

論文94-31B-9-10

Hidden Markov Model을 이용한 필기체 한글 및 영·숫자 오프라인 인식

(Off-line Recognition of Handwritten Korean and Alphanumeric Characters Using Hidden Markov Models)

金 禹 成*, 朴 來 弘**

(Woo Sung Kim and Rae-Hong Park)

要 約

본 논문에서는 이산 hidden Markov model (HMM)을 이용하여 제한된 필기체 한글 및 영·숫자를 인식하는 알고리즘을 제안한다. HMM은 doubly stochastic한 모델로서, 패턴의 다양한 클래스들 사이의 패턴의 왜곡과 유사성을 표현할 수 있는 모델이다. 문자에서 추출된 특징들의 통계적 특성에 의거하여, 입력 패턴의 변화를 흡수함으로써 인식 시스템을 구축할 수 있다. 퍼지 추론에 의하여 한글을 6가지 유형으로 분류하고, 각 유형마다 자소들의 특성에 의거하여 특징을 양자화하고 구별력이 강한 특징들을 먼저 배열시켜 HMM의 학습과 인식을 수행한다. 또한 한글 인식에 사용했던 특징들을 이용하여 제한된 영·숫자 인식 실험을 수행한다. 제한된 한글 및 영·숫자의 학습 및 인식에 forward-backward 알고리즘과 Viterbi 알고리즘 및 Baum-Welch 재추정 알고리즘을 이용한다. 제안된 방법이 필기체 한글 및 영·숫자 인식에 효율적임을 실험을 통해서 알 수 있었다.

Abstract

This paper proposes a recognition system of constrained handwritten Hangeul and alphanumeric characters using discrete hidden Markov models (HMM). HMM process encodes the distortion and similarity among patterns of a class through a doubly stochastic approach. Characterizing the statistical properties of characters using selected features, a recognition system can be implemented by absorbing possible variations in the form. Hangeul shapes are classified into six types by fuzzy inference, and their recognition is performed based on quantized features by optimally ordering features according to their effectiveness in each class. The constrained alphanumeric recognition is also performed using the same features used in Hangeul recognition. The forward-backward, Viterbi, and Baum-Welch reestimation algorithms are used for training and recognition of handwritten Hangeul and alphanumeric characters. Simulation result shows that the proposed method recognizes handwritten Korean characters and alphanumerics effectively.

*正會員, 湖西大學校 컴퓨터工學科
(Dept. of Computer Eng., Hoseo Univ.)

(Dept. of Elec. Eng., Sogang Univ.)
接受日字: 1993年 11月 22日

**正會員, 西江大學校 電子工學科

1. 서론

문자의 자동 인식 (automatic recognition)에 관한 연구가 지난 20년간 활발히 진행되어 왔다.^{[1][2]} 그러나 아직까지도 필기체 문자의 실시간 인식의 경우에는 문자 영역에 많은 제한을 두어야만 가능하고 인쇄체 문자도 문자수의 제한을 두는 수포나 영수증에 있는 숫자의 실시간 자동 인식 시스템이 개발된 정도이고 활자체의 종류 및 크기에 제한을 두지 않는 실시간 문자 인식 시스템은 개발되지 않은 실정이다. 본 연구는 음성 신호 처리에 매우 효율적인 것으로 입증된 Hidden Markov Model (HMM)^[3]을 이용한 필기체 한글 인식에 그 초점을 두고자 한다.

최근에 HMM은 음성 인식에 많이 이용되어 왔으며 다른 방법보다 동적인 데이터를 모델링하기에 적합하기 때문에 1980년대에 많은 연구가 이루어졌다. 이와 같은 장점때문에 근래에는 문자 인식 분야에서도 HMM을 이용하여 필기체 영문 단어를 인식하고자 하는 연구가 진행되었다.^[4] 필기체 영문 단어에 HMM을 적용한 연구 결과를 보면 먼저 단어 단위로 훈련 모델을 구성하고 인식하였는데 이 때는 모든 단어에 대한 학습이 선행되어야 하고, 그 단어들의 데이터베이스를 가지고 있어야 하는 단점이 있다.^[5] 또한 다중 활자체 영·숫자^[6] 및 한글^[7] 인식에 HMM을 이용한 연구 결과가 있다. 이밖에도 필기체 한글의 오프라인 인식에^[8] HMM을 적용한 결과가 있는데, 이는 한글을 문자 단위로 정합하는 방법이며, 문자의 수평, 수직 및 대각선 방향 기준에 대한 투영값의 윤곽선을 추적하여 관측열과 심벌 수를 구하였다. 최근에 온라인 필기체 한글 인식에 HMM을 신경회로망과 결합하여 적용한 연구^[9]도 있다.

다중 폰트 인쇄체 및 필기체 한글의 경우, 그 변화 특성이 음성신호의 stochastic한 변화 특성과 유사하다. 따라서 본 논문에서는 HMM의 오프라인 필기체 한글 및 영·숫자 인식에의 적용 타당성과 그 효율성을 고찰하고자 한다. 해상도가 높은 문서들을 인식할 수 있는 상용화된 인식 시스템이 많이 나와 있지만 다수의 문자 폰트들을 완전히 인식할 수 있는 인식 시스템은 그리 많지 않다. 인식속도와 정확성이 필수적인 인식 시스템에 있어서, HMM은 그 모델의 선형 특성 때문에 학습과 인식이 빠른 장점을 가지고 있다.

본 논문의 구성은 Ⅱ장에서 HMM의 이론적 배경을 설명하고, Ⅲ장에서 필기체 문자인식을 위한 특징 추출을, Ⅳ장에서는 HMM을 이용한 제한한 필기체 한글 및 영·숫자 인식기법을, Ⅴ장에서는 필기체 한

글 및 영·숫자 인식 실험 결과를, 마지막으로 Ⅵ장에서 결론 및 앞으로의 과제에 대해 서술한다.

Ⅱ. HMM의 이론적 배경

HMM은 신호를 doubly stochastic process로 나타낼 수 있는 stochastic 모델이다. 즉 관측열이라 불리는 특징의 연속열 $O_1, O_2, O_3, \dots, O_T$ 와 감추어진 (hidden) 부분인 상태열 s_1, s_2, \dots, s_T 로 나타낼 수 있다. HMM은 천이 (transition)에 의해 서로 연결된 상태들의 모임으로서 각 천이에는 2가지의 확률이 관련되어 있다. 하나는 현재의 천이가 이루어질 천이 확률이고, 다른 하나는 천이가 이루어졌을 때 유한개의 관측 대상으로부터 각 출력 심벌을 내는 조건부 확률을 규정하는 출력 확률이다. HMM을 나타내기 위해서는 먼저 다음과 같은 기호들을 정의한다.

상태수 : N

관측 심벌수 : M

상태 집합 : $Q = \{q_1, q_2, \dots, q_N\}$

심벌 집합 : $V = \{v_1, v_2, \dots, v_M\}$

관측열의 길이 : $t = 1, 2, \dots, T$

t번째 관측 심벌열이 상태 q_i 에 있고, t+1 번째 관측 심벌열이 상태 q_j 를 선택할 확률은

$$A = \{a_{ij}\}, a_{ij} = \Pr(q_j \text{ at } t+1 | q_i \text{ at } t)$$

로 주어지며 t번째 관측 심벌열이 q_i 상태에서 심벌 v_k 를 선택할 확률은

$$B = \{b_i(k)\}, b_i(k) = \Pr(v_k \text{ at } t | q_i \text{ at } t)$$

이다. 초기 상태에서 상태 q_i 를 선택할 확률은

$$\pi = \{\pi_i\}, \pi_i = \Pr(q_i \text{ at } t = 1)$$

이며, 관측열은 $O = O_1, O_2, \dots, O_T$ 로 주어진다. 이상의 정의를 이용하면 HMM은 (A, B, π) 로 표시할 수 있다.

1. HMM의 형태

HMM은 그림 1에서와 같이 세가지 형태가 있다. 그림 1(a)의 ergodic 모델은 어떤 상태에서 다른 어떤 상태로도 천이가 가능한 모델이며, 그림 1(b)의 left-to-right 모델은 단지 낮은 상태 번호에서 높은 상태 번호로의 상태 천이만 가능한 모델로 음성과 같이 시간에 따라 변화하는 신호를 모델화하는데 적합한 모델로 알려져 있다. 그림 1(c)의 parallel 모델은 그림 1(b)와 같이 좌에서 우로 상태가 천이되나 몇개의 다른 경로를 가질 수 있어, 몇개의 left-to-right 모델을 합친 효과를 기대할 수 있다. 본 연구에서는 계산량이 적고 음성인식에서 널리 쓰인 모델인 left-to-right 모델과 ergodic 모델을 사용하였다.

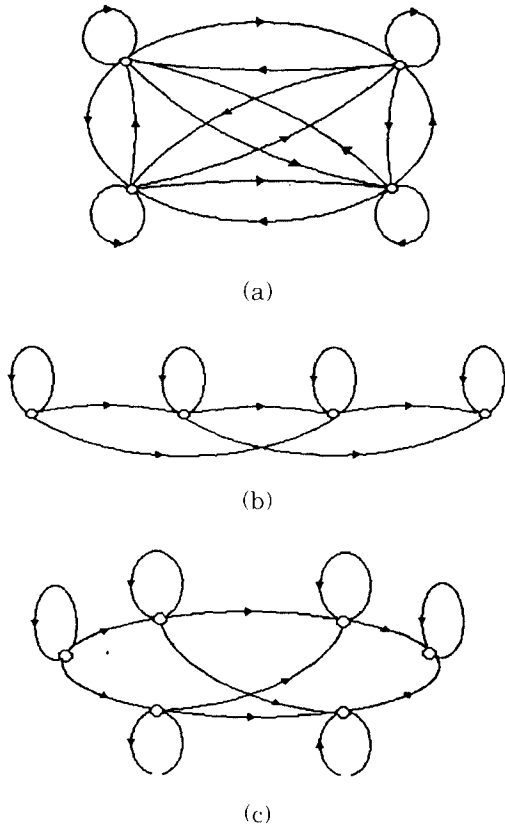


그림 1. HMM의 세가지 형태
 (a) 4 상태 엘고딕 모델 (b) 4 상태 left-right 모델 (c) 6 상태 Parallel path left-right 모델

Fig. 1. Three types of HMM.
 (a) 4-state ergodic model (b) 4-state left-right model (c) 6-state parallel path left-right model.

2. HMM 모델의 학습

모델 파라메타 $\lambda = (A, B, \pi)$ 가 주어졌을 때 관측열 $O = O_1, O_2, \dots, O_T$ 의 확률, 즉 $P(O | \lambda)$ 를 계산하는 방법으로서 효율적인 forward-backward 알고리즘이 Baum에 의해 제안되었다.

^[11] Forward-backward 알고리즘에 대해서 간단히 살펴보면 다음과 같다. 먼저 forward-backward 알고리즘에서 forward variable $\alpha_t(i)$ 를

$$\alpha_t(i) = \Pr(O_1, O_2, \dots, O_t, q_t = q_i | \lambda)$$

로 정의한다. 이는 주어진 모델 λ 에 대해서 시간 t 에서 관측열이 O_1, O_2, \dots, O_t 이고 상태가 q_i 일 확률이며 다음과 같은 절차에 의해서 구할 수 있다. 초기 확

률 $\alpha_1(i)$ 는 상태 q_i 에서 관측심볼이 나타날 확률이다. 전진확률 $\alpha_t(j)$ 는 시간 t 일 때 현재상태가 q_j 일 확률이다. 따라서 모든 관측열이 나타날 시간 $\alpha_t = T$ 일 때 상태 q_i 에 있을 확률은 $T(i)$ 이고, 전체확률은 시간 $t = T$ 일 때 각 상태에 있을 확률의 합이므로 식 (3)이 된다. 그림 2는 전진 및 후진 확률 변수를 구하는 과정을 나타낸다.

단계 1. 초기화

$$\alpha_1(i) = \pi_i b_i(O_1), 1 \leq i \leq N \tag{1}$$

단계 2. $t = 1, 2, \dots, T-1$ 에 대해 반복 ($1 \leq i \leq N$)

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}) \tag{2}$$

단계 3.

$$\Pr(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \tag{3}$$

같은 방법으로 backward variable $\beta_t(i)$ 는

$$\beta_t(i) = \Pr(O_{t+1}, O_{t+2}, \dots, O_{T-1}, O_T | q_t = q_i, \lambda)$$

로 정의할 경우 다음 절차에 의해 구할 수 있다. 즉 초기확률 $\beta_T(i)$ 는 시간 $t = T$ 일 때 상태 q_i 에서 관측심볼이 나타났을 때의 확률이며, 후진 확률 $\beta_t(i)$ 는 시간 t 에서 상태 q_i 를 출발하여 관측열이 모두 나타날 확률이다.

단계 1. 초기화 $\beta_T(i) = 1, 1 \leq i \leq N$ (4)

단계 2. $t = T-1, T-2, \dots, 1$ 에 대해 반복 ($1 \leq i \leq N$)

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \tag{5}$$

이상과 같이 서술한 forward-backward 알고리즘을 이용하여 $\Pr(O|\lambda)$ 을 계산하기 위한 계산량을 대폭 감소시킬 수 있다. 즉 $\Pr(O|\lambda)$ 를 직접 계산하기 위해서는 $2TN^T$ 의 계산량이 필요하다. forward-backward 알고리즘을 이용하면 N^2T 의 계산량만이 필요하다. 예를들어 $N = 5, T = 100$ 일 때 직접 계산에 의한 계산량은 10^{72} 이나, forward-backward 알고리즘을 이용하면 약 3000의 계산량만이 필요하므로 10^{69} 정도의 계산량을 감축할 수 있다. 초기 파라메타들로부터 $\Pr(O|\lambda)$ 를 최대화 하는 $\lambda = (A, B, \pi)$ 를 Baum-Welch 재추정 (reestimation) 알고리즘을 이용하여 구할 수 있다. 여기서, 현재 모델을 $\lambda = (A, B, \pi)$ 라 하고 다시 추정된 모델을 $\lambda' = (A', B', \pi')$ 라 하면 새로운 모델 파라메타는 아래 식들에 의해 구해진다. 여기서 a'_{ij} 와 $b'_j(k)$ 는 각각

재추정된 천이 확률과 관측 심벌 확률이다.

$$\pi_i' = \gamma_i(i), \quad 1 \leq i \leq N \quad (6)$$

여기서, $\gamma_i(i)$ 는 시간 $t = 1$ 일 때 상태 q_i 에 있을 확률이다.

$$a_{ij}' = \frac{\sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(0_{t+1}) \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i) \beta_t(i)} \quad (7)$$

$$b_j'(k) = \frac{\sum_{t=1, 0 \leq t \leq V_k}^T \alpha_t(j) \beta_t(j)}{\sum_{t=1}^T \alpha_t(j) \beta_t(j)} \quad (8)$$

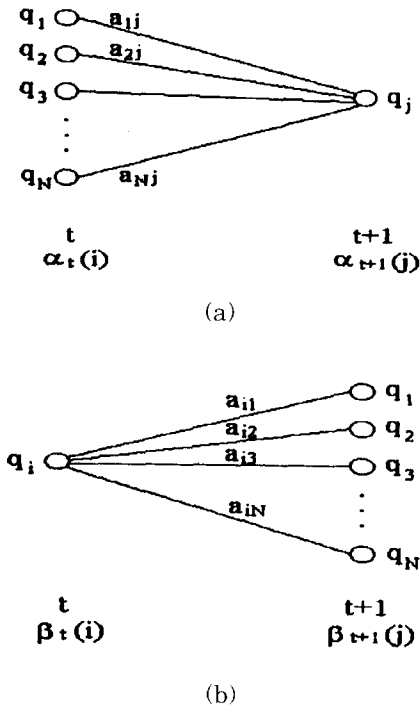


그림 2. 전·후진 확률 변수
(a) 전진 확률 변수 $\alpha_{t+1}(j)$
(b) 후진 확률 변수 $\beta_t(i)$

Fig. 2. Forward and backward variables.
(a) Forward variable $\alpha_{t+1}(j)$
(b) Backward variable $\beta_t(i)$

3. 인식 알고리즘

인식 알고리즘으로는 forward 알고리즘, back-

ward 알고리즘 혹은 Viterbi 알고리즘 등을 사용할 수 있다. 본 실험에서는 left-to-right 모델에 적합한 Viterbi 알고리즘을 사용하였다.^[12] Forward 알고리즘이나 backward 알고리즘이 모든 노드들 사이의 모든 경로에 대한 확률값을 전부 계산하는 반면에 Viterbi 알고리즘은 가장 확률이 높은 하나의 경로를 찾기 때문에 계산량이 감축되며, 자기자신의 상태와 상위의 상태로만 천이가 가능한 left-to-right 모델에 적합한 알고리즘이다. 즉 어느 시점에서부터 임의의 노드에 도달하는 확률은 직전 노드에서의 확률에다 거기서 다시 현재의 노드로 천이할 때의 추가 확률을 합하는 방식으로 확률을 계산하고, 이런 계산을 반복적으로 함으로써 어떤 시점의 현재 노드까지 오는 경로중 최적인 것 하나만을 기억한다.

III. 필기체 문자인식을 위한 특징 추출

특징 추출 단계에서는 전처리 과정에서 넘겨 받은 개별문자로부터 인식에 필요한 문자의 특징을 추출한다.^[13] 기존에 많이 사용된 방법으로는 입력 문자에 대하여 획을 한 화소의 굵기로 나타내는 세션화 과정을 거친 후 문자의 마디점을 특징점으로 하여 표준 모델의 마디점과의 유사성을 따져봄으로써 문자를 인식하는 방법이다. 이 방법은 인쇄체 문자뿐만 아니라 필기체 문자 인식에 많이 사용되는 방법인데, 다음과 같은 문제점을 가지고 있다. 두꺼운 획을 가진 문자를 처리하는데 있어서 마디점 추출이 어렵기 때문에 이 방법은 특히 해상도가 나쁜 문서인식에 적합하지 않다. 다음으로 많이 쓰이는 방법으로 문자 자체에 기준을 두지 않고 배경에 중점을 두어 문자의 오목, 볼록, 그리고 폐곡선을 이용하는 방법이 있는데 이것도 좋은 방법이 될 수 없다. 왜냐하면 이 방법을 쓸 경우 문자에 약간의 잡음이 있어도 특징점을 구하는데 오류가 발생하며, 특히 폐곡선이 뭉그러질 때는 인식할 수 없는 결정적 요인이 있다.

본 논문에서는 빠른 인식속도와 높은 인식률을 얻기 위하여 한글의 경우 초성, 중성, 종성 영역별로, 영·숫자의 경우 전체 문자 영역에서 다음과 같은 특징을 추출하였다.^[14, 15]

1. 문자의 부분 수직투영값과 수평투영값

전처리 단계에서는 문자가 차지하는 영역 전체에 대해 투영한 결과를 이용했는데, 특징 추출 단계에서는 각 영역에 대해서 윗부분, 중간, 아랫부분에서의 수평투영값과 오른쪽, 왼쪽 끝부분 및 중간 부분에서의 수직 투영값을 이용하였다. 그림 3(a)는 문자의

부분 수직투영값을 보여준다.

여러 부분에 대해 부분투영값을 이용하면 전체투영값을 사용하는 경우보다 특징값을 구하는데 있어, 많은 시간 단축을 얻을 수 있으며 또한 인식에 있어서도 전체투영값을 이용하는 것보다 자소의 왜곡이나 크기 변화에 무관하게 특징을 추출할 수 있다. 이 특징은 흑백화소 변화수와 함께 자음뿐만 아니라 모음을 분류하는데 중요한 역할을 한다. 이와 같은 부분투영 방법에 의한 특징 추출은 자소 영역을 x축과 y축 등의 직선상에 각 축에 평행한 mesh열의 흑점의 개수를 세어 구할 수 있으므로 하드웨어 구현이 간단하다.

2. 흑백화소의 변화수

전처리 과정에서 넘겨 받은 문자의 자음과 모음 영역에 대해 그림 3(b)와 같은 수평방향에서의 흑백화소의 변화수를, 영,숫자의 경우 수직방향에서의 흑백화소 변화수를 특징으로 이용하였다. 그림 3(b)에서 수평 번호 1, 2, 3, 4는 초성의 아래 행에서 수평으로 탐색하면서 화소값이 변하는 지점을 표시하였다. 흑백화소의 변화수에 의한 대표적인 자음 분류 예를 보이면 다음과 같다.

아랫부분 수평 행의 변화수 6 이상 : ㅕ, ㅗ, ㅛ, ㅜ, ㅠ, ㅡ, ㅝ, ㅞ, ㅝ, ㅞ, ㅝ, ㅞ 등

아랫부분 수평 행의 변화수 4 : ㅛ, ㅜ, ㅠ, ㅡ, ㅝ, ㅞ, ㅝ, ㅞ, ㅝ, ㅞ 등

아랫부분 수평 행의 변화수 2 : ㄱ, ㄴ, ㄷ, ㄹ, ㅋ, ㆁ 등

수직방향 흑백화소의 변화수에 의한 영,숫자의 분류 예는 다음과 같다.

중간 부분 수직 열의 변화수 6 이상 : B E G Q S Z a e g s e 2 3 5 6 8 9 &

중간 부분 수직 열의 변화수 4 이상 : A C D F J K O P Q R b c d f k o q p t y 4 7 0 # () []

중간 부분 수직 열의 변화수 2 이상 : H I J L M N T U V W X Y f h i j l m n r t u v w x y 1 7 . .

3. 흑화소까지 거리

문자의 흑화소까지 거리는 위의 두 특징값과 같이 입력 문자들을 유사한 부류로 분류할 수 있는 특징은 아니지만 각 부류 내에서 문자를 최종인식하는데 중요한 특징이다. 그림 3(c)와 같이 중성의 경우, 중성 영역의 왼쪽끝에서 중성이 처음 나타나는 흑화소까지의 거리를 특징으로 삼았다. 그런데 이 특징은 화소의 수를 기준으로 삼는다면 같은 문자라 하더라도 문

자의 크기에 따라 거리가 달라지므로 크기에 따른 거리의 변화를 최소화하기 위해 각 문자의 가로, 세로의 길이를 넷 또는 다섯 등분으로 나누어 그 값과 흑화소까지의 거리와의 비를 특징으로 삼았다. 문자의 여러 부분이 파손된 경우, 이 특징이 약간의 오류를 범하는 경우도 나타나므로 어떤 한 부분의 거리뿐만 아니라 주위에서의 거리도 구한 후 비교하여 오류를 줄이도록 하였다.

이 특징을 사용함으로써 다른 특징으로 오인식되는 자소도 인식 가능하며, 입력 데이터의 해상도가 낮아 각각의 문자내에서 흑화소가 붙어버리는 경우에도 효율적으로 문자를 구별할 수 있다.

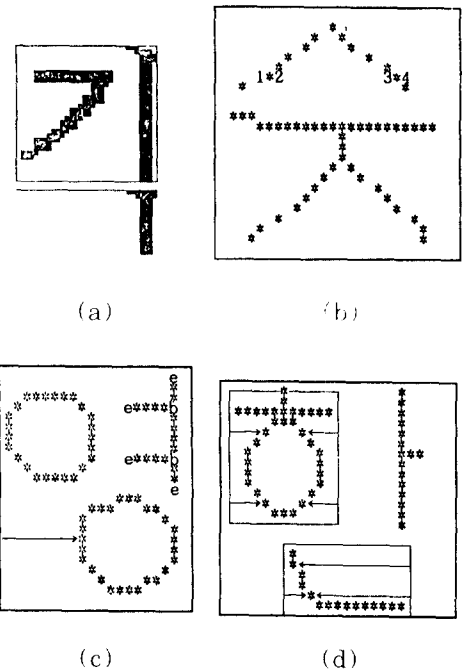


그림 3. 인식에 이용한 특징

(a) 부분 수직 투영값 (b) 수평 흑백 화소 변화수 (c) 흑화소까지 거리 (d) 부분 화소 간격

Fig. 3. Features used for recognition.

(a) Partial vertical projection value (b) Number of horizontal crossing points (c) Distance to a black pixel (d) Partial pixel width

4. 세선화 특징

각 자소의 세선화 특징으로서 끝점, 굴곡점, 분기점, 교차점 등을 추출할 수 있다그러나 굴곡점을 찾

는 경우에 잡음으로 인한 오류가 발생할 수 있으며, 이를 위한 알고리즘의 수행에 많은 시간이 필요하다. 본 논문에서는 특징점들 중에서 굴곡점과 교차점을 제외한 끝점과 분기점의 개수 및 위치를 이용하였다. 그림 3(c)에서 "e"는 끝점을 "b"는 분기점을 각각 나타낸다.

5. 부분 화소 간격

자소가 약간 기울어진 경우에 나타나는 오인식을 줄이기 위하여 쓰는 특징으로 사용한 부분 화소 간격의 예를 그림 3(d)에 보였다.

IV. HMM을 이용한 필기체 한글 및 영.숫자 인식

본 장에서는 III장에서 서술한 여러가지 특징을 이용하여 제한을 둔 필기체 한글 및 영.숫자를 인식하기 위한 알고리즘에 대해 서술하였다. 본 연구에서는 HMM을 이용하여여러가지 특징을 학습시키고, 인식하는 방법을 사용하였다.

1. 필기체 한글 인식

한글 인식에 대한 연구는 1969년부터 시작되었다. 초기의 인식방법으로는 index-window 방법을 사용하여 초성, 중성, 종성을 분리하여 인식하였으나, 필기체의 경우 모음 판정시 자음의 일부가 모음으로 판정되는 경우가 있고 초성과 중성이 붙어 있는 경우에는 분리하기가 어렵다. 또 세선화 과정을 거치지 않고 윤곽선 추적을 이용하여 인식한 경우도 있는데 이 경우에는 세선화에 따른 계산시간을 감소시킬 수 있으므로 수행속도가 빠른 반면, 글자체의 변형 등에 민감한 반응을 보인다. 또한 'ㅎ', 'ㅈ'과 같이 몇 개의 획으로 이루어진 기본자소의 경우에는 오류가 생길 수 있으므로 이 방법을 필기체 문자에 적용시키기는 부적합하다. 또 다른 한글 인식방법은 구문론적 방법에 의한 트리 문법으로서 패턴 그래프를 하향(top-down)적으로 순차 인식하는 방법이 있다.^[16] 19]

현재까지 진행되어온 한글인식 방법은 문자인식의 기본단위에 따라 문자단위, 영역단위, 자소단위의 인식으로 나눌 수 있다.^[26] 225] 영문자와 달리 사용문자의 개수가 많고 활자체들끼리의 변화가 많은 한글의 특성 때문에 자소단위의 분할로 인한 오류를 감소시키고자 본 연구에서는 분할을 하지 않고 한글의 각 유형마다 초성, 중성, 종성 영역단위로 앞 장에서 기술한 여러가지 특징을 HMM을 이용하여 학습하고, 인식한 결과를 종합하여 인식 문자로 분류하는 영역 단위 인식방법을 사용하였다. 그림 4에 HMM을 이

용한 필기체 한글 문자인식 시스템을 보였다.

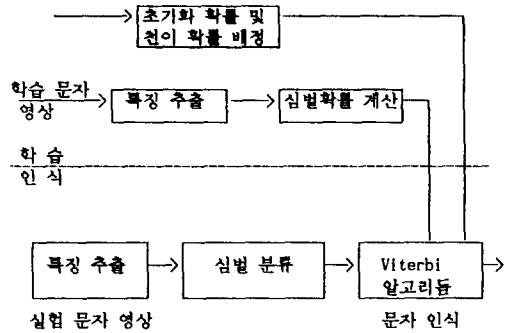


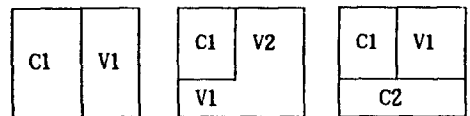
그림 4. HMM을 이용한 필기체 한글 문자인식 시스템

Fig. 4. Handwritten Hangeul recognition system using HMM.

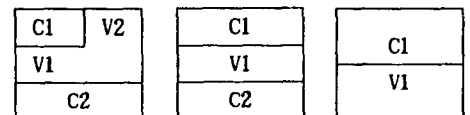
1) 한글의 구조 및 유형

한글의 모음은 수평모음과 수직모음으로 나눌 수 있는데, 수평(수직) 모음은 다시 긴 수평(수직) 모음과 보통 수평(수직) 모음으로 나눌 수 있다. 수평 모음 중에서 긴 수평모음은 크기의 변화는 심하지 않으나 위치의 변화가 있으며, 수직모음은 반대로 위치의 변화는 심하지 않으나 크기의 변화가 종성의 존재 여부에 따라 달라진다.

C1 : Head Consonant
V1, V2 : Vowel Consonant
C2 : Bottom Consonant



(a) 유형 1 (b) 유형 2 (c) 유형 3



(d) 유형 4 (e) 유형 5 (f) 유형 6

그림 5. 한글의 여섯가지 유형

Fig. 5. Six types of Hangeul.

한글의 유형은 모음을 구성하는 수직, 수평획과 자음의 배치에 따라 그림 5처럼 6 가지 유형으로 분류한다. 본 논문에서는 정확도가 높은 유형 분류를 위해 한글의 구조적 특성과 필기체 특성상 다음과 같은 특징을 고려한다.

- ① 긴 수평모음이 존재하고 수직모음이 존재하지 않는 경우는 유형 5 또는 6이라 볼 수 있다.
- ② 보통 수평모음이 존재하면, 유형 2 또는 4로 볼 수 있다.
- ③ 보통 수직모음이 존재하면, 유형 3 또는 4로 볼 수 있다.
- ④ 긴 수직 모음이 존재하면, 유형 1 또는 유형 2라 볼 수 있다. 즉 대부분 중성이 존재하지 않는다.
- ⑤ 각 자소들 중에서 수평모음은 가장 왼쪽에서 시작된다.
- ⑥ 중성이 존재하면, 유형 3, 4, 5로 볼 수 있다.

2) 퍼지 이론을 이용한 유형 분류

필기 변동에 따른 유형분류의 애매함을 극복하고자 퍼지 추론을 이용하여 유형을 분류하였다.^[23] 퍼지 추론에 의한 유형 분류를 위해 퍼지 규칙이 설정되어야 한다. 퍼지 규칙의 형태는 "If (조건) Then (결론) Else If" 로 주어지며, 퍼지 사실에 대한 결과를 추론하기 위해 Mamdani의 min-max 방식을 사용하고^[24], 추론 결과를 최대값 방법에 기초한 비퍼지화를 수행하여 유형분류를 행한다. 명확한 보통 변수를 퍼지 값으로 바꾸기 위해 그림 6과 같이 소속값이 "1"인 보통 변수를 중심으로 한 삼각형 퍼지변수를 사용하여 입력 패턴과 기준패턴과의 유사도를 min-max 방식에 의한 퍼지 함수값으로 나타낸다. 여기서 기준패턴은 3장에서 서술한 특징값의 각 유형에서의 기준값이며 사다리꼴 퍼지변수를 사용하여 퍼지 함수값으로 나타내고, 입력 패턴은 비교 대상의 특징값으로 삼각형 퍼지 변수를 사용하여 퍼지 함수값으로 나타낸다. 표 1에 나타난 것처럼 한글은 구조적 특성상 모음의 형태와 중성과의 관계에 따라 그 유형이 결정된다. 본 논문에서는 아래와 같은 6개의 퍼지규칙을 적용하여 퍼지추론을 하였다.

표 1. 여섯가지 퍼지 규칙에 따른 모음과 중성의 존재 여부

Table 1. Existence of vowel and final consonant by six fuzzy rules.

유형	LVV	LHV	SVV	SHV	FC
1	o	x	x	x	x
2	o	x	x	o	x
3	x	x	o	x	o
4	x	x	o	o	o
5	x	o	x	x	o
6	x	o	x	x	x

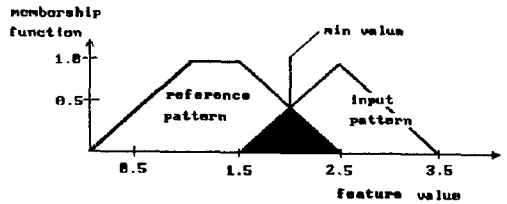


그림 6. 기준 패턴과 입력 패턴의 삼각형 퍼지함수
Fig. 6. Triangular fuzzy function for reference pattern and input patten.

• 퍼지추론을 위한 6개의 퍼지 규칙

- R1 : If VV = LVV and HV = NE and FC = NE Then TC = 유형 1
- R2 : If VV = LVV and HV = SHV and FC = NE Then TC = 유형 2
- R3 : If VV = SVV and HV = NE and FC = E Then TC = 유형 3
- R4 : If VV = SVV and HV = SHV and FC = E Then TC = 유형 4
- R5 : If VV = NE and HV = SHV and FC = E Then TC = 유형 5
- R6 : If VV = NE and HV = LHV and FC = NE Then TC = 유형 6

여기서 사용된 약어는 다음과 같다.

- VV: vertical vowel (수직모음)
- HV: horizontal vowel (수평모음)
- TC: type of character (문자 유형)
- FC: final consonant (중성)
- LVV: long vertical vowel (긴 수직모음)
- SVV: short vertical vowel (짧은 수직모음)
- LHV: long horizontal vowel (긴 수평모음)
- SHV: short horizontal vowel (짧은 수평모음)
- E: existing (존재함) NE: not existing (존재하지 않음)

3) 필기체 한글 인식

유형 분류후, 유형 1의 초성 영역을 인식하기 위한 학습 및 인식 알고리즘을 예를 들어 수식화하여 설명하면 다음과 같다. 먼저 18개의 특징값 (끝점 및 분기점 수 2개, 흑백화소의 변화수 2개, 부분 수직투영값과 수평 투영값 6개, 흑화소까지 거리 4개 및 부분 화소 간격 4개)을 특징열 (관측열)로 하여 앞의 식 (2) - (3), (5)식을 이용하여 전진 확률 $\alpha_{i+1}(j)$ 과 후진 확률 $\beta_i(i)$ 를 구한다. 여기서 T는 관측열의 길이, 즉 18이 되고, a_{ij} 는 천이 확률이며, $b_j(O_{i+1})$ 은 j 상태에서 관측되는 심벌 확률, 즉 0 - 10사이의 11개 관

측 심벌 확률값이다. 전진 확률과 후진 확률을 이용하여 II 장의 (6)-(8)식을 이용하여 각각 A, B 파라미터를 재추정한다. 위에서 재추정된 A, B 파라미터를 이용하여 Viterbi 알고리즘을 수행한 뒤 최적 확률값 p^* 와 상태열 $\psi_t(j)$ 를 구한다. $\delta_t(j)$ 는 시간 t에서 유일한 경로상의 최대 확률값이며 상태열을 구하기 위해서 각각의 t, j에서 (9a)식을 최대화하는 상태열을 (9b)식의 $\psi_t(j)$ 를 이용하여 추적한다. 최적 경로상의 확률값 p^* 와 최종상태 q^* 를 (10a)식과 (10b)식을 이용하여 구할 수 있으며 식 (11)은 최적 상태열을 구하기 위해 역추적하는 과정이다. 가장 확률이 높은 상태열을 찾는 과정은 다음과 같다.

$$\delta_t(j) = \max[\delta_{t-1}(i)a_{ij}b_j(O_t), 2 \leq t \leq T, 1 \leq j \leq N \quad (9a)$$

$$\psi_t(j) = \arg \max[\delta_{t-1}(i)a_{ij}, 2 \leq t \leq T, 1 \leq j \leq N \quad (9b)$$

$$p^* = \max[\delta_T(j)] \quad (10a)$$

$$q_T^* = \arg \max[\delta_T(i)] \quad (10b)$$

$$q_t^* = \psi_{t+1}(q_{t+1}^*) \quad t = T-1, T-2, \dots, 1. \quad (11)$$

위에서 구한 최적 확률값 p^* 가 설정한 값보다 작을 때까지 반복을 계속하여 최종 A, B 파라미터를 구한 뒤 학습을 끝마친다.

2. 필기체 영·숫자 인식

필기체 영·숫자 인식의 첫 단계에서 개별 문자의 몇 개의 부분에서 수직투영비와 수평투영비^[25]를 이용하여 5개의 그룹으로 나눔으로써 후보 문자를 줄이는 방법을 취하는데, 이는 인식에 소요되는 계산량을 줄이기 위함이다. 5개 그룹에 대해 필기체 한글 자소 인식에 이용된 끝점과 분기점의 개수 2개, 수직 및 수평 방향 흑백화소 변화수 2개, 수직 및 수평 방향 투영값 6개, 흑화소까지의 거리 4개, 부분 화소 간격 4개 총 18개의 특징을 학습하였으며, 인식과정에서는 Viterbi 알고리즘을 이용하여 각 부류에서 학습시에 생성된 각 문자의 상태 전이 및 심벌 관측 확률값과 입력 패턴의 특징열 (관측열) 18개에 대한 Viterbi 확률값이 최대가 되는 문자, 즉 $P(O|\lambda)$ 를 최대화하는 문자를 인식 문자로 판정한다.

위에서 서술한 바와 같이 몇 개의 특징을 이용하여 영문자 52개와 숫자 10개 및 특수문자 10개, 총 72개 후보문자를 분류해 보면 다음과 같다. 각 후보문자는 필기시 왜곡으로 인한 오분류를 최소화하기 위해 때로는 중복될 수 있도록 고려하였다.

1) 양쪽 투영비가 큰 것

H M N U I i l O u m n l &

(1) 수평 방향의 흑·백 변화수

한개 : I i l l

두개 : H U O u n

나머지 : M N m &

2) 위·아래 투영비가 큰 문자

C D E I S Z a c s 2 3 5 8

(1) 수직 방향의 흑·백 변화수

한개 : I

두개 : C D c

(2) 수평 방향의 흑·백 변화수

한개 : I

두개 : D S s 8

(3) 나머지

3) 중간 부분 투영비가 큰 문자

A B F G R e f h t 4 6

(1) 수직 방향의 흑·백 변화수

한개 : f t

두개 : A F G f t 4

세개 : B G e 6

(2) 나머지

4) 나머지 문자

D J K L P T b d f g h j k m n p q r 7 9

() []

(1) 왼쪽변의 투영비가 큰 문자

D K L P b h k p r ([

(2) 오른쪽변의 투영비가 큰 문자

J d g j q 7 9)]

(3) 윗쪽의 투영비가 큰 문자

J T f m n

(4) 나머지 문자

5) 크기가 작은 문자

" " . . -

V. 실험 결과 및 분석

본 논문에서는 HMM 모델을 이용하여 한글 및 영·숫자 인식에 대해 실험하였으며, 제한한 특징 양자화 방법이 특징을 그대로 적용한 방법에 비해 우수한 성능을 나타냄을 알 수 있었다. 한글과 영·숫자 인식에 대한 실험결과는 다음과 같다.

1. 필기체 한글 유형 분류

본 논문에서 사용한 실험 데이터는 해상도가 200 dpi (dot per inch)인 출력의 정도가 적은 제한된 한글 필기체 문자를 대상으로 유형 분류를 행하였다. 문자의 유형 분류에 용이하도록 32 × 32 크기의 이진 데이터로 정규화하였다. 또한 실험 데이터는 한글

찾기 순위^[26] 중 누적 빈도가 높은 상위 480자를 대상으로 하였으며, 성인 5명에 의해 쓰여졌다. 필기된 1400여자를 대상으로 실험한 결과 94%의 유형 분류를 얻을 수 있었으며, 대표적인 오분류는 초성이 "표"인 경우에 유형 1, 3에서 수평모음이 존재하는 유형 2, 4로 잘못 분류되었으며, 유형 2의 "의"가 유형 3의 "익"자로 잘못 분류되었다. 그림 7에 실험에 사용된 필기체 한글 데이터 및 유형 분류 결과의 예를 각각 보였다. 그림 7(a)의 입력 문서에 대해서는 유형 분류의 오류가 없었으나, 흘림의 정도가 심한 그림 7(b)의 입력문서 글자 중 "및"자의 유형을 분류하지 못하여 0으로 표시하였다.

외 에 한 과 와 로 들 서 이 관	2 1 3 2 2 6 5 1 1 4
는 리 으 어 가 른 하 데 기 및	5 1 6 1 1 5 1 1 1 3
지 인 그 위 아 나 스 며 저 및	1 3 6 2 1 1 6 1 1 3
사 다 고 구 라 우 자 치 수 미	1 1 6 6 1 6 1 1 6 1

(a)

외 에 한 과 와 로 들 서 이 관	2 1 3 2 2 6 5 1 1 4
는 리 으 어 가 른 하 데 기 및	5 1 6 1 1 5 1 1 1 0
지 인 그 위 아 나 스 며 저 및	1 3 6 2 1 1 6 1 1 3
사 다 고 구 라 우 자 치 수 미	1 1 6 6 1 6 1 1 6 1

(b)

그림 7. 실험에 사용된 필기체 한글 데이터 및 유형 분류 결과

(a) A 입력 문서 (b) B 입력 문서

Fig. 7. Example of Hangul data used in experheir type

(a) Input document A (b) input document B.

2. HMM을 이용한 필기체 한글 인식

각 유형으로 분류된 한글 문자에 대해 초성, 중성, 종성 영역별로 앞에서 기술한 특징을 이용하여 특징값의 개수, 상태 수, 모델, 오차률 등을 변화시키며 실험하였다.

1) 특징값 변화에 따른 인식률 고찰

각 유형의 중성 영역은 유형 3, 4, 5에서 나타나며

각 유형마다 중성영역은 제한되어 있다. 앞에서 서술한 특징 즉 끝점, 분기점 개수, 흑백 화소의 변화수, 자소의 부분 수직, 수평 투영값 자소의 부분 화소 간격 등 18개의 특징을 HMM을 이용하여 학습시키고, 인식실험을 수행하였다. 즉 18개의 관측값과 31개의 심벌 수가 존재하게 된다. 31개의 심벌 수가 존재하는 이유는 자소의 부분 수직, 수평 투영값이 0 - 30의 범위안에 있기 때문이다. 0 - 30의 특징값들을 그 분포에 따라 세개 레벨로 양자화를 수행하였다. 예를 들어 설명하면 끝점의 경우 그 개수가 2개 이하면 0으로, 3개이면 1로, 4개 이상이면 2로 값을 할당한다. 18개의 특징이 있으므로, 먼저 각 특징의 분포를 균일하게 양자화하면 0, 1, 2 세개의 심벌과 18개의 관측값이 존재하게 된다. 다음에 각 특징값을 전부 추출한 뒤, 그 분포에 따라 특징값의 밀도가 조밀한 부분은 세밀하게 0, 5, 10 세개의 심벌로 비균일하게 양자화하여 11개의 심벌수와 18개의 관측값을 갖도록 한뒤 학습을 시키고 인식률을 고찰하였다. 11개의 심벌은 0 - 10 사이의 값이며, 18개의 관측값은 끝점 및 분기점의 수, 수평 및 수직 흑백화소 변화수, 자소의 수평 및 수직 부분 투영값, 부분 화소 간격 등이다. 표 2는 양자화 레벨 크기를 변화시킨 경우의 인식률 변화를 보여주고 있다. 끝점, 분기점, 흑백화소 변화수, 학습에 사용된 중성은 각 유형별로 10개, 총 250개를 학습시키고 200여개를 학습하지 않은 테스트 패턴으로 사용하였다.

표 2의 실험 결과와 같이 특징의 분포에 의거한 제안한 특징 양자화 방법이 특징값을 그대로 학습시킨 경우보다 자소의 왜곡이나 잡음 등으로 인한 특징값의 변동을 흡수하여 보다 신뢰성 있는 모델을 만들게 하였으며 HMM의 학습시간을 단축시키는 결과를 가져왔다. 또한 0, 1, 2 세가지 레벨보다 0 - 10의 11개 심벌을 만들어 줌으로써, 8개의 가상 심벌이 존재하고, 이 가상 심벌들이 중성 25개 HMM의 심벌 관측 확률을 더욱 신뢰성 있는 확률 분포로 갖게 함을 알 수 있었다. 여기서 중성 27개 중 거의 쓰이지 않는 7, 8, 9, 10은 제외하였다.

표 2. 특징값의 양자화 레벨에 따른 인식률
Table 2. Recognition rate depending on the quantization level.

패턴	기준 이용한 경우	특징값을 그대로 세개 레벨로 양자화	특징값을 0, 5, 10 세개 레벨로 양자화
학습패턴	80%	88%	96%
테스트패턴	70%	80%	90%

심벌 관측 확률 B 행렬에 아래의 식¹²⁷⁾을 적용하여 새로운 심벌 관측 확률 B 을 구성하여 학습된 자소와 학습되지 않은 자소의 인식에 이용하였다.

$$b_j'(k) = n_1 b_j(k) + n_2 b_j(k-1) + n_3 b_j(k+1) \quad (12)$$

여기서 $b_j(k)$ 는 j 상태에서 k번째 심벌 관측확률 $b_j(k)$ 와 k-1 번째 심벌 관측확률 $b_j(k-1)$ 및 k+1 번째 심벌 관측 확률 $b_j(k+1)$ 에 평활화 상수 n_1, n_2, n_3 를 곱하여 생성된 j 상태에서 갱신된 k번째 심벌 관측 확률이다. 여기서 n_1, n_2, n_3 는 실험적으로 각각 0.15, 0.7, 0.15의 값을 할당하였다.

심벌 관측 확률 B 행렬을 평활화하여 얻어진 B 를 사용한 결과 학습된 패턴에 대해서는 인식률의 향상이 없었으나, 학습되지 않은 패턴에 대해서는 3-5%의 인식률이 향상됨을 볼 수 있었다. 이는 테스트 데이터에서 관측열 (특징열)의 변화가 있을 수 있으므로, 이를 평활화함으로써 관측열의 변화를 보상하여 더욱 신뢰성있는 관측확률 분포가 형성됨을 나타내고, 일반적인 실험에서 학습데이터의 부족으로 인하여 학습되지 않은 데이터에 대한 충분히 고려되지 못함을 보완하는 것이다.

2) 상태수에 따른 인식률 고찰

특징을 0, 5, 10 세개의 레벨로 양자화한 경우에 대해서, 모델은 left-to-right 모델로 학습률 증가분은 0.00001로 반복횟수는 300으로 각각 고정시키고, HMM의 상태수를 5에서 15로 변화시키며 인식실험

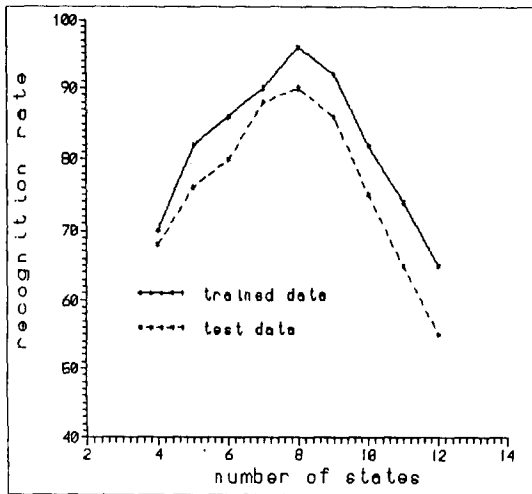


그림 8. 상태수에 따른 인식률
Fig. 8. Recognition rate as a function of the number of states.

을 수행한 결과 8개 상태에서 최대 인식률을 보이고 그 이후의 상태에서는 인식률이 감소함을 관찰할 수 있었다. 이는 HMM이 주어진 특성을 모델링하는데 8개의 상태가 최적임을 나타내는 것이며, 관측 심벌 수보다 적은 상태수에서 최적상태를 나타내는 HMM의 일반적인 특성과 일치한다. 그림 8에 상태수의 변화에 따른 인식률의 변화를 보였다. 그림에서 볼 수 있듯이 인식률은 학습된 데이터 및 학습되지 않은 데이터에 대해 8개 상태에서 최대이다.

3) 모델 변화에 따른 인식률 고찰

특징값을 고정시키고 HMM의 모델을 ergodic 모델과 left-to-right 모델로 변화시키며 인식실험을 수행한 결과 left-to-right 모델의 경우가 ergodic 모델의 경우보다 8개 상태에서 인식률이 3-4% 향상됨을 볼 수 있었다. 이것은 left-to-right 모델이 특징열을 상태에 할당할 때 자기자신의 상태나 그 다음 상태로 천이하고, 그 이전 상태로 천이하지 않기 때문에 특징열에서 특징값이 위치에 따라 약간 변화해도 유일한 모델을 결정할 수 있기 때문이다. 학습률 증가분은 0.00001로 반복횟수는 300으로 각각 고정시켰다. 그림 9는 학습된 데이터의 모델 변화에 따른 인식률 변화를 보였다. 그림에서 볼 수 있듯이 상태수가 증가함에 따라 left-to-right 모델의 경우가 ergodic 모델의 경우보다 인식률 차이가 증가하며, 8개 상태에서 두가지 모델의 인식률이 최대가 되며, 8개 상태 후에는 두가지 모델의 인식률 차이가 거의 없다.

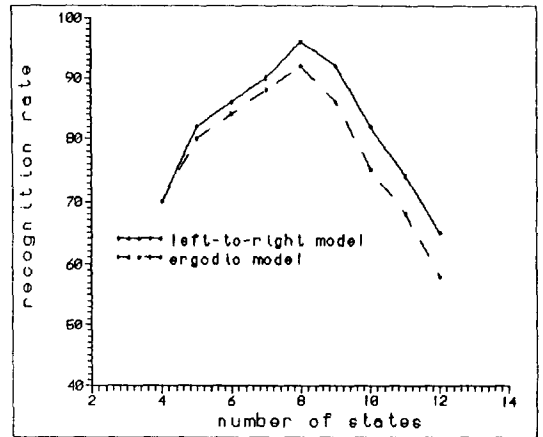


그림 9. 모델 변화에 따른 인식률
Fig. 9. Recognition rate depending on model types.

4) 학습률 증가분 변화에 따른 인식률 고찰
HMM의 학습시에 모델 파라미터를 재추정하기 위

해 Viterbi 확률값을 갱신하였는데 이 학습률 증가분(문턱값)을 0.005 - 0.00001로 변화시키며 인식률을 고찰한 결과 0.00002에서 인식률이 최대가 되었으며, 학습률 증가분을 더 줄인 경우에도 인식률은 최대 상태에서 변하지 않음을 알 수 있었다. 여기서 학습률 증가분은 재추정된 Viterbi 확률값과 이전 확률값과의 오차이다. 이는 주어진 데이터에 대해서 학습률 증가분 0.00002가 HMM의 최대 학습률 증가분임을 나타낸다. 또한 반복 횟수는 모델 파라메타를 재추정하기 위한 최대 허용 반복 횟수이며 350번 이후에는 인식률의 변화가 없다. 그림 10은 학습 반복 횟수 및 학습률 증가분 변화에 따른 인식률 변화를 보여주고 있다. 그림의 세가지 곡선은 학습률의 변화에 따른 인식률의 변화를 보이며, x축은 반복횟수를 보인다.

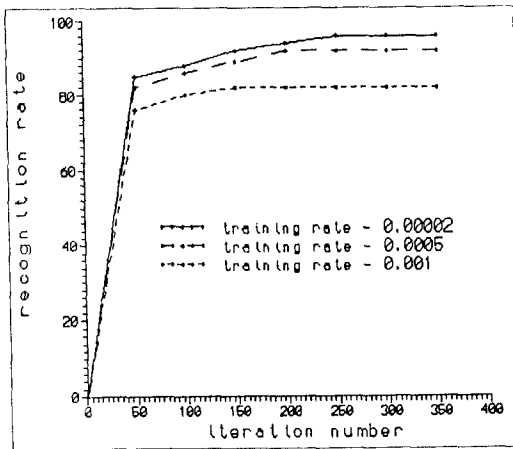


그림 10. 반복 횟수 및 학습률 증가분 변화에 따른 인식률

Fig. 10. Recognition rate as a function of the number of iterations and training rate increment.

5) 각 상태에서의 특징값 분포를 예측하여 초기화 한 경우

천이 확률 및 관측 심벌 확률에 고정된 값을 할당하지 않고 입력 특징에 따라 특징값을 예측하여 초기화한 후 학습시키고 인식률을 고찰한 결과 HMM의 학습시간이 감소되었으며 인식률이 증가함을 알 수 있었다. 초기화는 학습 데이터에 대해서 특징열들의 변화가 크게 일어나는 지점을 상태수에 따라 크러스트링하여 각 심벌의 빈도수를 구한 뒤 이를 초기 심벌 관측 확률값으로 할당하여 학습을 시작하는 과정이다. 결국 초기화가 HMM의 파라메타들을 수렴시키는데 기여함을 알 수 있다. 표 3은 천이 확률 및 관

측 심벌 확률을 초기화한 경우와 하지 않은 경우의 결과를 보여 주고 있다. 마지막으로 표 4는 각각의 유형별로 위에서 서술한 특징을 이용하여 학습된 패턴과 학습되지 않은 테스트 패턴의 인식률을 고찰한 결과를 보여준다.

대표적인 오인식의 경우를 고찰해 보면 자음의 경우 ㄴ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ , ㄹ 과 ㄹ 등이다.

표 3. 천이 및 심벌 관측 확률을 초기화 한 경우와 초기화하지 않은 경우의 인식률

Table 3. Comparison of recognition rate without initializing state transition and symbol observation probabilities.

패턴 기준	초기화한 경우	초기화하지 않은 경우
학습패턴	96%	94%
테스트패턴	90%	88%

표 4. 각 유형별 인식률

Fig. 4. Recognition rate for each type.

		유형 1	유형 2	유형 3	유형 4	유형 5	유형 6
초성	학습패턴	96%	94%	96%	94%	96%	96%
	테스트패턴	92%	88%	90%	88%	90%	92%
중성	학습패턴	98%	96%	98%	94%	98%	98%
	테스트패턴	93%	92%	96%	90%	94%	94%
종성	학습패턴			96%	90%	97%	
	테스트패턴			90%	85%	92%	

6) 특징 분석

각 특징값의 분포를 정규화하여 통계적으로 분석한 결과, 화소 간격 및 거리 특징 분포의 분산값이 가장 크며, 다음에 끝점, 분기점 개수 및 수평, 수직 투영값이고, 수평, 수직 흑백 화소 변화수 특징의 분산값이 가장 작았다. 결과적으로 자소의 패턴을 묘사하는 정규화된 특징값의 기여도는 화소 간격 및 거리, 끝점, 분기점 개수 및 수직, 수평 투영값, 수평, 수직 흑백 화소 변화수의 순서이다. 본 논문의 실험 결과에서 화소 간격 특징을 제외한 14개 특징을 이용한 인식 결과와 화소 간격 특징을 포함한 18개 특징을 이용한 인식결과의 차이는 화소 간격 특징이 구별력이 강한 특징으로서 자소의 패턴을 결정하는 기여도가 크기 때문이다.

7) 기존의 방법과의 비교

단순히 각 자소들의 특징열을 크러스트링^[28] 하여 인식한 결과, 단순히 유클리디언 거리 척도를 이용하

여 크러스트링했기 때문에 특징열의 정보가 중요한 정보와 중요하지 않은 정보의 구분없이 동등하게 이용되어 오인식되는 경우가 많이 발생하여 중성의 경우 80%의 인식률을 얻었다. 구문론적 (syntactic) 방법에 의해 각 자소들의 패턴을 생성 규칙 (production rule)을 이용하여 표현하고 인식한 결과, 필기된 자소의 패턴이 다양하고 특징의 의미있는 수치적 정보를 이용하지 못하기 때문에 인식률이 낮아진다. 각각의 특징열들을 동적 프로그래밍^[29] (dynamic programming)방법으로 정합한 결과 한글 자소의 유사성 때문에 한 자소가 다른 여러 자소들과 혼동을 하였으며, 학습을 하지 못하는 단점 때문에 특징열의 변화를 흡수하지 못하여 88%의 인식률을 얻을 수 있었다. 각 자소들의 특징열을 크러스트링한 결과보다는 향상되었으나, 제안한 특징 양자화 방법보다 높지 못하였다. 또한 신경회로망의 역전파 학습 알고리즘은^[30] 단순한 모델로서 수렴 가능성이 높으나, 학습에 많은 시간이 필요하고 학습에 의해 파라미터가 반드시 최적점으로 간다고 보장할 수 없다. 본 연구에서 추출한 특징들을 역전파 학습 알고리즘을 이용하여 학습, 인식한 결과 특징열의 변화에 민감하지 못한 특성때문에 250개 중성에 대해 92%의 인식률을 얻을 수 있었다. 그러나 특징을 양자화하여 신경회로망을 학습시킨 경우에 오차가 수렴하지 않아 학습이 되지 않는 경우도 발생하였다.



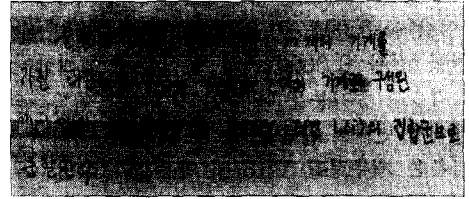
(a)

음역은 인간의 청서를 표현하는 가장 훌륭한 예술중의 하나입니다. 많은 사람에게 최노역의 감정을 느끼게 하고 또 어느 정도의 순화까지 가능하디면

(b)

그림 11. 한글 입력 문서 1 및 인식 결과
 (a) 한글 입력 문서 1
 (b) (a)의 인식 결과

Fig. 11. Korean input document 1 and its recognition result.
 (a) Korean input document 1
 (b) Recognition result of (a).



(a)

분할의 개념은 다음 같다. n 개의 기계를 가진 라인 L을 하나의 버퍼와 2대의 기계로 구성된 n+1개의 기계집합방으로 구성된 라인 L(i)의 집합군으로 분할한다.

(b)

그림 12. 한글 입력 문서 2 및 인식 결과
 (a) 한글 입력 문서 2
 (b) (a)의 인식 결과

Fig. 12. Korean input document 2 and its recognition result.
 (a) Korean input document 2
 (b) Recognition result of (a).

그림 11(a)는 제한을 두어 쓴 한글 입력 문서의 예이며, 그림 11(b)는 이의 인식 결과를 보여준다. 그림 11(b)의 입력 문서 1에 대한 인식 결과를 보면, 맨 밑줄 "정"자의 초성 윗을 가오로 오인식하였으며, 맨 끝자 "면"의 초성 링을 아오로 오인식 하였다. 또한 그림 12(a)는 한글과 영·숫자가 혼합된 문서이다. 그림 12(b)의 인식 결과를 고찰해 보면, 맨 윗줄 "과"자의 유형을 옳게 분류하지 못하였으며, 인식 결과를 ""로 표시하였다. 또한 두번째 줄의 숫자 "2"를 영문자 "z"로 오인식하였으며, 세번째 줄 "느"자의 유형을 분류하지 못하였다.

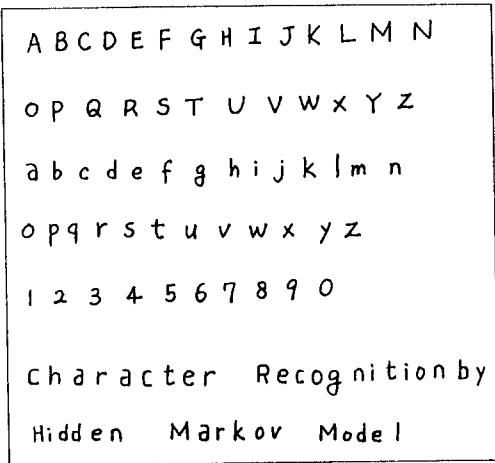
3. HMM을 이용한 필기체 영·숫자 인식

필기체 한글의 경우에 적용했던 특징값을 이용하여 제한된 필기체 영·숫자에 대해서 인식실험을 하였다. IV장 2절에서 기술하였듯이 인식할 모델의 수를 줄이고, 오류를 줄이기 위해 문자들을 양쪽 투영비, 위, 아래 투영비 및 중간부분 투영비에 따라 크게 5개 그룹으로 나누었다. 인쇄체 영·숫자의 경우에 g, p, q, y 등을 한개의 그룹으로 추가할 수 있고, 문자의 크기에 따라 분류할 수 있으나 제한된 필기체 영·숫자의 경우에는 문자의 크기가 인쇄체의 경우에 비해 변할 수 있으므로 투영비에 따라 분류한다.

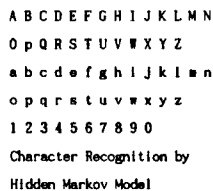
그림 13(a)는 제한을 두어 쓴 문서를 스캐너를 통

해서 취득한 입력문서의 예를 보여주고 있다. 그림 13(b)는 이의 인식 결과를 보여준다. 그림 13(b)의 인식 결과를 보면 제한을 두어 쓴 문서는 잘 인식하였다. 그림 14(a)는 약간 기울어진 문서의 예이다. 그림 14(b)의 인식결과를 고찰해 보면, 영문 소문자 "a"가 영문 소문자 "o"와 구별이 되지 않아 오인식이 일어났다. 이는 두가지 형태의 소문자 "a" 중 한 형태와 소문자 "o"와 유사한 특징값을 갖기 때문인데, 이를 해결하기 위해서는 필기시에 제한을 두어 한가지 형태의 "a"만 쓰도록 하면 해결할 수 있다. 그림 15(a)와 그림16(a)는 각각 다른 필자에 의해 쓰여진

문서이며 그림 15(b)와 그림 16(b)는 각각 그 인식 결과이다. 그림 16(b)의 인식 결과를 보면 역시 영문 소문자 "a"가 영문 소문자 "o"와 구별이 되지 않아 오인식이 일어났다. 그림 16(a)와 그림 17(a)는 다른 필자에 의해 쓰여진 특수 문자와 영 숫자가 혼합된 문서의 예이다. 그림 16(b)와 그림 17(b)는 각각 특수 문자가 혼합된 문서의 인식 결과를 보여준다. 인식 결과를 보면 자주 쓰이는 특수 문자는 잘 인식하였으며, 영문자 소문자 u, v를 혼동하였다. 문서 인식을 위해 문서가 시작되는 처음과 끝 위치 정보 및 글자 사이 간격에 대한 정보를 알고 있어야 하며, 문자 인식 결과의 출력시에 인식된 문자와 함께 출력되어 원래의 입력 문서와 형태가 같아야 한다. 필기체 한글의 경우와 마찬가지로 특징값의 변화에 따라 인식률을 고찰한 결과 예상했던대로 특징값의 수가 18개일 때가 14개의 경우보다 인식률이 향상되었고, 특징값을 양자화하여 학습 인식한 경우가 양자화하지 않은 경우보다 인식률이 향상되었으며, 최적의 상태에서 학습된 데이터에 대해서 제 1 그룹에서 94%, 제 2 그룹에서 98%, 제 3 그룹에서 96%, 제 4 그룹에서 92%, 제 5 그룹에서 98%의 인식률을 얻어 전체적으로 96%의 인식률을 보이며, 학습되지 않은 테스트 패턴에 대해서는 학습된 데이터에 비해 5% 정도 인식률이 감소되었다.



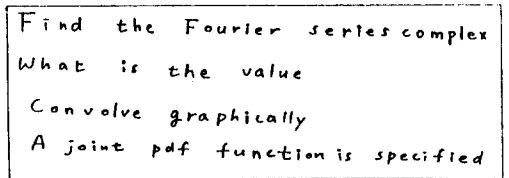
(a)



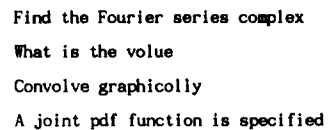
(b)

그림 13. 실험에 사용된 필기체 영·숫자 입력 문서 3 및 인식 결과
(a) 필기체 영·숫자 입력 문서 3
(b) (a)의 인식 결과

Fig. 13. Handwritten alphanumeric input document 3 and its recognition result.
(a) handwritten alphanumeric input document 3
(b) Recognition result of (a).



(a)



(b)

그림 14. 입력 문서 4 및 인식 결과
(a) 입력 문서 4
(b) (a)의 인식 결과

Fig. 14. Input document 4 and its recognition result.
(a) Input document 4
(b) Recognition result of (a).

Hangeul shapes are classified into 6 types by fuzzy inference and their recognition based on quantized features is performed by optimally arranging features

(a)

Hangeul shapes are classified into 6 types by fuzzy inference and their recognition based on quantized features is performed by optimally arranging featvres

(b)

그림 15. 입력 문서 5 및 인식 결과
 (a) 입력 문서 5
 (b) (a)의 이식 결과

Fig. 15. Input document 5 and its recognition result.
 (a) Input document (5)
 (b) Recognition result of (a).

G. Baptista and K.M. Kukani, "A high accuracy algorithm for recognition of handwritten numerals," Pattern Recognition, vol. 21, no. 4, pp. 287-291, Mar. 1988.

(a)

G. Baptista and K. M. kukani, "A high accuracy algorithm for recognition of handwritten numerols," Pattern Recognition, vol. 21, no. 4, PP. 287-291, Mar. 1988.

(b)

그림 16. 입력 문서 6 및 인식 결과
 (a) 입력 문서 6
 (b) (a)의 이식 결과

Fig. 16. Input document 6 and its recognition result.
 (a) Input document (6)
 (b) Recognition result of (a).

In this paper, we present Hidden Markov Model(HMM) and fuzzy set theory for optical text recognition[3,4]. Currently, we achieve the recognition accuracy of 96%.

(a)

In this paper, we present Hidden Markov Model (HMM) and fuzzy set theory for optical text recognition [3, 4]. currently, we achieve the recognition accvcracy of 96 %.

(b)

그림 17. 입력 문서 7 및 인식 결과
 (a) 입력 문서 7
 (b) (a)의 이식 결과
 Fig. 17. Input document 7 and its recognition result.
 (a) Input document (7)
 (b) Recognition result of (a).

VI. 결론 및 앞으로의 과제

본 논문에서는 HMM을 이용하여 제한을 둔 필기체 한글 및 영·숫자 인식 시스템을 구현하였다. 퍼지 추론에 의해 한글을 6가지 형태로 구분하였고, 한글 자소 특징의 분포에 따른 양자화 및 중요한 특징들의 순서화에 의거한 HMM의 구성을 제안하였고, 이의 효율성을 컴퓨터 시뮬레이션을 통해 입증하였다. 또한 HMM의 상태수, 모델의 변화, Viterbi 확률 오차값의 변화 및 각 상태에서 천이 확률과 관측 심벌 확률의 초기화에 따른 인식률을 고찰하였다.

앞으로의 과제는 오인식을 줄이기 위한 보다 효율적인 특징에 대한 연구가 이루어져야 하며, 더욱 신뢰성있는 천이 확률 및 관측 심벌 확률을 구축하는 방법을 모색하여야 할 것이다. 또한 한글, 영숫자 및 그래픽이 혼용된 문서를 각 부류로 분류하여 인식하는 통합 문서 인식 시스템에 HMM을 적용하여 효율성을 향상시키는 연구도 이루어져야 할 것이다.

參 考 文 獻

[1] V. K. Govindan and A. P. Shiv-

- aprasad, "Character recognition- A review," *Pattern Recognition*, vol. 23, no. 7, pp. 671-683, Feb. 1986.
- [2] M. Shiridhar and A. Badreldin, "Recognition of isolated simply connected handwritten numerals," *Pattern Recognition*, vol. 19, no. 1, pp. 1-12, Feb. 1986.
- [3] S. Kahan, T. Pavlidis, and H. S. Baird, "On the recognition of printed characters of any font and size," *IEEE Trans. Pattern Analysis Machine Intell.*, vol. PAMI-9, pp. 274-289, Mar. 1987.
- [4] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257-284, Feb. 1989.
- [5] A. Kundu, Y. Ye, and P. Bahl, "Recognition of handwritten word: First and second order hidden Markov model based approach," *Pattern Recognition*, vol. 22, no. 3, pp. 283-297, June 1989.
- [6] A. Vlontzos and S. Y. Kung, "A hierarchical system for character recognition with stochastic knowledge representation," in *Proc. IEEE Int. Conf. Neural Networks*, pp. 340-345, San Diego, CA, July 1988.
- [7] 김 우성, 함 영국, 박 래홍, "Hidden Markov Model을 이용한 다중 활자체 영·숫자 인식," 제 5회 신호처리합동학술대회 논문집, 제 5권, 제 1호, pp. 299-302, 1992년 10월
- [8] 김 우성, 함 영국, 박 래홍, "Hidden Markov Model을 이용한 인쇄체 한글 인식," 제 5회 영상 처리 및 이해에 관한 워크샵 발표논문집, pp. 178-184, 서울, 1993년 2월
- [9] 이 성환, 박 희선, "Hidden Markov 모델을 이용한 필기체 한글의 오프라인 인식," 제 2회 인공지능 신경망 및 퍼지시스템 종합학술대회 논문집, pp. 298-309, 서울, 1992년 11월
- [10] 조 성배, 김 진형, "온라인 필기문자의 인식을 위한 은닉 마르코프 모델과 신경망 분류기의 결합," 정보과학회논문지, 제 20권, 제 3호, pp. 328-337, 1993년 3월
- [11] L. E. Baum, "A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains," *Ann. Math. Statist.*, vol. 41, no. 4, pp. 164-171, Apr. 1970.
- [12] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268-270, Mar. 1973.
- [13] S. Mori, K. Yamamoto, and M. Yasuda, "Research on machine recognition of handprinted characters," *IEEE Trans. Pattern Analysis Machine Intell.*, vol. PAMI-6, no. 4, pp. 386-405, July 1984.
- [14] 함 영국, 도 상운, 김 우성, 박 래홍, 이 창범, 김 상중, "저해상도 영·숫자 데이터의 고속인식," 춘계종합발표회 논문집, 한국통신학회, 제 10권, 제 1호, pp. 106-109, 서울, 1991년 5월
- [15] Y. K. Ham, C. B. Lee, W. S. Kim, S. Y. Doh, R.-H. Park, and S. J. Kim, "A simple sequentially designed rule-based alphanumeric recognition algorithm for OCR document processing using a thinning process," in *SPIE Proc. Intelligent Robots and Computer Vision X: Algorithms and Techniques*, vol. 1607, pp. 146-157, Boston, MA, Nov. 1991.
- [16] 이 균하, "한글문자인식에 관한 사례연구," 정보과학회지, 제 9권, 제 1호, pp. 45-53, 1991년 1월
- [17] 고 건, 이 대영, "중첩운곽 형상에 의한 한글 패턴의 정점검출," 한국통신학회 논문지, 제 15권, 제 2호, pp. 112-123, 1990년 2월
- [18] T. Agui, M. Nakajima, T. K. Kim, and E. T. Takahashi, "A method of recognition and representation of Korean characters by tree grammars," *IEEE Trans. Pattern Analysis Machine Intell.*, vol. PAMI-1, no. 1, pp. 245-250, July 1979.
- [19] 이 주근, 남궁 재찬, 김 영건, "한글 pattern에서 subpattern 분리와 인식에 관한 연구,"

- 전자공학회논문지, 제 18권, 제 3호, pp. 1-8, 1981년 6월
- [20] 최 병욱, 市川忠男, 蔡田廣一, "한글 인식에 있어서의 자소 추출," 전자공학회 논문지, 제 18권 제 2호, pp. 36-43, 1981년 4월
- [21] 이 회건, 방 승양, "Backpropagation 신경 회로망을 사용한 한글 필기체 오프라인 문자 인식," 한국정보과학회 춘계 학술발표회 논문집, 제 19권 제 1호, pp. 681-684, 포항, 1992년 4월
- [22] 하 진형, 김 진형, "학습 기능을 이용한 필기 한글의 인식에 관한 연구," 인공지능연구회 춘계 학술발표회 논문집, 제 4 권 제 1호, pp. 3-23, 서울, 1989년 4월
- [23] 김 석중, 김 기두, "오프라인 필기체 한글 문자 인식을 위한 유형 분류에 관한 연구," 대한 전자공학회 회로 및 시스템 연구회 춘계 학술 발표대회 논문집, 제 5권 제 2호, pp. 19-22, 대전, 1992년 9월
- [24] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," *IEEE Trans. Computers*, vol. C-26, no. 12, pp. 1182-1191, Dec. 1991.
- [25] 함 영국, 도 상운, 김 우성, 박 래홍, "퍼지이론을 이용한 필기체 영·숫자 인식," 1991년도 제 4회 신호처리 합동학술대회 논문집, 제 4권 제 1호, pp.127-130, 서울, 1991년 9월
- [26] 한글 기계화 연구소, 한글 기계화 연구, 1975
- [27] B. S. Jeng, C. H. Shih, and S. W. Sun, "On the use of discrete Markov process for chinese character recognition," in *SPIE Proc. Visual Communication and Image Process.*, vol. 1360, pp. 1663-1668, Urbana Champaign, IL, Mar. 1990.
- [28] Y. L. Linde, A. B. Buzo, R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Communications*, vol. COM-28, no. 1, pp. 62-74, Jan. 1980.
- [29] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, no. 2, pp. 43-49, Mar. 1978.
- [30] R. O. Lippmann, "An introduction to computing with neural networks," *IEEE ASSP Magazine*, vol. 3, no. 2, pp. 4-22, Apr. 1987.

 著 者 紹 介

金 禹 成(正會員) 第 20卷 B編 第 10號 參照
 1993年 서강대학교 전자공학과 공
 학박사 학위취득, 현재 호서대학교
 컴퓨터공학과 부교수

朴 來 弘(正會員) 第 23卷 第 6號 參照
 현재 서강대학교 전자공학과 교수