다중 동적 위상보존 자기구성 지도의 결합을 통한 필기숫자 데이타의 분류율 향상

(Improvement of Classification Rate of Handwritten Digits by Combining Multiple Dynamic Topology-Preserving Self-Organizing Maps)

김 현 돈 † 조 성 배 ††

(Hyun-Don Kim) (Sung-Bae Cho)

요 약 자기 구성지도는 데이타 시각화, 위상보존 매핑 등의 분야에서 널리 사용되고 있지만, 학습이 되기 전에 위상을 미리 고정시켜야 하기 때문에 실제 문제에 적용하기 어렵다는 것과 클러스터링 능력에 비해 분류율이 낮다는 결점이 있다. 이를 해결하기 위해서 자기구성 지도의 출력 노드를 동적으로 분화하고 분화된 노드를 학습하는 동적 위상보존 자기구성 지도를 제안하고, 이를 다중 결합함으로써 분류율을 향상시켰다. 동적 위상보존 자기구성 지도의 결합 방법으로는 자기구성 지도의 K개 노드가 출력을 내도록하는 K-Winner 방법 및 K-Winner+가중치 방법이 제안되었는데, 이는 다수결 투표, 가중치, BKS, Bayesian, Borda, Condorect, 신뢰값 합산 등의 기존 결합 방법보다도 우수한 결과를 나타내었다. 동적 위상보존 자기구성 지도를 통해서 위상을 고정 시켜야 하는 결점을 해결할 수 있었고, 서로 다른 특징으로 학습된 동적 위상보존 자기구성 지도들을 결합하여 분류 능력을 향상시킬 수 있었다. 필기 숫자데이타로 실험한 결과, 제안한 방법이 자기구성 지도의 결점을 효과적으로 해결하여 98.1%의 높은 인식률을 보였다.

Abstract Although the self-organizing map (SOM) is widely utilized in such fields of data visualization and topology preserving mapping, since it should have the topology fixed before trained, it has some shortcomings that it is difficult to apply it to practical problems, and classification capability is quite low despite better clustering performance. To overcome these points, this paper proposes the dynamic topology preserving self-organizing map(DTSOM) that dynamically splits the ouput nodes on the map and trains them, and attempts to improve the classification capability by combining multiple DTSOMs. K-Winner method has been applied to combine DTSOMs, which produces K outputs with winner node selection method. This produces even better performance than the conventional combining methods such as majority voting, weighting, BKS, Bayesian, Borda, Condorect and reliability sum. DTSOM remedies the shortcoming of determining the topology in advance, and the classification rate increases significantly by combining multiple maps trained with different features. Experimental results with handwritten digit recognition indicate that the proposed method works out the problems of conventional SOM effectively so to improve the classification rate to 98.1%.

1. 서 론

자기구성 지도는 비교사 학습 방법을 통해서 자신을 스스로 학습시키고, 지도의 구조가 위상을 보존하는 특성이 있다. 자기구성 지도는 이러한 특성 때문에 비교사 학습이나, 데이타 시각화(visualization) 및 위상보존 매핑(topology-preserving mapping)이 필요한 분야에서 이용되고 있다. 그러나 학습이 되기 전에 위상을 미리 고정시켜야 하기 때문에 실제 문제에 적용하기 어렵다

· 이 연구는 과학기술부가 지원하는 뇌과학연구 프로그램에 의하여 지원받은 것임.

† 비 회 원 : 연세대학교 컴퓨터과학과
neoace@candy.yonsei.ac.kr
†† 종신회원 : 연세대학교 컴퓨터과학과 교수
sbcho@cs.yonsei.ac.kr

논문접수 : 2001년 1월 3일
심사완료 : 2001년 8월 24일

는 것과 클러스터링 능력에 비해 분류율이 낮다는 문제점을 안고 있다. 이러한 문제점들을 해결하기 위해서 동적 위상보존 자기구성 지도, K-Winner 방법, 다중 신경망의 결합 방법 등을 제안한다.

고정된 위상 구조에 대한 결점을 해결하기 위해 동적으로 구조를 변화시키고자하는 연구가 수행되어 왔는데, 트리 형태로 노드를 확장하는 방식이나 점증적으로 구조를 변화시키는 방법 등이 있다[1]. 트리 형태의 노드 확장 방식에서 학습은 하위 네트워크의 노드들에 대해서만 일어나고, 따라서 하위 네트워크의 노드들의 순서가 지역적으로 보존된다. 반면에, 구조를 변화시키는 방법에서는 모든 노드들이 학습에 관여하므로 모든 노드들의 순서를 반영할 수 있다.

본 논문에서는 점증적으로 구조를 변화시키는 방식을 통해 자기구성 지도의 구조를 변화시키는 방법을 사용하였다. Fritzke는 지도의 구조를 표현하는데 용이한 사각형 구조를 유지하기 위해 하나의 행이나 열을 삽입하여 노드의 위상 구조를 변화시키는 방법을 사용하였다. 그러나, 이 경우는 실제 문제에 필요한 노드보다 많은 노드가 추가되는 단점이 있다[2]. 따라서, 필요한 노드만을 분화시키는 동적 노드 분화 방법이 필요하다. 본 논문에서는 LVQ의 교사 학습 방법과 SOM의 비교사 학습 방법을 결합하여 위상을 보존하면서 데이타에 따라 적절하게 그 구조를 변화시키는 동적 위상보존 자기구성 신경망을 제안한다.

또한, 위상보존 특성을 이용한 KW(K-Winner) 방법을 제안한다. 일반적으로 자기구성 지도는 가장 큰 하나의 노드를 출력하지만, KW방법의 경우 K개의 노드 값을 출력한다. K개의 출력을 비교하여 다수의 노드가 출력으로 하는 값을 최종적인 출력 값으로 한다. K개의 출력은 첫 번째 승리자 노드와 그 노드로부터 가까운 거리에 있는 이웃들 중 승리자에 가까운 K-1개로 구성된다. 자기구성 지도가 위상보존 특성에 의하여 적당한 K에 대해서는 유사한 결과를 내기 때문에 K개의 출력이 의미를 가질 수 있다. 이 방법을 통해 하나의 출력을 내는 동적 위상보존 자기구성 지도보다 높은 분류율을 얻을 수 있을 것이다.

동적 위상보존 자기구성 지도의 분류율 향상을 위해서 서로 다른 특징으로 학습된 신경망들을 결합하여 그 결과를 살펴보았다. 결합 방법으로는 추상 레벨(단순한 투표, KW 투표, BKS 방법, 가중치 투표, KW 가중치 투표), 순위 레벨(Borda함수, Condorect 함수), 측정치 레벨(신뢰도 합산) 등의 방법을 이용하였다. 여기서 KW 투표, 가중치 투표, KW 가중치 투표 방법은 본 논문에서 제안한 방법이다. 먼저, KW 투표는 각 동적 위상보존 자기구성 지도에서 K개의 승리자 노드를 얻어서 투표에 참여시키는 방법이다. 그리고 가중치 투표는 동적 위상보존 자기구성 지도의 각 노드에 정답에 대한 신뢰지수를 가중치로 부여하여, 가중치의 합을 통해서 결과를 얻는 것이다. KW 가중치 투표는 KW에 대해서 가중치를 계산하는 방법이다. 이러한 결합 방법을 이용하여 오프라인 필기 숫자 데이타에 대한 실험을 하였는데 상당히 향상된 성능을 얻을 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 자기구성 지도에 관해서 전반적으로 설명하고, 위상보존의 평가기준, 다중 신경망 결합 등에 대해 살펴본다. 3장에서는 본 논문에서 제안하는 동적 위상보존 자기구성 지도 신경망의 결과로 K개의 승리자 노드를 산출하는 KW 방법을 살펴본다. 4장에서는 다중 동적 위상보존 자기구성 지도의 결합에 대해서 설명하고 5장에서 실험을 통해 제안한 방법들의 유용성을 검증한다.

2. 관련연구

2.1 자기구성 신경망

지도 학습방법을 사용하는 신경망은 입력값과 그에 상응하는 출력값을 가지고 학습을 수행한다. 학습은 각 입력값에 대해 올바른 출력값이 나올 수 있도록 가중치를 변화시키는 것이다. 신경망은 생물학적 근거에 기반을 두고 있지만, 학습 도중에 올바른 출력값이 주어져야 한다는 것은 이러한 생물학적 의미에 맞지 않다. 튜보 코호넨(Teuvo Kohonen)이 제안한 자기 구성 지도는 어떻게 신경망이 자기 스스로 그 구조를 생성해 낼 수 있는 지에 대한 답을 비교적 간단하게 제시해 주고 있다[3]. 자기 구성이라는 말은 신경망이 주어진 입력에 대해 올바른 출력값이 제공되지 않아도 학습됨을 의미한다. 뿐만 아니라, 반응하는 순서나 위치를 통해 위상을 보존한다는 특성을 가지고 있다.

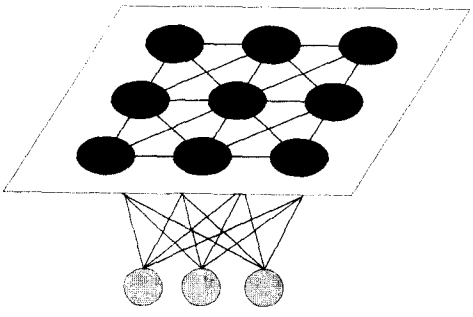

그림 1 자기구성 지도의 구조

자기구성 지도는 그림 1에서 보는 바와 같이 입력층과 출력층, 두 개의 층으로 구성되어 있다. 출력층은 N개의 노드로 구성되며 각 노드는 입력 벡터와 같은 차원을 가지는 가중치 벡터로 구성되어있다. 신경망의 각 노드는 가중치 벡터에 임의의 실수를 할당함으로써 초기화된다. 이 후, 신경망에 하나의 입력 벡터를 가하고, 신경망의 노드들로부터의 거리값을 구한 다음, 이로부터 가중치 벡터를 갱신하는 세 단계로 학습이 이루어진다.

입력 벡터가 신경망에 들어오면, 입력 벡터와 모든 노드들과의 유클리드 거리(Euclidian distance)를 계산한다. 최소 거리를 갖는 노드가 승자로 선택되는데 다음 수식에서처럼 m_c로 나타낼 수 있다.

$$\|x-m_c\| = \min_i \{\|x-m_i\|\} \quad (1)$$

그 다음에 승리자 노드를 중심으로 이웃 노드들과 승리자 노드에 대한 가중치 벡터 갱신이 일어난다. 그 과정은 다음의 수식과 같다.

$$m_i(t+1) = m_i(t) + \alpha(t) \times n_{ci}(t) \times \{x(t) - m_i(t)\} \quad (2)$$

여기서 $\alpha(t)$는 학습률을 나타내는 함수, $n_{ci}(t)$는 이웃 함수, $m_i(t)$는 노드의 가중치, $x(t)$는 입력 벡터 값이다 [3, 4]. $n_{ci}(t)$에서 c는 승리자 노드의 인덱스이다. 일반적으로 학습률과 이웃 함수는 학습이 반복되면서 감소되어야 한다.

2.2 위상보존 평가척도

자기구성 지도의 가장 대표적인 특징은 위상보존이다. 위상보존이란 입력 벡터의 위상 구조를 출력층의 노드들이 유클리드 거리에 따라 반영함을 의미한다. 즉, 유사한 벡터들이 근접하게 위치하여 위상학적 구조가 반영되는 것이다. 이러한 위상보존의 척도에 관한 연구는 다음과 같다.

Topographic 오류[5]는 입력 벡터 공간의 연속성을 계산하는 기준이다. 입력 벡터 x에 대해서 x의 가장 가까운 출력층의 가중치 벡터를 w_i라고 하고, 두 번째로 가까운 가중치 벡터를 w_j라고 하자. 이 때, 입력 벡터 공간의 벡터들 중, x와 w_i 값 사이의 모든 입력 벡터들이 w_i 에 매핑 되고, 나머지 값들이 w_j에 매핑 될 때, w_i와 w_j에 해당하는 노드들이 근접하면 벡터 x에 대해서 위상이 연속하고, 그렇지 않은 경우 위상은 불연속하게 되어 오류가 발생하게 된다. 시각 t에서의 Topographic 오류를 수식으로 표현하면 식 3과 같다.

$$\varepsilon_t = \frac{1}{N} \sum_{k=1}^{N} u(x_k) \quad (3)$$

여기서, N은 입력 공간의 샘플 벡터의 개수이며, $u(x_k)$값은 위상이 연속할 경우 1이 되고 연속하지 않을 경우 0이 된다.

Topographic 함수[6]는 근접한 수용 영역을 가진 출력층 노드들 중에서 실제 두 노드 사이의 거리가 특정 s 값 이상 되는 경우를 오류로 설정하고 그 오류 값을 누적시킨 값이다. 여기서 수용 영역 R_i 는 $V_i \cap M$ 과 같다. 여기서 M은 입력 스페이스를 나타내고 V_i는 Voronoi 다면체를 말한다. Topographic 함수는 다음 식 4와 같이 표현된다.

$$\Phi_L^M(s) = \sum_{\substack{i \in L, \|n_i - n_j\| > \\ s,\ n_i와\ n_j는\ 서로\ 이웃함}} \#\{ n_i | i \in L, \|n_i - n_j\| > \} \quad (4)$$

여기서, #은 집합의 원소의 개수를 의미하고, L은 지도상의 노드들의 인덱스 집합이다.

Koenig의 위상보존 평가 함수[7]는 n개의 최근접 이웃(nearest neighbor), NN_{ji} ($i \in [1,n]$, $j \in [1,N]$)를 이용한다. 여기서 N은 노드의 수를 말한다. 입력 공간 X와 출력 공간 Y에 대하여 입력 벡터 v_j에 대한 위상보존은 다음과 같이 평가된다.

$$\begin{cases} 3, & \text{if } NN_{ji} \text{ on X} = NN_{ji} \text{ on Y} \\ 2, & \text{if } NN_{ji} \text{ on X} = NN_{jl} \text{ on Y} \quad l \in [1,n] \quad i \neq l \\ 1, & \text{if } NN_{ji} \text{ on X} = NN_{jt} \text{ on Y} \quad t \in [n,k] \quad n < k \\ 0, & \text{else.} \end{cases} \quad (5)$$

이러한 방식으로 입력 벡터 v_j에 대한 평가값 q_{mj}를 구하여 합한 q_m이 위상보존의 정도를 나타낸다. q_m은 다음과 같다.

$$q_m = \frac{1}{3n \times N} \sum_{j=1}^{N} q_{mj} \quad (6)$$

완벽한 위상보존이 이루어 졌을 때, q_m 값은 1.0이 된다. 그리고 일반적으로 n은 4로 k는 10으로 둔다. 본 논문에서는 이 척도를 통해서 위상보존 정도를 평가하였다.

2.3 다중 신경망의 결합

하나의 신경망으로 풀리지 않았던 문제들이 여러 개 신경망의 결과를 결합하였을 때 해결되는 경우가 많기 때문에 다중 신경망의 결합에 대한 연구가 활발히 진행되어 왔다. 신경망의 결합은 하나의 문제를 여러 신경망이 풀고 그 결과를 결합하는 방식과 하나의 문제를 하위 문제로 나누어 각각의 하위 문제를 신경망들이 나누어 해결하는 방식이다. 그러나, 일반적으로 신경망의 결합은 하나의 문제에 대해서 여러 신경망이 같은 문제에 접근하여 그 결과를 결합하는 연구를 말한다. 이러한 신경망의 결합은 결합 방법에 따라 세 가지 레벨로 나눌 수 있는데, 각각 추상 레벨, 순위 레벨, 측정치 레벨이다[8].

추상 레벨은 각 인식기가 하나의 인식 결과만을 출력하는 방식인데, 대표적인 방법으로는 다수결 방법[9], BKS 방법, 베이시안 방법[10, 11] 등이 있다. 이 방법의 경우, 제공되는 정보가 적기 때문에 인식기의 과거 성향들이 결합에 크게 관여하게 된다. 순위 레벨[12]은

각각의 인식기가 분류에 참여하는 모든 클래스의 순위를 결과로 출력하여 모든 클래스들의 순위를 비교하여 결과를 산출하는 방식이다. 이 방법은 보다 많은 정보를 바탕으로 결합을 수행하고, 과거의 인식 성향이 결합에는 거의 사용되지 않는다는 특성이 있다. Borda 함수, 가중화 Borda함수, Condorect 함수를 통한 결합이 순위 레벨에 속한다. 측정치 레벨의 경우, 각 인식기가 클래스들에 대한 순위와 더불어 각 순위에 대한 신뢰값까지 함께 출력하는 방식이기 때문에, 가장 많은 정보로 결합을 수행한다. 그러나, 각 인식기 별로 제공되는 신뢰값이 일관성을 유지해야 한다는 어려움이 있다. 대표적인 방법으로 신뢰값 합산이 있다.

표 1 다중 신경망의 결합 방법

레벨	장점	단점
추상 레벨	인식기들의 성향이 결합에 반영됨	인식기들이 제공하는 정보의 양이 적음
순위 레벨	충분한 정보를 결합에 이용할 수 있음	인식기들의 과거 인식성향이 반영되지 않음
측정치 레 벨	충분한 정보를 결합에 이용할 수 있음	신뢰값의 일관성을 유지하는 것이 어려움

3. 동적 위상보존 자기구성 지도

이 장에서는 위상 구조를 미리 고정해야 하는 자기구성 지도의 단점을 해결하기 위해 학습 중에 스스로 구조를 변화시키는 동적 위상보존 자기구성 지도에 대해서 살펴보고, 위상보존 특성을 이용한 K-Winner 방법에 대해서 살펴본다. 동적 위상보존 자기구성 지도의 구조도는 그림 2와 같다.

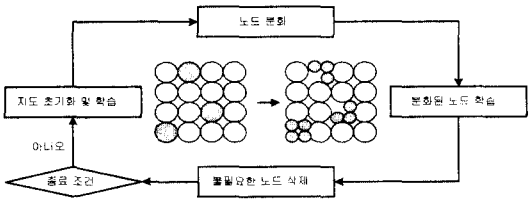

그림 2 동적 위상보존 자기구성 지도

3.1 동적 위상보존 자기구성 지도

이 모델은 표 2의 알고리즘을 기본으로 하여 완성되었다. 이 알고리즘은 각각 지도 초기화 및 초기 학습 단계, 노드 분화 단계, 분화된 노드 학습 단계의 세 부분으로 나뉘어진다[13]. 여기서 두 가지 학습이 필요한데,

표 2 DTSOM의 기본 알고리즘

① 지도를 4×4 크기로 초기화한다.
② SOM 알고리즘으로 학습시킨다.
③ 지도의 노드들 중 여러 클래스의 데이타가 섞인 노드를 찾는다.
④ 찾아낸 노드들을 2×2 크기의 노드로 분화시킨다.
⑤ 분화된 노드들을 LVQ 알고리즘으로 학습시킨다.
⑥ 분화된 노드들 중, 학습에 참여하지 않는 노드를 삭제한다.
⑦ ③~⑥의 과정을 종료 조건이 만족될 때까지 반복한다.

첫 번째는 일반적인 SOM 알고리즘을 이용하여 학습하는 것이고, 두 번째는 교사 학습 방법을 혼합한 LVQ 방식의 학습이다.

3.1.1 지도 초기화 및 초기학습

이 단계에서는 지도의 크기를 임의로 설정하여 초기화하고 코호넨 알고리즘에 의해서 학습시킨다. 지도는 4×4의 크기로부터 시작하여 2.1 절에 나온 수식에 의해 학습된다.

3.1.2 노드 분화

이 단계는 초기화된 지도를 토대로 분화되어야 할 노드를 찾아내는 역할을 한다. 분화되어야 할 노드를 찾아내는 방법은 다음과 같다. 먼저 지도의 모든 노드들에 대해서 최상의 매칭 클래스를 구한다. 초기에는 대부분, 하나 이상의 최상의 매칭 클래스를 가지는 노드가 발생할 것이다. 노드가 하나의 클래스에 대해서 반응하는 것이 아니라 다수의 클래스에 대해서 반응하게 되면 잘못된 결과를 산출하게 된다. 그러므로 이러한 노드들을 찾아서 분화시킨다.

이 모델에서는 분화되어야 할 노드를 찾아내기 위해서 hit ratio 값을 이용하는데, 이것은 i번째 노드에서 빈도수가 가장 높게 매칭되는 클래스의 빈도수를 i번째 노드에 매칭되는 클래스들의 빈도수 합으로 나눈 것이다.

여기에서는, hit ratio가 100% 미만을 나타내는 노드들을 찾아서 분화시키는 방식을 사용하였다. 분화되어야 할 노드들은 2×2 의 노드로 분화시킨다. 이때, 분화된 하위 노드들의 가중치는 분화되기 전 부모 노드의 가중치 값을 기반으로 이웃한 노드들의 가중치 값들을 고려하여 다음의 식과 같이 산출하였다.

$$C=\frac{(P\times 2)-\sum N_C}{S} \tag{7}$$

여기서, C는 자식 노드의 가중치, P는 부모 노드의 가중치, N_c는 자식 노드의 이웃 노드의 가중치, S는 (N_c의 개수+2)를 나타낸다. 즉, C는 이웃노드와 부모 노드의 평균값으로 결정된다.

3.1.3 분화된 노드 학습

일단 분화가 일어나면 그 노드들에 대해서 학습을 시켜야 한다. 이 단계에서의 학습은 순수한 코호넨 알고리즘에

LVQ 알고리즘을 결합한 형태의 교사 학습이다. 학습식은 다음과 같다.

$$m_i(t+1) = m_i(t) + \alpha(t) \times n_{ci}(t) \times h_{ci}(t) \times \{x(t) - m_i(t)\} \quad (8)$$

여기서 $h_{ci}(t)$는 다음과 같이 계산된다.

$$\begin{cases} h_{ci}(t)=1, & \text{if } x(t)\text{가 이전 단계 학습에서 } m_i(t)\text{나 } m_i(t)\text{의 부모 노드에 할당된 경우} \\ h_{ci}(t)=0, & \text{if } x(t)\text{가 이전 단계 학습에서 } m_i(t)\text{나 } m_i(t)\text{의 부모 노드에 할당 안된 경우} \end{cases} \quad (9)$$

원래 LVQ 알고리즘은 최상 매칭 클래스에 대해서는 가중치를 증가시키고, 그 이외에는 가중치 값을 오히려 감소시켜서(즉, $h_{ci}(t) = -1$), 학습이 빠르게 진행되도록 하였다. 그러나, 이 경우 오히려 최적의 해를 찾지 못하는 경우가 발생할 수 있기 때문에 위에서 보는 것과 같이 패배 노드들에 대해서는 학습을 시키지 않았다. 또한 기존의 LVQ 알고리즘에서는 이웃 함수($n_{ci}(t)$)가 없이 전체의 지도에 대해서 학습을 시킨다. 그러나, 본 논문에서는 이웃 함수를 사용하여 가까이 있는 최상 매칭 클래스들에 대해서 학습을 시켰는데, 이것은 자기구성 지도의 특성인 위상보존이 손상되는 것을 방지하기 위해서이다. 또한, 과도한 노드 분화를 막고 시간을 향상시키기 위하여 노드가 임계치 이상의 깊이 만큼은 분화되지 않도록 설정하였다.

일단 위의 학습 알고리즘으로 학습시킨 후, 불필요한 노드를 삭제한다. 삭제 평가 기준 역시 hit ratio를 이용한다. 분화와는 달리 노드 삭제의 경우에는 불필요한 노드를 찾아야 하므로 hit ratio가 0, 즉 어떤 데이타에 대해서도 승리자로 할당되지 못한 노드를 삭제한다. 그러나, 무조건 hit ratio가 0인 노드를 삭제하면, 다음과 같은 번거로움이 발생할 수 있다. 다음 단계 학습에서 삭제된 노드를 필요로 하면, 그 노드가 존재하지 않으므로 다시 주변 노드를 분화하여 그 노드와 비슷한 가중치를 가지도록 학습시킨다. 이 경우, 하나의 노드가 필요해서 다시 3개의 노드를 더 삽입한 결과가 된다. 이러한 잘못된 삭제를 피하기 위해서 각 노드마다 age를 두었다. age는 hit ratio가 0이어도 삭제되지 않는 횟수를 말한다. 즉, age가 3일 경우, 3번까지는 hit ratio가 0이어도 그 노드를 삭제하지 않는다.

3.2 K-Winner 방법

이 방법은 최종 승리자를 구할 때, 자기구성 지도의 가장 큰 특성인 위상보존을 이용한 것이다. 일반적으로 자기구성 지도는 하나의 노드를 출력으로 가진다. 그러나, 만약 신경망이 비슷한 신뢰도를 갖는 여러 노드를 결과로 낼 수 있다면 더 나은 결과를 보일 수 있을 것이다. KW방법의 경우 K개의 노드를 출력으로 가진다. 자기구성 지도는 위상보존 특성에 의하여 적당한 K에 대해서는 유사한 결과를 내기 때문에 K개의 출력이 의미를 가질 수 있다. K개의 출력은 첫 번째 승리자 노드와 그 노드로부터 가까운 거리에 있는 이웃들 중 승리자에 가까운 K-1개로 구성된다. 일단 K개의 결과가 나오면, 다수결의 원칙을 적용하여 최종 결과를 산출하였다. 예를 들어, K=5일 경우 입력 패턴이 들어오면 자기구성 지도의 출력층에서 승자 노드 1개를 결정한 후, 그 노드에 가장 가까운 4개의 이웃 노드를 추가적으로 승자 노드로 선택한다. 이렇게 선택된 승자 노드들의 레이블값들로부터 다수결의 원칙에 의거하여 입력 데이타의 최종 클래스를 결정하게 되는 것이다. 일반적인 자기구성 신경망은 사각형의 위상 구조를 가질 경우 4개의 노드를 최근접 이웃으로 가지고, 육각형 위상 구조를 가질 경우 6개의 노드를 최근접 이웃으로 가진다. 그러므로 K값은 이러한 최근접 이웃들의 개수를 기반으로 하여 정할 수 있다.

4. 다중 동적 위상보존 자기구성 지도의 결합

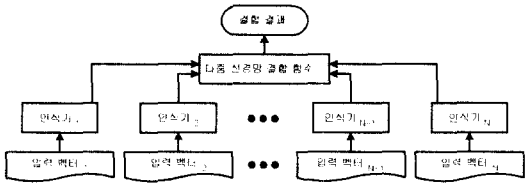

그림 3 다중 신경망의 결합

실세계의 문제를 해결하기 위해 신경망을 사용하려다 보면, 하나의 신경망으로 해결하기 어려운 경우가 많다. 이런 경우 여러 개의 다중 신경망의 결합을 통하여 더 나은 성능을 얻을 수 있는데, 그림 3에서 보는 바와 같이, 여러 개의 인식기(다중 신경망)가 각각의 입력 벡터에 대하여 결과를 내고, 이를 결합 함수를 사용하여 최종 출력을 결정한다. 본 논문에서도 동적 위상보존 자기구성 지도의 성능향상을 위해서 다중 신경망을 결합하였다. 다중 신경망을 결합하는 데는 크게 두 가지 방법이 있다. 하나의 문제를 여러 신경망이 풀고 그 결과를 결합하는 방식과 하나의 문제를 하위 문제로 나누어 각각의 하위 문제를 신경망들이 나누어 해결하는 방식이다. 본 논문에서는 하나의 문제에 대해서 여러 신경망이 낸 결과를 결합하는 방식을 이용하였다.

4.1 결합 방법

결합 방법은 추상 레벨, 순위 레벨, 측정치 레벨로 나눌 수 있다. 각 레벨은 하위의 방법들로 구성되어 있는데, 그 내용은 다음과 같다.

4.1.1 추상 레벨

(1) 다수결 방법

다중 신경망에서 출력되는 결과를 모아서 다수결의 원칙

으로 해답을 얻어내는 가장 단순한 방법이다. 신경망의 결합 방법 중 가장 기본이 되는 방법이기 때문에 이 방법을 사용하였으나, 투표에 참여한 신경망의 수가 적기 때문에 투표의 결과가 기각되는 경우가 많았다.

(2) KW(K-Winner) 투표 방식

이 방법은 KW 방법을 응용한 방법인데, N개의 인식기들이 각각 K개의 답을 내고 $N\times K$의 결과로부터 다수결의 원칙을 적용하여 해답을 얻어내는 방식이다. 기본적으로 다수결의 원칙을 적용하는 단순한 방식을 사용했지만, 많은 투표 참여자를 산출해서 투표 결과가 기각되는 경우를 줄였고 신뢰도 또한 높였다.

(3) 베이시안 방법[14]

다수결 방법은 각 인식기의 출력 결과만으로 결합하는 반면, 베이시안 방법은 각 인식기의 오류 가능성도 최종 결과에 영향을 미치도록 한다. 혼동 행렬(Confusion matrix)을 통해서 각 인식기의 오류 가능성을 구한 후, 이를 통해 조건부 확률 $P(x\in C_i \mid e_k(x)=j_k)$를 이끌어 낸다. 조건부 확률의 의미는 k 번째 분류기가 입력 x를 j번째 클래스에 소속시켰을 때 입력 x가 클래스 C_i에 속할 확률이다. 이러한 확률값을 이용해서 신뢰도 함수를 구할 수 있는데, M개의 클래스를 갖는 문제의 경우 수식은 다음과 같다.

$$BEL(i) = \eta \prod_{k=1}^{K} P(x\in C_i \mid e_k(x)=j_k), \quad \text{for } j=1,..,M \tag{10}$$

단, 여기서 $e_k(x)$는 x에 대한 인식기 k의 결과를 나타낸다. 위의 신뢰도 함수를 이용해 베이시안 결합 함수 F를 다음과 같이 정의한다.

$$F(e_k(x)) = \begin{cases} j & \text{if } BEL(j)=\max_{i\in A} BEL(i)\geq \alpha \ (0<\alpha\leq 1) \\ M+1 & otherwise \end{cases} \tag{11}$$

여기서 α는 임계치다.

(4) BKS 방법[15]

각 인식기의 행동 결과를 지식 기반으로 하여 각각의 결과를 결합하는 방식이다. 각 인식기가 서로 독립이 아니라도 학습시킬 수 있다는 장점이 있지만, 기억 장소가 많이 필요하다는 단점이 있다. BKS는 각 인식기의 결과를 인덱스로 가지는 셀(Cell)의 집합이다. 따라서 k개의 인식기와 M개의 클래스가 있다면, 필요한 셀의 수는 k^M이 된다. BKS의 결합 규칙은 다음과 같다.

$$F(e_k(x)) = \begin{cases} R_{e_1(x)\ldots e_K(x)}, \text{ if } T_{e_1(x)\ldots e_K(x)}>0 \\ \quad \text{and } \dfrac{n_{e_1(x)\ldots e_K(x)}(R_{e_1(x)\ldots e_K(x)})}{T_{e_1(x)\ldots e_K(x)}\geq\lambda} \\ M+1 \quad , otherwise \end{cases} \tag{12}$$

$BKS(e_1(x), \ldots, e_K(x))$: BKS의 셀. 단, 인덱스는 $(e_1(x), \ldots, e_K(x))$.

$n_{e_1(x)\ldots e_K(x))}(m)$: $BKS(e_1(x), \ldots, e_K(x))$ 일 때 클래스 m에 속하는 데이타의 수.

$T_{e_1(x)\ldots e_K(x))}$: $BKS(e_1(x), \ldots, e_K(x))$에 속하는 데이타의 총수. 즉, $\sum_{m=1}^{M} n_{e_1(x)\ldots e_K(x)}$

$R_{e_1(x)\ldots e_K(x))}$: $BKS(e_1(x), \ldots, e_K(x))$를 대표하는 클래스. 즉, $\{j \mid n_{e_1(x)\ldots e_K(x)}(j) = \max_{m=1}^{M} n_{e_1(x)\ldots e_K(x)}(m)\}$

(5) 가중치 투표 방법

각 신경망은 서로 다른 특징으로 학습되기 때문에 잘 분류할 수 있는 부분이 서로 다르다. 그러나 다수결의 원칙으로 결과를 산출할 경우 각 신경망의 특징이 반영되지 않기 때문에 각 신경망의 노드마다 다른 가중치를 가지도록 하였다. 각 노드가 승리자가 되었을 때 그 승리자의 투표권을 [0, 1]사이의 실수가 되도록 하였다. 기본적인 알고리즘은 표3과 같다. 여기서 $Vote\ Result$는 투표 결과를 나타내고, $DTSOM_i$는 각각의 동적 위상보존 자기구성 지도를 나타낸다. 그리고 w_{ij}는 i 번째 신경망의 j 번째 노드를 나타낸다.

표 3 가중치 투표의 기본 알고리즘

```
while(LoopNumber < MaxLoop) {
    if (VoteResult == TRUE) continue;
    if (VoteResult == FALSE) {
        if (DTSOM1 == TRUE) w1i = w1i + α
        else w1i = w1i - α
                                        ...
        if (DTSOMn == TRUE) wni = wni + α
        else wni = wni - α
        }
    }
```

(6) KW+가중치 투표 방법

이 방법은 KW방법에 가중치 투표 방법을 적용한 것으로, (5)의 가중치 투표 방법은 각 신경망 별로 학습 데이타에 대한 성능을 평가하여 가중치를 부여하는 방법인 반면, KW+가중치 방법은 K개의 승리자 노드들에 대하여 각각 가중치를 부여하여 결과를 결합하는 방법이다. 즉, 각 인식기당 K개의 결과에 대한 가중치를 (5)와 같은 방법으로 구하고 가중치 합이 가장 큰 클래스가 최종 출력으로 결정된다.

4.1.2 순위 레벨

(1) Borda 함수

클래스 i에 대하여 인식기 k가 출력한 순위 r_k^i에 따라서 M- r_k^i를 구하고 이 값들을 합산하여 가장 점수가 큰 클래스가 결과로 결정되는 방식이다. Borda 함수는 다음과

같다.

$$F(e_k(x)) = \max_{i \in A}(B_i(e_k(x))) \quad (13)$$

단, $B_i(e_k(x)) = \sum_{k=1}^{K}(M - r_k^i(x))$.

(2) Condorect 함수

각 클래스에 대하여 자신을 제외한 다른 모든 클래스와 순위를 비교하여 높은 순위의 인식기 개수를 합하여 최소값을 구한 후, 그 값들을 비교하여 높은 값을 가지는 클래스가 상위의 순위를 가지는 방식이다. Condorect 함수는 다음과 같다.

$$F(e_k(x)) = \max_{i \in A}(Con_i(e_k(x))) \quad (14)$$

단, $Con_i(e_k(x)) = \min_{j \in M-i} \#(K : r_k^i > r_k^j)$.

여기서, $\#(K : r_k^i > r_k^j)$는 클래스 i가 클래스 j보다 높은 순위라고 판별한 인식기의 총수다.

4.1.3 측정치 레벨

(1) 신뢰값의 합산법

비교적 간단한 방법으로 각 클래스 i마다 주어진 신뢰값 m_k^i를 모두 합하여, 값이 큰 클래스를 높은 순위로 결정하는 방법이다. 수식으로 표현하면 다음과 같다.

$$F(e_k(x)) = \max_{j \in A} S_j \quad (15)$$

단, $S_i = \sum_{k=1}^{K} m_k^i$

5. 실험 결과

실험은 3개의 구조 적응형 자기구성 지도를 이용해서 오프라인 숫자 필기 데이타 인식문제를 해결하고자 하였다. 데이타는 Concordia대학의 필기숫자 데이타 중, A(2000개)와 C(2000개)에서 Contour, Kirsh Mask, Gradient 특징을 추출하여 각각 학습 데이타와 테스트 데이타로 사용하였고, B(2000개)를 검증 데이타로 사용하였다. MLP는 은닉노드(10~30개), 출력노드(10개), 학습률 및 모멘텀을 여러 가지 조합으로 실험한 후 각 경우에 대하여 가장 좋은 성능을 보이는 값을 사용하였다. 코호넨의 자기구성 지도(10×10)와 동적 위상보존 자기구성 지도에 대한 실험에는 학습 횟수 100,000번, 학습률 0.02를 적용하였다. 또한 모든 특징을 한꺼번에 사용한 복합 특징에 대한 실험도 3개의 동적 위상보존 자기구성지도를 결합한 경우와의 비교를 위하여 수행되었다. 모두 3회 수행후 평균값을 계산하였다. 동적 위상보존 자기구성지도는 4×4에서 초기화되어 7~10회 분화되었다.

표 4에서 보는 것처럼, 코호넨 알고리즘보다 구조 적응형 알고리즘이 월등한 분류율을 보인다. 기존의 전통적인 패턴 분류방법 중의 하나인 역전파알고리즘을 사용한 다층 퍼셉트론(MLP, Multi Layer Perceptron)과의 결과를 비교해 보면, Contour와 Gradient에서 각각 1.6%와 1.15% 정도 차이로 DTSOM이 우수한 인식률을 보였으며 평균적으로 MLP보다 나은 성능을 보였다

표 4 인식기별 인식률 [%]

	Contour	Gradient	Kirsh Mask	복합 특징
SOM	63.25	75.4	67.5	11.5
DTSOM	90.9	94.05	90.15	92.4
MLP	89.3	92.9	91.6	92.9

그 외, DTSOM 자체의 객관적인 성능을 비교하기 위하여 UCI 데이타베이스의 호주 신용카드 승인 데이타를 사용하여 SOM과의 성능을 비교하였다. 전체 690개 샘플중 500개를 학습데이타로 190개를 테스트 데이타로 사용하였다. 위 실험과 같은 실험 환경 하에서 7회 반복실험을 수행한 결과 DTSOM은 평균 68.9%, SOM은 평균 66.4%의 인식률을 보였다.

이러한 결과를 바탕으로 신경망 결합 실험을 수행하였다. 그림 4, 5, 6은 각 특징별 신경망의 위상구조를 보이고 있다. 각각의 숫자들이 군집을 이루고 있는 것은 그만큼 위상보존이 잘 일어나고 있음을 보여준다. 또한, 표 5는 Koenig의 위상보존 평가 척도 값을 보이는 데, 이를 통해 수치적으로도 동적 위상보존 자기구성 지도가 위상 보존이 잘 이루어지고 있음을 알 수 있다.

표 5 위상보존 평가

	SOM	DTSOM
$\sum q_{mj}$	8801	16600
위상보존 정도	0.3667	0.692

표 6은 각 레벨별로 신경망 결합의 실험 결과를 보여준다. KW 투표, KW+가중치 투표의 경우, K는 5로 고정하였다. 표 4의 실험 결과와 표 6의 실험 결과를 비교해 볼 때, 신경망의 결합을 통해서 인식률에 큰 향상을 얻을 수 있음을 알 수 있다. 또, 방법론적으로 본다면, 추상 레벨 결합이 순위레벨이나 측정치 레벨보다 나은 결과를 보여준다는 것을 알 수 있다.

추상 레벨 결합 방법 중, 다수결, KW+가중치, BKS 방법등이 좋은 결과를 보이고 있는데, KW+가중치 방법이 인식률과 기각율에서 가장 좋은 결과를 보이고 있고, BKS가 가장 적은 오류율을 보이고 있다. 특히 다수결 방법의 경우, 가장 간단한 방법인데도 상위의 인식률을 보였다. 표 7은

가장 좋은 결과를 보인 3가지 방법에 대해서 숫자별 결과를 나타내었다. 그리고 그림 7은 인식률을 막대 그래프로 보이고 있다. 그림에서 보듯이 인식률에서 KW+가중치 방법이 전 숫자에 걸쳐 높은 결과를 보이고 있고 또한 숫자별 인식률 편차가 제일 작음을 알 수 있다. 일반적으로 각 결합 방법들은 숫자 3, 5, 8에서 오인식률이 가장 높다. 왜냐하면, 3을 5나 8로, 5를 3이나 6이나 8로, 8을 3이나 5나 6으로 잘못 인식하는 경우가 많기 때문이다. BKS 방법의 경우, 5는 잘 인식하였고, 8은 평균 정도로 인식하였지만, 3을 제대로 인식하지 못했다. 그러나 KW+가중치의 경우, 3이나 8을 잘 인식하였다. 또한 5도 평균 정도로 인식하였고, 9는 다른 인식기들과는 달리 거의 완벽하게 인식하였다.

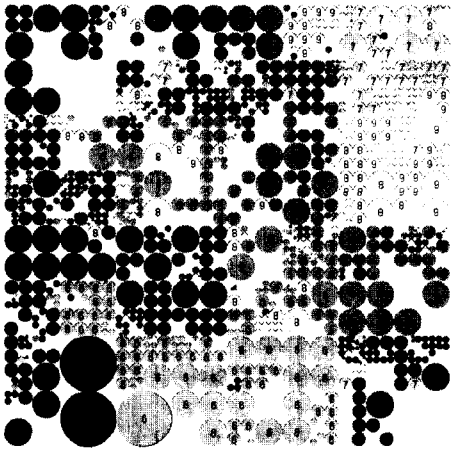

그림 4 Contour 특징의 위상 구조

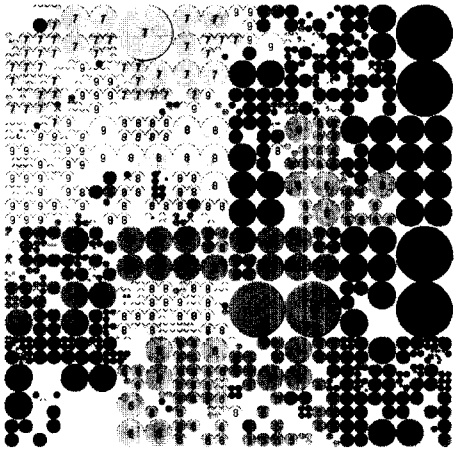

그림 5 Kirsh Mask 특징의 위상 구조

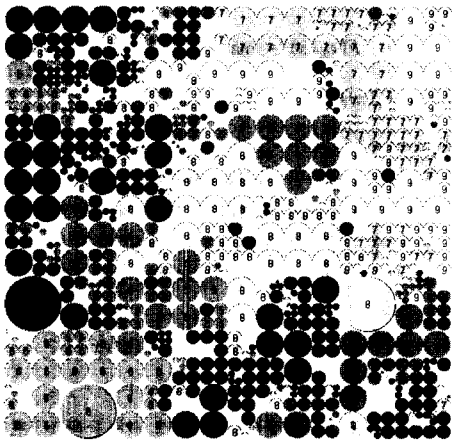

그림 6 Gradient 특징의 위상 구조

표 8과 9는 다수결 방법과 BKS 방법이 오인식 하거나 기각한 데이타들을 KW+가중치 방법이 얼마나 잘 인식하는 지를 나타낸다. 또한 표 10은 KW+가중치 방법이 여타의 결합에서 오인식되는 데이타를 바르게 인식하는 예를 KW를 사용하지 않은 결과와 대조하여 보여준다. KW+가중치 방법에서 K가 5이기 때문에 각 인식기들은 5개의 결과에 대한 가중치의 합을 구하게 되고, 그 합이 가장 큰 값이 승리자로 선택된다. 이 경우, 하나의 승리자를 구할 때 승리자가 아니었던 값들이 승리자로 되는 경우가 생겼다. 데이타 7의 경우 Countour, Kirsh와 Gradient 신경망의 출력이 단일 승리자 노드를 사용하였을 때는 각각 2, 2, 3의 결과를 내었으나, KW를 적용한 후에는 5, 2, 5의 결과를 나타내었고 가중치를 적용해 최종적으로 5를 출력하게 되었다. 또한, 데이타 5, 6, 9의 경우, KW적용 후 투표 방법을 사용하면 기각할 수밖에 없으나, 가중치의 크기를 비교하여 최종 출력을 올바르게 결정 할 수 있게 되었다. 이러한 결과를 통해, KW+가중치 방법이 가장 높은 성능을 보인다는 것을 알 수 있다.

표 6 결합 방법별 실험 결과

방법		기각률 (%)	인식률 (%)	오류율 (%)
추상 레벨	다수결	2.4	96.9	1.25
	K Winner	1.5	96.7	1.82
	가중치	2.4	95.65	2.00
	BKS	1.15	97.5	1.37
	Bayesian	2.85	95.1	2.11
	KW+가중치	0.2	98.1	1.7
순위 레벨	Borda	1.45	95.9	2.69
	Condorect	0.6	96.4	3.02
측정치 레벨	신뢰값 합산	0.4	96.8	2.81

표 7 숫자별 결합 결과

숫자	다수결			BKS			KW+가중치		
	인식률	기각률	오류율	인식률	기각률	오류율	인식률	기각률	오류율
0	99.5	0	0.5	99.5	0.5	0	99.5	0.5	0
1	100	0	0	99.5	0.5	0	100	0	0
2	97	0.5	2.51	97.5	2	0.51	98	0	2
3	97	2	1.02	93.5	1.5	5.08	97.5	0	2.5
4	98	0	2	99	0.5	0.50	99	0	1
5	90.5	3.5	6.22	96.5	2.5	1.03	95.5	0	4.5
6	98.5	0	1.5	99	1	0	99	0	1
7	99	1	0	98	1	1.01	98	0	2
8	92	4	4.17	95.5	1	3.54	95.5	1.5	3.46
9	97.5	1.5	1.02	97.5	1.15	2.02	99	0	1

표 8 다수결 방법에 대한 KW+가중치의 성능 향상 정도

숫자	다수결					KW+가중치					
	오답	기각	인식기별 오답 및 기각 수			오답된 결과에 대한 향상 정도			기각된 결과에 대한 향상 정도		
			Contour	Kirsh	Gradient	총수	정답	향상률(%)	총수	정답	향상률(%)
0	1	-		1	-	1	1	100			-
1	-	-	-		-	-	-				
2	5	1	1	5	-	5	5	100	1	1	100
3	2	4	2	4		2	0	0	4	3	100
4	4	0	2	2	-	4	3	75			-
5	12	7	1	16	2	12	9	75	7	1	14.3
6	3	0	-	3	-	3	2	66	-	-	-
7	0	2	-	1	1	-	-		2	1	50
8	8	8	1	12	3	8	3	37.5	8	5	62.5
9	2	3	1	3	1	2	1	50	3	3	100
합계	37	25	8	47	7	37	24	64.87	25	14	56

표 9 BKS에 대한 KW+가중치의 성능 향상 정도

숫자	BKS					KW+가중치					
	오답	기각	인식기별 오답 및 기각 수			오답된 결과에 대한 향상 정도			기각된 결과에 대한 향상 정도		
			Contour	Kirsh	Gradient	총수	정답	향상률(%)	총수	정답	향상률(%)
0	-	1	-	1		-	-		1	1	100
1	-	1	-	-	-	-	-		1	1	100
2	1	4	1	4	-	1	1	100	4	1	25
3	10	3	3	9	1	10	5	50	3	3	100
4	1	1	-	2		1	0	0	1	1	100
5	2	5	1	6	-	2	1	50	5	1	20
6	0	2	-	2	-	-	-		2	0	0
7	2	2	-	3	1	2	0	0	2	1	50
8	7	2	1	8	-	7	3	42.86	2	2	100
9	4	2	1	4	-	4	3	75	2	1	50
합계	27	23	7	39	2	27	13	48.15	23	12	52.17

표 10 KW+가중치 방법의 실제 인식 예

데이터	정답	KW+가중치						
		Contour		Kirsh		Gradient		결합 결과
		출력	가중치	출력	가중치	출력	가중치	
1	0	0	2.5	0	2	0	2.5	0
2	0	0	2.5	0	1.5	0	2.5	0
3	0	0	1.17	0	2.5	6	1.33	0
4	2	8	1.34	2	2.7	2	1.0	2
5	2	2	2.7	3	1.39	4	1.33	2
6	4	1	1.18	4	1.89	-	-	4
7	5	5	1.17	2	1.45	5	1.32	5
8	0	5	1.49	0	1.0	0	1.5	0
9	3	3	1.5	5	1.34	0	1.19	3
10	5	5	1.0	2	1.45	5	1.0	5

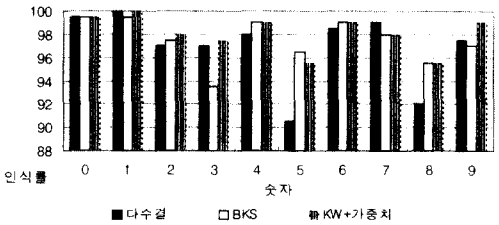

그림 7 결합 결과 비교

6. 결 론

본 논문에서는 자기구성 지도의 성능 향상 방법에 대해서 살펴보았다. 먼저, 문제에 따라 지도 구조가 스스로 변화하는 동적 위상보존 자기구성 지도 모델을 제안하였고, 자기구성 지도의 위상보존 특성을 이용하여 K-Winner 방법을 제안하였다. 그리고 여러 개의 동적 위상보존 자기구성 지도를 결합하여 성능 향상을 이루고자 하였다. 크게 추상 레벨, 순위 레벨, 측정치 레벨의 세 방법으로 나누어 실험하였다. 본 논문에서는 동적 위상보존 자기구성 지도에 적합한 가중치 투표 방법을 제안하였고, K-Winner 방법과의 혼합을 통해 가장 좋은 결과를 얻었다.

참 고 문 헌

[1] B. Fritzke, "Growing grid - A self-organizing netwroks with constant neighborhood range and adaptation strength," *Neural Processing Letters*, Vol. 2, No 5, 9~13, 1995.

[2] B. Fritzke, "Growing self-organizing netwroks - Why?," *ESANN'96*, p.61~72, 1996.

[3] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cyb.*, 43:59-69, 1982.
[4] T. Kohonen, Self-Organizing Maps, *Springer*, Berlin Heidelberg, 1995.
[5] K. Kiviluoto, "Topology Preservation in Self-Organizing Maps," *IEEE Transactions on Neural Networks*, 1996.
[6] T. Villmann, R. Der, M. Herrmann, and T. M. Martinetz, "Topology Preservation in Self-Organizing Feature Maps : Exact Definition and Measurement," *IEEE Transactions on Neural Networks*, vol 8, no. 2, 1997.
[7] A. Konig, "Interactive Visualization and Analysis of Hierarchical Neural Projections for Data Mining," *IEEE Transactions on Neural Networks*, vol 11, no. 3, 2000.
[8] 백종현, "다중 인식기의 다단계 결합을 통한 무제약 필기숫자 인식", *연세대학교 대학원*, 1996.
[9] L. Lam, C. Y. Suen, "A Theoritical analysis of the application of majority voting to pattern recognition," Proceeding of the 12th ICPR, pp. 418-420, 1994.
[10] L. Xu, A. Krzyzak, C. Y. Suen, "Methods of combining multiple classifier and their applications to handwriting recognition," *IEEE Transactions on Systems*, vol. SMC-22, no 3, pp. 418-435, 1992.
[11] C. Y. Suen, Y. S. Huang, "Multi-Expert Systems for pattern Recognition," *Proceedings of 2nd pacific Rim International Conference on Artificial Intelligence*, pp. 15-20, 1992
[12] T. K. Ho, "On multiple classifier systems for pattern recognition," *Proceeding of 11th ICPR*, vol.1, pp. 84-87, 1992
[13] S. B. Cho, "Self-organizing map with dynamical node splitting: Application to handwritten digit recognition," *Neural Computation*, Vol. 9, 1345~1355, 1997.
[14] J. C. Bioch, O. V. D. Meer, R. Potharst, "Classification using bayesian neural nets," *IEEE International Conference on Neural Networks*, vol. 3, pp. 1488-1493, 1996.
[15] A. Khotanzad, and C. Chung, "Hand written digit recognition using BKS combination of neural network classifiers," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 94-99, 1994.

김 현 돈
1998년 연세대학교 컴퓨터과학과(학사). 2001년 연세대학교 컴퓨터과학과(석사).

조 성 배
정보과학회논문지 : 소프트웨어 및 응용 제 28 권 제 6 호 참조