

# 자소 탐색 방법에 의한 온라인 한글 필기 인식

終身會員 조 범 준\*

## Online Korean Character Recognition using Letter Spotting Method

Bum-Joon Cho\* *Regular Member*

※본 연구는 1994년도 조선대학교 학술 연구비를 지원 받아 수행 되었음.

### 요 약

한글 필기는 항상 초성, 중성, 종성의 순으로 씌어진다. 본 논문은 이점을 이용하여 자소 탐색 모델을 설계하고 그 탐색 결과에 의거하여 글자를 인식하려는 온라인 필기 인식 방법을 제시하고자 한다. 기본 자소 모델은 은닉 마르코프 모델을 이용하고 자소 탐색 모델은 HMM의 망으로 구성한다. 자소 탐색은 Viterbi 알고리즘에 의한 정합으로 이루어지며 글자 인식은 이들 자소 가설 격자의 탐색으로 이루어진다.

인식 실험 결과는 간단한 인식기 구조에도 불구하고 정자체의 경우 87.47%에 달하는 상당한 인식률을 보였으며, 특히 자연스럽게 쓴 필기에서도 매우 훌륭한 자소 분할 결과를 얻을 수 있었다.

### ABSTRACT

Hangul character always consists of consonants-vowel-consonants in order. Using this point, this paper proposes an approach to design a model for spotting each letter in Hangul, and then recognize characters based on the spotting results. The network model consist of a set of HMMs. The letter search is carried out by Viterbi algorithm, while character recognition is performed by searching the lattice of letter hypotheses.

Experimental results show that, in spite of simple architecture of recognition, the performance is quite high reaching 87.47% for discrete regular characters. In particular the approach shows highly plausible segmentation of letters in characters.

### I. 개 요

역사가 타분야에 비해 상대적으로 짧은 온라인 필기 인식 분야는 방법론 상으로 체계 정립이 아직 미비한 상태에 있지만, 문제의 특성에 맞는 해결책들이 이미 몇가지 제안된 바가 있다. 인공 신경망[1,2], 은닉 마르코프 모델[3] 등 새로운 모형화 방법론이나 특정

\*조선대학교 공과대학 컴퓨터공학과  
論文番號:95365-1016  
接受日字:1995年 10月 16日

분석 및 트리 분류 등 전통적인 방법[4] 등에 기초를 둔 시스템 들이 대표적이라 할 수 있다.

이들 인식 시스템 들이 공통적으로 사용하고 있는 인식 알고리즘은 모두가 시간적 순서에 따라서 첫획부터 차례로 분석해 나가는 순차적 디코딩(decoding) 방식을 채택하고 있다. 만약 인식 과정에서 초성, 중성, 종성 중의 하나가 변형이 심하여 불분명한 경우에 그 단계에서 내린 결정이 틀렸을 경우 뒷 부분이 아무리 명확하더라도 회복이 불가능하다. 또한 선행 지식 등을 이용하는 후처리가 있어도 오류 회복의 여지가 별로 없다는 것이 이러한 순차적 탐색 알고리즘의 가장 근본적인 약점이다.

본 연구에서는 이와 같은 분계점을 해결하기 위해서 온라인 필기의 특징을 모형화하는데 뛰어난 통계적 모델을 기초로 하여 한글 필기 모델을 설계하였다. 그리고 인식 알고리즘으로는 자소 탐색과 글자 탐색으로 분리하여 실시하는 형태를 갖도록 하였다. 이 방법의 특징은 순차적 디코딩의 효율성과 island-driven 탐색의 최적성 등을 묶은 방법으로 격자(lattice) 탐색 기법의 한 형태를 구현하는 점에 있다고 할 수 있다.

음성 인식 분야에서 쓰이고 있는 키워드 탐색 기법과 마찬가지로[5,6,7] 문자 인식 분야에서의 글자 탐색은 입력 텍스트 이미지 또는 필기 데이터 상에서 특정 글자를 찾는 문제를 말한다[8,9,10,11]. 찾고자 하는 글자를 키 글자라고 하며, 그것이 자소이면 키 자소라고 부른다. 최근에 이러한 글자 탐색 기법이 영어 단어 인식 분야에 응용하여 높은 인식률을 발표한 바가 있다[12]. 이 방법에서는 입력 단어 필기 상에서, 모든 가능한 알파벳들을 찾아서 이들을 알파벳 가설 격자로 구성한 다음 거기서 알파벳 가설을 순차적으로 최적 연결하는 알파벳 열을 찾아낸다. 이때 알파벳 열이 사전에 있는 단어가 되도록 한다.

한글 문자 인식은 일반적으로 자소 단위의 정합을 기초로 이루어지는 것이 보통이다[1,4]. 그러나 변형이 심하고 본질적으로 일차원적 시계열 신호인 흘려 쓴 온라인 필기에 있어서는 자소의 경계를 모르는 경우가 많기 때문에 자소 단위의 정합을 그대로 적용할 수가 없다. 따라서 [18]처럼 HMM을 이용하여 글자 단위로 인식하는 방법이 나오게 되었다.

한글 필기는 항상 초성-중성-종성의 순으로 씌어진

다. 이것은 글자 필기에서 각 자소의 필기 위치에 대한 정보를 암시해 주는 것이므로, 만약 주어진 문제가 자소 탐색이라면 매우 유용한 정보가 된다. 본 연구에서는 이점에 착안하여 입력 필기에서 각 자소를 가설 격자 생성 및 글자 인식 문제를 접근하였다. 본 논문에서는 매끄럽게 흘려 쓴 한글 온라인 필기에서 자소의 경계를 찾기 위한 네트워크 구조의 자소 탐색 HMM 모델과 효과적인 자소 탐색 방법을 제안하였다. 실험 결과 자소 탐색 결과는 홀림체에 대해서도 매우 놀라운 결과를 보여주었다. 이 방법은 어떤 알파벳이 몇개 존재하는지 알 수 없는 영어 단어와 달리 한글에 매우 적합한 방법이라고 할 수 있다.

본 논문의 구성은 다음과 같다. 제 2절에서 은닉 마르코프 모델과 필기의 특징에 따른 필기 모형 구축을 설명한다. 제 3절에서 필기 자소 탐색 모델을 설명한 다음, 제 4절에서 탐색 과정을 기술한다. 그리고 제 5절에서 몇가지 실험 결과와 예를 제시하고, 결론을 맺는다.

## II. 필기 모형화

### 2.1 온라인 필기 특성

온라인 필기는 점들의 열로 샘플링된다. 샘플 필기의 각점을 가상의 기본꼴의 어떤 점으로 대응한다고 하면 이때 적절한 기준에 따라서 계산되는 거리 함수를 설정하게 된다. 한편 많은 필기 데이터로부터, 이들을 대표하는 샘플 평균 즉 기본 패턴을 찾았을 때, 이 평균점과 각 샘플 사이에는 샘플 편차가 있고, 샘플 편차의 평균이 존재한다. 그리고 기본꼴 상의 모든 점은 확률적으로 존재하고 그 위치는 미지의 분포 함수를 갖는다. 이 분포 함수의 밀도가 입력 필기 샘플점의 우도(likelihood), 또는 앞서 말한 거리 함수를 결정하게 된다. 이에 따라 입력 글씨획의 모든 점은 기본꼴의 한 점에 적절한 거리 함수에 따라서 사상시킬 수 있게 된다.

입력 필기의 한 점이 주어졌을 때 시간적인 순서에 따라서 다음 점의 위치를 정확하게 예측한다는 것은 불가능하다. 그런데 위와 같은 확률적 특성 외에도 온라인 필기 데이터는 시계열 형태의 일차원 점열로 나타나고, 각 점은 뒤따르는 점의 위치에 제약을 가한다는 특징이 있다. 즉, 모든 점은 무작위로 나타나

는 것이 아니라, 현재 필획의 진행 패턴-예를 들어, 방향-에 제약을 받게 된다. 그러므로 적절한 필기 모델을 세울 수만 있다면 적어도 확률적으로는 예측이 가능하다. 통계 모델은 이와 같이 여러가지 확률적 요소를 이용하여 목적 패턴의 특징을 표현하는 방법이다. 이러한 두 가지 예의 확률적 특성 외에도 여러가지 미지의 확률적 요소를 기술하는 방법이 확률적 모델이다.

온라인 문자 인식 시스템은 입력 필기 데이터와 그 시스템이 갖고 있는 유한개의 모델을 적절한 기준에 따라 비교하고 판단을 내리는 전형적인 패턴 인식 시스템의 구조를 갖고 있다. 문자 인식에서 모델이란 정합의 기본 단위에 해당하며, 대상이 되는 패턴의 이상적이고 가장 대표적인 꼴, 즉, 기본꼴의 표현이라고 할 수 있다. 기본꼴은 모든 글씨에 포함된 공통된 일반적 특징을 갖는 공통 분모로서 모든 필기의 평균 패턴이라고도 할 수 있다. 통계적 모델링은 이러한 관점에서 확률 모델링 방법을 통해, 불확실하고 불완전한 정보를 처리하는 방법이다. 온라인 필기 인식에서 불확실성과 불완전성은 애매모호한 글자 모양, 개인적인 필체, 주위 문맥의 효과 등에 따라 발생한다. 이런 점을 수학적 기반 위에서 모델링하는 통계적 모델은 음성 및 필기를 인식하기 위한 모델로서도 많이 응용되고 있다.

현재 통계적 방법으로서 가장 널리 알려져 있고 성공적인 것 중의 하나는 은닉 마르코프 모델(Hidden Markov Model, 이하 HMM)이 있다. HMM은 유한 상태의 마르코프 체인과 마르코프 체인의 함수인 심볼 출력 프로세스로 구성된다. 상태 전이 파라미터의 집합인 마르코프 체인은 입력 신호의 시간축의 변형을 모델링하고, 마르코프 체인의 함수인 출력 프로세스는 시간축을 따라 발생하는 출력 심볼의 변이를 모델링하는데, 이 두가지 통계 프로세스가 결합되어 다양한 형태를 가진 온라인 필기 등 현실 세계의 복잡한 프로세스를 모델링하는 것이다.

## 2.2 은닉 마르코프 모델

불완전하면서도 다양한 변화를 가지는 입력 패턴을 확률 모델로 표현하여 처리하고자 하는 노력이 지난 30여년 간 음성 인식 분야에서 널리 이루어져 왔다. 이러한 시도 중에서 은닉 마르코프 모델은 가장

널리 연구되고 음성인식 등을 위한 대표적인 방법이 되었다. 은닉 마르코프 모델(이하 HMM)은 J. Baker가 처음으로 연속 음성 인식 문제에 적용할 수 있음을 [13] 보였는데, 모델 자체는 그보다 앞서 Baum 등에 의하여 효율적인 알고리즘이 출현하면서부터 각광을 받기 시작했다[14]. HMM은 정형화된 수학적 도구로써, 순차적으로 발생하는 신호의 다양한 시간적 변형을 통계적으로 모델링하는데 뛰어나다. 따라서 시간적 제약을 받는 온라인 필기 인식, 음성 신호 처리 및 인식에서 성공리에 활용되어 왔고, 통계적 접근 방법 중에서 오늘날 가장 널리 알려져 있고 성공적인 방법이다.

HMM은 마르코프 체인과 그 함수로서 관찰 심볼 확률 분포로 정의할 수 있는데, 확률적 마르코프 체인 또는 마르코프 체인의 확률 함수라고 정의할 수 있다. 일반적으로 은닉 마르코프 모델은 다음과 같이 정의한다[15]:

- 초기 상태 확률 분포

$$\pi = \{\pi_i | \pi_i = P(q_0 = S_i), 1 \leq i \leq N\}, \sim \sum_i \pi_i = 1.$$

- 상태 전이 확률 분포

$$A = \{a_{ij} | a_{ij} = P(q_{t+1} = S_j | q_t = S_i), 1 \leq i \leq N\}, \sim \sum_j a_{ij} = 1.$$

- 관측 심볼 확률 분포

$$B = \{b_i(k) | b_i(k) = P(v_k | S_i), 1 \leq i \leq N\}, \sim \sum_k b_i(k) = 1, \\ 1 \leq i \leq N, v_k \in V.$$

그리고 이밖에 모델의 상태의 집합  $S = \{s_1, s_2, \dots, s_N\}$ 와 관찰 심볼의 집합  $V = \{v_1, v_2, \dots, v_M\}$ 가 있다. 그러나 보통 위에 설명한 모델의 동적 특성을 기술하는 세 가지 파라미터로

$$\lambda = (A, B, \pi)$$

와 같이 표시한다.

HMM을 실제 문제에 사용하기 위해서 해결해야 할 세가지 중요한 문제가 있다. 그 첫째가 모델 evaluation 문제인데 Baum 등에 의하여 제안된 forward-backward 알고리즘으로 해결 되었다[14]. 이 알고리즘은 관측

심볼열  $O=O_1O_2...O_T$ 와 모델  $\lambda$ 가 주어졌을 때, 주어진 모델의 관측 확률  $P(O|\lambda)$ 을 효과적으로 계산하는 방법이다. 그리고 이 알고리즘을 기반으로 하여 모델의 파라미터를 추정해내는 방법 또한 개발되었는데 그것이 EM(Estimation-Maximization) 기법에 기반한 두 번째 문제의 해결책이다. 한편, 관측열  $O$ 와 모델  $\lambda$ 가 주어졌을 때, 이에 대응되는 최적 상태열  $Q=q_1q_2...q_T$ 를 찾는 문제가 세 번째 문제이다. 이에 대해서는 별도로 동적 프로그래밍 기법에 의한 Viterbi 알고리즘이 이미 존재한다[16]. Viterbi 알고리즘은 forward-backward 알고리즘의 forward 절차와 매우 유사하다. 이 알고리즘은 격자 형태의 계산구조로 쉽게 구현할 수 있다. 본 논문에 기술된 글자 탐색 방법도 이 알고리즘에 기반을 두고 있다.

### 2.3 글자 모델

한글은 훈민정음 24자를 기본으로 하여 복자음, 복모음이 포함된 초성 19자, 중성 21자, 그리고 종성 27자 등 총 67 자소의 조합으로 이루어진 과학적인 글자 체계이다. 각 글자는 초성, 중성, 종성의 결합으로서, 숫자 또는 영어와는 달리 특별한 경우를 제외하고는 자소 독립적으로 쓰이지 않는다. 그러므로 한글에 있어서의 기본 모델은 음절 모델이 된다.

2.1절에서 설명한 온라인 필기의 특징에 따르면 한글 필기는 초성, 중성 및 종성의 연결로 정의할 수 있다. 그러나 손의 물리적 운동에 따라 한글 음절 필기를

〈글자〉:=〈초성〉〈연결획〉〈중성〉〈연결획〉〈종성〉

으로 정의된 바가 있다[3]. 여기서 〈연결획〉은 다음 소절에서 설명할 자소간의 패턴을 의미한다. 이 정의에 따라서 초성, 중성, 종성 각각에 대해서 하나씩 총 67개의 은닉 마르코프 모델을 설계한 다음 이들을 ‘초성+연결획+중성’ 또는 ‘초성+연결획+중성+연결획+종성’의 실시간 연결로 음절 모델을 구성할 수 있다.

### 2.4 자소 모델

일반적으로 로마자 또는 숫자 등의 필기에서와 같이 왼쪽에서 오른쪽으로 진행하면서 필기하는 구조의 필기 체계에는 글자 또는 단어의 공간적 배치 구

조에 대한 특별히 엄격한 제약 조건은 없다. 그리고 이러한 필기 체계는 숫자 또는 알파벳이라는 최소 단위로 필기를 분석할 수 있다. 숫자 또는 알파벳 각각을 모델링하는 HMM은 숫자는 각각 하나씩 총 10개, 로마자인 경우에는 대문자, 소문자 각각 26개씩의 모델을 설계한다. 이들 모델의 구조는 왼쪽에서 오른쪽으로 전이하는 전진 구조(그림 1)를 갖도록 하였다. 온라인 필기의 시간 정보를 구조적으로 표현하고, 또 계산상의 효율을 기할 수 있기 때문이다.

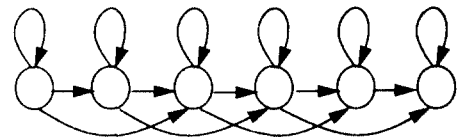


그림 1. 자소, 숫자, 알파벳의 우향 전이 구조의 마르코프 모델.

한글의 초성, 중성, 종성의 자소도 비슷한 구조의 모델을 만들 수 있다. 한글의 구조 분석에 따르면 여섯가지의 형식 분류가 있으며, 이에 따라 글자 모델을 만드는 연구 결과를 많이 볼 수 있다. 이에 따르면 초성은 각각 6가지의 변형, 중성은 각각 2가지, 그리고 종성은 각각 3가지씩의 변형으로 세분하여 표준 패턴 모델을 설계한다. 하지만 일반적으로 필기에서 그러한 구분이 적용될 수 있는가에 대해서는 의문의 여지가 있다. 그래서 본 연구에서는 초성 19개, 중성 21개, 그리고 종성 27개 등 자소당 모델을 하나씩 설계하였다.

### 2.5 연결획 모델

일반적으로 필기에는 인쇄체에서는 볼 수 없는 특징이 있다. 그 중 가장 두드러진 것이 흘려 쓴 한글이나 영어 필기체에서와 같이 여러 획을 하나로 이어 쓸 때 자소와 자소, 알파벳과 알파벳 사이에 나타나는 연결획이 그것이다. 연결획은 한글 필기, 특히 흘려 쓴 글씨에서도 볼 수 있는데, 공간적으로 떨어져 있는 두 획을 한데 이어쓰기 때문에 나타나는 특징이다.

필기에서는 자소간 연결획이 글자 모양의 주된 변형 요인으로 작용한다. 음성 인식에서의 조음 결합(co-

articulation)과는 달리 온라인 필기 인식에서는 이러한 변형의 근원을 모델링하는 체계적 연구가 없다. 최근에 연결획 모델을 도입함으로써 무제한 자유 필기 인식 방법론이 발표된 바 있다. 본 연구에서는 [3]에서 제안된 바와 같은 형식의 연결획 모델을 설계하였다. 그리고 모델의 구조 또한 비슷하게 우향 전진 구조이며 모델의 상태수는 연결획 패턴의 단순함을 고려해서 그림 2와 같이 3개로 하였다.

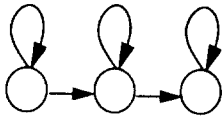


그림 2. 연결획 모델의 구조.

즉 연속해서 쓴 필기에서 숫자와 숫자, 영자와 영자, 그리고 한글 글자 사이의 펜 이동 궤적에 대해서 하나의 연결획 모델을 구축하는 것이다. 일반적으로 연결부의 펜 이동 궤적은 대체로 일정하여 거의 직선 형태이며 일관된 방향을 보인다. 따라서 본 연구에서 설계하는 연결획 모델은 극히 단순한 구조를 채택하였다.

### 2.6 통합 모델

모든 자소, 숫자, 알파베트 등 모든 글자와 정합될 수 있는 범용 모델로서 통합 모델을 정의한다. 특정 숫자 또는 알파베트 모델과는 달리 이것은 모든 훈련 데이터를 한데 모아 훈련시킨다. 특정 글자에 대응된 모델 보다 모델링 능력은 많이 떨어지지만, 그외 다른 글자에 대해서는 어느 정도의 수용 능력을 갖고 있어, 해당 글자 뿐만 아니라 모든 글자를 흡수할 수 있게된다.

이 모델을 훈련하는 것은 앞서 말한 바와 같이 모든 훈련 데이터를 한데 모아 훈련시키는 방법 외에 전혀 훈련 시키지 않고 무작위로 설정한 초기 파라미터를 그대로 쓰는 방법도 있다. 본 연구에서는 후자의 방법을 채택하였다. 그리고 모델의 크기는 키 자소를 제외한 자소 모델 평균 상태수의 합으로 하였다. 구조는 자소 모델과 같은 우향 전진 구조를 갖는다.

### Ⅲ. 자소 탐색 네트워크

글자 탐색을 위한 글자 키 모델의 구조는 앞에서 정의한 HMM을 기초로 적절한 정합 모델을 구축하는 것이다. 숫자 또는 알파베트 하나를 탐색하는 문제라면 목적 알파베트 모델을 키 모델로 하고, 알파베트 사이의 연결획 모델(ligature HMM), 그리고 통합 모델(wild HMM)을 결합하여 그림 3과 같은 구조로 된 글자 탐색 네트워크 모델을 구성할 수 있을 것이다. 그림은 숫자 '4'를 탐색하기 위한 네트워크 모델로서 하나의 싸이클을 구성하고 있는데, 다음 절의 탐색 알고리즘에 따라서 입력 데이터와 최적의 구간 매칭을 시도함으로써 통계적으로 최적의 영역을 찾아낼 수 있다.

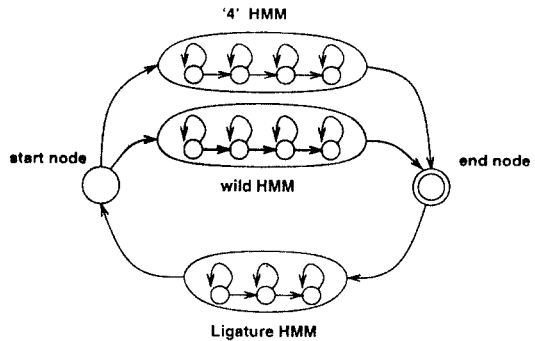


그림 3. 글자(숫자 '4') 탐색 모델: Wild HMM과 Ligature HMM은 각각 통합 모델, 연결획 모델을 나타낸다.

그러나 한글은 좌에서 우로 필기가 진행되는 영어 단어나 숫자와는 달리 이차원 공간 배열 구조를 갖고 있다. 이러한 배열은 한글 해독에 큰 역할을 하는 만큼 글자 인식에 중요한 정보로 활용된다. 따라서 그림 3과 같은 단순한 구조의 네트워크로는 다양한 모양의 필기 인식 탐색에는 무리가 있다. 한편, 자소 탐색에 있어서 또 하나의 문제는 그림 4와 같이 거짓 자소(false alarm) 또는 탐색 오류가 많이 생긴다는 것이다[12].

이러한 거짓 자소는 종종 높은 확률을 나타내는데, 그에 따라 다음 단계의 탐색 공간이 크게 확장된다. 실제로 탐색 오류를 최소화하는 문제가 단어 탐색 연구자들의 커다란 이슈로 되어왔다[17]. 이와 관련된 문제로 그럴듯한 자소의 경계를 찾기가 쉽지 않다는 점도 있다. 특히 흘려쓴 필기체에서 자소의 경계가



그림 4. 거짓 자소(false alarm) 또는 탐색 오류: 'ㄹ'의 탐색 결과.

부드럽게 선형으로 이어진 부분에서 경계는 매우 애매해지기 때문이다. 올바른 인식은 글자 내의 모든 자소 및 그 경계를 정확하게 찾아야만 한다. 따라서 자소 경계의 좋고 나쁨은 자소 탐색에 근거한 인식기

의 성공을 크게 좌우하게 된다.

위 문제점을 해결하기 위한 자소 탐색 모델은 2.3절의 정의에 따라 구성된 글자 모델을 기초로 하며, 이를 변형하여 각 자소에 대해 하나씩 구성한다. 글자 모델은 앞 절의 자소 및 연결획 HMM을 이용하여 네트워크 구조로 표현한다(그림 5). 본절에서 제안하는 각 자소별 탐색 모델의 구조는 그림 6과 같다.

초성 탐색 네트워크는 키 자소로써 초성 HMM(C-Key HMM)과 그 초성에 연결되는 연결획 모델(Ligature HMM), 그리고 통합 모델(Wild-HMM)에 연결되어 있다. 중성 모델은 이와는 달리 키 자소 중성 HMM(V-Key HMM)을 가운데 두고 그림 6(b)와 같이 앞(좌측)에 통합 모델과 연결획 모델을 붙이며, 뒤

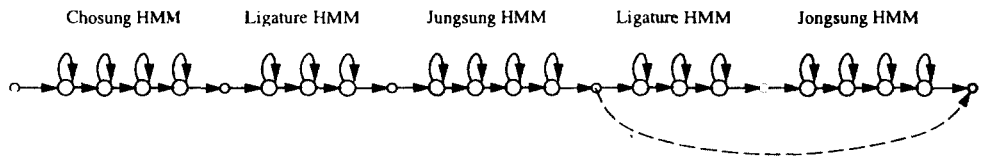


그림 5. 글자 모델.

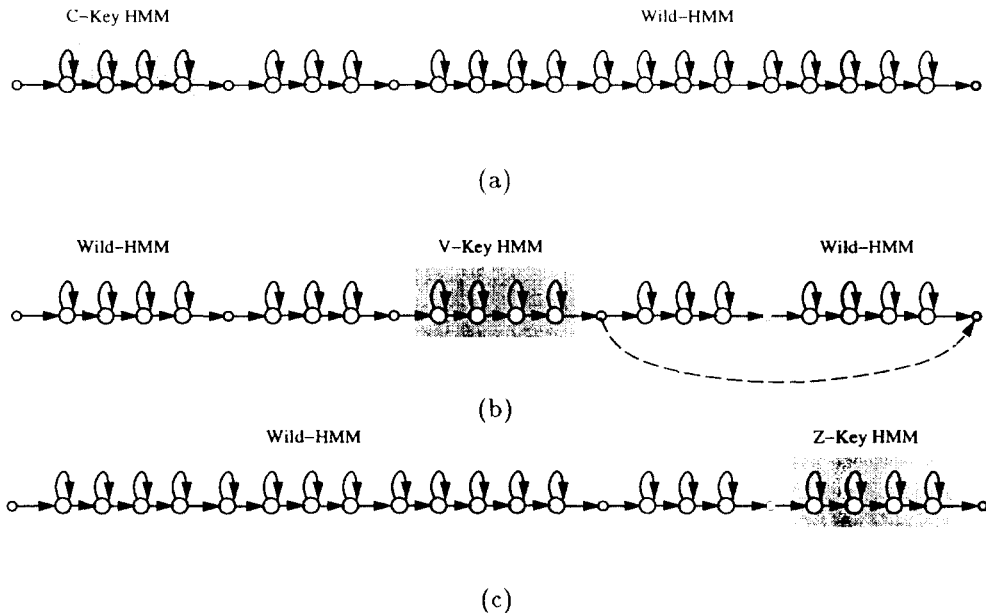


그림 6. 자소 탐색 모델: (a) 초성 탐색 모델, (b) 중성 탐색 모델, (c) 종성 탐색 모델. C-Key HMM, V-Key HMM, Z-Key HMM은 각각 키 자소 초성 HMM, 중성 HMM, 종성 HMM이며, Wild-HMM은 통합 모델을, Ligature-HMM은 연결획 모델을 나타낸다.

(우측)에는 연결획과 통합 모델을 차례로 연결한다. 이는 한글 필기가 초성, 중성, 종성 순이라는 데서 나온 것으로 앞쪽의 통합 모델은 임의의 초성을 수용하고, 뒷쪽의 통합 모델은 중성을 수용하는 것이다. 한편 받침이 없는 글자의 경우는 탐색과정에서 해결하도록 한다. 마지막으로 중성 탐색 모델은 초성 모델과 유사하다. 중성 HMM(Z-Key HMM)이 탐색 네트워크의 좌우 천이 구조에서 최우측에 있다는 점 외에는 다른 점이 없다.

위와 같은 구조의 모델을 이용했을 때 탐색 결과 얻을 수 있는 자소의 경계는 홀림체에 대해서도 뛰어난 결과를 보여준다. 전형적인 예로써 그림 7은 그림 4의 오류와는 달리 매우 개연성 있는 초성 탐색 결과를 보여준다.

랑

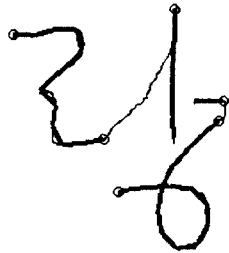


그림 7. 올바른 탐색 결과: 자소 탐색 결과는 굵은 선으로 표시.

#### IV. 인식 알고리즘의 구조

##### 4.1 자소 탐색

자소 탐색 구조는 효율성을 높이기 위해서 DP 기법에 의한 순차적 디코딩 구조를 갖도록 하였다. 그러나 표준 비터비(Viterbi) 알고리즘과 같이 확립적으로 순방향 탐색 구조를 가질 필요는 없고, 역방향 순차 탐색 또는 순방향과 역방향 탐색이 혼합된 양방향 탐색 구조를 가질 수 있다. 한글의 자소 탐색은 초성, 중성, 및 종성 탐색의 세개의 독립적인 소문제로 나뉘어지며 이들은 병렬 처리가 가능하다. 각각의 탐색 방법은 다음과 같다.

자소 탐색은 입력  $X = x_1 x_2 \dots x_T$ 와 탐색 모델  $M$ 을 정합하는 과정인데  $P(X, Q | M)$ 을 최대화 하는 Viterbi

순방향 디코딩 알고리즘을 사용한다.  $Q = q_1 q_2 \dots q_T$ 는 입력  $X$ 에 배열된 모델  $M$ 의 상태열이고,  $T = |X| = |Q|$ 는 열의 길이를 가리킨다. 베이스 정리에 따라

$$P(X, Q | M) = P(X | Q, M) P(Q | M)$$

과 같이 쓸 수 있고, 자소 탐색의 결과는

$$\max_Q P(X | Q, M) P(Q | M)$$

에서 얻을 수 있다. 위식에 Markov 가정과 입력 심볼 간 독립 가정을 적용하면 입력과 HMM 경로 간의 배열 함수

$$\max_Q \prod_{t=1}^T a_{q_{t-1}q_t} \prod_{t=1}^T b_{q_t}(x_t)$$

가 된다.  $a_0, q_0 = \pi_{q_0}$ 로 표현한다. 위 함수를 최대화 하는  $Q$ 를  $\bar{Q}$ 라고 하자.

[초성 탐색]

$\bar{Q} = \bar{Q}^C \bar{Q}^{LW} = \bar{Q}^C \cdot \bar{Q}^{LW}$ . 여기서  $\bar{Q}^C$ 는 초성 키 모델에 속한 상태열이고  $\bar{Q}^{LW}$ 는 나머지 부분, 즉 연결획과 통합 모델에 속한 상태열이다.  $0 < |\bar{Q}^C| = \tau < T$ 이라고 하면 초성 가설은 다음과 같은 자료 구조로 정의한다.

$$\left( 1, \tau, \prod_{t=1}^{\tau} a_{q_{t-1}q_t}, \prod_{t=1}^{\tau} b_{q_t}(x_t) \right).$$

[중성 탐색]

$\bar{Q} = \bar{Q}^{WL} \bar{Q}^V \bar{Q}^{LW} = \bar{Q}^{WL} \cdot \bar{Q}^V \cdot \bar{Q}^{LW}$ . 여기서  $\bar{Q}^V$ 는 중성 키 모델에 속한 상태열이고  $\bar{Q}^{WL}$ 과  $\bar{Q}^{LW}$ 는 전후의 나머지 부분, 즉 각각 통합 모델-연결획 모델, 연결획 모델-통합 모델에 속한 상태열이다.  $|\bar{Q}^{WL}| = \tau_1 > 0$ ,  $|\bar{Q}^V| = \tau > 0$ ,  $|\bar{Q}^{LW}| = T - \tau_1 - \tau > 0$ 이라고 하면 중성 가설은 다음과 같다.

$$\left( \tau_1 + 1, \tau_1 + \tau, \prod_{t=\tau_1+1}^{\tau_1+\tau} a_{q_{t-1}q_t}, \prod_{t=\tau_1+1}^{\tau_1+\tau} b_{q_t}(x_t) \right).$$

[종성 탐색]

$\bar{Q} = \bar{Q}^{WL} \bar{Q}^Z = \bar{Q}^{WL} \cdot \bar{Q}^Z$ . 여기서  $\bar{Q}^Z$ 는 종성 키 모델에 속한 상태열이고  $\bar{Q}^{WL}$ 는 나머지 부분, 즉 통합 모

델과 연결획 모델에 속한 상태열이다.  $0 < |Q^x| = \tau < T$  이라고 하면 중성 가설은 다음과 같다.

$$\left( T-\tau+1, T, \prod_{l=T-\tau+1}^T a_{\bar{q}_{l-1} \bar{q}_l}, \prod_{l=T-\tau+1}^T b_{\bar{q}_l}(x_l) \right).$$

시간 축상에 각 자소 가설의 영역을 도시하면 그림 8과 같다. 이러한 그림을 격자라고 부른다. 그림에서 가로 빗장은 각각 자소의 영역을 나타내며  $l_i$ 는 자소 분할점을 나타낸다.

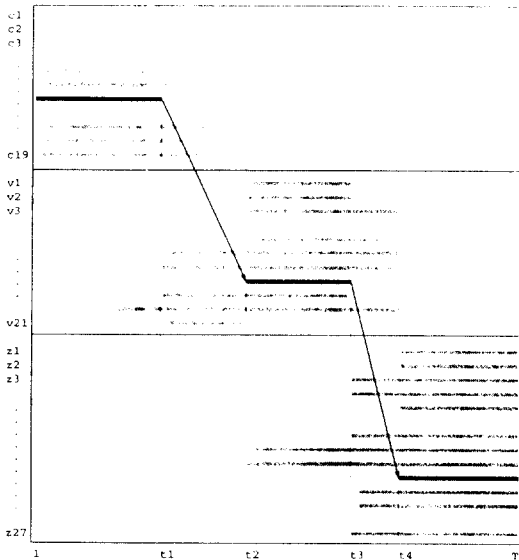


그림 8. 자소 가설의 격자.

자소 탐색에는 초성, 중성, 종성 모든 자소 탐색 모델과 한번씩의 정합 과정을 거치게 된다. 각 탐색 모형은 전체 크기(상태수)가 평균  $N$ 이고, 각 상태에  $k$ 개의 아크가 있는 우향 전진 구조의 HMM이라고 하자. 그러면 소요되는 총 계산량은 탐색 모델의 수를  $M$ 이라 할 때  $O(MkNT)$ 의 크기를 갖는다. 이때 자소 탐색 모델의 크기는 고정되어 있으므로, 입력에 정비례하는 복잡도를 갖는다. 만약 HMM의 상태 전이 구조와 입력 자소 길이의 변이를 고려하여 탐색 공간을 제한한다면 보다 효과적으로 탐색이 가능할 것이다.

### 4.2 격자 탐색

한글은 단순한 자소의 나열이 아니라 제자 원리에

따르는 자소 조합 규칙이 존재한다. 그러므로 글자 인식을 위해서는 탐색된 자소를 결합하여 하나의 글자로 인식하는 또 하나의 탐색이 뒤따라야 한다. 그 과정은 자소 탐색의 결과로 나오는 자소 가설을 이용하여 최적 정합값을 내는 자소 조합을 찾는다.

자소열은 연결획을 위한 영역을 사이에 두고 자소의 영역을 이어서 만들게 된다. 이 때 자소 영역 간에는 중첩이 없어야 한다. 실제 탐색은 기존에 알려진 그래프 탐색 알고리즘을 쓸 수 있다. 그리고 중첩 불허 조건을 제외하면 초성은  $t=1$ 에서 시작하고, 종성은  $t=T$ 에서 끝난다는 끝점 조건만 고려하면 된다. 이 과정은 가설의 격자 구조의 표현 공간 상에서 탐색을 하게 되므로 격자 탐색 구조라 불린다.

결합 가설은 하나의 글자를 이룰 수 있는 것이어야 한다. 즉 한글의 제자 원리에 어긋남이 없어야 한다. 또한 한글 완성형만을 대상으로 한 경우에는 완성형 글자 세트에 포함되는가의 여부를 검사하는 여과 과정도 격자 탐색 과정에 병행토록 한다. 격자 탐색 자체는 동적 프로그래밍 기법 또는 A\* 알고리즘 등 그래프 탐색에 쓸 수 있는 일반적인 알고리즘을 응용하면 간단히 해결할 수 있다. 그러나 한글 자소 가설 격자는 매우 간단하므로 순차적인 exhaustive search를 해도 크게 부리가 없다.

### 4.3 언어 모델

패턴 인식이 인간과 같이 뛰어난 인식 능력을 갖 추기 위해서는 이미 인식 대상에 대한 사전 정보를 많이 갖고 이를 적절히 활용할 수 있어야만 한다. 문자 인식에서 사전 정보는 언어적 지식이라고 할 수 있는데, 일반적으로 사전을 쓰거나 확률을 추정하는 통계적 언어 모델을 사용한다. 한글 음절의 경우 조합형 11,172자 중 완성형 2,350자만이 실생활에 쓰이고 있으므로 글자 언어 모델의 구축은 그 난이도에 있어서 단어 모델과 마찬가지로 할 수 있다. 본 연구에서는 매끄러운 결합을 위해 확률 모델을 사용하되 간단한 bigram 모델을 선택하였다. bigram은 일차 마르코프 체인 언어 모델이므로 HMM의 성질과 일치한다. 따라서 두 모델을 결합하는데 이론적으로 아무런 문제가 없다.

## V. 실험 결과



### 5.1 시스템 구성

필기 모델과 언어 모델이 구성되고 인식 알고리즘이 구현되면 비로소 하나의 인식 시스템으로 구현할 수 있다. 인식기의 구조와 기능을 설명하면 다음과 같이 단계별로 나뉘어 볼 수 있다.

- 전처리: 하드웨어 장치의 결함이나 손운동의 불완전성 등에 기인하는 각종 잡음을 제거하고 깨끗한 형태의 필기 원형으로 복구시켜 준다.
- 정규화: 사람에 따라 크기가 다른 필기를 이와 무관하게 인식할 수 있도록 표준 형태로 변환하는 과정으로 표준 크기로 축소 또는 확대하고, 이를 다시 일정 간격으로 필기 궤적을 resampling 한다.
- 데이터 코딩: 정규화된 데이터를 은닉 마르코프 모델이 처리할 수 있는 심볼열로 변환한다. 코드로는 방향각을 양자화한 체인 코드를 쓴다.
- 자소 탐색: 자소 탐색은 상기한 바와 같이 개별적으로 탐색을 실시한다. 순차적 디코딩의 Viterbi 알고리즘을 사용한다. 초, 중, 종성 병렬 탐색이 가능하다.
- 격자 탐색: 자소 탐색의 결과를 가설로 입력하여 제자 원리에 따른 결합 가설을 만들고 이 중 최적의 글자와 영역 분리 형태를 선택한다. 이때 일반적으로  $N \setminus g_{c1} \setminus 1$  개의 후보 글자를 생성토록 한다.
- 후처리: 언어 모델 및 그외 다른 사전 지식을 적용한다. 그 결과 격자 탐색에서 나온 후보들의 순서가 뒤바뀔 수 있다.
- 코드 변환 및 출력: 최적의 글자가 결정되면 초성, 중성, 종성이 결합된 조합형 글자를 조합형 코드 또는 완성형 코드로 변환하고, 화면에 출력한다.

### 5.2 모델 훈련

필기 모델은 다양한 필기에도 고르게 높은 수용 능력을 보이도록 해야 한다. 이점은 훈련과 밀접한 관계가 있는데 될수록 많은 양의 훈련용 데이터를 수집해야 한다. 자소와 연결획 모델의 훈련 데이터는 깨끗이 정서한 필기와 자연스럽게 흘려쓴 자연스런 필기 총 40명분을 한데 모아 사용하였다. 필기 데이터는 필순은 지키되 그외의 특별한 제약은 가하지 않고 쓰도록 하였다.

필기는 1/100 초당 한점의 속도로 샘플링하였으며, 체인 코드는 360도를 16 등분한 16 방향 코드 체계를 사용하였다. 자소별로 분할한 다음 체인 코드로 변환한 후 forward-backward 알고리즘으로 훈련을 실시하였다[14]. 훈련 반복은 100회 이내에서 끝내도록 하였다. 그리고 HMM은 Moore 타입의 모델로서 상태에서 심볼 출력이 있다.

### 5.3 인식 실험

글자 인식 실험에 사용한 데이터는 대학생으로부터 받은 정서한 필기 4인분과 자연스럽게 쓴 8명분의 필기로써 각각 총 2218자, 756자이다. 이것을 각각 데이터 I, 데이터 II라고 부르자.

인식 실험 결과는 표 1과 같다. 예상대로 정서한 데이터 I에서 높은 인식률 87.47%(후보 5개를 인정할 때 92.70%)을 얻었고 데이터 II에서는 상대적으로 낮은 80.82%(87.96%)의 인식률을 얻었다. 이것은 정서한 경우 자소의 경계 탐색 오류가 발생할 가능성이 훨씬 낮기 때문이라 판단된다. 표 하반부의 수치는 [18]의 연구 결과를 본 실험과 동일한 조건에서 실시했을 때 얻은 것이다. 비슷한 성능과 성능 변화의 추이를 볼 수 있다. -따라서 본 논문의 간단한 방법으로도 우수한 성능을 얻을 수 있다는 가능성을 확인할 수 있다.

표 1. 평활화 전후 모델의 인식 실험 결과. 숫자 쌍은 각각 후보 1개와 후보 5개에 의한 적중률(%)이다.

모 델	평 활 화	데 이 터 I	데 이 터 II	비 고
제안된 방법	평활화 전	87.47/92.70	80.82/87.96	
	평활화 후	81.38/87.29	83.07/88.62	1:2:1 평활화
[18]의 방법	평활화 전	87.51/93.24	84.13/91.40	
	평활화 후	86.70/92.52	85.98/91.27	1:1.5:1 평활화

HMM을 이용한 기존의 연구 결과[18]에서는 훈련이 충분치 못하여 확률 분포 함수를 평활화하여 큰 효과를 보았다. 그러나 본 연구에서는 예상치 못한 엇갈리는 결과를 얻었다. 표 1에서 출력 확률 분포 함수를 평활화한 모델의 성능은 데이터 II의 경우는 [18]의 연구 결과와 마찬가지로 약간의 증가를 보였으나, 데이터 I의 경우에는 인식률이 오히려 크게 떨어져 48.6%에 달하는 오류의 증가율을 보인 것이다.

위 현상의 주된 이유는 [18]의 연구와 달리 상태에

서 출력하는 모델을 사용한다. 즉, 상태 출력의 경우 훈련 파라미터의 수가 그리 크지 않기 때문에 훈련이 충분히 될 가능성이 높으며, 따라서 훈련된 모델을 평활화하는 경우 최적화된 모델을 깨뜨리는 결과를 가져오기 때문이다. 그리고 이점은 정서한 필기에 더욱 민감했다고 판단된다. 그리고 흘림 필기에는 거의 훈련이 안된 상태였기 때문에 자연스런 필기에 있어서는 평활화의 효과가 어느 정도 있었다고 판단된다.

위 결과에 따르면 구형 시스템의 성능이 충분히 높지는 않지만 간단한 인식 구조를 감안할 때 개선의 가능성은 매우 높다고 판단된다. 특히 현 시스템에서는 각 자소당 단지 하나씩의 가설을 만든다. 따라서 자소 가설 결합 과정에서 약간의 영역 중첩이 발생하면 정인식될 가능성이 전혀 없어질 수도 있다. 그럼에도 불구하고 표 1에서 보는 바와 같이 서로 필적할 만한 결과를 낸 것은 방법론 상의 우수성을 입증하는 것이라 할 수 있다.

표 1의 모델은 모두 간단한 bigram 언어 모델을 사용한 것이다. 참고로 언어 모델을 사용하지 않았을 때의 성능을 비교해보면 표 2와 같다. 평활화 전의 기본 모델을 이용하여 본 성능 차이는 언어 모델의 효과가 대단함을 보여준다.

표 2. 언어 모델의 효과; 평활화 전 모델에서 언어 모델 사용하지 않은(uniform) 경우와 bigram 언어 모델 사용한 경우.

언어 모델	데이터 I	데이터 II	비 고
uniform	71.10/92.43	59.13/87.70	언어 모델 없음
bigram	87.47/92.70	80.82/87.96	

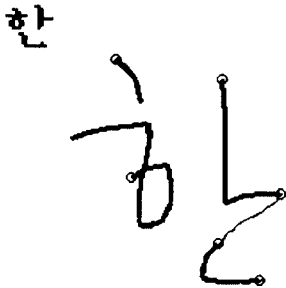


그림 9. 글자 인식 및 자소 분할의 예. 자소의 영역은 굵게 표시되어 있다.

그림 9는 흘려쓴 필기를 정확하게 자소 분할하고 인식해낸 예를 보여준다. 특히 종성 시작 부분의 경계점은 매우 애매모호 하지만 본 시스템에서는 훌륭한 경계점을 찾아내었다.

## VI. 결 론

자소 탐색에 기반한 글자 인식 방법의 성패는 키 자소의 탐색의 결과 질에 크게 좌우된다. 따라서 자소 탐색 분야의 주요한 초점은 어떻게 강력한 모델을 설계하고 높은 분별력을 가지도록 훈련시킬 것인가에서 찾을 수 있는 것이다.

앞에서 간단히 언급하고 넘어간 문제 중에 향후 과제로 넘긴 것은 탐색 알고리즘에 있다. 현재는 자소 결합시 자소간에 약간의 중첩이라도 허용하지 않는다. 그리고 자소별로 가설을 한개씩만 만들고 다른 가능성은 부인하고 있다. 이점은 다중 후보를 생성하도록 개선함으로써 어느 정도 해결이 가능하다고 본다.

또하나 남아 있는 관련 문제점은 자소 가설 생성시 정합 우도에 따라 가설을 기각할 수 있도록 하는 것이다. 가설의 수를 줄여서 탐색 공간을 축소시켜 주변 성능에 영향을 미치지 않고 속도를 크게 개선할 수 있게 될 것이다.

## 참 고 문 헌

1. 성태진, 방승양, "문자 조합 규칙 학습에 의한 한글 온라인 필기 인식기의 설계," 한국정보과학회 추계 학술 발표 논문집, pp. 223-226, 1991. 10.
2. S.-H. Lee, K.-Y. Lee, and Y. Lee, "A generalized character recognition using SEJONG-NET," in *Proc. of the 2nd Pacific Rim Conf. on AI*, (Seoul, Korea), pp. 947-952, Sep. 1992.
3. 신봉기, 김진형, "통계적 방법에 의한 온라인 한글 필기 인식," 제4회 한글 및 한국어 정보처리 학술 발표 논문집, pp. 533-542, 1992. 10.
4. 정봉만, 권오석, 김태균, "문법적 추론과 특징매치법을 이용한 학습-인식 시스템 개발," 정보과학회 논문지, 제 16권 제 1호, pp. 84-91, 1989. 1.
5. M.-W. Feng and B. Mazor, "Continuous word spotting for applications in telecommunications," in

*Proc. IEEE ICASSP*, pp. 21-24, Oct. 1992.

6. R. C. Rose and D. B. Paul, "A hidden Markov model based keyword recognition system," in *Proc. IEEE ICASSP*, pp. 130-132, Apr. 1990.
7. J. G. Wilpon, L. R. Rabiner, C.-H. Lee, and E. R. Goldman, "Automatic recognition of keywords in unconstrained speech using hidden Markov models," *IEEE Trans. on Acoustics, Speech and Signals Processing*, vol. 38, pp. 1870-1878, Nov. 1990.
8. F. R. Chen, L. D. Wilcox, and D. S. Bloomberg, "Word spotting in scanned images using hidden Markov models," in *Proc. IEEE ICASSP*, pp. 1-4, 1993.
9. M. Cheriet and C. Y. Suen, "Extraction of key letters for cursive script recognition," *Pattern Recognition Letters*, vol. 14, pp. 1009-1017, 1993.
10. G. E. Kopec and P. A. Chow, "Document image decoding using Markov source models," in *Proc. IEEE ICASSP*, pp. 85-89, 1993.
11. S.-S. Kuo and O. E. Agazzi, "Machine vision for keyword spotting using pseudo 2D hidden Markov models," in *Proc. IEEE ICASSP*, pp. 81-84, 1993.
12. S. -H. Lee and J. H. Kim, "On-line cursive script recognition by a letter spotting technique based on HMMs," in *Proc. of the 2nd Workshop on Character Recognition*, (Seoul, Korea), pp. 93-104, Sep. 1994.
13. J. K. Baker, "The DRAGON system-an overview," *IEEE Trans. on Acoustics, Speech and Signals Processing*, vol. 23, pp. 24-29, Feb. 1975.
14. L. E. Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," *Inequalities*, vol. 3, pp. 1-8, 1972.
15. L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257-285, Feb. 1989.
16. A. J. Viterbi, "Error bounds for convolution codes and an asymptotically optimum decoding

algorithm," *IEEE Trans. on Information Theory*, vol. 13, pp. 260-269, 1967.

17. R. A. Sukkar and J. G. Wilpon, "A two pass classifier for utterance rejection in keyword spotting," in *Proc. IEEE ICASSP*, pp. 451-454, Apr. 1993.
18. 신봉기, 김진형, "은닉 마르코프 모델 네트워크에 의한 온라인 홀림 필기 한글 인식," *정보과학회 논문지*, 제 35권, 제 5호, pp. 1737-1745, 1994. 9.



조 범 준(Bum-Joon Cho)정회원

1980년: 조선대학교 졸업  
 1988년: 한양대학교 대학원 졸업  
 (공학박사)  
 1989년~1990년: University of Connecticut(Post-Doc)  
 1990년~1991년: University of Massachusetts(Post-Doc)

1993년~현재: 조선대학교 전자계산소장  
 1980년~현재: 조선대학교 공과대학 컴퓨터공학과 부  
 교수  
 ※주관심분야: 패턴인식, 컴퓨터 비전