

문자인식 시스템을 위한 신경망 입력패턴 생성에 관한 연구

A Study on Input Pattern Generation of Neural-Networks for Character Recognition

신명준*, 김성종**, 손영익***
(Myong-Jun Shin, Sung-Jong Kim, Young-Ik Son)

Abstract - The performances of neural network systems mainly depend on the kind and the number of input patterns for its training. Hence, the kind of input patterns as well as its number is very important for the character recognition system using back-propagation network. The more input patters are used, the better the system recognizes various characters. However, training is not always successful as the number of input patters increases. Moreover, there exists a limit to consider many input patterns of the recognition system for cursive script characters. In this paper, we present a new character recognition system using the back-propagation neural networks. By using an additional neural network, an input pattern generation method is provided for increasing the recognition ratio and a successful training. We firstly introduce the structure of the proposed system. Then, the character recognition system is investigated through some experiments.

Key Words : character recognition, neural-networks, input patterns, image processing, back-propagation

1. 서론

일반적으로 인공신경망은 입력층과 은닉층, 그리고 출력층으로 구분되며 이들을 구성하고 있는 뉴런들 사이의 연결강도를 최적화하는 것이 관건이다.

단순하거나 또는 소규모의 신경망은(예를 들면 2입력 XOR 연산을 수행하기 위해 입력층 2개, 은닉층 2개, 그리고 출력층이 1개의 뉴런으로 구성된 신경망)[1] 입력과 출력을 그래프로 표현하여 연결강도의 최적 여부 판단이 가능하지만 좀더 복잡한 신경망의 경우(일반적인 경우)에는 연산결과로 얻어진 연결강도가 최적인지 그 여부조차 알 수 없으며 잘못된 결과를 출력할 가능성이 있다.

또한 감독학습을 하는 역전파 신경망의 경우 프로그래밍을 통한 반복학습으로 최적의 연결강도를 찾게 되는데 오차의 제공함미 기준이하이거나 학습의 반복이 지정된 횟수를 넘기는 경우에 학습을 종료하게 된다.[1][2][3] 입력층과 출력층, 그리고 은닉층의 수가 많아질수록 또 교육시키고자하는 패턴의 종류가 다양할수록 오차를 줄이는 것이 어렵고 일정 기준 이하의 오차를 얻기 위해서 많은 횟수의 가중치수정이 필요하다.

본 논문에서는 한 번의 가중치 연산에 많은 시간을 소비하고 그 결과 또한 만족스러울 확률이 높지 않기 때문에 보다 효율적인 신경망 시스템의 설계를 위해 입력패턴을 최소화하고 패턴의 학습 횟수를 줄이는 방법에 대해 기술하고 있다.

2. 본론

2.1 기존 문자인식 시스템의 구조

신경망을 이용한 문자인식 시스템은 크게 영상처리 부분과 신경망 부분으로 나눌 수 있다. 그림 1은 이러한 시스템의 특성과 실험을 위해 구성한 시스템의 개념도에 대해 설명하고 있다. 신경망의 구성과 학습은 매트랩을 사용하였으며 영상처리 부분의 인터페이스는 C++Builder를 사용하였다.[4]

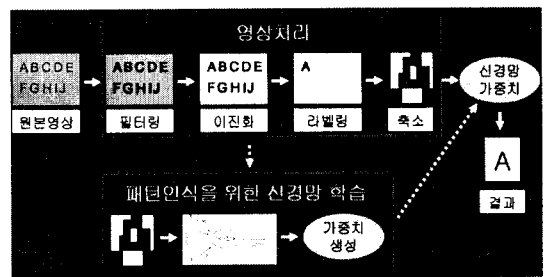


그림 1 문자인식 시스템의 구조 및 개념도

영상처리부분은 신경망 학습을 위해 샘플패턴을 만드는 과정(점선)과 실제 영상처리를 수행하는 과정(실선)으로 나뉜다. 최초 신경망 연결강도가 결정되고 나면 이후부터는 원본 영상의 추출부터 결과까지 영상처리를 수행하게 되며 인터페이스를 통해 진행과정을 모니터링 하도록 구성하였다.

참고문헌에[5][6] 인공 신경망을 사용하여 문자(알파벳)를 인식하는 시스템을 구현한 예가 있다. 본 논문에서 제안하는 신경망과 비교를 위해 입력층과 출력층 등을 일부 수정하여

* 辛明俊: 明知大學校 電氣工學科 碩士課程
** 金晟鍾: 明知大學校 電氣工學科 碩士課程
*** 孫瑛翼: 明知大學校 電氣工學科 助敎授·工博

새롭게 구성된 역전파 신경망을 그림 2에 나타내었다.

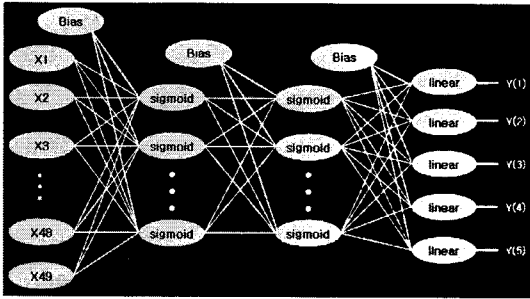


그림 2 문자인식 시스템의 신경망 구조

알파벳 대문자(A~Z)를 인식하기 위한 시스템이며, 한 문자당 49개의 픽셀을 사용한다. 입력층은 49개의 뉴런으로 1은닉층과 2은닉층은 각각 11개, 7개의 뉴런으로 구성하였으며 출력층은 5개의 뉴런을 가진다. 영상에서 라벨링 과정을 거쳐 추출한 문자는 신경망의 입력 패턴으로 사용되기 위해 7×7=49픽셀의 영상으로 축소하여 사용한다. 두 개의 은닉층은 반복 실험을 통하여 결정된 뉴런의 개수를 적용하고 출력은 5개의 뉴런으로 구성하였다. 이진수출력은 다음과 같이 구할 수 있으며 보다 자세한 내용은 그림 3에 있다.

$$\text{output} = Y(1) \times 2^4 + Y(2) \times 2^3 + Y(3) \times 2^2 + Y(4) \times 2^1 + Y(5) \times 2^0$$

(output=1이면 인식결과=A, output=2이면 인식결과=B, ..., output=26이면 인식결과=Z)

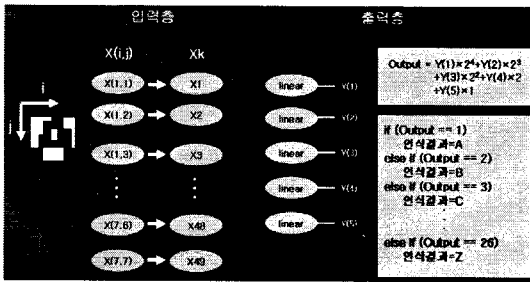


그림 3 문자인식 시스템의 입출력 구조

2.2 기존 문자인식 시스템의 실험결과

학습률은 0.1로 고정하고 반복횟수는 100000이내 대문자 A부터 Z까지 26개 문자에 대해 각각 20개의 패턴을 사용해 실험을 수행하였다. 입력패턴과 원하는 결과의 관계는 그림 4에 있다.

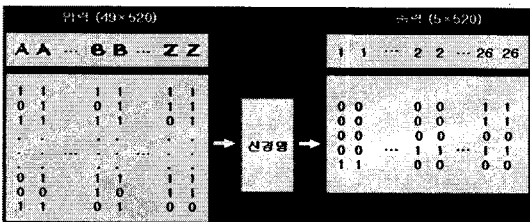


그림 4 패턴학습의 입출력 관계

문자의 축소영상은 7×7=49픽셀을 가지며 26개의 문자에 대해 20개의 패턴을 적용하였으므로 49×520의 행렬이 입력이 되고 출력은 5×520의 행렬이 된다.

학습에 사용되는 입력패턴의 종류와 수가 많아서 100000회 이내에 주어진 오차범위로 수렴하지 못하는 경우도 발생하였으며 산출된 연결강도를 이용하여 문자를 인식했을 경우 문자를 잘못 인식하는 경우가 발생하였다. 예를 들면 A(이진출력 0 0 0 0 1)를 E(이진출력 0 0 1 0 1)로 인식하는 문제가 발생하였다. 결과는 그림 5와 같은 C++Builder로 작성한 인터페이스를 통해 확인이 가능하다.

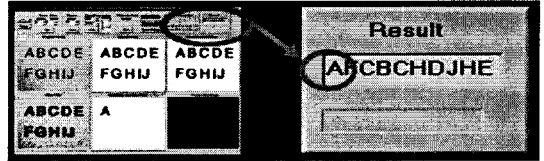


그림 5 문자인식 결과

2.3 새로운 문자인식 시스템의 구조

기존의 방법을 사용할 경우 학습에 사용한 데이터(학습데이터)에 대해서는 100%의 정확도를 보였으나 학습에 사용되지 않은 입력(비학습데이터)에 대해서는 약 70%의 정확도를 보였다. 입력패턴은 이진화 축소영상을 사용하므로 49개의 0과 1로 구성된 픽셀을 사용한다. 오차의 원인은 학습데이터에서는 1이지만 비학습데이터에서는 0인 값을 가지는 입력픽셀 또는 학습데이터에서는 0으로 사용되었지만 비학습데이터에서는 1의 값을 가지는 픽셀이 존재하기 때문이다.

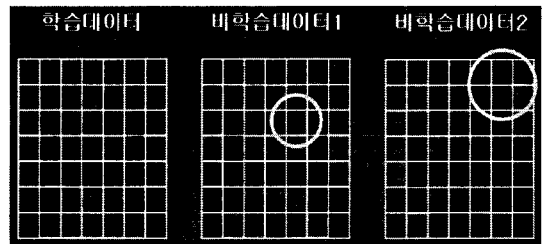


그림 6 오차 출력의 원인 분석

예를 들어 그림 6은 7×7픽셀로 축소한 문자 A의 그림이다. 학습에 사용된 데이터와 동일한 픽셀을 가진 문자가 실제 시스템의 입력 패턴으로 사용된다면 문자 A로 인식한다. 그러나 비학습데이터1과 같은 패턴이 시스템에 입력될 경우 오차가 발생할 수 있다. A(3,5)의 입력(A(3,5)는 A(3행,5열)을 의미)이 학습에서는 초기값 1로 사용되었으나 비학습데이터1에서는 0으로 사용되기 때문이다. A(3,5)가 A출력 또는 다른 문자에 영향을 미치는 중요한 픽셀이라면 비학습데이터1과 같은 패턴은 잘못된 결과를 출력하게 된다. 또, A(1,6)과 A(2,6) 픽셀은 학습과정에서는 초기값 0으로 사용되었지만 실제 시스템에서는 초기값 1로 사용되며 오차출력의 원인이 된다.

본 논문에서는 출력의 정확도를 높이기 위해 비학습데이터 패턴을 학습데이터 패턴과 일치시키고자 한다. 비학습데이터

를 학습데이터의 패턴과 일치시키기 위한 시스템이 그림7에 있다.

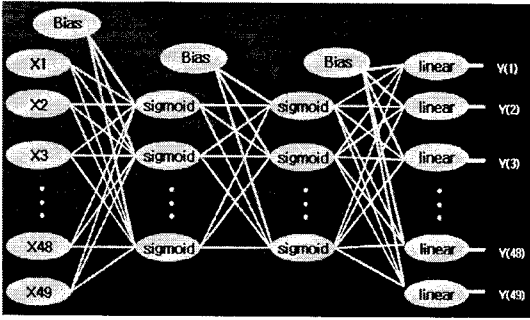


그림 7 입력패턴 생성시스템

입력패턴 생성시스템은 출력이 49개이고 이 시스템의 출력은 기존 문자입력 시스템의 입력패턴으로 사용된다. 이 시스템의 기능은 기존 시스템에 사용되는 패턴을 단일화시켜서 보다 정확한 출력을 만드는 것이다. 즉 시스템 출력이 오차를 가질 때 직렬로 연결된 또 다른 시스템의 학습과정을 통해서 그 오차를 최소화하는 것이다.

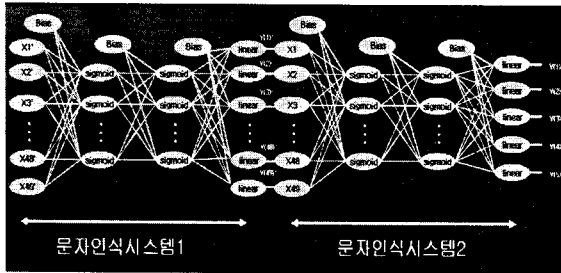


그림 8 문자인식 시스템의 연계

그림 8은 이러한 문자인식 시스템의 구조적 연계를 나타내고 있다. 문자인식시스템1은 영상으로부터 문자데이터를 받아서 원하는 패턴을 생성하는 기능이 있다. 이 시스템은 A(이진출력 0 0 0 0 1)를 E(이진출력 0 0 1 0 1)로 인식하는 기존 시스템의 오류처럼 목표출력은 부분적인 오차를 가진다. 이렇게 오차를 가진 각각의 1차 출력패턴은 다시 문자인식 시스템2(기존 문자인식 시스템)의 입력패턴으로 사용된다.

문자인식 시스템1과 2사이에는 목표값이 존재하기 때문에 시스템 전체에 역전파 알고리즘을 적용하지 않고 전반부와 후반부로 나누어서 학습을 수행한다.

2.4 실험결과

문자인식 시스템1에서 15개의 패턴으로 학습을 시킨 후 5개의 비학습데이터 입력에 대한 출력결과가 표 1에 있다. 이 데이터는 기준출력과 목표출력의 차가 0이 아닌, 즉 오차가 발생한 픽셀의 개수를 나타낸다. 기존의 문자시스템에서 인식이 낮았던 문자(M, N, R, S, T)에 대한 결과이며 이렇게 오차를 출력한 패턴은 문자인식시스템1의 학습데이터로 추가하여 최종 출력을 구한다.

비학습패턴 문자	1	2	3	4	5
M	0	1	4	0	0
N	0	0	1	3	5
R	0	1	1	0	0
S	1	1	1	0	0
U	0	0	0	1	0

표 1 비학습데이터의 출력 결과

실험결과 문자인식 시스템2(기존 문자인식 시스템)에 입력되는 패턴의 종류는 14개로 문자인식 시스템1이 없을 때 20개의 패턴을 사용하는 것보다 약 30% 적은 입력패턴을 사용하며 최종 인식률은 약 90%이다.

3. 결 론

다양한 입력패턴이 하나의 목표 출력으로 수렴하고자 할 때 입력패턴의 종류가 많을수록 오차가 크다. 최종단의 입력패턴 종류를 줄일 수 있다면 그만큼 최종출력이 정확해지기 때문에 기존 문자인식 시스템의 전단에 전처리기 시스템을 연계하여 입력패턴의 종류를 최소화하는 방법을 제안하였다. 제안하는 시스템을 기존 시스템에 추가함으로써 입력패턴의 수를 줄일 수 있었고 이에 따라 출력 오차가 기준 이내로 쉽게 수렴함을 알 수 있었다.

다소 복잡해 보일 수 있지만 각 문자인식 시스템의 구조는 동일하다. 시스템을 연계하기 위해서 문자인식 시스템1의 출력개수와 문자인식 시스템2의 입력개수를 동일하게 설정하는 과정이 필요하다.

감사의 글

본 연구는 과학기술부 / 한국과학재단
우수연구센터육성사업의 지원으로 수행되었음
(차세대전력기술연구센터)

참 고 문 헌

- [1] 한학용, "패턴인식개론", 한빛미디어, 2005
- [2] Martin T. Hagen, Howard B. Demuth, Mark Beale, "Neural Network Design", PWS Publishing Company, 2002
- [3] 임영도, 이상부, "퍼지 신경망 유전진화", 인솔미디어, 1996
- [4] 정태영, "볼랜드 C++ Builder 정복", 가남사, 1997
- [5] 이현엽, 문경일, "MATLAB을 이용한 퍼지-뉴로", 아진, 1999
- [6] 김진형, 조성배, 오세창, "문자인식에 있어서 기존의 방법과 Neural Network 방법의 비교분석", 한국 전자통신 연구소 최종보고서, 1990.6