

論文97-34C-7-8

## 효과적인 패턴분할 방법에 의한 하이브리드 다중 컴포넌트 신경망 설계 및 학습

(Hybrid Multiple Component Neural Network Design and Learning by Efficient Pattern Partitioning Method)

朴贊虎\*, 李顯洙\*

(Chan Ho Park and Hyon Soo Lee)

### 요약

본 논문에서는 다수개의 전체 입력패턴을 효과적으로 분할하는 새로운 패턴 분할 신경망 알고리즘을 다중 컴포넌트 신경망에 부가하여 학습성능을 향상시킨 하이브리드 다중 컴포넌트 신경망(HMCNN : Hybrid Multiple Component Neural Networks)의 구조 및 학습알고리즘을 제안한다. 제안한 신경망에서 입력패턴의 분할을 위해서 부가된 신경망은 Kohonen신경망과 유사한 학습을 수행하지만 전체 패턴 부류의 수 및 하나의 부류에 속하는 패턴의 수를 자율적으로 조정하는 알고리즘을 적용함으로써 출력뉴런의 수를 동적으로 결정하는 적응적 신경망이다. 따라서 입력패턴의 수가 많아지고 신경망 입력패턴의 차원 및 전체 크기가 커지는 문제에 대해서 효과적으로 적용가능하다. 학습실험 결과 제안한 패턴 분할 신경망의 우수한 성능을 얻었으며, 일반화와 같은 기존 다중 컴포넌트 신경망이 갖는 문제점을 해결하였다. 또한 병렬학습을 수행함으로써 기타 교사학습을 수행하는 신경망에 비하여 학습속도가 크게 향상됨을 알 수 있었다.

### Abstract

In this paper, we propose HMCNN(Hybrid Multiple Component Neural Networks) that enhance performance of MCNN by adapting new pattern partitioning algorithm which can cluster many input patterns efficiently. Added neural network performs similar learning procedure that of Kohonen network. But it dynamically determine it's number of output neurons using algorithms that decide self-organized number of clusters and patterns in a cluster. The proposed network can effectively be applied to problems of large data as well as huge networks size. As a result, proposed pattern partitioning network can enhance performance results and solve weakness of MCNN like generalization capability. In addition, we can get more fast speed by performing parallel learning than that of other supervised learning networks.

### I. 서론

신경망은 패턴 분류 및 인식, 비선형 시스템 식별 및 제어, 그리고 로봇제어 등과 같은 매우 다양한 응

용분야에서 활발히 사용되고 있다.<sup>[1-4]</sup> 문제에 따라 적용되는 신경망의 모델은 매우 다양하며 존재하고, 그중 교사학습을 수행하는 대표적인 신경망으로는 다중 퍼셉트론 신경망을 들 수 있으며, 비교사 학습 즉 패턴을 분할하는 알고리즘으로는 Kohonen 및 LVQ 신경망<sup>[5-8]</sup> 등을 들 수 있다.

실제 적용에 있어서 신경망은 대규모의 크기 및 매우 복잡한 문제의 해결을 요구한다. 따라서 앞으로의

\* 正會員, 慶熙大學校 電子計算工學科

(Dept. of Computer Engineering Kyunghee University)

接受日字: 1997年5月30日, 수정완료일: 1997年7月2日

신경망은 이러한 문제들에 대하여 유연하게 대처할 수 있는 능력을 지녀야 한다. 이러한 문제를 해결하는 방법으로서 고려해야 할 점은 첫째, 속성들이 상이한 다양한 문제에 대하여 네트워크의 크기가 적응적으로 변화되어야 한다. 즉 입력패턴의 수와 복잡도에 따라 전체 신경망의 크기가 가변적으로 적응 변화되어야 한다. 그러나 현재 대부분의 신경망은 이러한 요구를 만족시키지 못하고 있는 실정이다.

둘째는 주어진 문제를 적절한 크기로 나누어 분할하여 병렬 학습할 수 있어야 한다. 이러한 문제점을 좀 더 효과적으로 해결하기 위해서는 신경망의 모듈화가 필수적이다.

세 번째로 입력패턴의 차원을 감소시켜 작은 크기의 신경망을 구성하는 것이 중요하다. 입력차원을 작게하는 방법으로는 신경망의 입력으로 전처리 과정을 거친 작은 차원의 특징벡터를 사용하는 것이 효과적이다.

현재 신경망 학습알고리즘으로 가장 널리 쓰이는 에러 역전파 학습알고리즘은 단일의 다중신경망에서 학습한다. 따라서 주어진 문제의 크기가 크고 복잡할수록 학습시간은 길어지고 문제 해결이 어려워진다. 이 때 주어진 입력패턴을 분할하여 다중의 신경망에서 분할학습 시킴으로써 학습 시간을 크게 단축시킬 수 있으며, 또한 매우 큰 크기를 갖는 패턴인식문제에 대하여 병렬학습처리에 의해 효과적으로 대응시킬 수 있다. 이미 제안한바 있는 MCNN (Multiple Component Neural Network)신경망은 기존 에러역전파 신경망이 갖는 문제점을 해결하기 위해서 PCA(Principal Component Analysis)방법을 이용하여 입력패턴의 차원을 줄이고, 전체 입력패턴의 수를 분할하여 학습함으로써 전체 신경망의 학습성능을 향상 시켰다<sup>[9]</sup>. 여기에 사용한 패턴 분할방법은 결정트리 알고리즘과 유사한 방법을 이용하였다.

실제 신경망의 모듈화에서 검토되어야 할 중요사항은 패턴의 분할방법으로써 필요에 따라서는 적절한 수의 패턴을 분할시켜 독립된 모듈신경망에서 학습하는 것이 필요하다. 이에 따라 전체 모듈 신경망의 크기를 결정할 수 있다. 패턴을 분할하는 방법 중 신경망으로 구현된 단일 패턴 분할 신경망은 Kohonen 과 LVQ 신경망등이 있으나 이들 신경망은 분할하고자하는 패턴부류의 수 즉 출력뉴런의 수가 결정적이므로 패턴부류의 수를 초기 학습단계에서 결정해야 하는 단점을 갖고있다. 또한 하나의 부류에 많은 패턴의 밀집현상

이 발생할 수 있게 되어 궁극적으로 전체 처리성능면에 저하를 초래한다. MCNN신경망의 경우 결정트리 분할방법을 이용하였으나 이는 입력패턴간의 거리 및 유사도와 관련하여 패턴을 분할하는데 있어서 경우에 따라서는 매우 큰 오차를 가질 수 있다. 또한 분할된 부류에 단지 하나의 패턴만이 존재하는 경우 이를 근접한 다른 부류에 포함시켜 학습을 해야하는 문제점을 들 수 있다.

본 연구에서는 위에서 언급된 문제를 해결하기 위해서 학습도중 적절한 수의 패턴을 분할할 수 있도록, 한 부류의 소속패턴수를 제한하고 이를 위반하였을 경우 이보다 작은 여러 개의 새로운 부류로 다시 분할하는 새로운 동적 패턴 분할 학습알고리즘을 제안한다.

또한 기존의 다중 컴포넌트 신경망에 동적패턴 분할 알고리즘을 부가하여 학습함으로써 기존 방법보다 일반화 해결 능력 및 네트워크 크기 결정에 있어 유연함을 갖는 하이브리드 다중 컴포넌트 신경망을 제안한다. 실험대상으로는 비교적 문제의 복잡도가 높은 Parity 문제 및 Two-spiral 문제와 일반 영문자 문제인식을 대상으로 하였으며 실험결과 에러역전파 알고리즘 및 비교된 신경망보다 우수한 성능을 나타내었으며 특히 문제의 크기가 매우 큰 신경망의 구현에 효과적임을 알 수 있었다.

본 논문의 제2장에서는 기존 패턴 분할 신경망과 제안한 패턴 분할 신경망에 대하여 분석 고찰한다. 제3장에서는 하이브리드 다중 컴포넌트 신경망에 대하여 설명하고 제4장에서는 시뮬레이션을 통하여 제안한 신경망의 우수성을 검증한다. 마지막으로 제5장에서는 본 연구의 결과를 기술하고 향후 연구방향을 제시한다.

## II. 패턴분할 신경망 알고리즘 제안

### 1. 패턴 분할 및 신경망 알고리즘

입력패턴의 공간상에 무리를 지은 다수의 패턴들이 존재할 때 이러한 패턴집합을 패턴군집(cluster)이라고 한다. 이때 주어진 패턴간의 거리 및 유사도를 이용하여 패턴들을 뮤는 과정을 패턴분할 과정 또는 군집화 과정이라고 한다. 패턴분할을 수행하는 대표적인 알고리즘으로는 최대-최소거리 알고리즘, k평균 알고리즘, Isodata 알고리즘 등이 있으며, 신경망을 이용한 대표적인 알고리즘으로는 Kohonen의 SOFM (Self Organizing Features Map) 및 LVQ (Learning Vec-

tor Quantization) 등이 있다. 이때 사용되는 군집화의 척도로써는 주로 패턴간의 거리를 사용하는데 가장 많이 적용되는 거리함수는 유클리디안(Euclidean) 거리와 패턴간의 공분산을 적용한 마할라노비스 (Mahalanobis) 리, 이외에 랜스-윌리암스 (Lance-Williams) 거리 및 내적거리 등이 있다. 이중 유클리디안 거리가 계산의 편이성 및 사용의 용이성으로 인하여 비교적 폭넓게 사용되고 있다<sup>[10-11]</sup>.

Kohonen 신경망을 이용한 패턴분할 알고리즘은 대상으로 하는 미지의 패턴들을 자동으로 분할하는 비교사학습을 수행하며 주로 입력층과 출력층이 완전연결된 2층의 신경망으로 구성된다. 이를 그림으로 나타내면 다음과 같다

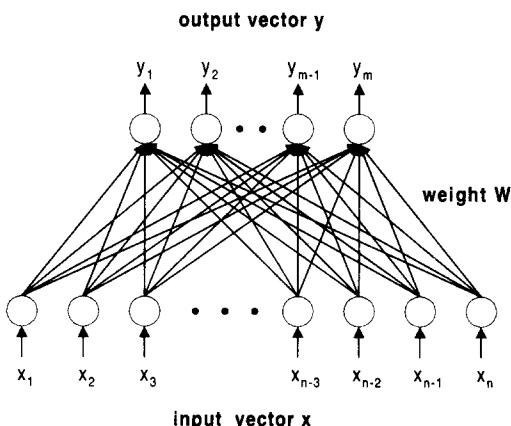


그림 1. 패턴 분할 신경망 구조

Fig. 1. Pattern partitioning neural network architecture.

위의 그림 1에서 입력패턴은 입력뉴런을 통하여 입력되며, 출력뉴런의 수는 분할하고자 하는 부류의 수를 나타낸다. 이때 각 뉴런은 연결강도 벡터와 입력벡터가 얼마나 가까운지를 계산한 후 각 출력뉴런은 학습의 권한을 부여받기 위해서 서로 경쟁학습을 수행한다. 이때의 승자뉴런은 주어진 입력벡터와 가장 가까운 뉴런으로 결정되는데 이를 수식으로 나타내면 다음의 식(1), 식(2)와 같다.

$$i(x) : \arg ; \min(D_i) \quad (1)$$

$i(x)$  : 최소거리를 갖는 승자 뉴런

$$D_i = \sum_{i=0}^n (X_i(t) - W_i(t))^2 \quad (2)$$

Kohonen신경망에서 사용된 가중치  $W$ 의 조정 수식은 다음의 식(3)과 같다.

$$W_{ij}(t+1) = W_{ij}(t) + \gamma(X_i - W_{ij}) \quad (3)$$

위의 식에서 학습상수  $\gamma$  값은 초기에 일정한 상수 값에서 학습이 진행되어 가면서 서서히 감소한다. 따라서 학습의 종반부에 가중치의 개신량은 거의 0에 근사한다.

LVQ 신경망은 Kohonen과 거의 유사하나 계산되어진 승자 뉴런이 요구되는 부류 즉 클러스터에 속하는가 아닌가를 판단하여 만약 원하는 부류에 포함되면 가중치의 값을 주어진 입력패턴과 가깝게 조정하고 만약 그렇지 않으면 가중치를 멀게 조정하는 개선된 알고리즘이다. 이를 요약 설명하면 다음과 같다.

식(1)과 식(2)에 의해 선택된 승자뉴런의 부류  $\omega_i$  와 원하는 부류  $\omega_c$  가 같은 경우 수식(3) 적용, 그렇지 않으면 다음의 식(4)를 적용한다.

$$W_{ij}(t+1) = W_{ij}(t) - \gamma(X_i - W_{ij}) \quad (4)$$

따라서 LVQ신경망의 적용을 위해서는 주어진 패턴이 어느 부류에 속해있는가를 미리 결정 해야 한다. 위의 두 가지 신경망이 갖는 장점은 계산의 단순성과 적용의 용이성에 있다. 그러나 입력패턴의 수가 대단히 많아지면 학습이 어려워지며, 학습에 소요되는 시간이 오래 걸린다. 그리고 분할하고자 하는 부류의 수가 결정적이므로 신경망이 갖는 장점인 적응성이 결여되는 단점을 가지고 있다.

위에서 언급된 2개의 신경망이 갖는 학습에서의 단점은 일반적인 패턴분할알고리즘과 마찬가지로 패턴의 군집화 특성에 따라 성능에 많은 영향을 미치는 점이다.

특히 패턴들의 분포가 적절하게 분포되지 않고, 밀집되거나 느슨하게 분포된 패턴부류가 같은 공간에 존재시 분할의 정확도가 떨어진다.

## 2. 개선된 패턴분할 신경망

앞서 기술한바와 같이 패턴 분할 신경망이 갖는 결점을 보완 해결하는 적응적 패턴분할 신경망 알고리즘을 제안한다. 적응적이라면, 신경망 구조가 학습도중에 동적으로 변화되어 문제에 적절한 네트워크가 형성됨을 의미한다.

패턴분할 신경망의 구조를 나타내면 그림 2와 같다.

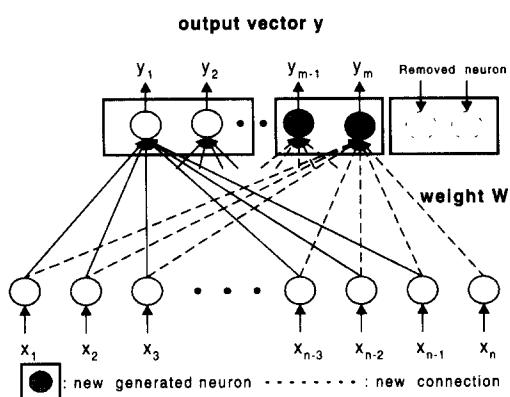


그림 2. 개선된 패턴 분할 신경망 구조

Fig. 2. Improved pattern partitioning neural network architecture.

위의 그림 2에서 출력층의 가운데 2개의 뉴런은 초기에는 존재하지 않았으나 학습진행 도중 새롭게 생성된 뉴런으로써, 학습도중 생성과 소멸과정을 거쳐 원하는 크기의 신경망으로 구성된다. 따라서 입력뉴런과 출력뉴런사이의 가중치의 연결 수가 시간의 변화에 따라 동적으로 변하게 된다. 학습이 종료된 후의 출력뉴런의 수는 최종적으로 분할된 적정한 부류의 수를 나타낸다.

학습알고리즘은 Kohonen신경망의 가중치 계산을 사용하였으며, 각 부류는 적어도 둘 이상의 패턴을 가지도록 하고 각 부류에 속한 패턴의 수를 적절하게 조절할 수 있도록 제한한다. 그리고 하나의 패턴이 자신이 속한 부류의 수와 관련되어 자신만의 독특한 영역을 가질 수 있도록 한다. 즉, 자신이 속한 부류와 다른 부류와의 유클리디안 최소거리를 구하여 만약 패턴 자신이 가지는 영역보다 유클리디안 거리가 작으면 자신을 새로운 부류로 결정하는 방법을 사용한다. 이때 결정되는 패턴 자신의 영역을 식으로 표현 하면 다음과 같다.

Distance 1

$$Dis_i(t+1) = Dis_i(t) * (1 + \mu(C_i)) \quad (5)$$

Distance 2

$$Dis_i(t+1) = Dis_i(t) * (1 - \mu(C_i)) / M \quad (6)$$

위의 식에서  $Dis_i(0)$ , 즉 초기 값은 임의로 결정하며 아주 작은값으로 결정하는 경우 부류의 수는 패턴의 수와 거의 동일하다. 큰 값으로 결정하는 경우의

패턴부류의 수는 작게 결정되어 진다. 식(5) 및 식(6)의 파라메터  $\mu$ 는 영역의 크기를 조절하는데 필요한 상수이며,  $C_i$ 은 패턴이 속한 부류가 갖는 패턴의 수를 나타내고  $M$ 은 전체 패턴의 수를 나타낸다.

신경망의 출력뉴런은 학습시 가변적으로 변화될 수 있으며, 각 부류의 크기와 부류에 속한 패턴의 수를 적절하게 조정함으로써 이상적인 패턴분할이 가능하다.

일반적으로 비교사학습이 갖는 단점의 하나로 학습의 종결조건을 명확히 결정하기 어려운 점을 들 수 있다. 따라서 가중치의 변화가 거의 없는 경우를 학습의 종결조건에 포함하거나 최대허용학습횟수를 결정하여 사용한다. 본 연구에서 제안한 신경망의 학습종결조건에는 이에 부가하여 전체 분할된 부류에 속한 패턴의 수가 전체 입력패턴의 수와 동일하게 될 때 학습을 종결한다. 이를 위해 부류에 속한 패턴의 중복 및 하나의 부류에도 속하지 않는 고립된 패턴의 존재를 피할 수 있도록 하였다. 부가된 학습 종결조건을 식으로 나타내면 식(7)과 같다.

$$C_N = \sum_{i=0}^n C_i = M \quad (7)$$

식(7)의  $C_N$ , 즉 전체 부류에 속한 패턴의 수가 전체 패턴의 수와 동일한 경우 학습을 종결한다.

본 신경망의 학습알고리즘은 다음과 같다.

#### <패턴분할 신경망 알고리즘 >

단계1) 연결강도 초기화 학습파라메터 초기화

단계2) 새로운 입력패턴 제시

단계3) 입력패턴과 출력뉴런간의 거리계산을 이용한 승자뉴런결정

: [수식1, 수식2] 사용

단계4) 승자뉴런이 속한 부류의 패턴 수를 검사하여 입력패턴의 영역결정

: 패턴수 2미만 [수식5]

패턴수 2이상 [수식6] 적용

단계5) 승자뉴런이 속한 부류가 갖는 최소 유클리디안 거리와 입력패턴이 갖는 영역의 비교 만약 승자뉴런의  $Min(D_j) < Dis_i$

가중치 생성 : [수식3] 적용

그렇지 않으면

새로운 출력뉴런의 생성

(새로운 클러스터로 분할)

### 단계6) 하나의 부류에 속한 패턴의 수 결정

만약 허용패턴수보다 작으면 단계7로 아니면 단계2)로 가서 재학습

### 단계7) 학습완료조건 판단

전체 허용 학습횟수 판단

전체 패턴 수 판단 : [수식7] 적용

만약 위의 조건을 만족하면 단계8)로

아니면 단계2)로 가서 재학습

### 단계8) 학습완료

## III. 하이브리드 다중 컴포넌트 신경망(HMCNN)

제안하는 하이브리드 다중 컴포넌트 신경망은 PCA(Principal Component Analysis)역활과 패턴분할을 동시에 담당하는 하나의 신경망인  $\text{PCA}^2$ (Pattern Clustering And Principal Component Analysis)와 에러역전파 학습을 수행하는 다수개의 CBP신경망 부분으로 구성된다. 하이브리드 다중 컴포넌트 신경망의 전체구조는 그림3과 같다.

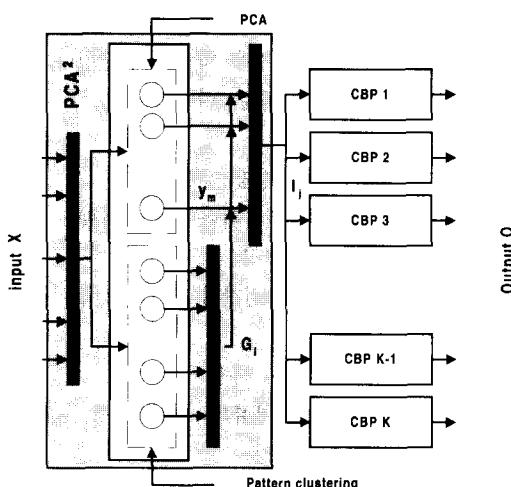


그림 3. 하이브리드 다중 컴포넌트 신경망 구조

Fig. 3. Hybrid Multiple Component neural networks architecture

입력패턴은 우선  $\text{PCA}^2$ 신경망으로 입력되어 이중 PCA부분에서는 낮은 차원으로 변화된 새로운 출력패턴을 만들어 내며, 패턴분할 부분에서는 입력된 패턴 간의 거리에 따른 부류를 결정한다. 이후 PCA의 출력은 해당되는 CBP신경망으로 전파되어 독립적인 교

사학습을 수행하게 된다.

그림 3에서 전체 CBP신경망의 수는 패턴분할 신경망에서 결정된 출력뉴런의 수 즉 패턴부류의 수와 동일하며 하나의 CBP신경망의 입력뉴런의 수는  $\text{PCA}^2$  신경망에서 결정되는 차원의 수와 동일하다.

### 1. $\text{PCA}^2$ 신경망 구조 및 학습

$\text{PCA}^2$ 신경망은 입력층과 출력층이 완전 연결을 가지며, 출력층은 PCA변환을 수행하는 부분의 출력뉴런과, 패턴분할을 수행하는 부분의 출력뉴런사이의 완전 연결을 가진다.  $\text{PCA}^2$  신경망의 내부구성은 그림4과 같다. 전체 입력뉴런의 수는 입력패턴 벡터차원과 같으며, 출력뉴런의 수는 PCA변환 후의 출력뉴런수에 패턴부류의 수를 더한 수와 동일하다.

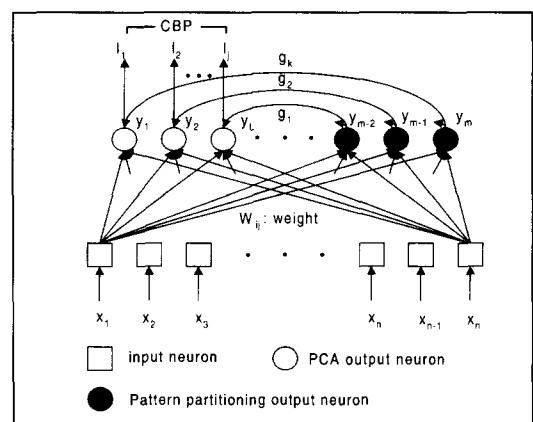


그림 4.  $\text{PCA}^2$  신경망 구조

Fig. 4.  $\text{PCA}^2$  neural networks architecture.

본 신경망에서는 각 PCA부분과 패턴분할 부분의 학습이 독립적으로 이루어지며 이후  $\text{PCA}^2$ 의 출력이 CBP신경망의 학습을 위하여 출력 값으로 전파되어진다.

우선 PCA변환을 위한 부분에서 사용한 알고리즘으로는 GHA (Generalized Hebbian Algorithm)보다 빠르고 작은 주성분(Principal Component)값을 추출하는데 효과적인 SGA (Stochastic Gradient Ascent)방법을 사용한다. 이때 입력패턴 벡터에 대한 주성분이 하나인 경우에는 NHR(Normalized Hebbian Rule)과 동일하다<sup>[12-14]</sup>. SGA를 이용한 출력값을 표현하면

$$Y_i = WX = \sum_{k=1}^n w_{ik}x_k \quad (1 \leq i \leq l) \quad (8)$$

$$(W = [w_1, w_2, w_3, \dots, w_l]^T \in R^{n \times l})$$

식(8)과 같다. 입력으로는  $n$  차원의 입력패턴벡터가 사용되었으며, 출력으로는  $l$  ( $l \leq n$ ) 차원의 출력패턴벡터가 사용된다. 또한 가중치의 갱신을 표현하면

$$w_{ij}(k+1) = w_{ij}(k) + \Delta w_{ij} \quad (9)$$

$$\Delta w_{ij} = \eta y_i(k) [y_j(k) - w_{ij}(k)y_i(k) - \alpha \sum_{h=1}^{l-1} w_{ih}(k)y_h(k)]$$

식(9)과 같으며, 사용된 파라메터  $\eta$  는 학습률을,  $\alpha$  는 주성분 출력  $y$  의 값사이의 조정을 위한 상수로 사용된다.

다음으로 패턴분할을 담당하는 부분에서의 가중치 갱신식 식(3)을 적용하여, 패턴분할 부분의 뉴런출력  $G = \{g_{m-l+1}, g_{m-l+2}, \dots, g_m\}$  는 식(10) 및 식(11)을 따른다. 즉 패턴분할 부분에서의 학습은 winner-take-all 경쟁에 의해 승자뉴런을 선택하여, 입력패턴에 대하여 승리한 뉴런의 출력을 1로, 나머지 뉴런의 출력을 0으로 변화시켜야 한다. 이러한 동작을 수행하기 위한 표현은 다음과 같다.

$$Y_i = WX = \sum_{k=1}^n w_k x_k \quad (m-l+1 \leq i \leq m) \quad (10)$$

$$\text{if } y_i = \max(y_i) \text{ then } g_i = 1 \\ \text{else } g_i = 0 \quad (11)$$

PCA변환을 거쳐 출력된 출력값  $Y$ 는 패턴 분할 부분의 출력값  $G$ 와 곱해져서 출력되며, CBP신경망으로 전파되어 자신이 속한 부류 안에서의 학습을 수행한다. CBP신경망의 입력으로 사용되는 PCA<sup>2</sup>신경망의 출력  $I_i^j$ 는 다음 식(12) 와 같다.

$$I_i^j = \sum_{i=m-l+1}^m g_i y_i \quad (1 \leq i \leq l) \quad (12)$$

PCA<sup>2</sup> 신경망에서는 2개의 서로 다른 학습을 수행하는 알고리즘을 하나의 신경망 구조 상에서 해결하였으며, 학습의 종료는 일반적으로 전체 출력뉴런이 안정된 상태에 도달하였을 때 학습이 끝나게 된다. 그러나 PCA학습부와 패턴분할부의 종료조건은 서로 다르므로 두 부분이 각각의 종료조건을 만족하였을 때 전체 학습이 완료되는 것으로 결정한다.

## 2. 컴포넌트 신경망 CBP의 학습

컴포넌트 신경망인 CBP는 3개층의 다층퍼셉트론 구조를 가지고 있으며, 에러역전파 신경망 알고리즘을 사용한다. 각 CBP신경망의 수는 PCA<sup>2</sup> 신경망에서 결정된 패턴부류의 수로 결정되어지며, 각 CBP신경망

의 크기는 입력패턴의 차원을 어느정도까지 감소시키는가에 따라 결정된다.

바꾸어 말하면, 각 컴포넌트 신경망의 크기는 입력뉴런의 크기에 따라서 결정되며, 입력뉴런의 수는 PCA변환시 주성분을 몇 개까지 출력해내는가에 따라 결정된다. 출력되는 주성분의 수가 작은 경우 미세한 값을 갖는 입력 패턴을 잘 구별할 수 없다. 따라서 입력패턴의 차원을 무조건 작은 크기로 줄일 수는 없다.

PCA<sup>2</sup> 신경망의 학습이 끝난 후 전파되는 패턴  $I_i^j$ 는 각 컴포넌트 신경망 중 해당되는 신경망으로 전파되어 학습을 수행하며, 이때의 학습알고리즘은 식(13)-식(15)로 표현한다.

$$O_j^k = f(\sum_{i=1}^n w_{ji}^k I_i^{j,k-1} + \theta_j^k) \quad (13) \\ = f(\text{net}_j)$$

이때  $O_j^k$ 는  $k$  층의  $j$  번째 출력값이고  $w_{ji}^k$ 는  $k$  층의  $j$  번째 뉴런과  $k-1$  층의  $i$  번째 뉴런의 연결가중치를 나타낸다. 또한  $I_i^{j,k-1}$ 은  $k-1$  층의 출력을 나타낸다. 출력함수로는 시그모이드 함수를 사용하며

$$w(t+1) = \lambda w(t) + \Delta w \\ = \lambda w(t) + \theta \delta_i I_i \quad (14) \\ \theta : \text{error rate}, \\ \theta : \text{learning rate}, \lambda : \text{momentum}$$

위의 식(14)에서 에러량  $\delta$ 는

$$\delta_i = (d_i - o_i)f(\text{net}) \text{ if } i : \text{output unit} \quad (15) \\ \delta_j = f'(\text{net}) \sum_i \delta_i w_{ij} \text{ if } j : \text{hidden unit}$$

로 나타내며,  $d_i$ 는 교사신호를 나타낸다.

제안한 신경망에서 하나의 컴포넌트 신경망은 주어진 다수개 입력패턴에 대한 하나의 부류로 정해지며, 이때 하나의 부류에 속한 패턴의 수는 PCA<sup>2</sup> 신경망의 패턴분할 부분에서 결정되어 진다. 따라서 전체 CBP신경망은 전체 패턴의 수를 적당한 크기로 나누어 병렬 학습이 가능하며 매우 빠르게 학습을 수행한다.

## 3. 학습알고리즘

HMCNN신경망의 학습은 우선 주어진 입력패턴에 대한 분할 및 차원감소과정을 병렬로 수행한 후, 다수의 CBP신경망에서 교사학습을 수행한다. 따라서 차원의 감소과정이 끝난 후 패턴분할을 수행하는 기준

MCNN신경망에 비하여 학습시간을 감소시킬 수 있다. 제안한 신경망 학습알고리즘을 나타내면 다음과 같다.

#### 단계1) 초기화 과정

(가중치  $w$  초기화, 학습파라메터 초기화)

PCA<sup>2</sup> 신경망 초기화

CBP 신경망 초기화

#### 단계2) PCA<sup>2</sup> 학습과정

학습시간을 더한값을 학습iteration으로 표현하였다. 그러나 위의 두 신경망의 매회 학습iteration에 소요되는 시간은 매우 다르며 일반적으로 PCA<sup>2</sup> 의 1회 학습 계산 시간이 학습방법의 차이에 의하여 CBP의 학습 보다 매우 빠름을 알 수 있다.

##### 1. 2-Spiral 문제

HMCNN에서 PCA<sup>2</sup>신경망의 패턴분리 능력을 실험하기 위해서 2-spiral문제를 대상으로 전체 200개의 패턴을 사용하여으면 이를 그림5

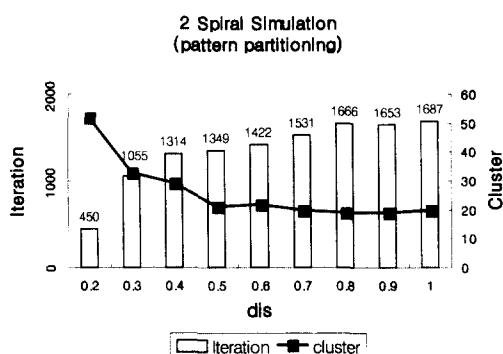


그림 6. 초기 Dis 값의 변화에 따른 부류의 수 및 학습 iteration

Fig. 6. Number of clusters and learning iterations results for various initial Dis.

위의 그림에서 초기Dis의 값이 0.2인 경우는 52개의 부류로 분리되지만 초기 Dis의 값이 0.5에서 1.0까지는 20내지 21개의 부류로 분리되어 후반부의 CBP 학습에 적용된다. 이유로는 초기 Dis의 값이 작은 경우에는 작으므로 하나의 부류에 속하는 패턴의 수가 작으므로 많은 부류들로 분할되나, 초기 Dis의 값이 큰 경우에는 초기 하나의 부류에 많은 패턴들을 가질 수 있기 때문이다.

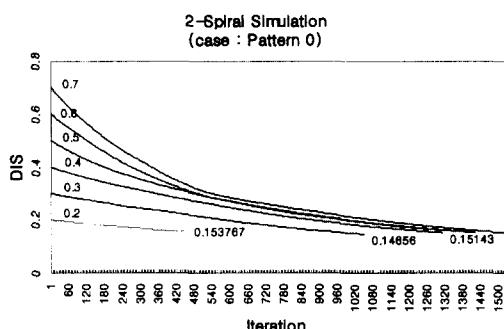


그림 7. 패턴0 (0.0000, -0.0531)의 초기Dis값 결정에 의한 수렴

Fig. 7. Convergence of initial Dis of pattern 0 (0.0000, -0.0531)

초기Dis값에 따른 학습iteration의 변화를 그림으로 나타내면 그림 7과 같으며, 각 Dis의 초기값들에 대하여, 학습이 진행됨에 따라 그림상의 결과와 같이 수렴되었다.

이때 각 패턴이 갖는 Dis의 값은 주어진 조건에 의해서 확대와 축소를 미세하게 반복하며, 최종적으로 최적의 패턴 개수를 가진 부류들의 수렴의 해를 얻게 되었다.

된다.

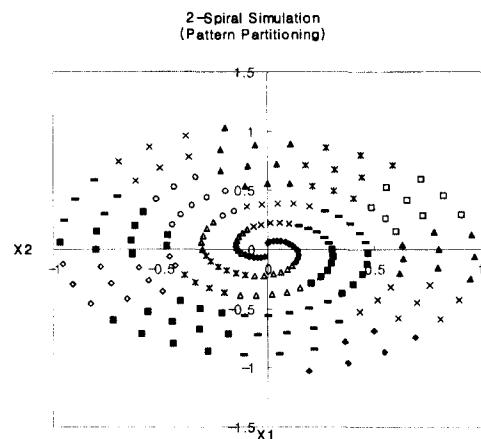


그림 8. 초기 Dis 값의 결정에 따른 패턴분할 (초기Dis = 0.7)

Fig. 8. Pattern partitioning by initial Dis. (Initial Dis = 0.7)

그림 8은 Dis의 초기값을 0.7로 하였을 때 분리된 20개 부류의 결과이다. 이때 분리된 각 부류는 최소 패턴 수 5개에서 최대수는 20개 까지를 갖는다.

그림 8에서와 같이 분리된 각 부류는 CBP신경망상에서 독립적으로 학습된다. 본 실험에서는 우선 2-Spiral문제에 대하여 2차원의 입력벡터를 PCA<sup>2</sup>신경망의 PCA과정을 거치지 않고 바로 입력벡터를 CBP로 전파한 후 학습하였다. 그 이유는 위의 문제가 매우 작은 입력차원을 가지기 때문에 차원의 감소를 통한 네트워크 크기의 감소가 필요로 하지 않았으며, PCA 과정을 거칠 경우에 더 많은 에러를 가질 수 있기 때문이다. 따라서 적은 차원의 입력벡터에 대하여 PCA 과정을 통한 신경망 학습은 본 실험에서는 고려하지 않는다. 그림 9는 초기Dis가 0.7일 때 분할된 패턴들에 대한 HMCNN신경망의 학습Iteration 및 error를 나타낸다. 그림 10은 전체패턴을 일반 EBP신경망에서 학습시킨 후 학습Iteration 및 error의 결과를 나타낸다. 사용된 학습상수  $\lambda$  및  $\mu$ 는 각각 0.1로 주었으며, 허용에러는 0.001로 허용 iteration은 100,000으로 하였다. 실험결과 CBP 학습이 EBP 보다 빠르게 수렴함을 알 수 있다. 위에서 사용된 신경망의 크기는 CBP의 경우 전체가 동일하게 3개의 중간 뉴런을 사용하여, EBP1은 10개, EBP2는 15개, EBP3는 20개의 중간뉴런을 가지고 학습한다. 전자인 CBP의 경우, 패턴분할에 사용된 연결수는 40개이고, 20개의 패턴에

대하여 동일한 수의 CBP신경망이 형성되므로 가중치 갱신을 위한 전체 연결수는 220개이다.

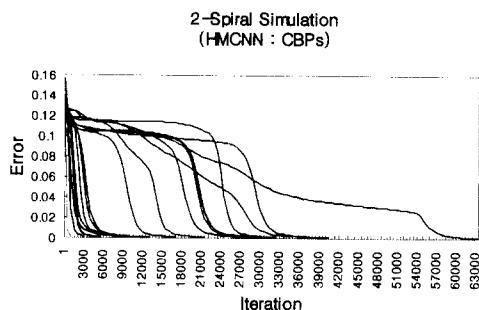


그림 9. 2-spiral 문제 CBP신경망 학습 error (20개의 CBP신경망 학습)

Fig. 9. CBP neural networks learning error for 2-spiral problem. (20 CBP networks)

그림 10에서 학습에 소요된 시간은 HMCNN 이 EBP에 비하여 매우 빠르며, 그 이유로는 EBP신경망의 경우 1번의 iteration에 200개의 패턴을 처리하는 반면에, HMCNN의 경우 최대 20개의 패턴만을 하나의 CBP에서 단독으로 처리하기 때문이다. 더 나아가 세안한 신경망을 병렬 하드웨어로 구현시 더욱 더 빠른 학습시간을 얻을 수 있다.

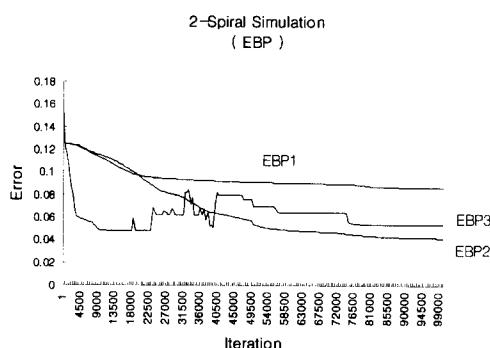


그림 10. 2-spiral 문제 EBP신경망 학습 error (전체 패턴에 대한 CBP신경망 학습)

Fig. 10. EBP neural networks learning error for 2-spiral problem.

## 2. 6-Parity문제

기존 EBP신경망에서 패리티문제의 해결은 매우 유사한 문제에 대하여 서로다른 출력을 내야하므로 단일의 신경망에서 해결하기 어렵다. EBP의 경우 일반적으로 중간뉴런의 수는 입력뉴런의 약 2배의 크기를 갖

는다.

HMCNN에서는 6-parity문제를 대상으로 전체 64개의 패턴을 분리하여 학습시켰으며 이때 최대 부류의 수는 32개로 자신과 1 bit의 차이가 있는 패턴을 순서에 따라 우선적으로 묶어서 학습됨을 알 수 있었다. 그러나 패턴부류의 수가 너무 많은 경우 전체 신경망의 크기가 너무 커지는 단점이 있으므로 이를 Dis의 초기 값으로 이용하여 조정하였다. PCA<sup>2</sup>신경망의 PCA 부분에서는 6차원의 입력벡터를 3차원으로 줄여서 CBP로 전파하였으며, Dis의 변화에 따른 부류의 수 및 패턴분할부분의 학습iteration을 나타내면 그림11.과 같다.

그림 11에서 초기Dis의 값이 3일 때 학습iteration 2회, 부류의 수 8개로 가장 적절하게 분할됨을 알 수 있었으며, 학습iteration이 작은 이유는 6-parity패턴의 분포가 일정하기 때문에 적절한 초기값이 주어진 경우 부류의 중앙값(center)의 큰 이동 없이 원하는 수의 부류를 얻을 수 있기 때문이다.

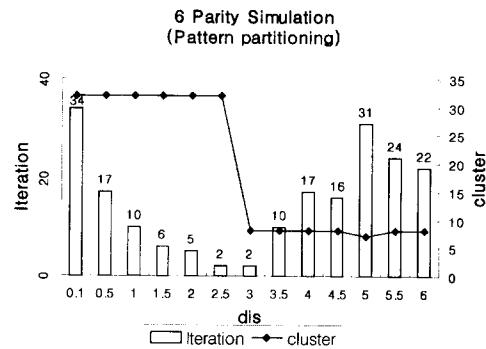


그림 11. 초기 Dis값의 변화에 따른 부류의 수 및 학습 iteration

Fig. 11. Number of clusters and learning iterations results for various initial Dis.

그림 11에서 초기 Dis의 값이 4.0일 때 차원의 감소과정을 거친 8개의 부류가 각각 8개의 CBP로 전파되어 학습한 결과와 및 기존 MCNN신경망, EBP신경망과의 비교를 나타내면 그림 12과 같다. 이때 MCNN, HMCNN신경망의 학습iteration수는 주어진 각각의 컴포넌트 CBP신경망 학습중 가장 길게 소요된 학습iteration과 이전 PCA 및 PCA<sup>2</sup>에서의 학습 iteration을 합친 것이 된다. 각각의 신경망에서 사용된 여러역전파 신경망의 학습상수  $\lambda$  및  $\theta$ 는 동일하게 각각 0.9 및 0.7로 주었으며, 허용에러는 0.001로

허용 iteration은 50,000이다.

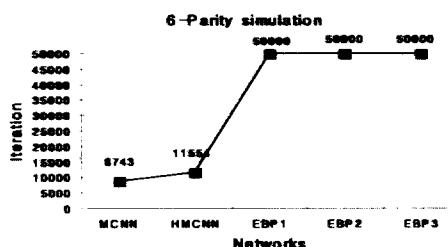


그림 12. 6-Parity 문제 학습속도 비교

Fig. 12. 6-Parity problem learning speed comparison results.

그림 12에서 패턴을 분할 학습한 MCNN 및 HMCNN신경망은 12,000회 이내에 학습을 완료한 반면 기존 EBP신경망은 허용iteration을 넘어 학습이 안됨을 알 수 있다.

표 1에서는 6-Parity 문제 해결을 위한 전체 신경망의 크기를 나타낸다.

표 1. 6-Parity 신경망 크기비교

Table 1. 6-Parity networks size comparison results.

Nets.	MCNN		HMCNN		EBP1	EBP2	EBP3
	PCA	CBP	PCA <sup>2</sup>	CBP			
Size. (con.)	18	64	90	64			
total Size. (con.)		82		154	105	140	175

표 1에서 MCNN보다 HMCNN의 크기가 더 큰 이유는 패턴의 분할을 좀 더 유연하게 해주는 패턴분할 부분이 추가되었기 때문이다.

HMCNN신경망은 MCNN신경망과 비교하여 패턴 분할시 패턴간의 유사도 또는 거리가 밀접함에 따라서 패턴을 분할하는 반면에 기존 MCNN신경망은 트리결정 방법에 따라 패턴을 분할한다. 따라서 MCNN의 경우는 서로 가까운 곳에 인접한 패턴일지라도 서로 다른 신경망에서 학습하는 경우가 발생하거나, 고립된 패턴이 발생하기도 한다. 그러므로 전체 신경망의 크기가 MCNN보다 좀더 크더라도, 패턴의 분할 및 일반화 능력에 있어서는 HMCNN이 우수하다.

### 3. 영문자 인식 문제

마지막으로 비교적 학습이 간단한 영문자 인식에 대

하여 실험한다. 영문자 인식에서는 우선 특징추출을 거치지 않은 25차원의 입력차원을 갖도록 임의로 만들어진 입력패턴을 PCA를 통하여 4차원으로 감소시킨 후 분할된 패턴별로 각각의 CBP신경망으로 전파되어 학습한다.

Dis의 변화에 따른 부류의 수 및 패턴분할부분의 학습iteration을 나타내면 그림13와 같다.

영문자 인식의 경우 주어진 패턴은 26개이다. 따라서 패턴간의 distance의 차이가 크고, 분할되어지는 부류의 수 또한 그려 크지 않음을 알 수 있다. 따라서 CBP신경망에서 가장 패턴의 수가 작은 부류는 2개, 가장 큰 부류는 4개의 패턴을 갖는다.

Alphabet character recognition simulation (pattern partitioning)

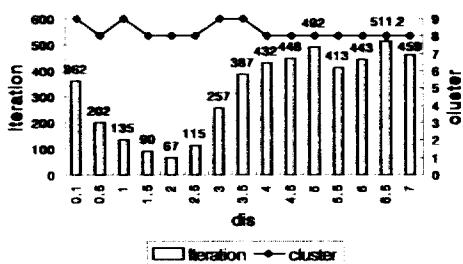


그림 13. 초기 Dis 값의 변화에 따른 부류의 수 및 학습 iteration

Fig. 13. Number of clusters and learning iterations results for various initial Dis.

그림 13에서 초기Dis의 값이 2일 때 67회의 학습 iteration으로 8개의 부류를 갖는다.

그림 14는 영문자 인식문제에 대한 각 신경망의 학습iteration을 나타낸다. 이전의 두 문제와는 달리 비교적 단순한 문제이므로 일반 EBP신경망에서의 학습이 가능하다.

그림 14에서 MCNN신경망은 최대 3개의 패턴을 갖는 12개의 CBP신경망에서 학습을 수행하였으며, 제안한 HMCNN신경망에서는 초기 Dis가 2일 때 8개의 CBP신경망에서 학습을 수행하였다. 그림14에서 학습상수  $\lambda$  및  $\theta$ 는 각각 0.9 및 0.7로 주었으며, 허용 에러는 0.001로 허용 iteration은 50,000으로 하였다. 그림 13에서 중간뉴런의 수가 5인 EBP3신경망은 허용학습횟수를 초과하였으나 중간뉴런의 수가 7개 및 10개인 EBP2 및 EBP1은 30,000번 이내에 수렴함을 알 수 있다.

표 2. 영문자 인식 신경망 크기 비교  
Table 2. Alphabet character recognition problem networks size comparison results.

Nets.	MCNN		HMCNN		EBP1	EBP2	EBP3
	PCA	CBP	PCA2	CBP			
Size. (con.)	75	160	299	100			
total Size. (con.)		235		399	510	357	255

표 2에서는 6-Parity 문제 해결을 위한 전체 신경망의 크기를 나타낸다. 표 2에서 MCNN은 최종적으로 235개의 연결을 가지며, 이중 CBP 컴포넌트 신경망은 초기 5 개로 구성된 후 최종 11개의 신경망으로 결정되어진다. 이에 반하여 HMCNN신경망은 초기 PCA<sup>2</sup>신경망의 패턴분할부분에서 출력뉴런의 수를 변화시키다가 PCA<sup>2</sup>의 학습이 완료된 시점에서 전체 신경망의 크기는 결정되어 진다.

MCNN신경망보다 HMCNN신경망의 연결 수가 많은 이유는 CBP신경망에서 학습되는 패턴을 적응적으로 분할시켜주는 패턴분할 영역이 첨가되었기 때문이다. 그러나 실제 학습을 담당하는 CBP컴포넌트 신경망의 전체 연결 수는 감소되었다.

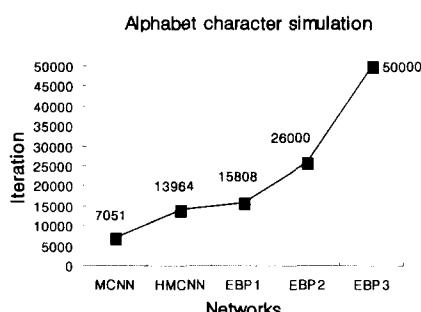


그림 14. 6-Parity 문제 학습속도 비교  
Fig. 14. Alphabet character recognition problem learning speed comparison results.

제안한 HMCNN신경망은 기존의 MCNN신경망에 부가하여 주어진 문제의 패턴분포에 따라 적절하게 패턴을 분할한 후 원 패턴의 크기보다 작은 입력차원을 갖는 소규모의 컴포넌트 신경망에서 학습을 수행함으로써 MCNN보다 좀 더 적응적이고 유연한 학습이 가능토록 하였으며 일반화 능력의 단점을 보완하였다. 또한 기존 EBP신경망보다 학습속도 및 전체 신경망

의 크기를 감소시켰으며, 제안한 신경망을 병렬 학습한 실제의 속도는 기존 신경망에 비하여 크게 향상됨을 알 수 있다. 결과적으로 제안한 신경망은 입력패턴의 크기가 크고 및 패턴의 수가 많은 복잡도가 높은 대규모 문제에 있어서 기존의 네트워크의 크기가 고정적이고 정형적인 신경망에 비하여 매우 효과적으로 적용할 수 있다.

## V. 결 론

본 논문에서는 다수개의 입력패턴을 효과적으로 분할하는 새로운 패턴 분할 신경망 알고리즘을 제안하고 이를 다중 컴포넌트 신경망에 부가하여 학습성능을 향상시킨 하이브리드 다중 컴포넌트 신경망(HMCNN)의 구조 및 학습알고리즘을 제안하였다.

패턴분할신경망은 기존의 패턴 분할 신경망과는 상이하게 학습도중 분할되는 부류의 수를 조정할 수 있도록 출력뉴런의 수를 적응적으로 변화시킬 수 있게 하였다. 또한 하나의 부류에 속하는 패턴의 수 또한 적절하게 조정하여 이를 이용하는 각각의 CBP신경망의 입력패턴의 수가 유사하게 결정되도록 하여 전체 CBP신경망의 학습속도가 균형을 이루도록 하였다.

제안한 신경망은 기존 MCNN신경망과 비교하여 패턴분할에 있어서 좀 더 적응적으로 분할가 가능하며, 기존 에러역전파 신경망과 비교하여 전체 학습속도의 향상 및 전체 크기를 감소시켰다. 특히 제안한 신경망은 입력패턴을 크게 감소시키고 이를 모듈화된 다수개의 신경망에서 학습시킴으로써 대규모의 데이터 및 신경망의 크기를 요구하는 문제에 적합하다. 앞으로의 연구방향은 제안한 하이브리드 다중 컴포넌트 신경망에서 PCA의 성능을 향상시키는 방법에 대한 연구와, 각 CBP신경망으로 부가되는 패턴들, 즉 하나의 부류에 속하는 패턴들만의 다중 PCA변환을 수행함으로써 학습의 일반화와 학습속도를 향상시키는데 있다.

## 참 고 문 헌

- [1] S.Haykin, Neural Network A Comprehensive Foundation, Mcmillan Publishing Co., 1993.
- [2] J.M. Zurada, Introductio to Artificial Neural Systems, WEST Publishing Co.,

- 1992.
- [ 3 ] K.A. MarKo, J.James, J.Dosdall, and J.-Murphy, "Automotive Control System Diagnostics Using Neural Nets for Papid Pattern Classification of Large Data Sets," in Proc. 2nd Int. IEEE Joint conf. On Neural Networks, Washington D.C., pp 13-17, 1989.
  - [ 4 ] M.M. Gupta, D.H. Rao, "Neuro-Control Systems: A Tutorial", IEEE Press, pp 1-44, 1994.
  - [ 5 ] D.E. Rumelhart, G.E., Hinton and R.J. Williams, "Learning Internal Representation by Error Propagation," in Parallel Distributed Processing, vol 1, MIT Press, MA, pp 318-362, 1986.
  - [ 6 ] DeMers, D.G. Cortell, "Nonlinear Dimensionality reduction," Advances in Neural Information Processing Systems, San Mateo, CA:Morgan Kaufmann, pp 580-587, 1993.
  - [ 7 ] T.Kohonen, Self-Organization and Associative Memory, 3rd ed., Springer-Verlag, Berlin, 1989.
  - [ 8 ] T.Kohonen, "Improved versions of learning vector quantization," Int. Joint Conf. Neural Networks, I, pp 545-550, 1990.
  - [ 9 ] 이현수, 박찬호, "PCA를 이용한 디중 컴포넌트 구조설계 및 학습," 대한전자공학회논문지, 33 권 B편 제10호, pp 107-119, 1996
  - [ 10 ] R.J. Schalkoff, Pattern Recognition: Statistical, Structural and Neural Approaches, John Wiley and Sons. inc., 1991.
  - [ 11 ] K.Fukunaga, Introduction to Statistical Pattern Recognition, Academic press, 1992.
  - [ 12 ] A.Cichocki, R.Ubenhauen, Neural Networks for Optimization and Signal Processing, WILEY Publishing co., 1992.
  - [ 13 ] E.Oja, "Principal components, minor components and linear neural networks," Neural Networks, vol 5, pp 927-935, 1992.
  - [ 14 ] T.D. Sanger, "An optimality principal for unsupervised learning," Advances in Neural Informatin Processing Systems, D.S., Touretzkey editor, Morgan Kaufman, pp 11-19, 1989.
  - [ 15 ] E., Oja, "A simplified neuron model as a principal component analyzer", J. Math. Biology, vol 15, pp 267-273, 1982.
  - [ 16 ] C.H. kwen, C.H. Park, H.S. Lee, "Efficient strategies for Error Updating to Improve Performance Backpropagation Learning," Proc. of ICNN, PerthAustralia, pp 2825-2830, 1995.
  - [ 17 ] J.Park, I.W. Sandberg, "Universal Approximation using Radial Basis Function Networks", Neural Computation, vol 3, pp 246-257, 1991.
  - [ 18 ] R.A. Jacobs, M.I. Jordan, A Competitive modular connectionist Architecture, Advances Neural Information Processing Systems, pp 763-773, SanDiego, CA:Morgan Kaufmann, 1991.
  - [ 19 ] Y.H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison Wesley Co., pp 197-222, 1989.
  - [ 20 ] J.hertz, A.Krogh and R.G. Palmer, Introduction to the Theory of Neural Computation, Addison Wesley Co., 1991.

## 저자소개

朴 賢 虎(正會員) 第33卷 B編 第10號 參照  
현재 경희대학교 전자계산공학과

李 顯 洊(正會員) 第33卷 B編 第10號 參照  
현재 경희대학교 전자계산공학과