

변형에 무관한 필기체 문자 인식을 위한 퍼지 신경망과 학습 알고리즘

유 정 수*

요약

본 논문에서는 퍼지 집합을 기반으로 한 새로운 신경망에 대해 기술하고 있다. 새로운 퍼지 신경망은 변형에 영향을 받지 않는 문자 인식에 적용하였다. 퍼지 신경망은 5개 층으로 구성되어 있다. 구현 결과 왜곡, 이동, 회전 및 필기체 문자의 크기가 서로 달라도 문자들을 정확하게 인식함을 보였다. 잡음(8~30%)이 있는 경우에도 정확하게 인식을 하였다. 이동, 왜곡, 서로 다른 문자 크기 및 잡음은 L2 층에 의해서 이루어졌으며, 회전에 영향을 받지 않게 하기 위해서 L5층을 구성하였다. 퍼지 신경망을 훈련하기 위해서 108개 문자를 사용하였으며 훈련 패턴에서 1- 또는 2-픽셀의 이동이 있는 경우에도 100%의 인식률을 보였다. $\pm 20^\circ$ 도 정도 회전된 문자인 경우에는 정확하게 인식하였다. 또한 제안된 퍼지 신경망은 학습된 문자인 경우 100% 인식률을 가지고 recall하였다. 제안된 퍼지 신경망은 구조가 간단하고 학습 속도와 recall 속도가 매우 빨랐다. 본 퍼지 신경망은 필기체 문자열의 분할과 인식에도 적용하였다.

A Novel Fuzzy Neural Network and Learning Algorithm for Invariant Handwritten Character Recognition

Jeong Su Yu*

Abstract

This paper presents a new neural network based on fuzzy set and its application to invariant character recognition. The fuzzy neural network consists of five layers. The results of simulation show that the network can recognize characters in the case of distortion, translation, rotation and different sizes of handwritten characters and even with noise(8~30%). Translation, distortion, different sizes and noise are achieved by layer L2 and rotation invariant by layer L5. The network can recognize 108 examples of training with 100% recognition rate when they are shifted in eight directions by 1 pixel and 2 pixels. Also, the network can recognize all the distorted characters with 100% recognition rate. The simulations show that the test patterns cover a $\pm 20^\circ$ range of rotation correctly. The proposed network can also recall correctly all the learned characters with 100% recognition rate. The proposed network is simple and its learning and recall speeds are very fast. This network also works for the segmentation and recognition of handwritten characters.

1. 서론

문자 인식은 패턴 인식의 한 분야로 시각 정보를 통하여 문자를 인식하고 나아가 그 의미를 이해할

수 있는 능력을 컴퓨터로 하여금 실현하려는 시도가 다. 문자 인식에 관한 연구는 오래 전부터 이루어졌으며 오늘날까지도 활발히 진행되고 있다.

최근에는 신경망을 이용하여 변형에 무관한 패턴 인식에 관한 연구가 이루어지고 있으며 연구 결과가 가장 유망한 분류기로 알려져 있다. Fukushima는 입

* 종신회원 : 전주교육대학교 실과교육과 교수

력 패턴의 위치, 크기 및 이동과 변형에 별로 영향을 받지 않는 neocognition을 개발하였고 필기체 문자를 인식하는데도 적용되었다. 인식 결과는 좋은 편이나 시스템이 너무 복잡하고 많은 뉴런들을 필요로 하는 단점을 지니고 있다[1]. 그리고 Pittman은 backpropagation 알고리즘을 필기체 숫자와 문자를 인식하는데 사용하여 좋은 인식률을 보였으나 훈련 과정에서 많은 시간이 요구되는 단점이 있다. 예를 들어, 한자나 한글을 인식하는 경우와 같이 많은 클래스를 포함하고 있는 인식 문제에 있어서는 특히 학습 시간이 주요 장애가 되고 있다[2].

인간의 생각을 모델링하기 위한 강력한 도구로 사용되고 있는 퍼지 논리(fuzzy logic)를 최근 일부 연구에서는 패턴 인식에서의 공통적인 문제인 불확실성(uncertainty)을 다루는데 적용하고 있다[3]. 입력 패턴의 고유의 특징들이 종속적이거나 클래스 밀도 함수들을 가정할 수 없을 때 퍼지 방법들은 특히 유용하다[5, 6]. 그러나 기존연구들에서는 입력 패턴의 클래스를 분류하는데 알고리즘의 파라메타 값을 조정해야 하는 단점이 있다. 파라메타 값에 따라 동일한 문자라도 다르게 인식되는 경우가 있다[3].

따라서, 본 논문에서는 퍼지 정보를 사용하여 전처리 과정을 거치지 않고 문자의 크기, 이동, 회전 및 잡음에 전혀 영향을 받지 않는 새로운 신경망 구조와 학습 알고리즘을 개발하였다. 제안된 퍼지 신경망은 5-층 전 방향(feedforward) 구조로 구성되었으며 3번째가 4번째 층의 뉴런들은 학습하는 동안 스스로 구성(self-organization)된다. 제안된 학습 알고리즘의 속도는 기존의 알고리즘 등에 비해 매우 빠르며 왜곡된 패턴에 대해서도 정확하게 인식한다.

본 논문은 다음과 같이 구성되었다. 2절은 문자 인식을 위해서 제안된 퍼지 신경망 구조와 학습 알고리즘에 대해서 설명한다. 3절에서는 2절에서 제안한 모델을 입력 문자의 이동, 회전, 크기, 손상 및 잡음 등의 변형이 가해진 여러 형태의 필기체 숫자와 영문자에 적용한 실험 결과를 보여 준다. 마지막으로 4절에서는 본 논문의 결론을 맺는다.

2. 새로운 퍼지 신경망과 학습 알고리즘

제안된 새로운 퍼지 신경망은 5개 처리 층으로 이루어 졌으며 구조는 그림1과 같다.

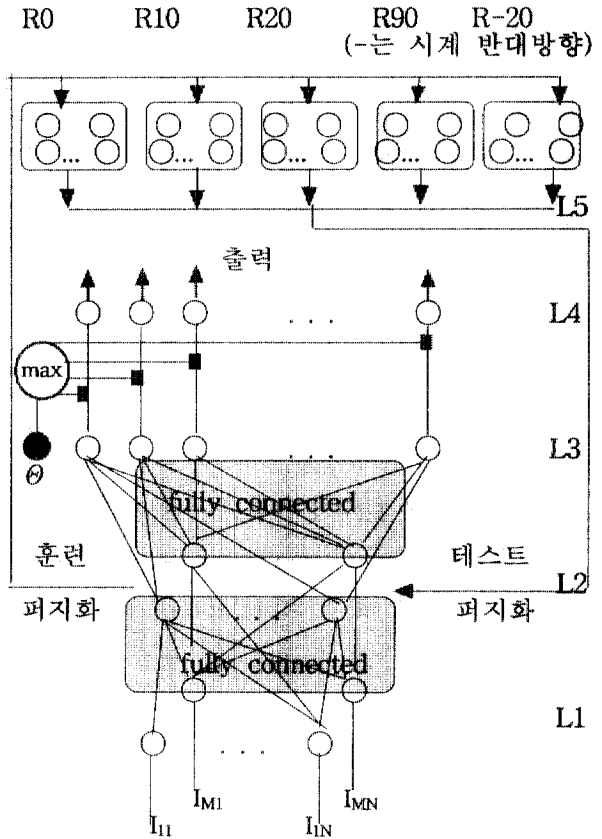


그림 1 새로운 퍼지 신경망 구조

첫 번째 층은 입력 층으로 M x N 2차원 2진(0 또는 1) 입력 패턴을 받아들인다. (i, j)번째 입력 뉴런의 state와 출력은 식(1)과 식(2)와 같다. I_{ij} 는 (i, j) 번째 입력 패턴을 의미한다. s^1_{ij} 는 L1에서 (i, j)번째 상태를 의미하여 y^1_{ij} 는 L1에서 (i, j)번째 출력이 다. y^1_{ij} 는 [0,1]의 값을 갖는다.

$$s^1_{ij} = I_{ij}, i = 1, \dots, M, j = 1, \dots, N \quad (1)$$

$$y^1_{ij} = s^1_{ij}, i = 1, \dots, M, j = 1, \dots, N \quad (2)$$

두 번째 층, L2는 입력 층과 동일하게 M x N개의 뉴런으로 구성되었다. L2는 서로 다른 이동 위치와 서로 다른 문자 크기에 도 문자를 인식할 수 있는 능력을 지니고 있다. 새로운 망은 self-scaling 성질을

갖고 있다. L2 층의 목적은 입력 패턴들을 가중치 함수, w_{mn} 을 이용하여 퍼지화하는 것이다(식3). 따라서, w_{mn} 을 퍼지화 함수라 한다.

가중치는 문자의 회전과 문자의 크기 및 이동에 무관한 구성요소(primitive)를 얻기 위해 사용된다. L2의 뉴런들은 렌즈 역할을 한다. 각 뉴런은 입력 패턴의 한 개 픽셀에 초점을 맞추고 중심이 되는 픽셀의 주위 픽셀들을 본다. 이를 위해서 입력 패턴의 중심으로부터 값이 퍼지도록 β 값을 설정해야 한다(그림 2). 만일 β 값이 너무 작으면 신경망은 서로 다른 문자들을 구별 할 수 없다. 반대로 β 값이 너무 크면 문자가 이동, 왜곡되었을 경우 인식을 할 수 없다. 따라서, 실제로 실험을 통해서 β 값에 따라 입력 문자들이 학습되는 과정을 살펴 볼 수 있다.

퍼지 신경망의 L2에서 퍼지화하기 위해서 s^2_{pq} 는 입력 문자를 non-boolean quantitative 로 해석한다(그림 3). s^2_{pq} 는 일치 정도의 퍼지 추정을 하기 위해서 사용된다.

$$w_{mn} = e^{-\beta^2(m^2 + n^2)}, \quad m=1, \dots, M, n=1, \dots, N \quad (3)$$

$$s^2_{pq} = \max_{i=1}^M (\max_{j=1}^N (w_{p-i, q-j} y_{ij}^1)), \quad p=1, \dots, M, q=1, \dots, N \quad (4)$$

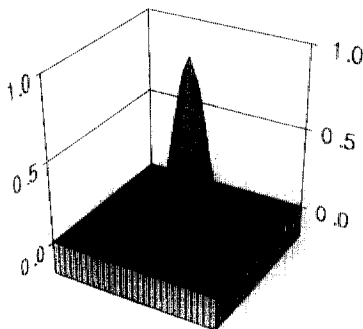


그림 2 가중치 함수($\beta=0.3$)



그림 3 L2의 상태, s^2_{pq}

L2 층에 있는 각 뉴런은 M개의 서로 다른 출력을 갖는다. M은 세 번째 층의 뉴런의 개수이다. 각 뉴런은 세 번째 뉴런중에 한 개의 뉴런에 지정된다. L2의 출력은 다음과 같다.

$$y^2_{pqm} = G_{fcn}(s^2_{pq}), \quad p=1, \dots, M, q=1, \dots, N, m=1, \dots, NF \quad (5)$$

$$G_{fcn}(s^2_{pq}) = \frac{1}{1 + \left[\left(\frac{s^2_{pq} - \Theta_{pqm}}{a} \right)^2 \right]^r}, \quad m=1, \dots, NF, p=1, \dots, M, q=1, \dots, N \quad (6)$$

여기서, Θ_{pqm} 은 G_{fcn} 의 중심점을 의미한다. G_{fcn} 은 적합 파라메타, Θ_{pqm} , a , r 들을 포함하고 있는 bell 함수이다. 각 파라메타들의 물리적 의미는 다음과 같다. Θ_{pqm} 은 소속성 함수와 일치하는 중심을 결정한다. Θ_{pqm} 은 입력 패턴들의 특징을 추출한다. 따라서, 두 번째 층은 특징 점을 추출하는 곳이다. a 는 인식을 요구하는 입력 패턴이 훈련 패턴에 대해서 어느 정도의 크기와 이동을 허용하는 파라메타이다. 따라서 a 값에 따라 동일한 패턴이라도 다르게 인식되기도 한다. 예를 들어, 훈련 패턴에 대해서 3-픽셀 정도 이동된 패턴을 동일한 패턴으로 인식하기 위해서는 a 값을 6정도로 설정하면 된다.

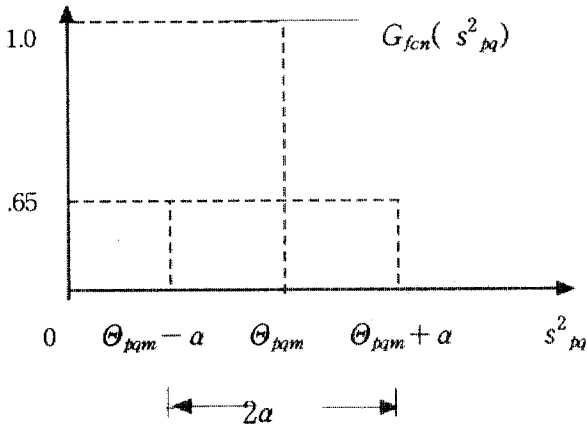


그림 4 벨 소속성 함수에서 파라메타들의 물리적 의미

NP는 훈련 패턴의 수를 의미한다. α , γ 는 crossover point에서 기울기를 제어한다. α , γ 값에 따라 동일한 문자가 다른 문자로 인식되기도 한다 (그림 4).

벨 함수는 학습 알고리즘에 의해서 결정된다. NP는 전체 훈련 패턴 수이다. L2에 있는 각 뉴런은 M개의 서로 다른 출력을 갖는다. L2의 뉴런은 특징점 추출 망과 유사하다.

세 번째 층은 한 번 학습된 패턴을 나타낸다. 따라서, 세 번째 층의 뉴런의 개수는 학습이 모두 끝난 후에 결정된다. 세 번째 층의 뉴런들의 상태와 출력은 다음과 같다.

$$y_m^3 = s_m^3 = \min_{p=1}^M (\min_{q=1}^N (y_{pqm}^2)) \quad (7)$$

$$y_m^3 = \begin{cases} \min_{pq} (1 - \frac{2}{\alpha} |s_{pq}^2 - \theta_{pqm}|) \\ , \max_{pq} (|s_{pq}^2 - \theta_{pqm}|) \leq \frac{\alpha}{2} \\ 0 \quad , \text{그밖의 경우} \end{cases} \quad (8)$$

$m = 1, \dots, NP$

식(7)은 한 번 학습된 패턴을 나타내며, 식(8)은 이미 학습된 모든 패턴들과의 유사성을 비교하는 것으로 학습된 패턴인 경우는 1의 값을 갖는다.

네 번째 층은 세 번째 층의 출력 y_m^3 값중에서 최대치를 갖는 뉴런을 활성화한다. 네 번째 층의 뉴런들의 상태와 출력은 식(9), (10)과 같다.

$$s_m^4 = y_m^3 \quad (9)$$

$$y_m^4 = G_{fcn}(s_m^4) = \begin{cases} 0 & s_m^4 < \max_{m=1}^{NP} (y_m^3) \\ 1 & s_m^4 = \max_{m=1}^{NP} (y_m^3) \end{cases} \quad (10)$$

마지막 층은 회전된 패턴을 인식하기 위한 것으로 훈련 과정에서 얻어진 θ_{pqm} 값을 가지고 있다. $\pm 10^\circ$, $\pm 20^\circ$ 로 회전하여 기울어진 패턴에 대해서는 다섯 번째 층이 없어도 인식이 가능하나 그 이상인 경우에는 인식률이 현저히 떨어진다. 따라서 이를 보완하기 위해서 본 층을 두었다. 다섯 번째 층은 새로운 패턴을 인식할 때 적용된다.

위의 식들을 사용한 학습 알고리즘은 다음과 같다.

Training()

```
{
  NP=1;
  for(k=0; k<K; k++) { /*K 전체 training 패턴의 수*/
    for(p=0; p<M; p++) {
      for(q=0; q<N; q++) {
        식(1);
        식(2);
      }
    }
    for(m=-(M-1); m<M; m++)
      for(n=-(N-1); n<N; n++)
        식(3);
    for(p=0; p<16; p++)
      for(q=0; q<16; q++)
        식(4);
    for(m=1; m<=NP; m++)
      for(p=0; p<M; p++)
        for(q=0; q<N; q++)
          식(5);
  }
}
```

```

for(m=1; m<=NP; m++) 식(8); 식(9);
for(m=1; m<=NP; m++) 식(10);
σ=1-max; /* max는  $y_m^3$  값중에서 최대치*/
if(σ> Tf) step3(); /*Tf는 허용 오차 */
}
}

step3()
{
NP++;
for(p=0; p<M; p++)
for(q=0; q<N; q++)
 $\Theta_{pqm} = S_{pq}^2$ 
for(m=1; m<=NP; m++)
for(p=0; p<M; p++)
for(q=0; q<N; q++)
 $y_{pqm}^2 = G_{fcn}(S_{pq}^2)$ 
for(m=1; m<=NP; m++) 식(7); 식(9);
for(m=1; m<=NP; m++) 식(10);
}
/* 마우스, 스캐너 및 파일에 의한 새로운 패턴 인식
과정 */
Recognizer()
{
/* 새로운 문자에 대해서 */
for(p=0; p<M; p++)
for(q=0; q<N; q++)
식(1); 식(2);
for(r=0; r<5; r++) { /* L5 */
for(m=1; m<=NP; m++)
for(i=0; i<M; i++)
for(j=0; j<N; j++)
식(4); 식(5);
for(m=1; m<=NP; m++) {
식(8); 식(9);
for(m=1; m<=NP; m++)
식(10);
}
}
}

```

max(y_m^3)을 갖는 뉴런 m을 결과로 출력한다.
}

3. 실험 결과

본 절에서는 새로운 퍼지 신경망을 필기체 문자 인식 문제에 적용했다.

제안된 모델을 평가하기 위해서 마우스를 사용하여 입력한 0에서 9까지의 숫자와 알파벳 대문자 26개를 훈련 패턴으로 사용하였다. 훈련 패턴의 크기는 16 x 16으로 하였다. 테스트 패턴으로 훈련 패턴을 8방향으로 1-, 2-픽셀 이동시킨 영숫자 180(그림 5) 문자와 스캐너를 통해 300dpi로 입력된 이진 문자를 전 처리 단계를 거치지 않고 실제의 입력 패턴을 그대로 사용하였다. 또한 마우스를 사용하여 입력한 필기체 문자도 사용하였다(그림 6).

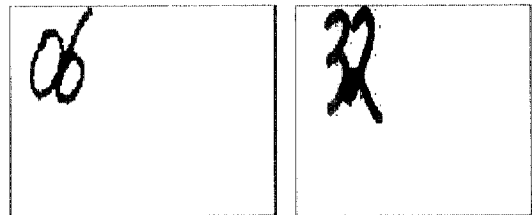


그림 6 스캐너를 통해 입력한 이진 필기체 문자

3.1 새로운 퍼지 신경망을 통한 이동과 크기에 무관한 인식

L2는 새로운 신경망에서 문자의 크기와 이동에 영향을 받지 않게 하기 위해서 사용되었다. 36개 훈련 문자들은 1-픽셀과 2-픽셀씩 8 방향(U, D, L, R, UL, UR, DR, DL)으로 이동하였다. 그림 7은 테스트 패턴 숫자 6을 8방향으로 이동한 예이다.

576문자를 가지고 테스트하였다. 학습 알고리즘에서 사용되는 파라미터는 2-픽셀 까지 이동된 패턴을 인식하기 위해서 $\alpha=4$, $\beta=0.3$, 그리고 $\gamma=0.65$ 로 하였다. 실험 결과 8방향으로 1-, 2-픽셀 이동된 문자들은 완벽하게 100%인식을 하였다.

표 1과 표2는 1-픽셀과 2-픽셀 이동된 테스트 패

턴들을 Kwan[4]과 Hamming net와 제안된 퍼지 신경망과의 인식률을 비교한 것이다.

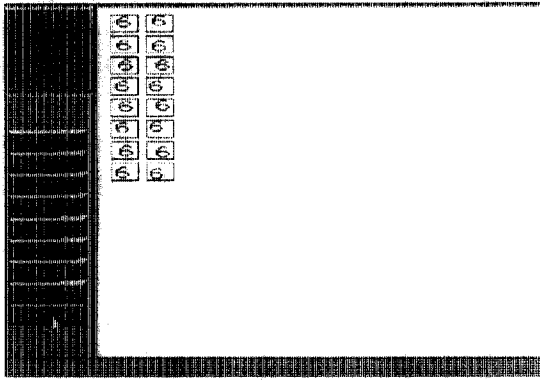


그림 7 1-픽셀과 2-픽셀씩 8방향으로 이동된 문자의 예(U: upward, D: downward, R: right, L: left, UR: up-right, UL: up-left, DR: down-right, DL:down-left)

표 1 1-픽셀 이동 패턴의 인식 결과
 $\alpha=4, \beta=0.3, \gamma=0.65$

R	L	U	D	UR	UL	DR	DL	평균
100	100	100	100	100	100	100	100	100 (*)
47.22	44.44	44.44	47.22	11.11	8.33	5.56	5.56	26.74 (**)
100	100	100	100	100	100	100	100	100 (***)

(*) : 참고문헌[4]
 (**) : Hamming net
 (***) : 제안된 퍼지 신경망

표 2 2-픽셀 이동 패턴의 인식 결과
 $\alpha=4, \beta=0.3, \gamma=0.65$

R	L	U	D	UR	UL	DR	DL	평균
100	100	91.67	91.67	88.88	91.67	77.77	94.44	92.01(*)
5.56	11.11	22.22	27.78	2.78	2.78	5.56	2.78	10.07(**)
100	100	100	100	100	100	100	100	100 (***)

(*) : 참고문헌[4]
 (**) : Hamming net
 (***) : 제안된 퍼지 신경망

패턴의 크기가 서로 다른 경우(훈련 패턴의 3배 또는 0.5배 범위)에 3개 모델 모두 100% 인식률을 보였다.

3.2 퍼지 신경망을 통한 잡음, 왜곡 및 회전에 무관한 인식

문자가 깨졌거나 찌그러진, 즉 왜곡된 문자에 대해서도 좋은 인식률을 보였다. 그림 8은 왜곡이 있는 문자의 인식 결과이다. 왜곡된 패턴의 경우 Hamming net은 43.89%정도의 인식률을 보였으며 [4]와 제안된 퍼지 신경망은 100%의 인식률을 보였다.

8에서 30%의 잡음이 있는 문자의 경우에도 100% 인식률을 보였다. 30%이상의 잡음이 있는 경우에는 인식률이 현저히 떨어졌다. 그림 9는 잡음이 있는 문자의 인식 결과이다.

L5층은 회전된 문자들을 인식하기 위해서 필요한 층이다. 회전을 위한 훈련 패턴들의 집합은 원래의 36개 패턴을 $\pm 10^\circ, \pm 20^\circ, \pm 90^\circ$ 회전을 하여 얻었다. 그림 5의 4행부터가 회전된 패턴들이다. 실험 결과 $\pm 10^\circ, \pm 20^\circ$ 정도 회전된 문자에 대해서는 $\alpha=2.0, \beta=0.3$, 그리고 $\gamma=0.65$ 로 다섯째 층이 없이도 완벽하게 인식하나 그 이상에 대해서는 오인 식이 있었다. $\pm 20^\circ$ 도 이상 회전된 문자에 대해서는 인식 과정에서 L5층을 사용하는 경우에 인식률이 현저히 증가 하였다. 그림 10은 회전된 문자의 인식 결과이다.

3.3 새로운 퍼지 신경망을 필기체 문자의 분할과 인식을 결합한 시스템에 적용

새로운 퍼지 신경망은 필기체 문자열을 인식하는데 성공적으로 적용하였다. 분할과 인식 과정은 고정된 위치에서 동시에 수행된다. 단일 망에서 32x16 이진 입력 문자로부터 분할 오퍼레이션을 수행한 다

음 마지막으로 인식 단계로 학습된다. 분할 과정은 참고 문헌 [4]의 방법을 사용하였다. $\alpha=2.0$, $\beta=0.3$, 그리고 $\gamma=0.65$ 를 가지고 분할과 인식을 수행한 결과는 그림 11과 같다.

4. 결론

본 논문에서는 퍼지 집합 이론을 적용하여 변형이 가해진 문자들을 인식 할 수 있는 새로운 신경망과 알고리즘을 개발하였다. 학습 후에 100%의 인식을 보였다.

실험 결과 문자의 왜곡, 이동, 회전 및 크기가 다른 필기체 문자에 대해서도 문자들을 정확하게 인식함을 보였다. 또한, 30%의 잡음이 있는 경우에도 인식을 잘 하였다.

기존의 인식 시스템들에 비해 시스템이 복잡하지 않고 학습 속도가 매우 빨랐다. 필기체 문자의 분할과 인식을 결합에도 좋은 결과를 얻었다.

참 고 문 헌

[1] K. Fukushima and N. Wake, "Handwritten alphanumeric character recognition by neocognition, " IEEE Trans. Neural Networks, vol. 2, no. 3, pp.355-365, May, 1991.

[2] G. L. Martin and J.A. Pittman, "Recognizing hand-printed letters and digits using backpropagating learning," Neural Computation, vol. 3, no. 2, pp258-267, Summer,1991.

[3] H. K. Kwan and Y. Cai,"A Fuzzy Neural Network Network and its Application to Pattern Recognition," IEEE Trans. on Fuzzy Systems, vol.2 , no. 3, pp185-192, August, 1994.

[4] O. Martin, Cristoper J. C. Burges, Yann LeCun and John S. Denker,"Multi-Digit Recognition Using A Space Displacement Neural Network, " Neural Information Processing System 4, pp488-495.

[5] Jeong Su Yu, "An Integrated Segmentation

and Recognition of Handwritten Characters using a New Fuzzy Neural Network," Proceedings of The Fourth International Workshop on Rought Sets, Fuzzy Sets, and Machine Discovery, pp307-314, Nov. 6-8, 1996.

[6] 유정수, 이원돈, "새로운 퍼지 신경망과 학습 알고리즘을 변형이 가해진 문자 인식에 적용, " 한국정보과학회 추계학술대회, pp - , 1996.

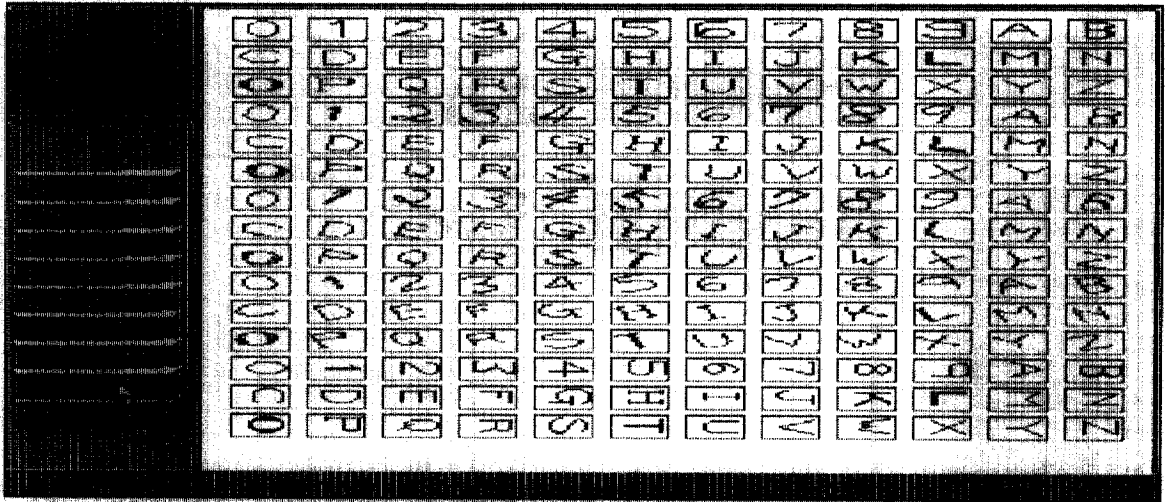


그림 5 변형된 문자 인식을 위한 훈련 패턴들과 테스트 패턴들

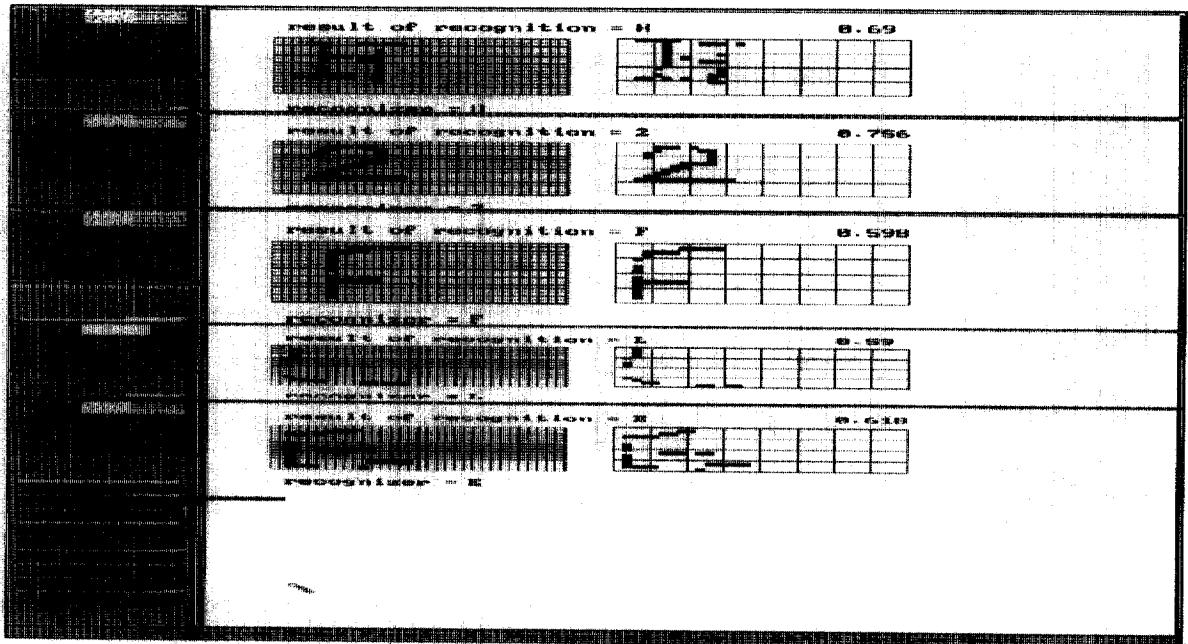


그림 8 문자가 깨져있거나 일그러진 문자의 인식 결과

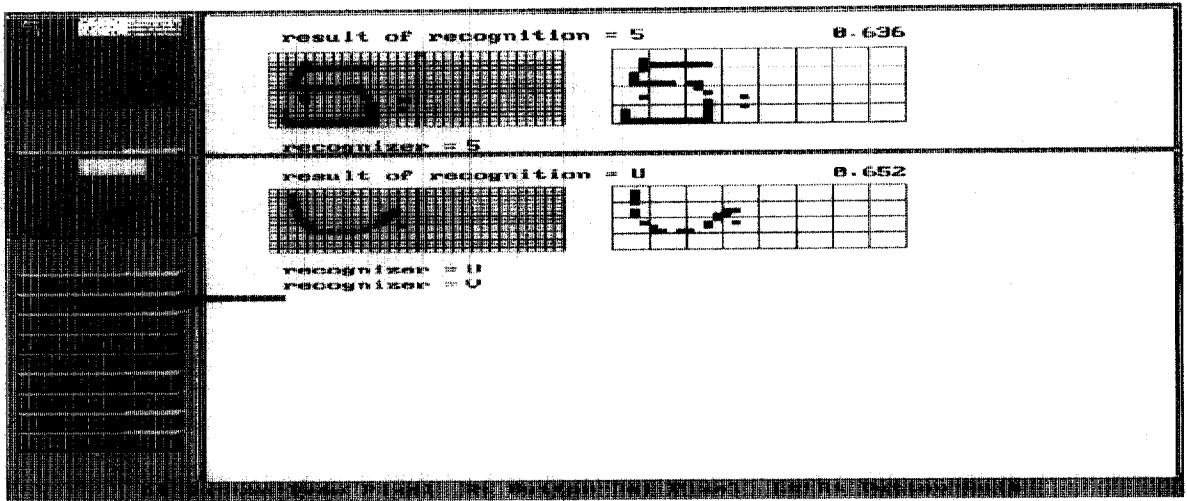


그림 9 잡음이 있는 문자의 인식 결과(8~30%)

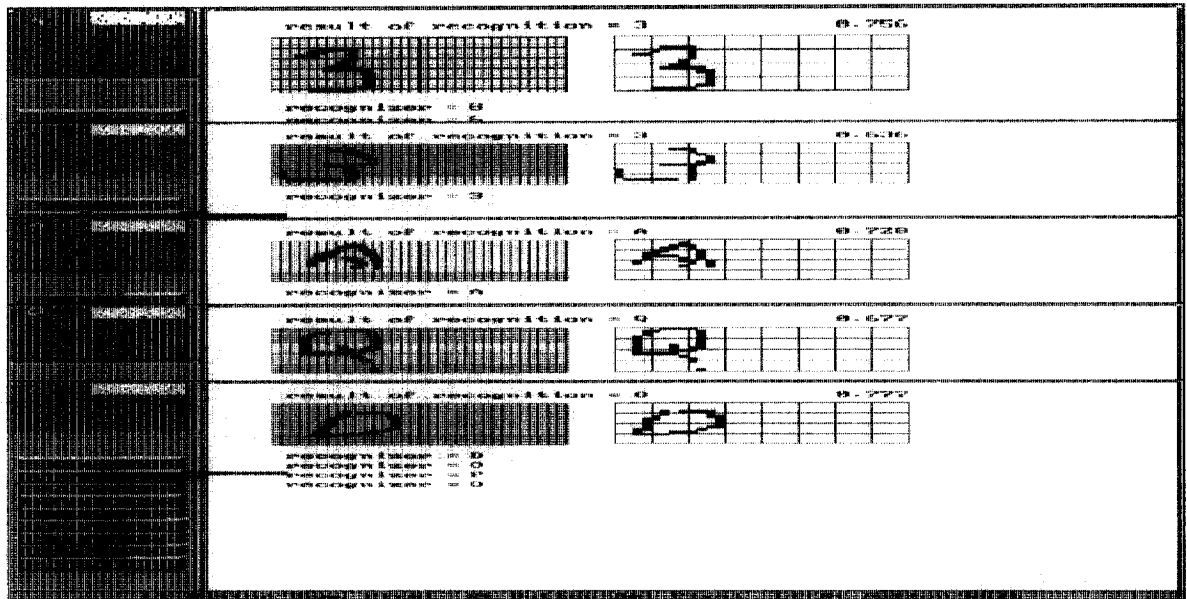


그림 10 회전된 문자들의 인식 결과

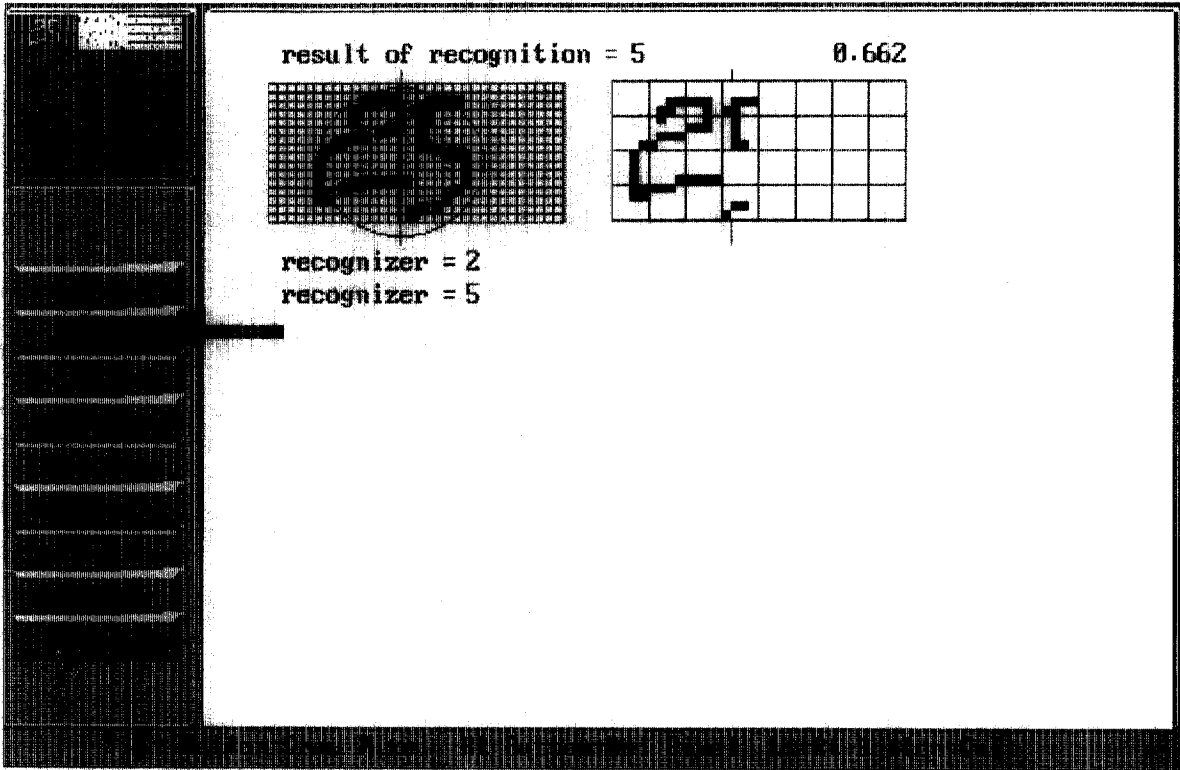


그림 11 필기체 문자의 분할과 인식을 결합한 시스템의 인식 결과