

두자 접촉 숫자열의 분할 자유 인식

(Segmentation-free Recognition of Touching Numeral Pairs)

최순만[†] 오일석^{**}

(Soon-Man Choi) (Il-Seok Oh)

요약 숫자열 인식은 문서 처리 자동화에서 매우 중요하다. 기존 방법들은 숫자열을 낱자 단위로 분할하는 단계와 분할된 숫자들을 인식하는 두 단계로 이루어져 있다. 그러나 이들 방법으로는 접촉 유형의 수많은 변형 때문에 만족할 만한 결과를 얻을 수 없다. 본 논문은 두자 접촉 숫자열의 분할-자유 인식 방법을 제안한다. 이 접근 방법에서는 두자 접촉 숫자열을 하나의 패턴으로 간주하여, 총 100개('00', '01', '02', ..., '98', '99') 부류를 대상으로 인식한다. NIST 데이터베이스의 숫자열 필드에서 두자 접촉한 숫자열을 추출하여 실험하였다. 부류수가 방대한 경우 나타나는 기존 신경망 인식기의 한계 때문에, 모듈러 신경망을 사용하였으며 인식 실험을 통하여 우수성을 입증하였다.

Abstract Recognition of numeral fields is a very important task for many document automation applications. Conventional methods are based on the two-steps process: segmentation of touching numerals and recognition of the individual numerals. However, due to a large variation of touching types this approach has not produced a robust result. In this paper, we present a new segmentation-free method for recognizing the two touching numerals. In this approach, two touching numerals are regarded as a single pattern coming from 100 classes ('00', '01', '02', ..., '98', '99'). For the test set, we manually extract two touching numerals from the data set of NIST numeral fields. Due to the limitation of conventional neural network in case of large-set classification, we use a modular neural network and prove its superiority through recognition experiments.

1. 서론

숫자는 일반적으로 금융 기관의 전표나 관공서의 각종 민원 서류, 일상 생활에서 사용되는 양식 등에서 많이 사용되고 있다. 때문에 문서 처리 자동화에서 다른 문자에 비해 그 중요도가 상대적으로 높다고 할 수 있다. 그래서 필기 숫자에 대한 인식 연구가 활발히 진행되어 상당한 인식률과 신뢰도를 갖는 낱자 숫자 인식기들이 발표되었다 [1, 2, 3, 4].

최근 들어서는 낱자 단위 인식과 더불어 연속 필드에 대한 인식 필요성이 크게 대두되고 있는데, 대부분 문

서는 낱자 단위가 아니라 연속 필드에 숫자를 기입하도록 설계되어 있기 때문에 연속 숫자 필드를 인식하는 기술이 필수적이다. 일상 생활에서 흔히 사용하고 있는 많은 양식에는 숫자를 낱자 단위로 기입하는 형태가 아닌 하나의 긴 박스나 밑줄 위에 연속적으로 필기하는 형태가 대부분이다.

대부분 연속 숫자열 인식 알고리즘은 연속 필드를 적당한 단위로 분할한 후 각각을 인식하는 방법을 취한다. 분할 결과로 나온 숫자들은 낱자, 두자 접촉열, 세자 접촉열 등이 있을 수 있다. 때문에 연속 숫자열 인식기를 구축하기 위해서는 분할된 숫자열 각각이 몇 자를 포함하는지 판단하는 모듈, 낱자 인식기, 접촉 숫자열 인식기 등이 필요하다. 본 연구는 이중 두자 접촉 숫자열을 인식하는 문제로 국한되어 있다. 실제로 두자 접촉 숫자열이 많이 발생하므로 이에 대한 연구가 80년대 중반부터 꾸준히 수행되어 왔다.

기존 방법은 두자 접촉 숫자열을 두 개의 낱자로 분할하고 각각을 인식하는 형태를 취하고 있으므로, 대부

· 이 논문은 1999년도 정보통신부 대학기초과제 지원에 의한 연구 결과임

† 학생회원 : 전북대학교 전산통계학과
smchoi@cs.chonbuk.ac.kr

** 종신회원 : 전북대학교 컴퓨터과학과 교수
영상·정보 신기술연구소
isoh@moak.chonbuk.ac.kr

논문접수 : 1998년 10월 12일

심사완료 : 2000년 2월 24일

본 논문이 두 개 낱자로 분할하는 문제에 초점이 맞추어져 있다. Shridhar는 아래 외곽선에서 시작점을 잡아 위로 진행하며 검은점을 만나면 주위 상황에 따라 왼쪽 또는 오른쪽으로 꺾어가며 분할 경로를 찾는 방법을 사용하였다 [5]. 이 알고리즘은 패턴의 아래 부분의 모양만을 사용하므로 근본적으로 한계를 안고 있다. Matan 등은 수직 방향 투영에서 분할 후보 지점을 선택한 후 인식기의 인식 결과에 따라 최종 분할 지점을 결정하는 분할기-인식기 협조 방법을 사용하였다 [6]. 인식과 분할을 동시에 수행하는 순환 신경망 구조도 제안되어 있다 [7]. 이 방법은 숫자열의 공간 정보뿐만 아니라 화소들 간의 공간적인 의존성도 학습에 고려하여 분할 및 인식 성능 향상을 모색하였다. Nishida와 Mori는 세션화된 골격선에서 특징점들을 중심으로 선분들로 분할한 후, 이들의 조합을 인식기에 입력하여 인식 결과에 따라 분할하는 방법을 제안하였다 [8]. Fenrich는 여러 분할 방법을 순서적으로 적용해가며 인식 결과에 따라 최적의 분할을 결정하는 방법을 사용하였다 [9]. 패턴의 위 부분과 아랫부분의 특징점 쌍을 분할 후보 경로로 하여 이들을 적당한 점수식으로 정렬하여 최고점을 얻은 경로를 최종 분할 경로로 사용하는 방법도 개발되어 있다 [10, 11, 12]. 이 방법에서는 점수식을 어떻게 만드느냐가 가장 중요한 문제이다.

이러한 방법들은 주로 경험적인 규칙들을 사용하며 나름대로의 장점과 한계를 가지고 있다. 단순한 형태의 숫자열 분리에는 어느 정도 만족할 만한 성능을 보이고 있으나, 두 개의 숫자가 두 곳 이상에서 접촉하거나 겹치는 경우, 또는 깨지거나 입력 영상에 잡음이 섞인 경우, 리가춰(ligature)가 존재하는 경우 등 좀더 복잡한 형태의 숫자열 분할에는 많은 문제점들을 안고 있다.

또 다른 문제점으로는, 숫자열 분할 성공률이 Westall에서 83%[10], Shi에서 78%[11]등으로 낮다는 것이다. 또한 분할이 성공적으로 이루어졌다 하더라도 분할된 낱자에 리가춰같은 여분의 획이 포함되는 경우가 발생하여 이를 낱자 인식기가 인식하기 어렵다는 것이다. 결국 이들 문제로 인해 연속 필드의 최종 인식률(즉, 분할 성공률 \times 숫자 1 인식 성공률 \times 숫자 2 인식 성공률)이 낮아 전체적인 성능을 저하시키고 있다.

최근에 Wang등은 두자 접촉한 숫자열을 분할하지 않고 인식하는 방법을 제안하였다 [13]. 이 방법은 두자 접촉 숫자열을 하나의 패턴으로 간주하여 100개의 부류를 대상으로 인식한다. 이들은 두자 접촉 숫자열 패턴의 중앙 부분은 접촉으로 인해 변화가 심하고 비교적 적은 정보를 갖는다고 판단하여 중앙의 일정 부분을 잘라내

고 좌우의 나머지 일부만으로 인식하는 방법을 사용하였다. 특징으로는 CEDAR(Center of Excellence for Document Analysis and Recognition)에서 개발한 GSC특징 [14], 인식기로는 k-NN을 사용하였다. 100개의 부류간 발생 빈도에서 많은 차이가 있는데 이 논문에서는 이에 대한 고찰이 없다. 또한 '8'과 '1'이 접촉한 경우 등은 왼쪽과 오른쪽 숫자가 차지하는 너비가 크게 다른데, 일률적으로 가운데 영역을 제거하는 것은 문제가 있다. 특히 숫자열이 기울어지게 쓰인 경우는 중요한 정보가 제거되는 문제도 안고 있다.

본 논문에서는 두자 접촉한 숫자열 인식 문제를 다루며, 접근 방법은 분할-자유(segmentation-free) 방법을 사용한다. 이 접근 방법에서는 두자 접촉한 패턴을 낱자 단위로 분할하지 않고 하나의 단일 패턴으로 간주하여 통째로 인식한다. 즉, 훈련과 인식 대상이 되는 부류 수가 100개이며, 각각은 '00', '01', '02', ..., '50', '51', ..., '98', '99' 이다. 이 방법은 분할과정을 거치지 않고 곧바로 인식을 함으로써 분할하기 위한 여러 가지 복잡한 알고리즘과 경험적인 지식들이 필요치 않아 인식기의 구조가 간단하다. 또한 여러 단계를 거치지 않음으로서 인식 시간을 단축시킬 수 있다는 장점이 있다.

두 자로 제한한 이유는, 숫자열 필드에서 나타나는 접촉 길이 빈도 수에서 두자 접촉이 가장 두드러지게 나타나고, 부류수가 우리가 다룰 수 있는 범위 내에 있기 때문이다. 높은 성능을 보장하는 두자 인식 모듈은 낱자 인식 모듈과 더불어 임의의 길이 숫자열 인식기를 구현하는데 중요한 모듈로 사용할 수 있다. 또한 앞에서 언급한 분할-기반 방법과 더불어 다중 인식기(multiple classifier) 기법을 적용하는데 사용할 수 있다.

본 논문은 인식기로서 신경망(neural network)을 사용하였다. 신경망은 패턴의 다양한 변화등을 비교적 잘 흡수하여 일반화(generalization) 능력이 뛰어나다고 알려져 있다 [15]. 또한 신경망은 훈련 시간은 비교적 길지만, 훈련을 마친 후 인식 단계는 매우 빠르게 작동한다. 하지만 기존의 신경망 구조는 대용량 부류 분류에서는 낮은 인식 성능을 보인다. 이 논문은 이러한 사실을 실험을 통하여 보이고, 해결책으로서 모듈러 신경망을 채택하여 사용한다. 모듈러 신경망은 10개 부류를 갖는 숫자에서도 우수한 성능을 보이며 [16], 한국 우편 주소에 나타나는 352종류의 한글 문자와 같은 대용량 분류에서도 높은 성능을 보임이 입증되었다 [17].

본 논문에서 사용한 분할-자유 방법과 기존 분할 방법과의 객관적인 비교는 쉽지 않다. 그 이유는 사용한 데이터베이스, 평가 방법 등이 서로 다르기 때문이다.

Westall은 자체 수집한 데이터베이스를 사용하여 83%의 분할 성공률을 얻었고 [10], Strathy는 USPS 데이터베이스를 사용하여 57.6%(1순위), 80.1%(2순위), 89.0%(3순위)의 분할 성공률을 얻었다 [18]. 최순만은 한국인을 대상으로 수집한 자체 데이터베이스를 사용하여 55%(1순위), 63%(2순위), 70%(3순위)의 분할 성공률을 얻었고 [12], Lu 등은 NIST 데이터베이스에서 추출한 패턴에 대해 79.9%의 분할 성공률을 얻었다 [19]. 이러한 사실을 종합할 때 분할 성공률은 70~80%대이며 무려러 있다할 수 있고, 여기에 두 자를 모두 인식해야 하는 경우 이보다 낮은 인식률에 그칠 것이다.

두자 접촉 숫자열 인식기의 훈련을 위해 고 품질의 훈련 데이터 집합이 매우 중요하다. 하지만 100개 부류에 대한 충분한 양의 데이터 수집은 현실적으로 어려운 문제이다. 또한 발생하는 빈도가 부류간에 심한 편차를 보이므로 부류간 균형 잡힌 훈련 데이터 수집이 매우 어렵다. 때문에 본 논문에서는 적은 수의 데이터 집합에서 인식기 성능 평가를 위하여 보통 사용하는 잭나이프(jackknife) [20] 기법을 사용하였다. 그리고 부류별 불균형과 적은 빈도수를 보이는 부류들로 인한 인식기 훈련의 어려움 점등을 해결하고자 낱자 숫자 패턴을 붙여서 만든 두자 접촉 숫자열 패턴을 훈련 집합에 추가하여 신경망을 훈련하고 인식 성능을 측정하는 실험도 수행하였다. 또한 훈련 샘플 수가 적은 부류에 대한 분석을 통해 샘플 수가 인식률에 미치는 영향을 파악하였다.

2장에서는 두자 접촉 숫자열을 위한 신경망 인식기의 구조와 신경망에 입력되는 특징을 소개한다. 3장에서는 훈련 집합이 가져야할 특성과 수집에 대해 설명한다. 4장에서는 본 논문에서 제안한 인식기의 성능 평가 결과를 보이고, 5장에서 결론 및 향후 연구 방향에 관하여 논의한다.

2. 두자 접촉 숫자열 인식기

본 논문에서는 신경망 인식기를 이용하였다. 그 이유는 두자가 접촉함으로서 나타나는 많은 변형들을 효과적으로 흡수할 수 있고, 신경망은 뛰어난 일반화 능력을 가지기 때문이다. 인식기는 내부 구조에 따라 비 모듈러(nonmodular)와 모듈러(modular)로 나눌 수 있다. 본 논문에서는 고전적 구조를 갖는 비 모듈러 신경망이 본 논문이 다루는 100-부류 분류와 같이 대용량 분류 문제에서 갖는 낮은 성능 문제를 실험을 통해 입증하고, 이를 극복하는 모듈러 신경망을 도입하여 사용한다. 4장에서는 실험을 통하여 인식 성능을 비교한다.

2.1 비 모듈러 인식기

첫 번째 인식기는 전통적으로 사용해온 비 모듈러 신경망으로 그 구조는 그림 1과 같다. 이 인식기는 n -차원 특징 벡터를 받기 위해 n 개의 노드를 갖는 입력층과 100개의 노드를 갖는 출력층으로 이루어져 있다. 은닉층의 노드 개수는 출력층 노드 개수의 두 배인 200으로 설정하였다. 이들 세 층은 완전 연결되어 있으며 훈련은 오류 역전파(error backpropagation) 알고리즘으로 수행한다 [15]. 초기 가중치는 0.0과 0.2 사이의 임의의 수로 설정되며, 훈련 계수는 반복(epoch)이 증가할수록 0.5에서 0.001씩 선형적으로 줄여 나간다. 훈련 데이터는 새로운 반복을 시작하기 전에 임의로 쉰는다.

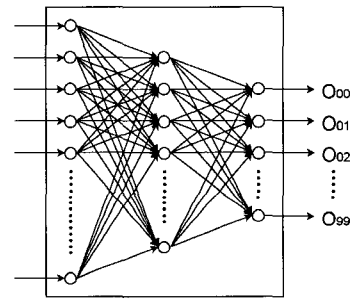


그림 1 비 모듈러 인식기

2.2 모듈러 신경망 인식기

두 번째 인식기로는 모듈러 신경망 구조를 사용하였다 [16, 17]. 모듈러 신경망 구조는 100개의 부망(subnetwork)으로 구성되고, 각 부망은 100개의 부류 중 하나를 책임진다. 부류 w_i 를 위한 부망은 그림 2와 같으며 하나의 입력층, 하나의 은닉층, 하나의 출력층을 갖는다. 이들 세 층은 완전 연결되어 있다. 이 부망의 기능은 두 개 부류 군, $\Omega_1=\{w_i\}$ 과 $\Omega_2=\{w_k \mid 00 \leq k \leq 99, k \neq i\}$ 를 분류하는 것이다. 입력층은 n -차원 특징 벡

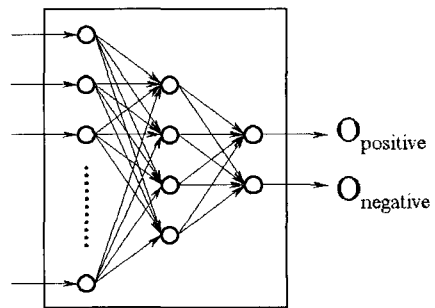


그림 2 모듈러 신경망을 위한 부 망

터를 받기 위해 n개의 노드를 갖는다. 출력층은 Ω_1 과 Ω_2 를 위한 두 개의 노드를 갖고 각각 $O_{positive}$ 와 $O_{negative}$ 라 부른다. 은닉층은 출력층 노드 개수의 두배인 네 개의 노드를 갖게 하였다.

100개의 부 망의 훈련은 각각 독립적으로 이루어지며, 오류 역전파 알고리즘으로 수행한다. 부류 ω_i 를 훈련하기 위해 훈련 집합을 두 개의 군 $Z_{positive}$ 와 $Z_{negative}$ 로 나눈다. $Z_{positive}$ 는 Ω_1 에 속한 샘플, $Z_{negative}$ 는 Ω_2 에 속한 샘플을 갖는다. 예를 들어, ω_i 가 부류 '00' 이라면 $Z_{positive}$ 는 부류 '00'에 속하는 훈련 샘플, $Z_{negative}$ 는 '00'을 제외한 나머지 99개 부류의 모든 훈련 샘플을 갖는다. 즉, 100개 부류가 동일한 개수의 훈련 샘플을 갖는다고 가정했을 때 $Z_{negative}$ 는 $Z_{positive}$ 의 99배의 샘플을 갖는다. 각 부망 입장에서 살펴보면 1:99의 불균형이 있음에도 2.1절의 비 모듈러 신경망에 비해 우수한 인식 성능을 가짐을 4장에서 실험으로 보인다. Anand 등이 [25]에서 제시한 불균형 훈련 집합에서의 성능 향상 기법을 도입하면 인식 성능을 개선할 여지가 있다고 생각된다. $Z_{positive}$ 의 샘플은 기대값으로 ($O_{positive}, O_{negative}$) = (1.0, 0.0)와 함께 입력되며, $Z_{negative}$ 샘플은 ($O_{positive}, O_{negative}$) = (0.0, 1.0)와 함께 입력된다. 초기 가중치는 0.0과 0.2 사이의 임의의 수로 설정되며, 훈련 계수는 반복(epoch)이 증가할수록 0.5에서 0.001씩 선형적으로 줄여 나간다. 훈련 데이터의 순서는 새로운 반복을 시작하기 전에 임의로 섞인다.

부류 ω_i 의 부 망을 M_{ω_i} 라 하면, 전체 인식기의 구조는 그림 3과 같다. 입력 패턴으로부터 특징 벡터를 추출하고, 이를 모든 부 망에 입력한다. 부 망 M_{ω_i} 는 $O_{positive}$ 와 $O_{negative}$ 의 값을 뺀으로써 단일 값을 출력한다. 최종적으로 입력 패턴은 최고 출력값을 갖는 부류로 분

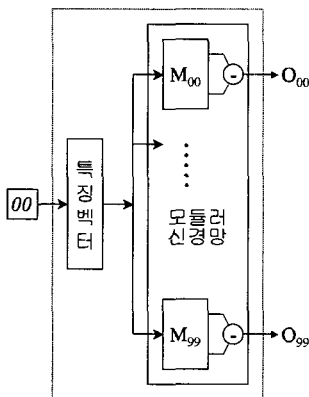


그림 3 모듈러 인식기의 전체 구조

류한다.

2.3 특징 추출

광학 문자 인식 시스템의 성능은 사용하는 특징에 따라 큰 영향을 받는다. 특히 본 논문에서는 인식 대상이 되는 부류수가 방대하기 때문에 분별력이 뛰어난 특징을 필요로 한다. 현재까지 여러 종류의 특징들이 개발되어 사용되고 있으나, 본 논문에서는 흑/백 분포와 방향 거리 정보를 모두 포함하고 있어 높은 분별력을 갖는 DDD(Directional Distance Distribution, 방향 거리 분포) 특징을 사용한다 [23].

입력 패턴 $P_{m \times n}$ 은 먼저 20×16 크기의 명암 메쉬, $R_{20 \times 16}$ 으로 크기 정규화 된다. 보통의 날자 단위에서는 정방형(16×16 , 20×20 등) 크기의 정규화를 주로 사용하는데 두자 접촉 숫자열은 날자와 다르게 두 개의 날자가 수평으로 놓여 있기 때문에, 입력 패턴의 왜곡을 최소화하기 위해 가로 길이가 세로 길이보다 큰 20×16 크기의 정규화를 사용하였다.

R은 $m \times n$ P 메쉬상에 20×16 격자 메쉬를 씌움으로써 계산된다. P의 몇 개 격자들은 서로 다른 비율로 R에 있는 한 개의 격자와 겹쳐지게 된다. R의 한 격자의 값은 P의 겹쳐진 모든 격자들을 사용해서 계산한다. R 격자의 값은 P의 겹치는 화소들 값에 가중치(겹치는 정도)를 곱한 값의 합으로 계산된다. R의 모든 격자 값이 계산된 후, 이 값들은 0.0과 1.0사이의 값으로 정규화된다. 모든 겹치는 격자들을 겹침 정도에 따라 고려하였으므로, 크기 정규화 과정에서 정보 손실이 최소화되었다고 말할 수 있다. 그림 4는 입력 패턴 P와 크기 정규화된 R 메쉬, 그리고 R의 이진화 결과를 보여주고 있다.

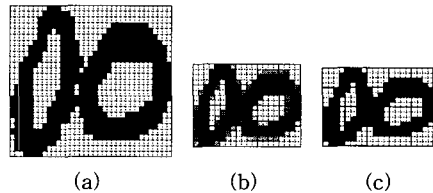


그림 4 입력 패턴의 크기-정규화 (a) 입력 패턴 $P_{35 \times 27}$, (b) 크기-정규화된 명암 메쉬 $R_{20 \times 16}$, (c) 이진화된 크기-정규화 메쉬 $R_{20 \times 16}$

DDD 특징은 그림 4 (c)의 이진 맵으로부터 계산한다. 입력 패턴 배열에 있는 각 화소에 W(흰) 집합과 B(검은) 집합이라 부르는 8 바이트 집합을 그림 5와 같이 할당한다. 흰 화소는 8 방향에 대해 가장 가까운 검은 화소까지의 거리값을 W 집합에 기록한다. B 집합은

거리 계산 없이 단순히 0으로 채운다. 같은 방식으로 검은 화소는 8-방향에 대해 가장 가까운 흰 화소까지의 거리 값을 B 집합에 기록하고 W 집합은 0으로 채운다. (8-방향 코드는 0(동), 1(북동), 2(북), 3(북서), 4(서), 5(남서), 6(남), 7(남동)이다.)

화소 \ WB	W							B								
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
(8, 3)	3	4	4	1	1	1	1	2	0	0	0	0	0	0	0	0
(8, 4)	0	0	0	0	0	0	0	0	1	1	1	5	3	3	8	1

그림 5 WB 인코딩 예

그림 5는 그림 4 (c)에 있는 예제 패턴에서, 좌측 가장 위의 화소를 (0, 0)이라 할 때 (8, 3)와 (8, 4)의 화소의 WB 인코딩 예이다. (8, 3)의 화소는 흰 화소이므로 B 집합은 0값을 갖는다. W 집합의 8 방향 거리를 계산하기 위해 이 화소는 각 방향으로 광선을 쏘며, 이 광선은 검은 화소와 충돌할 때까지 진행한다. 충돌하면, 진행 거리를 광선 방향에 해당하는 바이트에 기록한다. 한 예로서, 방향 0의 광선은 (8, 3)W → (9, 3)W → (10, 3)W → (11, 3)B의 경로를 밟는다. 결국 진행 거리 3를 갖는다. 그리고 (8, 4) 화소의 경우는 검은 화소이므로 W 집합은 0값을 가지고, B 집합의 8방향 거리가 계산되어 진다.

모든 화소에 대한 WB 코딩을 계산한 후, 배열을 5×4 블록 매쉬로 나눈다. 한 블록은 4×4 화소로 구성된 부 영역이다. 각 블록은 16 바이트의 WB 코딩에 대해 16 화소의 평균을 구한다. 마지막으로 0.0과 1.0사이로 정규화한다. 최종적으로 320-차원(20 블록과 16값/블록)의 DDD 특징 벡터를 얻는다. 그림 6은 그림 4 (c)에서 추출한 DDD 특징 벡터이다.

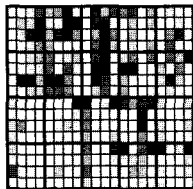


그림 6 DDD 특징 벡터

3. 패턴 집합

훈련 집합의 품질이 인식기의 일반화(generalization)

능력을 좌우하기 때문에 패턴 인식에서 훈련 집합은 매우 중요하다 [15]. 이 장에서는 훈련 집합이 갖추어야 할 요건과 두자 접촉 숫자열 수집 방법에 대해서 설명한다.

3.1 훈련 집합의 요건

다음은 훈련 집합이 높은 품질을 유지하기 위해 갖추어야 할 요건이다.

- 1) 충분한 양 : 훈련 집합의 개수가 적으면 훈련이 제대로 이루어지지 않아서 높은 인식률을 기대할 수 없으며, 그 개수가 너무 많으면 훈련 시간이 너무 많이 소요되며 개수를 늘리는데 따르는 인식률 향상 효과가 미미하기 때문에 빠른 훈련 시간과 높은 인식률을 얻을 수 있는 적당한 양의 집합이 필요하다. Jain은 훈련집합의 크기는 특징 벡터 차원수의 5~10배 가되어야 한다는 개략적인 규칙을 제시하였다 [24].
- 2) 부류별 균형 : 부류간 균형을 이루지 못한다는 것은 어느 특정 부류만을 훈련하는 효과가 나타나므로 부류간 정인식률에 큰 편차가 나타나고, 전체적인 인식 능력이 약하다 할 수 있다. 때문에 분별력을 향상시키기 위해서는 부류별 균형 잡힌 개수가 필요하다 [25].
- 3) 다양한 형태의 패턴 : 입력으로 사용될 패턴들은 문서의 종류나 입력 형태 그리고 사람들의 필기 습관에 따라 패턴의 크기와 모양이 달라지는데, 이들 다양한 형태를 포함해야 한다.

3.2 표준 데이터베이스에서 실제 패턴 수집

제안한 방법의 성능 검사를 위해 NIST의 숫자열 필드에서 두자 접촉한 숫자열 5,405개를 추출하였다. 그림 7은 NIST 숫자열 필드와 여기에서 추출한 두자 접촉 패턴들이며, 그림 8은 이렇게 모은 패턴의 일부를 보여 준다. 표 1은 추출한 패턴들의 부류별 발생 빈도수이고, 표 2는 빈도수에 따라 부류들을 나열하였다. 가장 많이 나타나는 부류는 '89'로서, 이는 NIST 데이터베이스를 1989년에 만들었으며 수집 양식에 연도를 기입하는 필드가 있었기 때문이다. 가장 적게 나타나는 부류는 '91'로서 한 개만 나타났으며, 10개미만 부류가 21개이다. 이처럼 실제적으로 부류간의 발생 빈도수에서 큰 편차를 보인다. 때문에 인식기 훈련이 쉽지 않다는 문제점을 안고 있다.

3.3 합성

날자 단위 숫자는 부류수가 10개로 적으며, 여러 문서들에서 나타나는 빈도수에 큰 차이가 없기 때문에 훈련 집합 수집이 수월하나, 두자 접촉 숫자열은 부류수가 100개로서 많으며 부류별로 나타나는 빈도수의 편차가 커서 훈련 집합 수집이 어렵다 (표 1과 표 2 참조).

본 논문은 훈련 집합의 부류간 균형과 충분한 양의

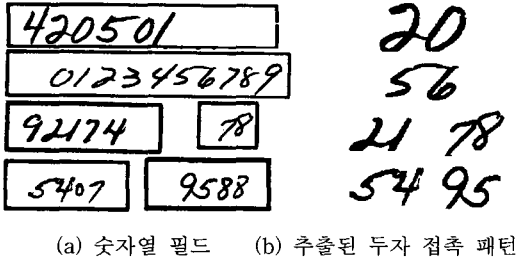


그림 7 NIST 숫자 필드 및 추출한 두자 접측 패턴

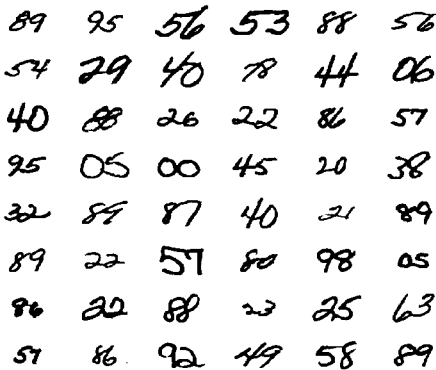


그림 8 NIST 데이터베이스에서 추출한 패턴들

표 1 실제 패턴 집합의 부류별 빈도수

L \ R	0	1	2	3	4	5	6	7	8	9	합계
0	118	9	36	45	43	67	29	6	43	84	480
1	3	2	9	14	4	19	2	9	4	25	91
2	192	64	149	179	27	171	142	9	95	53	1,081
3	80	4	47	45	47	80	58	12	42	48	463
4	151	16	25	52	132	129	103	14	75	96	793
5	123	6	65	51	80	63	169	41	121	114	833
6	52	4	33	31	23	123	27	2	25	15	335
7	22	4	7	15	14	36	10	16	31	61	216
8	54	4	24	43	33	60	34	21	75	563	911
9	7	1	10	19	11	44	3	9	30	68	202
합계	802	114	405	494	414	792	577	139	541	1,127	5,405

확보를 위하여 낱자 숫자 두 개를 붙여서 두자 접측 패턴을 합성하는 기법도 사용하였다. 실제 패턴 집합에서 샘플수가 적은 부류에 대해 합성 패턴을 추가하여 성능을 측정하는 인식 실험도 수행하였다.

표 2 빈도수별 부류

빈도수	부 류
10개 미만	01 07 10 11 12 14 16 17 18 27 31 51 61 67 71 72 81 90 91 96 97 (21개)
10~19	13 15 37 41 47 69 73 74 76 77 92 93 94 (13개)
20~29	06 19 24 42 64 66 68 70 82 87 (10개)
30~39	02 62 63 75 78 84 86 98 (8개)
40~49	03 04 08 32 33 34 38 39 57 83 95 (11개)
50~59	29 36 43 53 60 80 (6개)
60~69	05 21 52 55 79 85 99 (7개)
70~79	48 88 (2개)
80~89	09 30 35 54 (4개)
90~99	28 49 (2개)
100개 이상	00 20 22 23 25 26 40 44 45 46 50 56 58 59 65 89 (16개)

두개의 낱자 숫자 패턴을 'A'와 'B'라 할 때 'A'와 'B'를 하나로 합한 'AB' 와 'BA' 패턴을 합성할 수 있다. 그림 9는 낱자 숫자 '2'와 '0', 그리고 합성된 두자 접측 숫자열 '20'와 '02'를 보여주고 있다. 접측시키기 위해서는 왼쪽 패턴의 기준선(위선, 아래선 또는 중앙선)을 중심으로 오른쪽 패턴과 높낮이를 맞춘 후, 접측이 발생할 때까지 오른쪽으로 이동시킨다.

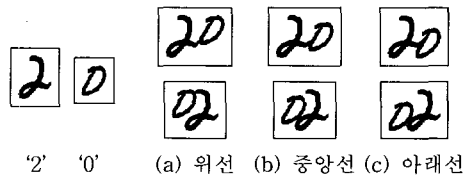


그림 9 낱자 숫자와 합성된 패턴

낱자 숫자는 NIST 데이터베이스에서 추출하였고, 숫자열 합성시 양쪽 낱자 단위의 높이의 차가 크면 이상한 모양의 패턴이 되므로 높이의 차가 $\pm 30\%$ 이내가 되도록 하였다. 그림 10은 합성한 패턴의 예를 부류별로 하나씩 보여준다. 이렇게 합성한 패턴 중에서 실제 패턴에 가까운 즉, 자연스러운 모양을 가진 패턴을 선택하여 사용하였다.

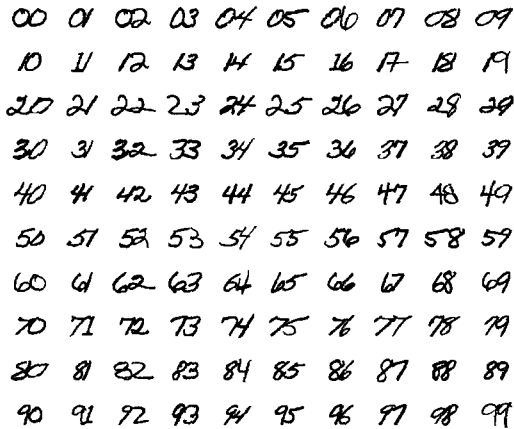


그림 10 합성된 두자 접촉 숫자열

4. 실험

제안한 방법에 대해 여러 측면에서 성능을 평가하기 위해 먼저 실험 설계를 하고, 이에 따라 실험을 수행하고 그 결과를 분석하였다.

4.1 실험 설계

우리가 NIST 데이터베이스에서 수집한 5,405개의 실제 패턴 집합은 100개 부류에 대해 평균 54개의 샘플만을 가지며, 부류간의 불균형도 심하다. 이러한 상황에서 보통 많이 사용하는 잭나이프(jackknife) 기법을 도입해 사용한다. 또한 샘플수가 너무 적은 부류에 대해 합성한 패턴을 보충하여 부류간 훈련 샘플수의 균형을 적절히 맞춘 후 실험도 수행하였다.

(1) 잭나이프 기법

총 N개의 샘플을 가지고 있을 때, 이중 P%를 검사를 위해 남겨놓고 (100-P)%를 가지고 인식을 훈련한다. 훈련을 마친 후 P%의 검사 샘플을 가지고 검사하여 인식을 기록한다. 이제 앞에서 검사에 사용한 샘플을 제외한 것들 중 P%를 검사 샘플로 새로이 선택하고, 훈련하고 검사하고 인식을 기록한다. 이러한 과정을 모든 샘플이 검사에 한번씩 참여하도록 반복한다. 기록된 모든 인식률의 평균을 구하고, 이 평균 인식률을 최종 인식률로 간주한다. 우리 실험에서는 P로 10%를 사용하였다. 즉 총 10번의 성능 실험이 수행되었고, 10개의 인식률 평균이 최종 인식률이 된다.

(2) 네 가지 실험

실제 패턴과 합성한 패턴을 사용하여 네가지 실험을 수행함으로써, 제안한 접근 방법과 모듈러 신경망 인식기의 성능을 여러 측면에서 평가하였다. 표 3은 네가지

실험을 요약하여 기술하고 있다. 각각에 대해 훈련과 검사를 위해 사용된 패턴에 대해 기술하고, 잭나이프 기법을 적용했는지 여부, 그리고 실험 목적을 설명한다. 각 실험에서는 이를 보다 자세히 기술하고 실험으로 얻은 성능 자료와 이를 분석한 결과를 제시한다.

표 3 네가지 실험

실험	훈련 집합	검사 집합	잭나이프	목적
실험 1	실제 패턴	실제 패턴	적용	실제 상황에 대한 성능평가 & 샘플 빈도수 변화에 따른 성능변화
실험 2	실제 패턴 + 합성 패턴	실제 패턴	적용	낮은 빈도수 부류에 대한 샘플 보충을 통한 성능향상
실험 3	합성 패턴	실제 패턴	적용 않함	충분한 양의 훈련 집합에 대한 성능평가
실험 4	합성 패턴	합성 패턴	적용 않함	충분한 양의 실제패턴이 주어진 상황의 성능 추정

(3) 비 모듈러와 모듈러 신경망 인식 성능 비교

2장에서 기술한 전통적인 비 모듈러 신경망과 최근 제안되어 사용되는 모듈러 신경망의 성능을 비교한다. 표 3에서 훈련 샘플이 충분하고 부류간 균형있는 실험 3에 대해 두 신경망의 성능을 비교한다. 특히 이 논문이 다루고 있는 100-부류의 대용량 분류에서 모듈러 신경망이 우수함을 성능 자료를 통해 입증한다.

(4) 필드 인식률과 낱자 인식률

인식률은 두 가지로 측정하였다. 첫째로 두자 접촉 숫자열 자체를 100개 부류 중 하나로 옮겨 분류했는지를 평가하는 필드 인식률(field recognition rate)이다. 두 번째로 두자 접촉 숫자열에 들어있는 두 개의 낱자를 개별적으로 보고 각각을 옮겨 분류했는지를 평가하는 낱자 인식률(individual numeral recognition rate)이다. 낱자 인식률에서는 CC, CE, EC, EE의 네가지 경우가 있는데, CC는 두자 모두 맞는 경우이고(예, 23을 23으로 분류), CE는 앞 숫자는 맞고 뒤 숫자는 틀린 경우(예, 23을 24로 분류), EC는 앞 숫자는 틀리고 뒤 숫자는 맞는 경우(예, 23을 43으로 분류), EE는 두자 모두 틀린 경우(예, 23을 44로 분류)이다. 이렇게 낱자 단위로 옮겨 분류한 개수를 세어 인식률을 측정한다.

4.2 실험 1

잭나이프 기법에서 검사에 10%의 샘플을 사용하기

때문에 10개미만의 샘플을 가진 부류는 제외하고 실험을 하였다. 표 4는 샘플을 T개 이상 가진 부류를 대상으로 훈련 집합과 검사 집합에 대한 필드 인식률과 남자 인식률을 측정할 결과를 보여준다. 여기서 T는 10에서 시작하여 100까지 10씩 증가시키며 실험하였다. T가 증가할수록 부류수가 적어지고 부류당 샘플 수가 많아지므로, 인식률이 꾸준히 증가한다. 이 실험은 표 3에서 보였듯이, 잭나이프 기법을 사용하였고, P=10%이므로 총 10번의 훈련과 인식을 통하여 얻은 인식률의 평균값을 구하여 표 4에 기록하였다.

표 4 실험 1에서 얻은 인식 성능

부류당 샘플개수(T)	부류수	총 샘플수	훈련 집합 평균 인식률(%)		검사 집합 평균 인식률(%)	
			필드	남자	필드	남자
10개 이상	79	5,297	92.0	94.94	77.13	86.62
20개 이상	66	5,112	94.72	96.58	79.97	88.39
30개 이상	56	4,864	95.41	97.0	81.81	89.47
40개 이상	48	4,600	96.34	97.63	83.41	90.45
50개 이상	37	4,112	97.20	98.18	86.55	92.28
60개 이상	31	3,792	97.71	98.54	88.48	93.41
70개 이상	24	3,344	98.09	98.83	90.43	94.50
80개 이상	22	3,194	98.57	99.11	92.42	95.65
90개 이상	18	2,870	98.73	99.24	93.28	96.10
100개 이상	16	2,679	98.88	99.20	94.25	96.51

표 5는 T=10일 때의 부류별 인식률을 보여준다. 이 표는 샘플 수가 인식률에 미치는 영향을 알아보기 위해 작성하였다. 표에서 각 부류에 해당하는 x/y 표시는, y는 잭나이프 기법에서 10번 인식 실험에 참가한 검사 샘플의 총수를 말하며 x는 이중 정인식한 샘플 수를 말한다. 총 5,405개중 10개미만을 갖는 21개 부류를 제외하여 총 5,297개에 대해 검사가 수행되었다. 표에서 샘플 개수가 많은 부류일 수록 인식률이 높고 적을수록 인식률이 낮음을 보여주고 있다. 특히 '76', '92', '47' 와 '69' 등 샘플수가 20개 미만인 부류들에서는 인식률이 0%로 나타나고 있다. 이것으로 볼 때 높은 성능을 얻기 위해서는 부류별 샘플수가 적은 부류에 일정수준의 샘플들이 보강 되어야함을 알 수 있다. 때문에 실험 2에서는 부류별 샘플수가 적은 부류에 대해 샘플 수를 보강한 실험이 이루어졌다.

4.3 실험 2

실험 1에서는 10개미만의 샘플을 가진 부류는 제외하고 79개의 부류만 가지고 (표 3에서 T=10인 경우) 인식 실험한 결과 검사 집합에 대해 필드 인식률 77.13%와 남자 인식률 86.62%를 얻었다. 이 결과 표 5에서 살펴본 바와 같이 샘플 개수가 적은 부류에서 발생하는 낮은 인식률이 전체 인식률을 낮추는 주요 원인중의 하나이다. 따라서 실험 2에서는 이들 부류에 대해 샘플 수를 보충하고 이에 따른 인식률 변화를 측정하여, 샘플 수가 인식률에 미치는 영향을 고찰하고자 한다. 이를 위

표 5 T=10일 때의 부류별 인식률(x/y: y는 총 샘플 수이고 x는 이중 정인식한 샘플 수)

L \ R	0	1	2	3	4	5	6	7	8	9	합계 (%)
0	110 /118	-	26 /36	28 /45	31 /43	49 /67	22 /29	-	30 /43	76 /84	372/465 (80.0)
1	-	-	-	0 /14	-	8 /19	-	-	-	20 /25	28/58 (48.3)
2	168 /192	53 /64	139 /149	150 /179	8 /27	145 /171	125 /142	-	75 /95	26 /53	889/1,072 (82.9)
3	66 /80	-	32 /47	31 /45	30 /47	66 /80	40 /58	0 /12	25 /42	24 /48	314/459 (68.4)
4	130 /151	0 /16	8 /25	32 /52	115 /32	115 /129	94 /103	0 /14	51 /75	67 /96	612/793 (77.2)
5	105 /123	-	48 /65	37 /51	53 /80	47 /63	156 /169	29 /41	101 /121	95 /114	671/827 (81.1)
6	46 /52	-	24 /33	23 /31	15 /23	108 /123	16 /27	-	19 /25	0 /15	251/329 (76.3)
7	14 /22	-	-	1 /15	0 /14	21 /36	0 /10	13 /16	21 /31	49 /61	119/205 (58.0)
8	39 /54	-	9 /24	33 /43	15 /33	42 /60	19 /34	8 /21	34 /75	536 /563	735/907 (81.0)
9	-	-	0 /10	5 /19	0 /11	27 /44	-	-	13 /30	50 /68	95/182 (52.2)
합계 (%)	678 /792 (85.6)	53 /80 (66.3)	286 /389 (73.5)	340 /494 (68.8)	267 /410 (65.1)	628 /792 (79.3)	472 /572 (82.5)	50 /104 (50.1)	369 /537 (68.7)	943 /1,127 (83.7)	4,086/5,297 (77.13)

해 100개미만의 빈도를 가지는 부류들의 샘플 수를 100개가 되도록 3.3절에서 합성한 패턴을 추가한 다음 잭나이프 기법을 이용하여 성능 평가를 하였다. 추가한 합성 패턴 중에서 비교적 실제 패턴에 가까운 것들을 주관적으로 선택하여 추가하였다. 이렇게 추가한 합성 샘플의 개수는 5,674개이며 실제 샘플 5,405개와 합하여 총 11,079개이다. 표 6은 실험 2에서 얻은 인식 성능을 요약하고 있다. 표에서 훈련과 검사 집합의 샘플 개수는 잭나이프 기법의 10번의 실험에 참가한 샘플을 모두 합한 것이다. 검사 집합에 대해 필드 인식률 82.76%와 날자 인식률 90.08%를 얻었다. 이는 표 4에서 79개 부류를 갖고 실험한 77.13%와 86.62%에서 크게 향상된 성

능이다. 실험 2의 경우 100개 부류 모두 참여했다는 점을 감안하면 향상 정도는 더 크다고 말할 수 있다.

4.4 실험 3

이번 실험은 훈련 집합을 위해 합성한 샘플을 사용하고, 실제 샘플을 검사에 사용하여 성능을 평가하였다. 훈련 집합은 부류별로 500샘플씩 총 50,000 개를 사용하였다. 표 7은 실험 3에서 얻은 인식 성능을 보여주고 있다. 표 8은 검사 집합에 대한 부류별 인식률을 보여준다. 표 5와 비교해 보면, 부류간 인식률 편차가 적은 편이다. 이는 비록 합성한 패턴들을 사용했지만 훈련 집합이 부류간 균형을 유지한 때문이다. 이 실험에서 훈련에서는 합성된 패턴 그리고 검사에서는 실제 패턴만을 사용하였고, 합성 패턴과 실제 패턴간에는 모양에서 근본적인 차이가 있을 수 있으므로 실험 2보다 낮은 78.0%라는 성능을 얻었다.

4.5 실험 4

실험 3에서는 합성 패턴을 이용하여 훈련하고 실제 패턴을 이용하여 검사하였으나, 실험 4에서는 검사 집합을 위해서도 합성 패턴을 사용한다. 즉, 합성한 100,000 샘플을 동일한 크기의 두 집합으로 나누어 하나는 훈련으로 다른 하나는 검사 집합으로 사용한다. 이 실험의 목적은 실제 패턴만으로 구성된 충분한 크기의 훈련 집합이 주어진 상황에 대한 모듈러 신경망 인식기의 인식 성능을 유추하기 위함이다. 즉, 실험 2와 실험 3은 실제

표 6 실험 2에서 얻은 인식 성능

데이터 단위	훈련 집합				검사 집합			
	개수	인식	거부	인식률 (%)	개수	인식	거부	인식률 (%)
필드	99,711	94,567	5,144	94.84	11,079	9,169	1,910	82.76
날자	199,422	192,887	6,535	96.71	22,158	19,959	2,199	90.08

표 7 실험 3에서 얻은 인식 성능

데이터	필드 인식률 (%)	날자 인식률 (%)
훈련집합	95.3	97.1
검사집합	78.0	86.9

표 8 실험 3에서 검사 데이터에 대한 인식 성능

L \ R	0	1	2	3	4	5	6	7	8	9	합계 (%)
0	102 /118	5 /9	32 /36	30 /45	33 /43	50 /67	27 /29	6 /6	34 /43	62 /84	381/480 (79.4)
1	1 /3	0 /2	7 /9	8 /14	3 /4	11 /19	1 /2	9 /9	3 /4	22 /25	65/91 (71.4)
2	161 /192	53 /64	135 /149	154 /179	21 /27	143 /171	126 /142	6 /9	61 /95	27 /53	887/1,081 (82.0)
3	60 /80	4 /4	39 /47	43 /45	34 /47	64 /80	49 /58	6 /12	30 /42	29 /48	358/463 (77.3)
4	113 /151	12 /16	19 /25	44 /52	109 /132	105 /129	92 /103	11 /14	55 /75	72 /96	632/793 (79.7)
5	104 /123	3 /6	55 /65	39 /51	54 /80	60 /63	153 /169	31 /41	94 /121	97 /114	690/833 (82.8)
6	46 /52	4 /4	27 /33	27 /31	18 /23	114 /123	24 /27	2 /2	23 /25	8 /15	293/355 (82.5)
7	10 /22	1 /4	4 /7	10 /15	13 /14	14 /36	7 /10	15 /16	20 /31	40 /61	134/216 (61.8)
8	33 /54	2 /4	20 /24	35 /43	22 /33	50 /60	19 /34	14 /21	46 /75	376 /563	617/911 (67.8)
9	5 /7	0 /1	8 /10	14 /19	9 /11	26 /44	2 /3	9 /9	20 /30	65 /68	158/202 (78.6)
합계 (%)	635 /802 (79.2)	84 /114 (73.7)	346 /405 (85.4)	404 /494 (81.8)	316 /414 (76.3)	637 /792 (80.4)	500 /577 (86.7)	109 /139 (78.4)	386 /541 (71.3)	798 /1,127 (70.8)	4,215/5,405 (78.0)

표 9 실험 4에서 얻은 인식 성능

L \ R	0	1	2	3	4	5	6	7	8	9	합계 (%)
0	493 /500	485 /500	484 /500	461 /500	485 /500	475 /500	475 /500	446 /500	450 /500	472 /500	4,726/5,000 (94.5)
1	483 /500	461 /500	435 /500	448 /500	488 /500	459 /500	493 /500	439 /500	472 /500	471 /500	4,649/5,000 (92.8)
2	473 /500	453 /500	454 /500	442 /500	459 /500	448 /500	462 /500	416 /500	440 /500	454 /500	4,501/5,000 (90.0)
3	470 /500	463 /500	456 /500	444 /500	459 /500	445 /500	456 /500	388 /500	435 /500	440 /500	4,456/5,000 (89.1)
4	478 /500	474 /500	469 /500	436 /500	459 /500	465 /500	470 /500	454 /500	447 /500	426 /500	4,578/5,000 (91.6)
5	463 /500	450 /500	435 /500	404 /500	455 /500	461 /500	462 /500	399 /500	407 /500	448 /500	4,384/5,000 (87.7)
6	485 /500	481 /500	450 /500	465 /500	466 /500	471 /500	490 /500	443 /500	451 /500	432 /500	4,634/5,000 (92.7)
7	469 /500	446 /500	438 /500	424 /500	444 /500	458 /500	473 /500	438 /500	413 /500	437 /500	4,440/5,000 (88.8)
8	461 /500	437 /500	448 /500	419 /500	443 /500	456 /500	448 /500	411 /500	402 /500	411 /500	4,336/5,000 (86.7)
9	466 /500	469 /500	455 /500	445 /500	430 /500	452 /500	467 /500	430 /500	432 /500	452 /500	4,498/5,000 (90.0)
합계 (%)	4,741 /5,000 (94.8)	4,619 /5,000 (92.4)	4,524 /5,000 (90.5)	4,388 /5,000 (87.8)	4,588 /5,000 (91.8)	4,590 /5,000 (91.8)	4,696 /5,000 (93.9)	4,264 /5,000 (85.3)	4,349 /5,000 (87.0)	4,443 /5,000 (88.9)	45,202/50,000 (90.4)

패턴과 합성 패턴을 혼합한 환경에서 실험하였으므로 실제 패턴과 합성 패턴간의 모양 차이를 내포하고 있고, 이 사실이 인식을 저하의 원인이 되었다. 실험 4에서는 훈련과 검사 모두 충분한 합성 패턴을 사용하여 인식 성능을 측정하여 모듈러 신경망 인식기의 100-부류 분류 문제에서 인식 능력을 파악하고자 한다.

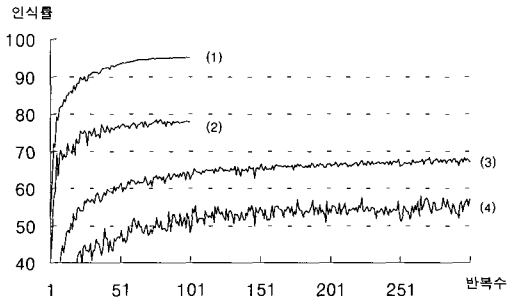
실험 결과 표 9에서 보는 바와 같이 검사 집합에 대해 필드 인식을 90.4%로서 높은 인식 능력을 보였다. 표 10은 낱자 인식을 94.5%를 보여주고 있다. 이러한 높은 인식을 보면 모듈러 신경망 인식기는 100-부류 분류 문제에 대해 높은 인식 성능을 가졌다고 볼 수 있다. 이를 기반으로 유추해 보면, 충분한 양과 부류간 균형 있도록 실제 패턴을 수집한다면 실험 4와 같은 높은 성능을 갖는 인식기를 만들 수 있을 것이다.

표 10 실험 4의 낱자 단위 인식률

인식 데이터	CC	CE	EC	EE	합계	인식률
필드 개수	45,202	2,049	2,081	668	50,000	필드 인식률 45,202/50,000 = 90.4%
정인식 낱자 개수	90,404	2,049	2,081	0	94,534	낱자 인식률 94,534/100,000 = 94.5%

4.6 비 모듈러와 모듈러 신경망 인식기의 인식 성능 비교

본 논문에서 사용한 모듈러와 비 모듈러 인식기의 성능을 비교하기 위하여 실험 3 환경에서 훈련 반복(epoch) 수를 증가시키면서 인식률을 측정하여 그림 11과 같은 결과를 얻었다. 그림에서 (1)과 (2)는 모듈러 인식기의 인식 성능을 나타내고 있고, (3)과 (4)가 비



- (1) 모듈러, 훈련 데이터
- (2) 모듈러, 검사 데이터
- (3) 비 모듈러, 훈련 데이터
- (4) 비 모듈러, 검사 데이터

그림 11 모듈러와 비 모듈러 인식기의 인식률 변화

모듈러 인식기의 인식 성능을 보여주고 있다. 그래프가 보여주듯이 모듈러 신경망은 인식률이 훈련 반복 수가 증가하면서 빠르게 수렴하고 있는 것을 보이고 있으나, 비 모듈러는 비교적 느리게 수렴하는 것을 볼 수 있다. 훈련은 인식률 증가가 거의 나타나지 않는 곳에서 중단하였다. 모듈러 신경망의 경우 반복 횟수 100에서, 비 모듈러 신경망의 경우 300에서 중단하였다. 수렴한 후의 인식률은 표 11에서 볼 수 있다. 훈련 집합에 대한 경우 28.3%, 검사 집합의 경우 20.9%의 큰 차이를 보이고 있다. 이 비교 실험에서 얻을 수 있는 사실은 100 정도의 부류를 갖는 방대한 분류 문제에서는 모듈러 신경망을 이용함으로써 높은 성능 향상을 얻을 수 있다는 것이다.

표 11 모듈러와 비 모듈러 인식기의 수렴 후 필드 인식률

인식기 종류	데이터	인식률
모듈러	훈련 집합	95.3%
	검사 집합	78.0%
비 모듈러	훈련 집합	67.0%
	검사 집합	57.1%

5. 결론 및 향후 연구 과제

본 논문은 두자 접촉 숫자열 인식 문제를 분할-자유 접근 방법으로 해결하였다. 인식기로는 신경망을 사용하였고, 특히 대용량 분류 문제에서 높은 성능을 보이는 모듈러 신경망을 사용하였다. 수집한 실제 패턴 개수가 적은 문제를 다루기 위해 합성 패턴도 사용하였으며, 다양한 측면에서의 실험을 통하여 분할-자유 기법과 모듈러 신경망 인식기의 유용성을 보였다.

앞으로의 연구 과제는 다음과 같다. 첫째, 실제 데이터의 지속적인 수집을 통하여 대용량 훈련 집합을 구성하고, 이를 통하여 실제 패턴에 대한 성능 평가 및 실제 상황에서 사용할 수 있는 인식기 구축을 수행하다. 둘째, 두자 접촉 숫자열에 적합한 특징 개발, 신경망 구조 및 훈련에 관련한 파라미터 설정등을 통하여 인식률 향상을 모색한다. 셋째, 임의 길이 숫자열 인식기를 위하여 낱자 숫자 인식기와 두자 접촉 숫자열 인식기를 이용하는 방안을 강구하는 것이다.

참고 문헌

[1] C.Y. Suen et al., "Computer recognition of

unconstrained handwritten numerals," *Proceedings of IEEE*, Vol. 80, No. 7, pp. 1162-1180, 1992.

[2] H. Bunke, *Advances in Structural and Syntactic Pattern Recognition*, World Scientific, 1992.

[3] H. Bunke, P.S.P. Wang, and H.S. Baird (Edited), *Document Image Analysis*, World Scientific, 1994.

[4] H.S. Baird, H. Bunke, and K. Yamamoto (Edited), *Structured Document Image Analysis*, Springer-Verlag, 1992.

[5] M. Shridhar and A. Badreldin, "Recognition of isolated and simply connected handwritten numerals," *Pattern Recognition*, Vol. 19, No. 1, pp. 1-12, 1986.

[6] O. Matan et al., "Reading handwritten digits: a ZIP code recognition system," *IEEE Computers*, Vol. 25, No. 7, pp. 59-63, 1992.

[7] 이성환, 이용재, "순환 신경망을 이용한 연속 필기 숫자열의 통합적 분할 및 인식", 한국정보과학회 1995 추계학술발표회지, pp. 447-450, 1995.

[8] H. Nishida and S. Mori, "A model-based split-and-merge method for character string recognition," in *Document Image Analysis* (Edited by Bunke et al.), World Scientific, pp. 209-226, 1994.

[9] R. Fenrich, "Segmentation of automatically located handwritten words," *Proceedings of IWFHR II*, pp. 33-44, 1991.

[10] J.M. Westall and M.S. Narasimha, "Vertex directed segmentation of handwritten numerals," *Pattern Recognition*, Vol. 26, No. 10, pp. 1473-1486, 1993.

[11] Z. Shi and V. Govindaraju, "Segmentation and recognition of connected handwritten numeral strings," *Pattern Recognition*, Vol. 30, No. 9, pp. 1501-1505, 1997.

[12] 최순만, 홍기천, 오일석, "무제약으로 필기한 숫자 필드의 인식에 대한 연구", 제8회 영상 처리 및 이해에 관한 워크샵 발표논문집, pp. 1-6, 1996.

[13] X. Wang, V. Govindaraju, and S. Srihari, "Holistic recognition of touching digits," *Proceedings of IWFHR VI*, pp. 295-303, 1998.

[14] J.T. Favata, G. Srikantau, and S.N. Srihari, "Handwritten character/digit recognition using a multiple feature/resolution philosophy," *Proceedings of IWFHR IV*, pp. 57-66, 1994.

[15] C.G. Looney, *Pattern Recognition using Neural Networks*, Oxford University Press, 1997.

[16] I.S. Oh, J.S. Lee, S.M. Choi, and K.C. Hong, "Class-expert approach to unconstrained handwritten numeral recognition," *Proceedings of IWFHR V*, pp. 95-102, 1996.

[17] I.S. Oh and C.Y. Suen, "A class-modular feedforward neural network for handwriting recognition," *IEEE Tr. on Neural Networks* (submitted).

- [18] N.W. Strathy, C.Y. Suen, and A. Krzyzak, "Segmentation of handwritten digits using contour features," *Proceedings of ICDAR*, pp. 577-580, 1992.
- [19] Z. Lu, Z. Chi, W.-C. Siu, and P. Shi, "A Background-thinning-based approach for separating and recognizing connected handwritten digit strings," *Pattern Recognition*, Vol. 32, No. 6, pp. 921-933, 1999.
- [20] E. Gose, R. Johnsonbaugh and S. Jost, *Pattern Recognition and Image Analysis*, Prentice Hall, 1996.
- [21] H. Fujisawa, Y. Nakano, and K. Kurino, "Segmentation methods for character recognition: from segmentation to document structure analysis," *Proceedings of IEEE*, Vol. 80, No. 7, pp. 1079-1092, 1992.
- [22] M. Cheriet, Y. Huang and C. Suen, "Background region-based algorithm for the segmentation of connected digits," *Proceedings of 11th ICPR*, pp. 619-622, 1992.
- [23] 오일석, C.Y. Suen, "광학 문자 인식을 위한 거리 특징," *한국정보과학회논문지*, Vol. 25, No. 7, pp. 1028-1043, 1998.
- [24] A. K. Jain and B. Chandrasekaran, "Dimensionality and sample size considerations in pattern recognition practice," *Handbook of Statistics*, Vol. 2, (Edited by P.R. Krishnaiah and L.N. Kanal), North-Holland Publishing Company, pp. 835-855, 1982.
- [25] R. Anand, K. Mehrotra, C.K. Mohan and S. Ranka, "An improved algorithm for neural network classification of imbalanced training sets," *IEEE Tr. on Neural Networks*, Vol. 4, No. 6, pp. 962-969, 1993.
- [26] 오일석, 이진선, C.Y. Suen, "부류 분별에 기반한 특징 분석 및 부류-의존 특징 결합", 제10회 영상 처리 및 이해에 관한 워크샵 발표논문집, pp. 169-175, 1998.



오 일 석

1984년 서울대학교 컴퓨터공학과 학사.
 1986년 한국과학기술원 전산학과 석사.
 1992년 한국과학기술원 전산학과 박사.
 1992년 ~ 현재 전북대학교 컴퓨터학과 교수. 관심분야는 컴퓨터 비전, 문서 및 문자 인식 등임.



최 순 만

1992년 전북대학교 전산통계학과 학사.
 1995년 전북대학교 전산통계학과 석사.
 1995년 ~ 현재 전북대학교 전산통계학과 박사과정. 관심분야는 컴퓨터 비전, 그래픽스, 숫자 및 문자 인식 등임.