

---

# 필기체 숫자의 인식과 비숫자의 기각을 위한 MLP 신경망의 구현 방법에 관한 연구

임길택\*

A Study on the Implementation Methods of MLP Neural Networks for the Recognition of  
Handwritten Numerals and the Rejection of Non-Numerals

Kil-Taek Lim\*

---

본 연구는 2005년도 경주대학교 학술연구비의 지원에 의해 이루어졌음

---

## 요 약

본 논문은 MLP (multilayer perceptrons) 신경망을 이용하여 필기체 숫자와 비숫자를 인식하거나 기각할 경우의 MLP 구현 방법에 관한 것이다. MLP는 인식률 및 속도, 메모리 요구량 등에 있어서 필기체 숫자인식에 매우 효율적인 인식기로 알려져 있다. 그러나 기존 연구에서의 MLP는 숫자 입력에 대해서만 인식에만 초점이 맞춰져 있으며 비숫자 입력 경우의 인식률, 기각률 및 동작 특성에 대해서는 연구된 바가 거의 없다. 본 논문에서는 숫자와 비숫자가 혼재하는 경우의 MLP 인식기의 구현방법에 대해서 논한다. MLP 인식기는 세 가지 방법으로 구현되며, 세 가지의 오류유형을 정의하여 각 인식 방법의 인식 특성을 분석하였다. 인식 실험은 66,705자의 필기체 숫자와 비숫자를 이용하여 이루어지며, 세가지 오류유형의 측면에서 숫자와 비숫자에 대한 가장 적절한 인식 방법이 논의된다.

## ABSTRACT

This paper describes the implementation methods of MLP (multilayer perceptrons) neural networks to recognize or reject handwritten numerals and non-numerals. The MLP has known to be a very efficient classifier to recognize handwritten numerals in terms of recognition accuracy, speed, and memory requirements. In the previous researches, however, researchers have focused on the only numeral inputs and have not payed attention to the non-numeral inputs with respect to recognition accuracy, rejection rates, and other characteristics. In this paper, we present some implementation methods of the MLP in the environments that numeral and non-numerals are mixed. The MLPs have been developed by three methods, and investigated with three error types introduced. The experiments have been conducted on a total of 66,705 images of numerals and non-numerals. The promising method to recognize numerals and reject non-numerals has been described in terms of the three error types.

## 키워드

numeral recognition, MLP, recognition rate, rejection rate

## I. 서 론

필기체 숫자인식은 우편물의 자동처리, 전장표 자동처리 등 각종 문서의 자동처리에 있어서 필수적으로 요구되는 기반 기술로 이에 대한 많은 연구가 진행되어왔다[1-5]. 숫자인식은 인식하고자 하는 대상 숫자를 추출하는 숫자 분리 과정을 거친 후 수행된다. 숫자의 추출 과정에서 숫자는 다른 숫자 또는 문자 등과 접촉되어 있지 않거나, 다른 숫자와 약간만 접촉된 상태로 존재하는 경우가 많다. 따라서 인식 대상 숫자의 추출 과정이 매우 용이하다고 가정하여 인식기는 온전히 분리된 숫자를 인식하게 된다. 기존의 대부분의 연구는 이렇게 온전히 분리된 고립 숫자의 연구 [1,2,3]에 초점이 맞추어져 있으며 분리가 제대로 되지 않거나 숫자가 아닌 비슷자 입력에 대한 연구는 거의 없다.

MLP 신경망 인식기 [7]는 숫자인식 문제에 있어서 가장 널리 사용되는 인식기의 하나로 그 효용성이 증명되었다. 그러나 MLP 신경망은 학습 상태에 따라 false positive 문제점과 같은 심각한 문제점을 나타내기도 한다. MLP 신경망의 입력으로 학습에 전혀 사용되지 않는 영상패턴이 입력되는 경우 전혀 예기치 못한 인식결과가 출력되기도 한다. 전술한 바와 같이 기존의 숫자인식 방법에서 사용하는 MLP 인식기는 숫자 영상만을 입력으로 하여 학습을 한다. 그러나 실제의 문제에 있어서는 인식기에 숫자 영상만이 입력되지 않을 수 있으며 비슷자 영상 입력도 경우에 따라서는 많이 발생한다. 왜냐하면 숫자인식 전단계로서 개별 숫자를 추출하는 숫자분리 과정에서 오류가 발생하므로, 올바른 숫자영상이 아닌 잘못된 영상이 인식기에 입력되는 것이다. 비슷자 영상이 입력되는 이러한 경우 MLP 인식기는 기각을 하여야 하지만 인식점수가 매우 높아 숫자로 인식을 하게 되는 문제점이 발생한다. 그러나 기존의 MLP 숫자인식기에 관한 연구에서는 이러한 문제점이 크게 논의되지 않았다. 또한 기존의 연구에서는 그 성능이 숫자 입력에 대한 인식률과 기각률 및 처리속도의 관점에서만 논의되어 왔으며 비슷자 입력에 대한 성능 분석은 논의되지 않았다.

본 논문은 숫자와 비슷자가 혼재하는 환경에서 MLP로 숫자인식기를 구현하는 방법에 관한 것이다. MLP를 이용한 숫자인식 방법을 제시하며 각 방법의 특성을 분석하여 적절한 방법을 제안한다. 각 방법의

체계적 분석을 위해 세 가지 종류의 오류유형을 정의한다. 이 오류유형에는 숫자 입력뿐만 아니라 비슷자 입력에 대한 성능 측정 기준도 포함되어 있다. 본 논문에서 정의하는 인식기의 오류유형을 사용함으로써 기존의 분석 방법에 비해 인식기의 인식 특성을 보다 명확하게 구분할 수 있다. 제시한 방법에 따른 숫자인식기의 구현 및 인식 실험은 약 53,000여자의 숫자 영상 데이터와 14,000여자의 비슷자 영상 데이터를 이용하여 이루어졌다. 실험 결과, 제안한 MLP 숫자인식기가 숫자와 비슷자가 혼재하는 경우에 적절하게 사용될 수 있음을 알 수 있었다.

## II. 인식기 구조 및 오류유형

### 2.1. 인식기 구조

본 연구에서 인식기로 사용하는 MLP 숫자인식기의 전형적인 구조를 그림 1에 나타내었다.

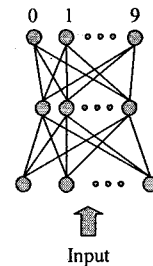


그림 1. 숫자인식 MLP의 전형적인 구조  
Fig. 1. Typical structure of the MLP for the numeral recognition

MLP의 입력층은 여러 개의 노드로 구성되며 각 노드는 인식대상 숫자영상에서 추출한 특징벡터의 각 성분을 입력받는다. 출력층은 0에서 9까지의 숫자 클래스 각각에 대응되는 노드들로 구성되며 각 노드의 값은 그 노드에 해당하는 숫자 클래스의 인식점수이다. MLP의 학습은 일반적으로 오류역전파 알고리즘을 이용하는데 목표벡터와 출력층의 출력벡터의 차이를 최소화하는 방향으로 가중치를 조정한다 [7].

본 연구에서는 이러한 구조의 MLP를 이용하여 숫자뿐만 아니라 비슷자도 처리하고자 한다. 입력이 숫자라면 0~9까지의 숫자 중 하나를 결과로 출력하고

비숫자이면 비슷자로 인식하여야한다. 비슷자는 숫자가 아닌 것이므로 기각을 하는 것을 제대로 인식한 것으로 한다. 이를 그림 2에 나타난 것과 같은 두 가지의 방법으로 구현한다.

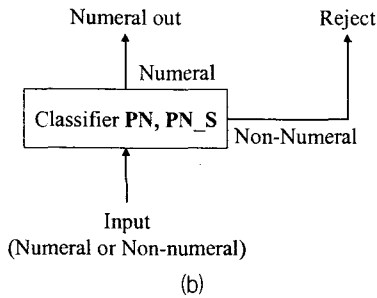
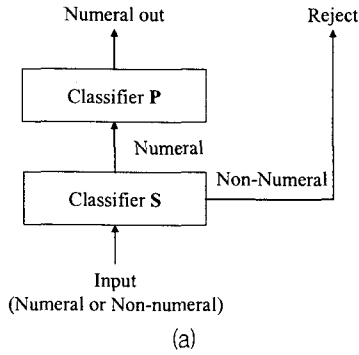


그림 2. 숫자와 비슷자의 두가지 인식 방법: (a) 인식방법1, (b) 인식방법2

Fig. 2. Two methods of the recognition of numerals and non-numerals : (a) method 1 and (b) method 2

인식방법1에서 S는 입력이 숫자 클래스인지 비슷자 클래스인지를 구분하는 이진분류기의 역할을 한다. 만약 S가 입력을 숫자 클래스로 분류한다면 입력은 P로 전달되고 P는 0~9의 숫자 중 하나를 출력하게 된다. 여기서 P는 숫자 클래스들만을 대상으로 학습하며, S는 숫자 클래스와 비슷자 클래스들을 모두 이용하여 구현된다. 인식방법2에서 PN과 PN\_S도 S와 마찬가지로 숫자 클래스와 비슷자 클래스들을 모두 이용하여 학습을 한다. PN은 최고 출력값이 임계치를 넘으면 숫자로 받아들여지고 그렇지 않다면 비슷자 클래스로 인정하여 기각을 한다. PN\_S의 경우는 출력층에 기각노드를 별도로 두어 기각노드에 해당하는 출력값이 가장 클 경우 비슷자 클래스로 인식을 한다. S, P, PN, PN\_S

모두 하나의 중간층을 가지는 MLP로 구현되는데, P와 PN은 10개의 출력노드, PN\_S는 11개의 출력노드, 그리고 S는 두 개의 출력노드로 구성되어있다. P와 PN은 그림 1과 동일한 구조를 가지며 S와 PN\_S의 구조는 그림 3과 같다.

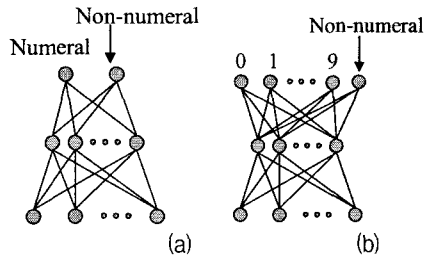


그림 3. (a) S와 (b) PN\_S의 구조  
Fig. 3. Structure of (a) S and (b) PN\_S

### 2.2. 세 가지 오류유형

일반적으로 인식기의 인식성능은 정확도와 속도의 관점에서 논의된다. 정확도는 다시 오류율(또는 인식률)과 기각률로 구분되기도 하는데, 기각이라 함은 인식기의 인식점수가 매우 낮은 경우 등을 이유로 인식 결과를 취하지 않는 것을 의미한다. 인식기의 정확도를 오류율과 기각률의 관점에서만 측정할 경우 인식대상 숫자의 추출이 완벽하지 않아서 발생하는 잘못된 입력에 대한 정확도의 측정이 어렵다. 따라서 이러한 오류율과 기각률을 포함하여 본 논문에서는 표 1과 같이 인식기의 오류유형을 세 가지로 정의한다. Type1 오류는 입력이 숫자 클래스이고 출력도 숫자 클래스이지만 잘못된 클래스로 인식을 하는 경우에 해당한다. Type2 오류는 입력이 숫자 클래스이지만 인식점수가 낮아 인식을 거절하는 것을 의미한다. Type3 오류는 비슷자 클래스 입력인데 숫자 클래스로 결정하는 경우의 오류를 의미한다. 이와 같이 오류유형을 정의하고 분석함으로써 숫자인식기의 정확도를 보다 분석적으로 명확하게 판단할 수 있다.

표 1. 숫자인식기의 세 가지 오류유형  
Table 1. Three error types of a numeral recognizer

오류유형	입력	출력	비고
Type1 오류	숫자	숫자	인식오류
Type2 오류	숫자	기각	기각오류
Type3 오류	비숫자	숫자	

표 1과 같이 정의된 오류유형을 이용하여 어떤 숫자 인식기의 우수성을 논의할 경우에 몇 가지 고려 사항이 있다. 모든 오류유형이 최소화 되는 인식기가 최적의 인식기이겠지만, 일반적으로 그러한 인식기의 구현은 쉽지 않다. 어떠한 인식기는 특정 유형의 오류가 낮게 나타나지만 다른 유형의 오류가 높게 나타날 수도 있다. 따라서 인식기가 적용되는 환경에 따라 각 오류유형별로 가중치를 주어 인식기의 전체적인 성능을 평가하여야 할 것이다. 예를 들어 기각이 아무리 많더라도(Type2 오류가 아무리 높더라도) 인식 오류(Type1 오류)가 최소화되어야 하는 경우에는 Type1 오류에 보다 더 많은 가중치를 두어 인식기의 성능을 측정하여야 할 것이다. 본 논문에서는 그림 1, 그림 2, 그림 3에 나타난 각 MLP 인식기와 인식 방법에 대한 성능을 표 1에 정의된 세 가지 오류유형의 관점에서 살펴보고자 한다. 전체적인 인식 성능은 본 논문에서의 중요 분석 대상이 아니다.

### III. 실험 및 분석

#### 3.1. 실험 데이터 세트

본 논문에서는 실험을 위하여 4가지 종류의 영상 데이터 세트를 사용하였다. 학습을 위해서 11,690개의 숫자 영상으로 이루어진 데이터 세트와 9,113개의 비숫자 영상으로 구성된 데이터 세트를 사용하였다. 학습된 인식기의 테스트를 위해서 41,193개의 숫자 영상으로 구성된 데이터 세트와 4,709개의 비숫자 영상으로 구성된 데이터 세트를 사용하였다. 숫자 영상과 비숫자 영상의 예를 그림 4에 나타내었다. 비숫자 영상 데이터 세트는 필기체로 작성된 우리나라의 우편봉투에 존재하는 주소 문자열에서 숫자간의 접촉에 의해 숫자 분리에 실패한 영상과 한글문자 및 잘못 분리된 한글문자들로 구성되어 있다.

그림 4. 숫자 영상 및 비숫자 영상 예

Fig. 4. Sample images of numerals and non-numerals

#### 3.2. 오류유형별 실험

그림 1, 그림 2, 그림 3에 나타난 방법에서의 P, PN,

PN\_S, S를 오류역전과 알고리즘[7]을 이용하여 학습하였다. 학습시의 각 MLP 인식기에는 영상이 그대로 입력되지 않고 [8]에서 제시된 것과 같은 방향각도 특징 벡터가 입력된다. 인식기 테스트시에도 마찬가지로 방향각도 특징 벡터가 이용된다. 각 MLP 인식기는 하나의 중간층을 가지며 중간층 노드의 개수를 30, 50, 70, 90개로 달리하여 구현하였다. 학습시의 학습율은 0.1, 관성항은 0.7로 설정하였으며 학습 반복회수는 100회로 제한하였다.

인식기들의 학습을 완료한 후 다음과 같은 인식 규칙과 오류율식을 이용하여 Type1 오류유형에 대한 인식기별 성능을 분석하였다.

recognition rule : If  $\underset{j}{\operatorname{argmax}} O_j = \text{true class}, j=0, 1, \dots, 9$ , then the recognition result is correct.

Type1 error (%) = the number of misrecognized numerals  $\times 100$  / the total number of numeral inputs

인식 규칙에서  $O_j$ 는 MLP 출력층 노드 중 클래스 j에 해당하는 노드의 출력값을 의미한다.

표 2는 숫자 테스트 데이터 세트에 대한 Type1 오류율을 나타낸다. 모든 인식기에서 중간층 노드의 수가 많아질수록 Type1 오류가 감소한다. 각 인식기별로 성능을 비교해보면, 동일한 중간층 노드의 수를 가지는 인식기에서 숫자만 학습한 P의 오류율이 가장 낮으며, PN의 오류율이 PN\_S보다 낮다. 그러나 PN과 PN\_S의 중간층 노드의 수를 증가시키면 작은 규모의 P의 성과 유사하게 나타남을 알 수 있다. 약 6.4%의 오류율을 기준으로 보면 P의 경우 30개, PN은 50개 PN\_S의 경우 90개의 중간층 노드가 있음을 확인할 수 있다.

표 2. 숫자 테스트 데이터 세트에 대한 Type1 오류율(%)  
Table 2. Type1 error rates for the numeral test data set

인식기 \ 중간층 노드 수	30	50	70	90
P	6.44	5.48	5.04	4.73
PN	8.05	6.34	5.75	5.43
PN_S	8.77	8.02	6.88	6.41

다음 실험으로 Type2의 오류율을 분석하였으며 그 결과를 표 3에 나타내었다. Type2 오류는 기각 오류로써 숫자가 입력되었음에도 인식을 하지 못하고 기각을 하는 경우이며 다음과 같은 규칙과 계산식으로 오류율이 계산된다.

**rejection rule1** : If  $\max O_j < 0.5, j=0,1,\dots,9$ , then reject the input.

**rejection rule2** : If  $\max O_j < O_{10}, j=0,1,\dots,9$ , then reject the input.

Type2 error (%) = the number of rejected numerals $\times$ 100/the total number of numeral inputs

일반적으로 인식기는 인식기의 출력값을 기준으로 인식결정에 대한 확신도를 나타내며 이에 따라 기각 결정을 한다. 다양한 방법으로 기각 규칙을 정할 수도 있겠지만 이는 본 논문의 주요 분석 대상에 포함되지 않는다. 여기에서는 최대 출력값이 0.5보다 낮으면 입력을 기각한다. 이를 기각 규칙1이라 한다. 기각 규칙2는 PN\_S에 대해 적용된다. PN\_S에서 마지막 열 번째 노드가 비슷자 클래스에 해당하는 노드이다. 따라서 숫자에 해당하는 처음 10개의 출력층 노드의 출력값의 최고값이 기각 노드의 출력값 보다 작으면 기각한다. Type2 오류율은 전체 입력된 숫자 중에 기각된 숫자의 비율을 의미한다. 표 3에서 PN\_S2에 대한 오류율은 기각 규칙2를 적용했을 경우의 오류율이다. Type2 오류의 측면에서 P가 가장 우수한 성능을 보이고, PN이 PN\_S보다 대체적으로 좋은 성능을 보이며, PN\_S2가 가장 나쁜 성능을 보인다. P보다 나머지 세 개의 인식기가 Type2 오류율이 높은 것은 단순한 기각 규칙을 적용했을 경우 P보다 나머지 세 개의 인식기가 보다 신중하게 인식을 시도한다는 것을 의미한다.

표 3. 숫자 테스트 데이터 세트에 대한 Type2 오류율(%)  
Table 3. Type2 error rates for the numeral test data set

인식기 \ 중간층 노드 수	30	50	70	90
P	3.87	4.32	4.09	4.33
PN	22.77	20.11	21.48	14.62
PN_S	27.27	18.37	18.19	17.58
PN_S2	31.24	21.71	16.71	16.26

표 3의 Type2 오류율에 포함되지 않은 즉, 기각되지 않은 숫자에 대한 인식률을 살펴보는 것도 의미있을 것이다. 본 논문에서는 다음과 같이 Type1\* 오류율을 정의하여 인식기별로 오류율을 분석하였으며, 그 결과를 표 4에 나타내었다. 이 측정값들은 인식기의 인식 결정에 대한 신뢰도를 반영한다.

Type1\* error (%) = the number of misrecognized numerals $\times$ 100/the total number of not-rejected numeral inputs

표 4. 숫자 테스트 데이터 세트에 대한 Type1\* 오류율(%)

Table 4. Type1\* error rates for the numeral test data set

인식기 \ 중간층 노드 수	30	50	70	90
P	4.51	3.53	3.19	2.74
PN	2.24	1.74	1.32	1.71
PN_S	2.10	2.35	1.96	1.73

Type1\* 오류의 측면에서 PN이 가장 우수한 성능을 보였으며, P가 가장 좋지 않은 성능을 보였다. 이는 Type1 오류의 측면에서는 P가 가장 좋은 성능을 보인 것과 다른 특성이다. 즉, 기각을 하지 않고 인식 결정을 하였을 경우의 신뢰도는 PN이 가장 좋다는 것을 알 수 있다. 인식결과의 신뢰도가 중요한 전장표 처리, 우편물 구분 등의 경우에는 기각이 없을 때 최고 인식성능을 나타내는 인식기보다는 기각을 적정 수준으로 하더라도 신뢰도 높은 인식기가 보다 더 효과적이다. 논문에 나타내지 않았지만 기각 임계치를 0.5로 하지 않고 변경하여 실험하였을 때도 동일한 양상이 나타났다.

최종적인 Type1 오류와 Type2 오류 성능뿐만 아니라 실제적인 인식기의 출력값 분포를 분석하는 것도 인식기의 특성 분석에 매우 유용할 것이다. 이를 위하여 본 논문에서는 P, PN, PN\_S 모두 중간층의 노드 수가 90개인 것을 선정하였으며, 숫자 데이터 세트에서 추출된 200개의 숫자 데이터를 각 인식기에 입력하였으며 그 출력 특성을 그림 5에 나타내었다.

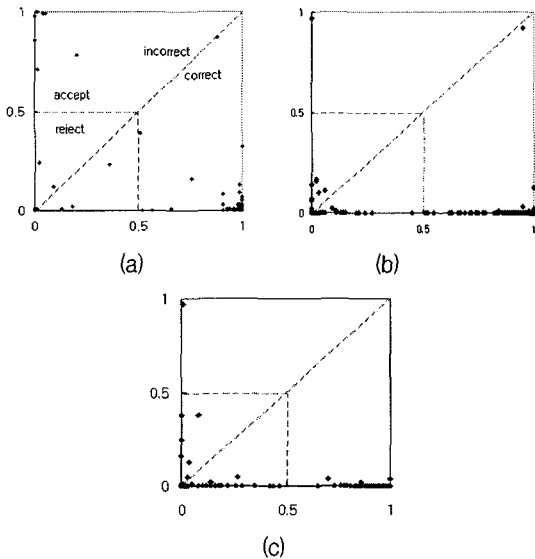


그림 5. 숫자 테스트 데이터 세트에 대한 (a) P, (b) PN, (c) PN\_S의 출력값 : x축-정클래스 노드의 출력값, y축-다른 클래스 노드 출력값들 중 최대값  
 Fig. 5. Out values of (a) P, (b) PN, and (c) PN\_S for the numeral test data set : x axis-out value of the true class, y axis-the maximum value among out values of remaining classes

그림 5에서 x축은 입력 숫자 클래스에 해당하는 출력층 노드의 값이고 y축은 다른 클래스 노드의 출력값들 중 최대값을 나타낸다. 따라서 그림의 우사선의 아래쪽에 위치한 점들은 올바르게 인식된 숫자들의 출력값이고 우사선의 위쪽에 위치한 점들은 잘못 인식된 숫자들의 출력값들을 의미한다. 점선으로 된 사각형 영역은 기각 임계치를 0.5로 하였을 경우의 기각 영역을 의미한다. 사각형 영역의 외부 영역은 인식 결정을 인정하는 영역이고 내부 영역은 인식을 거절하는 영역을 의미한다. 그림을 살펴보면 P와 PN, PN\_S의 출력값 분포 특성의 차이가 뚜렷하게 나타남을 알 수 있다. PN과 PN\_S의 경우 정클래스 이외의 다른 클래스의 출력값(y 축의 값)은 대부분 0.5이하에 위치하고 있으며 정클래스의 값들도 기각 영역에 다수 포함되어 있음을 알 수 있다. 특히 PN의 경우 정클래스 이외의 클래스에 해당하는 출력값들은 0과 인접한 곳에 집중 분포하고 있음을 주목할 만하다. 이러한 출력값 분포 특성을 이용하면 신뢰도 높은 인식결정을 할 수 있을 것이라 예상할 수 있다. 만약 인식 후처리기를 활용할 수

있는 경우라면 기각 영역에 포함된 입력들을 후처리 대상으로 선별하거나, 인식점수를 후처리기의 입력 요인으로 반영함으로써 인식률을 높일 수 있을 것이다. 이러한 분포 특성은 P에 대비한 PN의 장점이라 할 수 있을 것이다.

Type3 오류의 측면에서 실험한 결과를 표 5에 나타내었다. 이 오류는 비슷자 입력을 숫자인 것으로 잘못 받아들여 인식한 오류이며 다음의 인정 규칙과 오류율 계산식에 의해 구해진다. 인정 규칙2는 PN\_S의 마지막 노드의 출력값을 이용한 것이다.

**acceptance rule** : If  $\max O_j > 0.5, j=0,1,\dots,9$ , then accept the input.

**acceptance rule2** : If  $\max O_j > O_{10}, j=0,1,\dots,9$ , then accept the input.

Type3 error (%) = the number of accepted non-numerals  $\times 100 /$  the total number of non-numeral inputs

표 5. 비슷자 테스트 데이터 세트에 대한 Type3 오류율(%)

Table 5. Type3 error rates for the non-numeral test data set

인식기	중간층 노드 수			
	30	50	70	90
P	74.47	77.15	75.60	76.32
PN	1.85	2.55	1.85	2.78
PN_S	1.32	2.91	1.83	1.89
PN_S2	0.87	1.57	1.81	1.57

P의 Type3 오류율은 최소 약 74%이상의 오류를 보인다. 이는 비슷자 입력에 대해 학습하지 못한 인식기는 비슷자 입력에 대한 기각 신뢰도가 현저히 낮아진다는 것을 의미한다. 반면 PN과 PN\_S는 1%~3%내외의 낮은 오류율을 보여 안정적으로 기각하는 것으로 나타났다. 이러한 양상은 인정 규칙에서의 임계치를 변경하여 실험을 하였을 때도 비슷하게 나타났다. 비슷자 데이터 세트에 대해서도 각 인식기의 출력값 특성을 살펴보기 위해 최대 출력값과 차상위 출력값을 그림 6에 나타내었다. 각 인식기의 중간층 노드의 수는 90개이며 샘플 데이터는 200개이다.

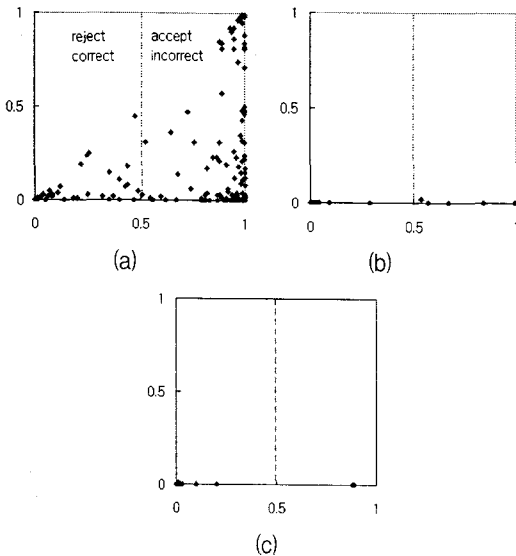


그림 6. 비슷자 테스트 데이터 세트에 대한 (a) P, (b) PN, (c) PN\_S의 출력값 : x축-최대 출력값, y축-차상위 출력값  
 Fig. 6. Out values of (a) P, (b) PN, and (c) PN\_S for the non-numeral test data set : x axis-the maximum out value, y axis-the runner-up out value

x축의 값이 0.5인 직선의 우측 영역에 존재하는 점들은 숫자로 인식된 것으로 Type3 오류가 발생한 것이고, 직선의 좌측 영역에 존재하는 점들은 숫자가 아닌 것으로 올바르게 기각된 것들이다. 그림을 살펴보면 P의 경우 상당수의 비슷자 입력이 숫자인 것으로 잘못 결정되었음을 알 수 있다. 반면 PN과 PN\_S의 경우 비슷자를 안정적으로 잘 기각함을 알 수 있다.

Type1 오류가 최적인 P의 장점을 취하고자 그림 2의 방법1에서와 같이 숫자와 비슷자만을 구분하는 S를 도입하는 경우는 어떠한 양상이 나타나는지 검토해볼 가치가 있다. 이는 비슷자가 존재하는 환경에서 S를 도입함에 따라 기존에 이미 구현된 숫자만을 대상으로 학습한 인식기 P를 활용하면 좋을 것이라라는 가정의 타당성을 검증하는 것이 될 것이다. 그림 3에 나타난 S의 Type2, Type3 오류율에 대한 실험 결과를 표 6에 나타내었다. 실험 결과 비슷자에 대한 오류는 적지만 숫자에 대한 오류는 매우 많다. 숫자인 것으로 올바르게 분류한 경우에 정확한 숫자 클래스를 인식할 수 있는지에 대해서도 실험해 보았다. S에서 바르게 분류된 숫자 입력을 P에 입력하여 인식한 결과 성능을

표 7에 나타내었다. 이 오류율과 표 4에 나타난 P의 Type1\* 오류율을 비교해보면 거의 유사하며, PN과 PN\_S보다는 나쁜 결과임을 알 수 있다. 이는 비슷자가 존재하는 환경에서 S를 도입하여 기존에 구현된 숫자 전용 인식기 P와 결합하는 그림 2의 인식방법1은 S를 사용하지 않고 단독으로 P를 사용하는 경우보다 못한 결과를 초래할 수 있다는 것을 의미한다.

전술한 바와 같은 오류들을 전체적으로 통합한 결과를 살펴보는 것도 의미가 있다. 인식기가 적용되는 환경에 따라 각 오류율의 중요성을 다르게 취급하는 것이 타당하므로 본 논문에서는 각 오류유형별로 다음과 같이 가중치를 설정하여 결과를 분석하고자 한다.

$$E_w = w_1E_1 + w_2E_2 + w_3E_3$$

여기서  $E_1$ 은 Type1\* 오류율,  $E_2$ 은 Type2 오류율,  $E_3$ 은 Type3 오류율이고,  $w_i$ 는 각 오류율에 대한 가중치이다. 또한  $w_1+w_2+w_3=1$ 이다. 만약  $w_1=w_2=w_3$  이면 평균 오류율이 되며,  $w_1=1$  이면 Type1\* 오류율이 곧 최종 오류율이 된다.

표 6. S의 Type2, Type3 오류율(%)  
 Table 6. Type2 and Type3 error rates of S

오류유형 \ 중간층 노드 수	30	50	70	90
Type2	30.77	28.31	30.29	31.44
Type3	1.76	1.74	1.57	1.53

표 7. 숫자 데이터 세트에 대한 S+P의 Type1\* 오류율(%)  
 Table 7. Type1\* error rate of S+P for the numeral test data set

오류유형 \ 중간층 노드 수	30	50	70	90
Type1*	4.42	3.62	3.19	3.03

그림 7에 가중화율을 다르게 했을 경우의 가중화 오류율에 대한 몇 가지 예를 나타내었다. 그림 7(a)에서 각 오류유형에 대한 가중치가  $(w_1, w_2, w_3) = (0, 0.5, 0.5)$ 로 각각 주어졌을 때 P의 오류율이 가장 낮다. 이는 Type1\* 오류를 무시한 경우로 인식 결정이 틀리더라도, 숫자를 기각하거나 비슷자를 숫자로 받아들이는

오류를 최소화하는 인식기는 **P**인 것을 의미한다. 오류 가중치가 (0.5, 0.0, 0.5)인 경우에는 숫자를 기각하는 비율이 높더라도, 즉, 이를 무시하고 숫자의 인식율과 비슷자를 제대로 걸러내는 비율이 높은 인식기는 **PN**임을 나타낸다. 오류 가중치가 (0.5, 0.5, 0.0)인 경우에는 비슷자를 숫자로 잘못 인식하는 것을 상관하지 않을 경우의 오류율은 **P**가 가장 좋음을 나타낸다. 그림 7-(b)에서는 세 가지 오류유형을 모두 고려하되 특정한 오류유형에 보다 큰 가중치를 주었을 때의 인식기 성능을 나타내고 있다. 숫자인식 신뢰도와 비슷자 기각율의 관점에서 **PN**이 가장 좋으며, 기각율의 관점에서는 **P**가 가장 좋게 나타났다.

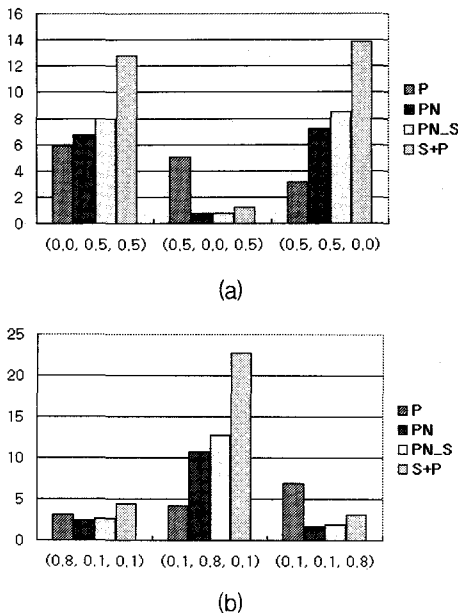


그림 7. 각 인식기의 가중화 오류율 : x축은 ( $w_1, w_2, w_3$ ), y축은 가중화 오류율(%)  
 Fig. 7. Weighted error rates of the recognizers : x axis-( $w_1, w_2, w_3$ ), y axis-weighted error rate(%)

#### IV. 결 론

본 논문에서는 숫자와 비슷자가 혼재하는 경우의 MLP 인식기의 구현방법에 대해서 논하였다. MLP 인식기를 세 가지 방법으로 구현하였으며, 세 가지의 오류유형을 정의하여 각 인식 방법의 인식 특성을 분석

하였다. 인식 실험은 약 67,000여자의 필기체 숫자와 비슷자를 이용하여 이루어졌다.

구현된 MLP 인식기는 숫자만을 학습한 **P**, 숫자와 비슷자를 학습한 **PN**, **PN**의 출력층에 비슷자 노드를 추가한 **PN\_S** 이다. 이들을 기본 인식기로 하여 **P**와 숫자와 비슷자를 구분하는 인식기인 **S**를 결합하여 인식하는 방법과 **PN**, **PN\_S** 단독으로 숫자와 비슷자를 인식하는 방법을 실험하였다.

논문에서 정의된 인식기의 오류유형은 숫자입력을 다른 숫자로 인식하는 Type1 오류, 숫자입력을 인식하지 못하고 기각하는 Type 2 오류, 비슷자입력을 숫자로 인식하는 Type3 오류이다.

본 논문의 실험 결과를 분석함으로써 다음과 같은 결론을 얻을 수 있다. 첫째, 비슷자 환경에서도 Type1 오류만을 최소화시키고자 하는 경우에는 기존에 숫자만으로 학습된 인식기 **P**를 사용하는 것이 더 적절하다. 둘째, **P**의 장점을 취하고자 숫자와 비슷자, 두 가지를 구분하는 새로운 인식기 **S**를 도입하더라도 **S**의 오류율이 높아 **P** 단독으로 사용하는 것보다 나쁜 결과를 초래할 가능성이 높다. 셋째, 입력을 기각하지 않고 인식결정을 내린 경우에는 숫자와 비슷자를 동시에 학습한 **PN**의 인식결정이 신뢰도가 가장 높으며, 출력값 특성도 후처리에 유리하게 나타난다. **PN**의 출력층에 비슷자를 담당하는 기각 노드를 새로이 추가한 **PN\_S**도 **PN**과 동작 특성이 유사하며 특별히 우수한 장점이 없다. 마지막으로, Type2와 Type3의 오류율이 낮아야 하는 경우에는 기존에 아무리 잘 구현된 숫자전용 인식기 **P**라도 비슷자를 포함하여 재학습을 하는 것이 보다 더 적절하다.

#### 참고문헌

- [1] S. Cho, "Neural-network classifiers for recognizing totally unconstrained handwritten numerals," *IEEE Transactions on Neural Networks*, vol. 8, pp. 43-53, 1997.
- [2] F. Kimura and M. Shridhar, "Handwritten numeral recognition based on multiple algorithms," *Pattern Recognition*, vol. 24, no. 10, pp. 976-983, 1991.
- [3] J. Cao, M. Ahmadi, and M. Shridhar, "Recognition of handwritten numerals with multiple feature and



- multistage classifier," *Pattern Recognition*, vol. 28, no. 2, pp. 153-160, 1995.
- [4] C. Y. Suen, C. Nadal, R. Legault, T. A. Mai, and L. Lam, "Computer recognition of unconstrained handwritten numerals," *Proc. of IEEE*, vol. 80, no. 7, pp. 1162-1180, 1992.
- [5] S.N. Srihari, E.J. Keubert, "Integration of hand-written address interpretation technology into the United States Postal Service remote computer reader system," *Proc. of 4th International Conference on Document Analysis and Recognition*, Ulm, Germany, pp.892-896, 1997.
- [6] R. Plamondon and S. N. Srihari, "On-line and off-line handwriting recognition: A comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, pp. 63-84, 2000.
- [7] D. E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing*, vol. 1, Eds. D. E. Rumelhart and J. L. McClelland, pp. 319-362, 1986.
- [8] 김호연, 임길택, 김두식, 남윤석, "서장 우편물 자동 처리를 위한 우편영상 인식 시스템," *정보처리학회 논문지*, Vol.10 No.4, pp.429-442, 2003.

## 저자소개

### 임길택(Kil-Taek Lim)



1993.2. 경북대학교 전자공학과  
(공학사)

1995.2 경북대학교 전자공학과  
(공학석사)

1999.2 경북대학교 전자공학과  
(공학박사)

1999.3 ~ 2004.2 한국전자통신연구원 우정기술연구  
센터 선임연구원

2004.3.~ 현재 경주대학교 컴퓨터멀티미디어공학부  
전임강사

※관심분야 : 패턴인식, 컴퓨터비전, 멀티미디어 신호  
처리, 신경망