노드는 각 가자소 인식 단계를 의미하고, 화살표는 다음의 가자소 인식 단계로의 천이를 의미한다. 그림 12의 인식 네트워크에서 가장 큰 인식 정도를 보이는 경로를 찾음으로써 하나의 글자를 인식할 수 있는데, 이는 수정된 비터비 알고리즘에 의해 구해진다. 알고리즘에서 $\delta_i(j)$ 는 (i-1)번째 입력에서 i번째 입력의 j 노드로 이동하면서 생성된 최대 인식값을 저장하고, $\phi_i(j)$ 는 최고 인식 확률을 생성한 j번째 노드의 인덱스를 저장하여 역추적 단계에 사용한다 비터비 알고리즘으로 인식 결과 그래프 상에서 최적의 경로를 찾을 때 인식 확률에 따라 여러 개의 인식 결과가 존재할 수 있으며 이 중에서 한글의 조합 규칙에 맞는 글자를 추려 낸다.

그림 13은 입력 문자 '에'에 대한 인식 결과인데 그림 13(b)는 그림 13(a)의 입력 문자에 대한 최고 인식 정도를 보인 자소를 보인 것이다. 인식시의 매칭 정도에 따라 정렬하면, [에/ 세/ 새/ 서/ 시]의 순서이다.



(a) 입력 문자
(a) input character



(b) 인식 결과
(b) recognition result

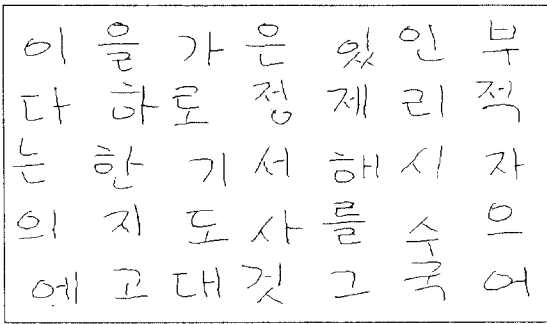
그림 13. 입력 문자 '에'에 대한 인식 결과
Fig. 13. Recognition result for the character '에'.

III. 실험 및 결과

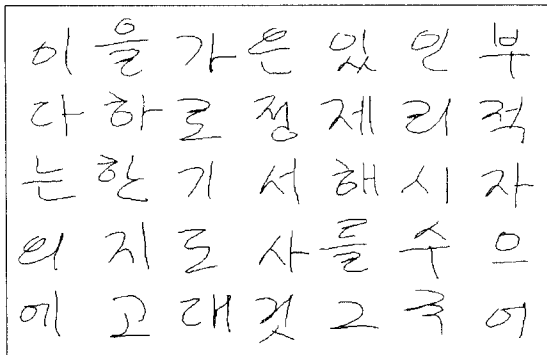
제안한 방법은 IBM PC Pentium-75상에서 C 언어로 구현하였으며, 온라인 문자 입력 장치로는 WA-COM HD-648A 디지털이저를 사용하였다. 그림 14는 실험에서 사용한 한글 사용 빈도순 300자이다.

이다는의에올하한지오가로기도대은정서사깃
 있제해름그인리시수국부적자어과나들상경
 아우라일전만보추면문화장할원무계당되위구
 야없치성민러관계개통비중방안회여조공선력
 실동금와신양세업요유러내히행때단거체발
 진설제된점분했모결본소각용직등마오물운역
 산강생간니입권더연명피차영오불반현처심스
 북외모엇책말임바법질년감새학건데평속표능
 두까터피출생될또에난갈김환특추후닛꼬최협
 지번확작련판태찰며변긴른급근격따혁못닛남
 래필합호절합청규던반총울음평집란양담핵과
 백스육검록측매배언토든천노음앞목절르울편
 향군달루복힘총범별투뿐결포피갖중크키중트
 글응울벌형많타준독럼초참드극황농약효린철
 논망완폭은취악술박삼건갈택프획편저패엔류

그림 14. 실험 데이터(사용 빈도순 300자)
Fig. 14. Experimental data(300 characters by frequency).



(a) 정서체 데이터
(a) handprinted characters



(b) 흘림체 데이터
(b) handwritten characters

그림 15. 실험 데이터
Fig. 15. Experimental data.

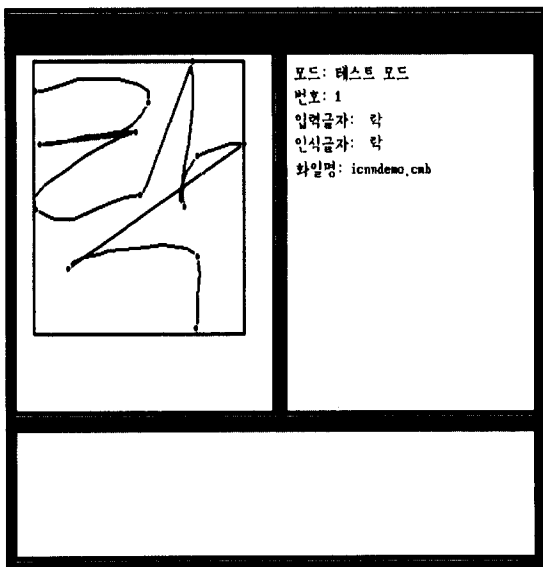


그림 16. 제안한 인식 시스템의 실행 화면
Fig. 16. Recognition system screen layout.

이는 3만여 자의 데이터에서 사용 빈도순으로 정렬한 결과이다. 학습용 데이터로 네 명으로부터 입력받은 1,200자의 정서체 및 흘림체 문자를 사용하였다. 테스트 데이터는 20명의 서로 다른 필기자에 의한 각각 300자씩의 정서체와 아무런 제약 없이 필기한 흘림체 데이터이다. 그림 15는 실험 데이터의 예이며, 그림 16은 구현한 온라인 흘림체 한글 인식 시스템의 실행 화면이다.

표 1은 실험에서의 문자 인식율이다. 이미 학습을 거친 데이터에 대해서는 97.9%의 인식율을, 학습하지 않은 흘림체 데이터에 대해서는 94.5%의 인식율을 보였다. 그리고 인식 속도는 문자 당 평균 1.4초 였다.

표 1. 인식율
Table 1. Recognition rate.

	정서체		흘림체	
	학습 데이터	시험 데이터	학습 데이터	시험 데이터
문자 개수	600	2400	600	2400
정인식 개수	593	2357	582	2268
인식률(%)	98.8	98.2	97.0	94.5

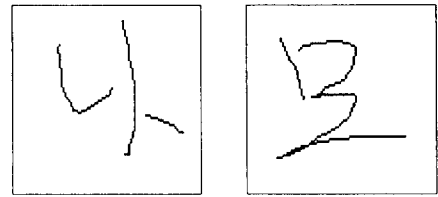
표 2. 필기자별 인식율(*1. 정서체인식율)
Table 2. Recognition rate according to writers.

필기자	필기문자(자)	정인식(자)*1	인식율(%)
1	300	295*1	98.3*1
	300	288	96.0
2	300	297*1	99.0*1
	300	292	97.3
3	300	296*1	98.7*1
	300	286	95.3
4	300	294*1	98.0*1
	300	278	92.7
5	300	295*1	98.3*1
	300	281	93.7
6	300	296*1	98.7*1
	300	291	97.0
7	300	296*1	98.7*1
	300	276	92.0
8	300	288*1	96.0*1
	300	273	91.0
9	300	297*1	99.0*1
	300	293	97.7
10	300	296*1	98.7*1
	300	292	97.3
계	3,000	2,950*1	98.3*1
	3,000	2,850	95.0

표 2는 각 필기자별 정서체 및 흘림체의 인식 결과를 나타낸다. 그림 17은 오인식의 예이다. 오인식의 주된 이유는, 첫째로 그림 17(e)의 예와 같이, 흘림체 문자에서 한 획을 정확하게 가자소들로 분리하기가 쉽지 않고, 둘째로 그림 17(b)의 예처럼, 흘림획에서 자소간을 연결하여 쓰면서 발생하는 연결획들과 자소의 일부를 인식기가 분리하기가 쉽지 않다는 점이다. 셋째로 그림 17(i), (j)의 예와 같이 문맥상에서 구분 가능한 모호한 글자의 오인식을 들 수 있다. 표 3은 인식 오류의 유형별 비율을 나타낸다. 현재의 인식율은 TDNN 자소 인식기를 더욱 많은 데이터로 학습함으로써 향상시킬 수 있을 것이다. 또한 문자 인식 결과는 인식 정확도에 따라 다수 개가 출력되기 때문에, 이들을 이용하는 문맥에 따른 후처리 모듈을 추가하여 보다 안정된 인식 시스템을 구축할 수 있을 것이다.

표 3. 상대적인 오인식 비율
Table 3. Relative error ratio of test character.

오인식 종류	오인식 발생 횟수(자)	비율(%)
자소 분리 가능점 추출 에러	56	34.3
흘림획에서의 연결획 인식	82	42.1
모호한 문자	26	13.6
과다한 획/끊어진 획	19	10.0
총 계	194	100.0



나 → 나 (e)

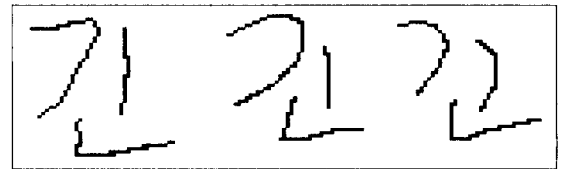
모 → 므 (f)



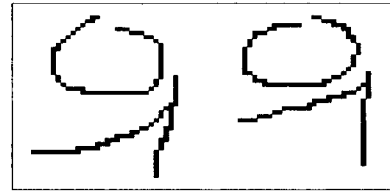
결 → 결 (g)



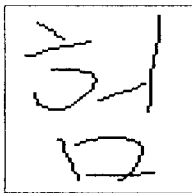
전 → 전 (h)



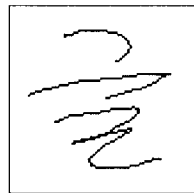
(i) [긴, 조, 꼬]



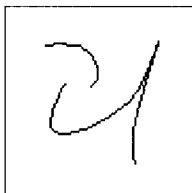
(j) [의, 우]



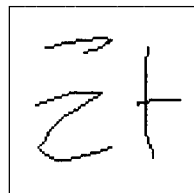
힘 → 힘 (a)



글 → 글 (b)



리 → 리 (c)



타 → 타 (d)

그림 17. 오인식의 예
Fig. 17. Examples of recognition error.

V. 결론

기존의 온라인 흘림체 한글 인식 방법들은, 2차원 상에서 자소들의 결합으로 하나의 글자를 구성하는 한글 고유의 구조적 특성을 반영치 못하여, 방대한 문자 집합을 효율적으로 표현하기 어려웠다. 또한 자소내의 흘림과 자소간의 흘림을 충분히 수용치 못하였다. 본 논문에서는 이러한 문제점을 개선한 새로운 온라인 흘림체 한글 인식 방법을 제안하였다. 제안한 인식 방법은 자소를 인식 단위로 하며, 입력된 흘림체 한글을 코너 점에 의한 가자소들로 분리하고 이를 TDNN상에서 인

식하면서 자소 단위로 분할한다. 그리고 인식된 자소들로부터 조합 가능한 자소의 결합을 파악하여 문자 단위의 인식을 수행한다. 기존의 연구와 비교하였을 때 제안한 방법의 장점은 다음과 같다.

1. 자소 단위의 인식을 수행함으로써 방대한 문자 집합을 효율적으로 표현한다.
2. 문자 인식 단계, 즉 자소 인식 결과 그래프 상에서 최적 경로를 찾을 때 한글의 구조적 특성을 이용할 수 있다.
3. 자소인식기로 사용된 TDNN은 자소간 및 자소내의 홀림을 수용한다.
4. 신경망의 학습 기능을 자소에 적용함으로써 특정한 필체에 쉽게 적용할 수 있다.

제안한 인식 방법은 우수한 성능을 보였지만, 인식 오류의 많은 비중을 차지하는 끊어진 획과 잡음을 처리하기 위한 전처리 기법의 연구, 유사한 문자 집합을 분류할 수 있는 알고리즘의 연구, 홀림체에서의 변형된 획과 자소와의 변별력, 문자마다의 상이한 인식 속도를 극복하기 위해서 코너점보다 효율적인 자소 분리 가능점의 선택 등에 관한 연구 등을 통해서 개선할 수 있을 것이다. 더불어 제안한 인식 방법을 기초하여, 연속되는 홀림체 문자를 문자 단위로 파악하여 인식할 수 있는 연속 필기 한글 인식 시스템의 개발, 한자와 같이 한글과 비슷한 구조적 특성을 가진 문자에 대한 인식 시스템의 개발, 한글과 영·숫자가 혼용된 입력에서의 문자 인식 시스템의 개발 등이 향후 연구 과제이다.

참 고 문 헌

[1] H. D. Lee, T. K. Kim, T. Agui, M. Nakajima, "On-line recognition of cursive Hangul by extended matching method," *Journal of KITE*, vol. 26, no. 1, pp. 29-37, 1989.

[2] P. K. Kim, J. Y. Yang, and H. J. Kim, "On-line cursive Korean character recognition using extended primitive strokes," *The 3rd PRICAI (Pacific Rim International Conference on A.I.)*, pp. 816-821, Aug. 1994.

[3] 권오성, 권영빈, "동적인 선분 생성을 이용한 온라인 한글 필기 인식," 한국 정보과학회 학술발

표논문집, 제20권, 제1호, pp. 151-154, 1993

[4] 김태균, 이은주, "한글에 적합한 획 해석에 의한 연속 필기 한글의 On-line 인식에 관한 연구," 한국정보과학회 논문지, 제15권, 제3호, pp. 171-181, 1988

[5] T. K. Kim, E. J. Rhee, "On-line recognition of successively written Korean characters by suitable structure analysis for Korean characters," *Journal of The Korea Information Science Society(KISS)*, vol. 20, no. 6, pp. 171-181, 1988.

[6] H. Weissman, M. Schenkel, I. Guyon, C. Nohl and D. Henderson, "Recognition-based segmentation of on-line run-on hand-printed words : input vs. output segmentation," *Pattern Recognition*, vol. 27, no. 3, pp. 405-420, 1994.

[7] I. Guyon, P. Albrecht, Y. Lecun, J. Denker and W. Hubbard, "Design of a neural network character recognizer for a touch terminal," *Pattern Recognition*, vol. 24, no. 2, pp. 105-119, 1991.

[8] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano and K. Lang, "Phoneme recognition using time delay neural networks," *IEEE Trans. Acoustics Speech Signal Process*, vol. 37, pp. 328-339, 1989.

[9] L. R. Rabiner, "Tutorial on hidden Markov model and selected application in speech recognition," in *Proc. the IEEE*, vol. 77, no. 2, pp. 257-285, 1989.

[10] C. C. Tappert, "Speed, accuracy and flexibility trade-offs in on-line character recognition," *IBM Research Division*.

[11] C. C. Tappert, and C. Y. Suen, "The state of the art in on-line handwriting recognition," *Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 8, pp. 787-808, 1990.

[12] C. J. C. Burges et al., "Shortest path segmentation: a method for training neural networks to recognize character string," *IJCNN'92*, Baltimore, vol. 3, 1992.

[13] Kee Chul Jung, Hang Joon Kim, "Recognition-based segmentation of on-line cursive Korean characters," in *Proc.*

ICNN'95, pp. 3101-3106 1995.

- [14] X. Li and N. S. Hall, "Corner Detection and Shape Classification of On-line Hand-

printed Kanji Stroke," *Pattern Recognition*, vol. 26, no. 9, pp. 1315- 1334, 1993.

저 자 소 개



鄭 基 哲(正會員)

1971년 3월 7일생. 1994년 2월 경북대학교 컴퓨터공학과 졸업(공학사). 1996년 2월 경북대학교 대학원 컴퓨터공학과 졸업(공학석사). 1996년 3월 ~ 현재 경북대학교 대학원 컴퓨터공학과 박사과정. 주관

심분야는 온라인 문자인식, 음성인식 등임

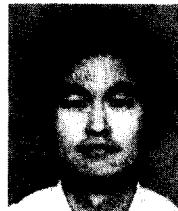


金 相 均(正會員)

1965년 8월 16일생. 1991년 8월 경북대학교 통계학과 학사학위 취득. 1994년 2월 경북대학교 대학원 컴퓨터공학과 석사학위 취득. 1996년 8월 경북대학교 대학원 컴퓨터공학과 박사학위 취득. 1996년 9월 ~ 현재 인제대학교 전산학과 전임 강사

李 鍾 國(正會員)

1989년 경북대학교 통계학과 졸업(학사). 1991년 경북대학교 대학원 컴퓨터공학과(석사). 1995년 경북대학교 대학원 컴퓨터공학과(박사). 1993년 ~ 1996년 동서대학교 컴퓨터공학과 전임강사. 1996년 ~ 현재 안동대학교 컴퓨터공학과 전임강사. 관심분야는 패턴인식, 게임 이론, 한국어정보처리 등임



金 恒 俊(正會員)

1954년 12월 30일생. 1977년 2월 서울대학교 전기공학과 학사학위 취득. 1979년 2월 KAIST 전기 및 전자 석사학위 취득. 1980년 ~ 현재 경북대학교 컴퓨터공학과 교수. 주관심분야는 인공지능, 문자인식,

음성인식, 병렬처리, 컴퓨터 구조 등임