

論文2000-37TE-3-16

블록 군집 신경 회로망을 이용한 분류

(Classification Using Convex Clustering Neural Network)

金榮俊*, 朴容震**

(Young-Jun Kim and Yong-Jin Park)

요약

본 논문에서는 기존의 Fuzzy C-Means, Nearest Neighbouring Classification, FMCCNN, Fuzzy-ART 등에서 사용하였던 정형에 근거한 분류에서 유기될 수 있던 판단 오류를 최소화하기 위해 단 한가지의 형태적 특징을 갖고 있는 정형에 의존하지 않고 분류를 수행하는 방법을 제안하고자 한다. 이를 위해 본 논문에서는 주어진 학습 데이터로 학습하는 과정에서 블록 다면체를 적응적으로 생성하고 다면체의 구조를 수정하는 퍼지 신경회로망을 설계하였다. 따라서, 본 방법은 순차적으로 입력되는 데이터를 분류하여 패턴 유형들을 생성하는 기능을 갖게된다. 본 방법의 유용성을 증명하기 위해, Hyperbox를 정형으로 하는 FMCCNN과의 다양한 시뮬레이션 비교를 수행하였다.

Abstract

This paper proposes a classification method using an amorphous prototype to minimize classification error caused by such fixed-prototype-based methods as Fuzzy C-Means, Nearest Neighbouring Classification, FMCCNN, and Fuzzy-ART. For this method, a new fuzzy neural network is introduced, in which a convex polytope is generated or adaptively reshaped to classify the given datum into a proper group. Thus, this method contains a function to classify sequential data set. To show the validity of this method, various numerical experiments including comparison results with FMCCNN are presented

1. 서론

패턴 분류는 많은 공학 문제들에 있어서 매우 중요한 요소이다. 음파탐지기, 레이더, 지진 감지, 로봇틱스, 화상 인식 및 모든 분석 응용시스템들은 상황을 정확하게 분석하기 위한 능력을 필요로 한다. 따라서, 시스

템의 입력과 출력의 관계를 결정하기 위하여 분류기(classifier)를 자주 사용한다. 이러한 넓은 응용범위 때문에, 패턴 분류¹⁻³⁾는 많은 응용 부분에서 중요한 연구 대상이 되어왔다. 기존의 FMCCNN⁴⁻⁷⁾이나 Fuzzy ART⁸⁾에서는 아래 그림 1과 2의 예에서 보이는 것처럼 유형의 정형이 Hyperbox로 고정됨에 따라서 부류의 표현이 많은 오차를 포함하게 된다⁴⁾. 본 논문에서는 이러한 오차를 줄이기 위해 학습하는 과정에서 입력값에 적응적으로 블록 다면체(Convex Polytope)⁴⁾를 생성하고 이러한 여러 다면체를 하나의 유형으로 발생시키는 퍼지 신경 회로망을 이용한 분류기를 제안한다. 여기서, 각각의 블록 집합은 반드시 같은 부류의 데이터들로 이루어져있으며, 이러한 여러 블록 퍼지 집합으로 이루어진 집합은 특정한 한 부류를 나타낸다. 이

* 正會員, 惠泉大學 정보시스템 계열
(Dept. of Schools of Information Systems Heychon College)

** 正會員, 漢陽大學校 電氣, 電子 工學部
(Division of Electrical and Computer Engineering, Hanyang University)

接受日字:2000年8月21日, 수정완료일:2000年9月18日

를 망 구조로 구현하기 위해 다음과 같은 용어들을 정의한다.

부유형 : 동일한 부류의 입력 데이터를 나타낼 수 있는 블록 집합.

유형 : 부유형 집합들로 이루어진 집합.

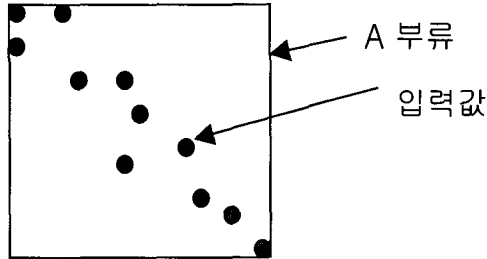


그림 1. hyperbox에 의한 부류표현
Fig. 1. Class Representation by hyperbox.

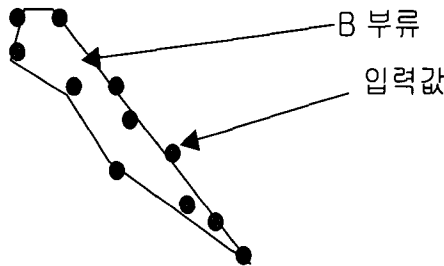


그림 2. 볼록 다면체에 의한 부류표현
Fig. 2. Class Representation by Convex Polytope.

이러한 계층적인 분류기를 신경망으로 구성하기 위해서 두 가지의 층이 필요하다. 즉, 매 입력에 대해 적응적으로 볼록 집합을 구하는 부유형층이 필요하며, 이렇게 구해진 각각의 부유형을 하나의 같은 유형으로 묶는 유형층이 필요하다. 부유형층에서는 외부에서 주어지는 각각의 입력패턴이 각 부유형에 얼마나 소속되는지를 나타내기 위해 비선형 소속도 함수가 사용되며, 부유형층으로부터의 출력은 각 부유형의 소속도가 된다. 이러한 각 부유형의 소속도는 유형층의 입력으로 사용되고, 유형층에서는 이러한 소속도들중 가장 큰 소속도를 입력이 해당 유형에 소속되는 소속도로 사용된다.

본 방법은 매 입력 패턴에 대해 순차적으로 수행되며, 온-라인 특성을 가진다. 구성된 볼록 집합은 신경 회로망의 출력 층인 유형층을 정의하기 위한 부유형이며, 해당 유형으로 완전히 소속되는 소속도 1.0을 가

지며, 해당 패턴 유형 공간의 일부 영역을 차지한다. 이러한 내용들의 각각은 본 논문에서 보다 상세히 설명된다. 본 논문은 다양한 공학 분야에서 널리 사용되고 있는 패턴 분류의^[1-3] 예로서 신경회로망을 이용한 볼록 부유형 집합들을 모음으로 유형들을 생성하는 볼록 집합을 기반으로 한 분류 기법을 제시 및 개발한다.

II. 볼록 다면체를 이용한 유형 생성

1. 입력 패턴이 볼록 부유형내에 포함되는지를 결정 및 내확장성 측정

어떤 부유형으로 부터 입력 패턴을 분리시키는 초평면의 존재성은 [4]에서 증명되었고, 이를 조사할 수 있는 알고리즘은 다음과 같이 요약되어 질 수 있다.

```

Separating Hyperplane Detection(SHD)
Given an input pattern x ;
FOR all vertex vectors in k-th sub-class DO
    Obtain expansion unit vectors  $e_i^k$  ;
END FOR
Arbitrarily select expansion vector  $e_i^k$ , as a
reference vector among m expansion unit
vectors ;
FOR all remaining expansion unit vectors DO
    Compute resultant unit vector  $f_i^k$  ;
END FOR
Set interior = FALSE
(The input pattern is initially assumed to lie
outside the k-th sub-class) ;
Set resultant unit vector pair number  $\iota = 0$  ;
WHILE(interior=FALSE AND  $\iota \neq$  total number
of resultant unit vector pairs)
    Increment  $\iota$  ;
    IF  $f_i^k \cdot f_j^k \leq 0$  THEN
        Set interior =TRUE
        (The input pattern is in the interior side of the
k-th sub-class) ;
    END IF
END WHILE
    
```

따라서, 입력패턴 x 가 블록 다면체의 안이나 밖에 놓이는지를 결정할 수 있다. 이 부유형 수정 후 블록 유지는 입력패턴 x 가 첨가 될 경우 부유형 변경의 형태를 결정하기 위한 방법을 설명한다. 이 Convexity 테스트 알고리즘은 다음과 같이 정리되어 있다^[4].

Convexity Test(CT) Algorithm

```
Set convexity = TRUE ;
Construct an arbitrary sub-class  $C_i$  with all vertices
except for  $a_i^k$ 
(the arbitrary sub-class need not be convex) ;
Apply the SHD algorithm with  $a_i^k$  and vertices of
 $C_i$  ;
IF interior = TRUE THEN
    Set convexity = FALSE ;
END IF
```

Convex sub class Modification (CCM) Algorithm

```
Construct a new sub-class with input pattern  $x$  as
a new vertex and all vertices of k-th
sub-class
Apply the CT algorithm with input pattern  $x$  and
the k-th sub-class ;
IF convexity = FALSE THEN
Discard vertex due to  $x$  from k-th sub-class ;
ELSE
FOR all remaining vertices of k-th sub-class DO
    Apply the CT algorithm ;
    IF convexity = FALSE THEN
        Store vertex  $a_i^k$  as a vertex to be removed ;
    END IF
END FOR
Discard all vertices that are stored to be removed ;
END IF
```

계속적으로 부유형을 확장하는 것은 바람직하지 않다. 그러므로, 대부분의 부류화 알고리즘에서 적용되어 온 것처럼, 앞에서 언급한 부유형 확장 제한 조건이 필요하다. 그러한, 제한조건을 얻기 위해서, 부유형의 크기를 얻기 위한 방법이 필요하다. 이를 위해서, 내 확장

성 측정으로서 부유형의 크기를 식(1)과 같이 정의한다^[1].

$$\varepsilon_{intra}(A^k) = \sum_{\ell_1=1}^{N_1} \cdots \sum_{\ell_n=1}^{N_n} m(\ell_1, \dots, \ell_n) \cdot \prod_{\gamma=1}^n I_{\gamma} \quad (1)$$

이는 블록 부유형의 내부에 위치한 크기 셀들의 수를 포함한다. 이것은 부유형의 각 차원을 위한 작은 축 간격 I 를 선택함으로써 수행되어지며, 그리고 실제와 측정된 크기사이의 에러가 최소인 내부 크기 셀들의 수를 누적함으로써 수행된다.

여기서, $I_{\gamma} = \max_{i \neq j} \{ |a_n^k - a_m^k| / N_{\gamma} \}$ 는 γ -번째 차원을 위한 셀 길이이다. N_{γ} 은 간격들의 수이다. 그리고 $m(\cdot)$ 은 n-D 이진화 값된 매트릭스이다. 위 식에서, $m(\cdot)$ 의 값은 내부 부유형 셀일 경우 1이고, 그렇지 않을 경우 0이다. 크기 계산의 이 방법은 아마도 낮은 차원성의 문제 일 경우 만족된다. 그림 3은 부유형의 크기 파라미터 값에 따른 부유형 내 확장성에 의한 변화된 결과를 보인다.

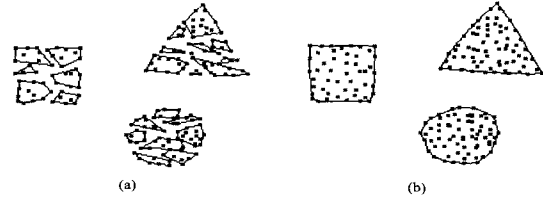


그림 3. (a) 크기 파라미터 : 0.95 (b) 크기 파라미터 : 0.84
Fig. 3. (a) size parameter : 0.95 (b) size parameter : 0.84

2. 퍼지 블록 분류기 신경망

신경회로망 분야에는 서로 다른 다양한 접근 방법들이 있는데, 본 논문에서는 패턴공간에서 부분 집합들을 생성 하므로서 유형 결정 경계들을 만드는 방법을 사용했다. 즉, 블록 집합 기반의 신경회로망은 이러한 방법을 기초로 한 신경회로망 분류기의 예이다. 각각의 블록 부유형 퍼지 집합 A^k 는 식(2)과 같이 순서화된 집합으로 정의 할 수 있다.

$$A^k = \{X, V^k, f(X, V^k)\} \quad (2)$$

여기에서, X 는 입력 패턴집합이고, V^k 는 k-번째 블록 다면체를 위한 꼭지점 집합이며, $f(X, V^k)$ 는 X 와 k-번째 부유형에 소속될 소속함수를 나타낸다. 따라서

j-번째 패턴 유형 c_j 는 식(3)과 같이 각 서브 유형 A^k 의 합집합으로 표현 될 수 있다.

$$C_j = \bigcup_{k \in j} A^k \quad (3)$$

여기서 소속함수 $f(X, V^k)$ 는 부유형의 내부와 외부에 따라 다르게 소속도가 할당되어 지도록하였다. 즉 외부에서는 $\mu^k(x)$ 를 사용하고 내부에서는 1.0을 할당하였다. 여기서, $\mu^k(x)$ 는, $0 \leq \mu^k(x) \leq 1$ 이며, 식(4)와 같이 정의될 수 있다.^[1]

$$\mu^k(x) = \frac{1}{1 + \frac{d(x, x_0)^{\gamma(x_0)}}{\lambda}} \quad (4)$$

$$d(x, x_0) = \text{dist}(x, A^k) \quad (5)$$

$$\gamma(x_0) = \frac{N_k}{1 + \frac{\epsilon_{\text{intra}}(A^k)}{e_0}} \quad (6)$$

여기서 식(5) $d(x, x_0)$ 는 소속도가 1.0인 멤버 x_0 에서 입력패턴 x 사이의 거리이며, λ 는 $\mu^k(x)=0.5$ 가 되는 x_0 로부터의 거리를 나타내고, 식(6) $\gamma(x_0)$ ($\gamma(x_0) > 0$)는 소속함수의 퍼지함(fuzziness), 혹은 기울기의 급격함을 나타낸다. 또한, $\text{dist}(x, A^k)$ 는 입력 패턴 x 에서 블록 부유형 A^k 로부터 최소거리를 나타내며, N_k 는 A^k 의 내부안에 있거나 A^k 상에 위치한 입력 패턴들의 수 이고, e_0 ($0 < e_0 < 1$)는 부유형 크기 측정 $\epsilon_{\text{intra}}(A^k)$ 를 위한 가중치 벡터이다. 결론적으로 다음식 (7)과 같이 k-번째 부유형을 위한 입력 패턴 x 의 소속도 값이 주어지게 된다.

$$f(X, V^k) = \begin{cases} 1 & \text{부유형내부} \\ \mu^k(x) & \text{부유형외부} \end{cases} \quad (7)$$

이제, 입력 패턴 x 가 최대로 소속하는 가장 확장 할 수 있는 부유형을 결정하기 위해, 동일한 유형 특성을 갖는 부유형들 사이의 가장 큰 소속도 값을 얻는 것이 필요로 하다. 이 것은 식(8)와 같다.

$$K^* = \arg \{ \max_k \{ f^k(x) \} \} \quad (8)$$

여기서 K^* 는 입력패턴 x 에 가장 큰 소속도를 가지는 확장할 수 있는 부유형을 나타낸다. 앞에서 언급한

소속도의 할당을 사용함으로써, 부유형은 다음과 같이 수행되어진다. 들어오는 입력 패턴을 위하여, SHD 알고리즘은 패턴이 현재 부유형들의 내부에 놓이는지를 결정하기 위해 사용되어 진다. 만일 패턴이 부유형들의 하나에 속한다면, 부유형 확장은 이러한 패턴일 경우 수행되지 않는다. 그렇지 않으면, 가장 확장할 수 있는 부유형은 K^* 를 구함으로써 얻을 수 있다. 이제, 만일 $f^{k^*}(x)$ 가 미리 정의된 소속도 임계값 θ 를 초과한다면, 블록 부유형 변형(CCM) 알고리즘은 부유형 K^* 를 확장하기 위해 적용되어진다. 다음, 확장된 부유형의 크기는 최대 확장할 수 있는 부류 크기 β 사이에 놓이는지를 보기 위해 검사되어진다. 마침내, 부유형 K^* 의 확장은 몇몇 이웃하는 다른 유형의 부유형들을 가지고 오버랩을 일으키는지를 보기 위해 검사된다, 그러한 상황을 검출하기 위해, K^* 의 내부에 놓여 있는지 어떤지를 조사함으로써 쉽게 결정되어 질 수 있다. 오버랩의 경우 이웃하는 부유형 꼭지점들의 몇몇 확장된 부유형의 이러한 경우에서, SHD 알고리즘은 여기서, 또한 이용되어 질 수 있다. 만일, 위조건의 하나라도 만족하지 않는다면, 확장할 수 있는 해당 부유형은 주어진 입력 패턴일 경우 확장하지 않으며, 패턴 x 는 새로운 점의 부유형이 된다. 위 방법을 정리한 퍼지 블록 부유형 확장(FCSE) 알고리즘은 아래와 같다.

Fuzzy Convex Sub Class Expansion (FCSE) Algorithm

Set all necessary initial parameter ; $(\lambda, e_0, \theta, \beta)$

Given an input pattern x ;

FOR all sub classes **DO**

 Apply SHD algorithm ;

END FOR

IF interior = **FALSE THEN**

 Set expandable sub class number $i = 0$;

FOR all sub classes **DO**

 Obtain $\gamma(x_0)$ using (7) ;

 Obtain $\mu^k(x)$ using (5) ;

IF $\mu^k(x) >$ threshold value **THEN**

 Increment i ;

 Sort sub class indexes in decreasing

 order according to corresponding

$\mu^k(x)$ values ;

```

    END IF
  END FOR
  Set expand = FALSE ;
  Set expandable sub class index i=0 ;
  WHILE (expand = FALSE AND i ≠ total
    number of expandable sub class)
    Increment i ;
    Expand sub class i using CCM algorithm ;
    Text following conditions for each sub class ;
    1) Determine if size of expanded
      sub class < β using (2) or (3) ;
    2) Determine if no overlap is detected
      between the expanded sub class and
      the other current sub classes using SHD
      algorithm ;
    IF the above two conditions are satisfied THEN
      Set expand = TRUE ;
      k* = expanded sub class i ; END IF
    END WHILE
    IF expand = FALSE THEN
      Create a new point sub class ;
    END IF
  END IF

```

신경회로망은 그림4에서 보여지는 것과 같이 퍼지 블록 분류기를 처리한다. 이 세층의 신경회로망안 에서 각각의 F_B 마디는 블록 퍼지 집합(블록 부유형)을 나타낸다. 여기서, F_A 와 F_B 연결들은 블록 집합의 꼭지 점들로 구성되어진다. 이러한 모든 블록 집합의 꼭지점 들은 V 매트릭스에 저장 된다. V 매트릭스는 다음과 같이 식(9)처럼 나타낼 수 있다.

$$V^k = [V_1, V_1, \dots, V_n] \quad (9)$$

여기서 각각의 V_n 은 블록 부유형들을 구성하며, 하나의 블록 부유형을 구성하는 꼭지점들은 식(10)과 같이 나타낼 수 있다.

$$V_1 = \{ v_{11}, v_{12}, v_{13}, \dots, v_{1n} \} \quad (10)$$

그리고 F_B 전달함수는 소속함수를 구하는 식(4)로부터 정의된 블록 소속함수 이다. k-번째 F_B 의 상세한

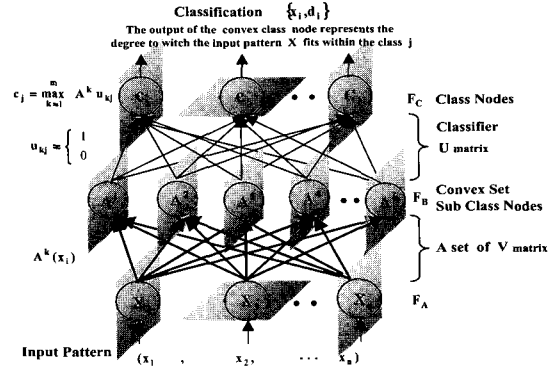


그림 4. 퍼지 블록 신경망 구조
Fig. 4. Fuzzy Convex Neural Network Architecture.

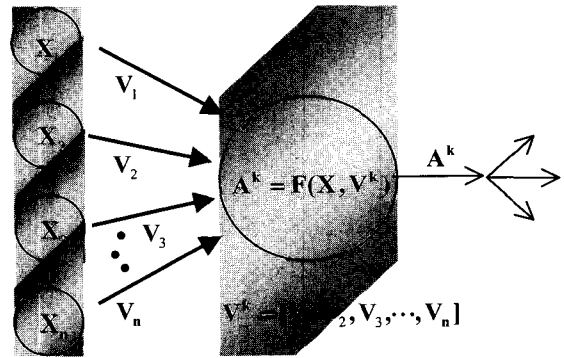


그림 5. F_b 마디를 위한 블록 집합의 처리
Fig. 5. Processing of convex set for F_b node

표시는 그림 5에서 보여진다. 그림 6은 F_B 를 통과한 분류기 처리의 예를 보여준다.

F_B 와 F_C 를 연결하는 마디들은 이진수 값들이고, U 매트릭스에 저장되어 있다. F_B 와 F_C 를 연결하기 위해 할당하기 위한 값들의 조건은 다음과 같다.

$$u_{kj} = \begin{cases} 1 & \text{if } A^k \text{ is a convex set for class } c_j \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

여기서, A^k 는 k-번째 F_B 마디이고, c_j 는 j-번째 F_C 마디이다. 각각의 F_C 마디는 해당 유형을 나타낸다. F_C 마디의 출력은 입력 패턴 x 가 j유형에 얼마나 해당되는지의 정도를 나타낸다. 즉 입력 패턴은 유형을 정의한다. F_C 마디들의 각각을 위한 전달함수는 블록 퍼지 집합 값들의 퍼지 합을 수행한다. 이 식의 정의는 다음과 같다.

$$c_j = \max_{k=1}^m A^k u_{kj} \quad (12)$$

위와 같이 이러한 신경회로망 처리를 이용하여, 들어오는 입력 패턴에 대하여 해당하는 신경회로망을 정의할 수 있다. 마지막으로 아래와 같이 본 알고리즘을 정리하였다.

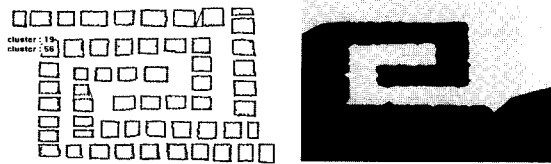


그림 6. F_B 를 통과한 분류기 처리
Fig. 6. processing of Classifier passed by F_B

Fuzzy Convex-Set-Based Classification (FCSC)
Algorithm

```

Learn to total input pattern  $x$  ;
Define class (  $C_j$  ) with learned total input
pattern ;
Create an initial point sub class(  $A^k$  ) for
first training input pattern  $x$  ;
FOR remaining training input patterns  $x$  DO
Apply FCSE algorithm ;
Create sub classes (  $A^k$  );
END FOR
FOR Given an total input patterns  $x$  DO
FOR created all sub classes(  $A^k$  ) DO

Apply SHD algorithm;
IF interior = TRUE THEN
FOR defined classes (  $C_j$  ) DO
IF sub class (  $A^k$  ) is a convex set
for class(  $C_j$  ) THEN
 $U_{kj} = \text{TRUE}$ ;
Set  $C_j = \max A^k$ ;
END IF
END FOR
END IF
END FOR
END FOR
END FOR
    
```

III. 실험 및 결과

제안된 방법의 효율성을 입증하기 위하여 몇 가지 실험 결과를 제시한다. 실험에서 사용된 두 알고리즘은 FCSC와 FMCCNN^[4]를 알고리즘으로 한 시뮬레이션 결과이다. 모든 입력패턴 벡터들은 1600개의 데이터 집합으로 그림 7과 같은 입력패턴들을 사용하였으며, 이 데이터 집합은 두 유형으로 나누어 학습을 시킨 다음 각각의 유형에 해당하는 모든 입력패턴에 유형 레이블을 주었다. 이는 그림 8에서 보여진다. 이러한 데이터 집합 중에서 순서에 상관없이 랜덤하게 10%, 15%,

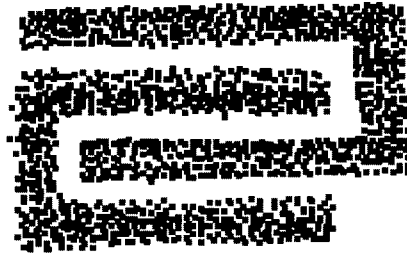


그림 7. 모든 데이터 집합
Fig. 7. All Data Set.

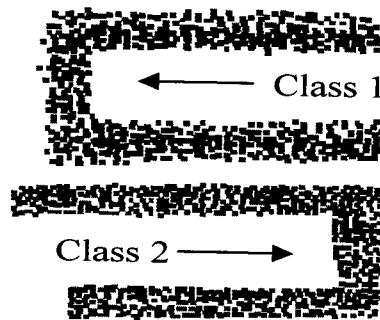


그림 8. 두 유형 데이터 집합
Fig. 8. Two Class Data Set.

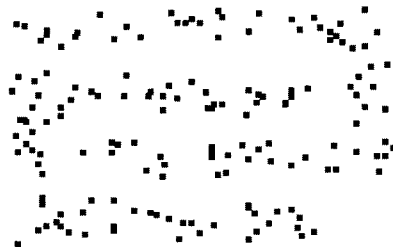
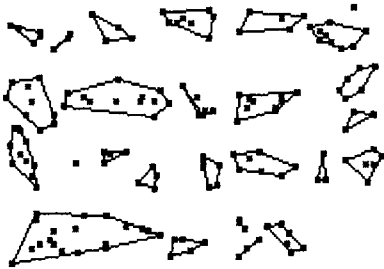


그림 9. 트레이닝 데이터 집합:160(10%)
Fig. 9. Training Data Set:160(10%).

30%, 50%의 트레이닝 데이터 집합을 추출하여 분류기를 구성해 분류한 결과를 얻었다. 그림 9는 10%의 트레이닝 데이터 집합의 예를 보여준다.

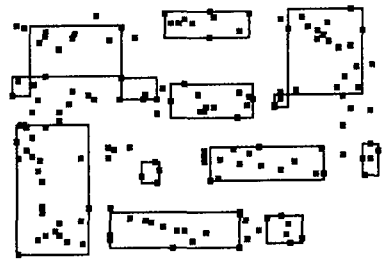
제일 먼저 트레이닝 데이터 집합 160(10%)을 두 알고리즘 각각의 부유형 크기 파라미터를 이용하여 적용한 결과 생성된 부유형들의 수에서 FCSC는 26개, FMMCNN는 32개를 생성하였다. 이는 그림 10, 11에서 보여진다. 두 결과에서 FCSC에서는 99.68%라는 정확도를 얻을 수 있었으며, FMMCNN에서는 91.37%의 정확도를 얻었다. 보다 명확한 실험 결과를 얻기 위하여 트레이닝 데이터 집합 30%를 사용하였다. 그림 12, 13에서와 같이 FCSC는 부유형 개수 37개에 99.93%라는 정확도를 얻었으며, FMMCNN은 부유형 개수 39개임에도 불구하고 94.5%라는 정확도를 얻었다. 트레이닝



Training Data Set :160
Sub Class Size : 0.9
Sub Class : 26
Correct rate : 99.68 %

	R	T	1	2
1			800	5
2			0	795

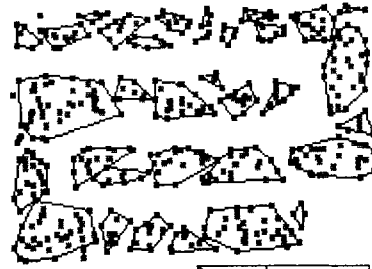
그림 10. FCSC : 10%을 이용한 표본 자료의 분류 결과
Fig. 10. FCSC : Classification result of 10% sample data.



Training Data Set :160
HyperboxSize : 0.135
Sub Class : 32
Correct rate : 91.37 %

	R	T	1	2
1			724	62
2			76	738

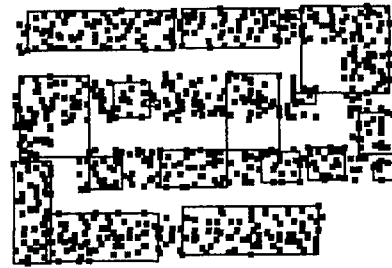
그림 11. FMMCNN : 10%을 이용한 표본 자료의 분류 결과
Fig. 11. FMMCNN : Classification result of 10% sample data.



Training Data Set :480
Sub Class Size : 0.945
Sub Class :37
Correct rate : 99.93 %

	R	T	1	2
1			800	1
2			0	799

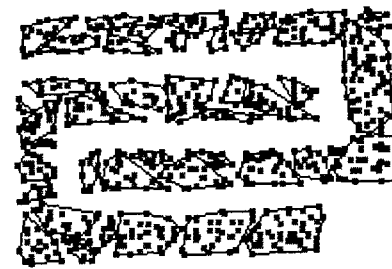
그림 12. FCSC : 30%을 이용한 표본 자료의 분류 결과
Fig. 12. FCSC : Classification result of 30% sample data.



Training Data Set :480
HyperboxSize : 0.155
Sub Class : 53
Correct rate : 91.43 %

	R	T	1	2
1			681	18
2			119	782

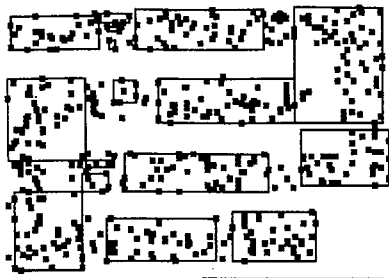
그림 13. FMMCNN : 30%을 이용한 표본 자료의 분류 결과
Fig. 13. FMMCNN : Classification result of 30% sample data.



Training Data Set :800
Sub Class Size : 0.965
Sub Class :47
Correct rate : 100 %

	R	T	1	2
1			800	0
2			0	800

그림 14. FCSC : 50%을 이용한 표본 자료의 분류 결과
Fig. 14. FCSC : Classification result of 50% sample data.



Training Data Set : 800
HyperboxSize : 0.155
Sub Class : 39
Correct rate : 94.5 %

		T	
R	1	765	53
	2	35	747

그림 15. FMMCNN : 50%을 이용한 표본 자료의 분류 결과
Fig. 15. FMMCNN : Classification result of 50% sample data.

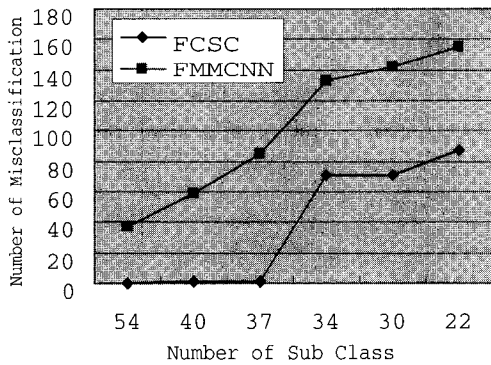


그림 16. 30% 표본자료를 이용한 경우의 소유형별 FMMCNN과 FCSC의 분류오차
Fig. 16. Classification error of each FMMCNN and FCSC for 30% sample data.

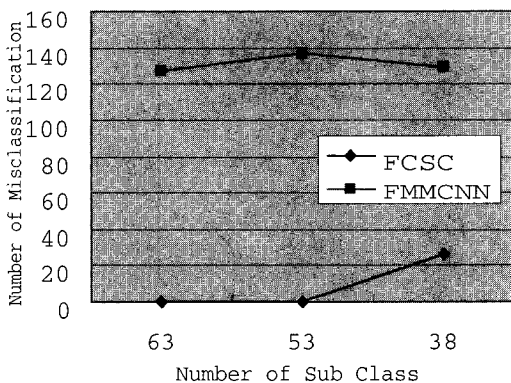


그림 17. 50% 표본자료를 이용한 경우의 소유형별 FMMCNN과 FCSC의 분류오차
Fig. 17. Classification error of each FMMCNN and FCSC for 50% sample data.

데이터집합을 보다 증가시킨 그림 14, 15는 50%를 사용했을 때의 결과를 보여준다. 마지막으로 그림 16, 17은 생성된 서브 유형 수에 따른 잘못된 분류 에러수를 보여 주며, 이는 그래프로 나타난다. 결과에서 FCSC 알고리즘이 FMMCNN에 비하여 보다 좋은 결과를 얻을 수 있음이 보여진다.

IV. 결론 및 연구과제

본 논문에서는 기존의 FMMCNN이나 Fuzzy ART에서 Hyperbox를 정형으로 이용한 분류방법보다 판단오차를 줄이기위해 입력패턴에 적응적으로 분류가 가능한 블록 다면체를 기반으로 한 새로운 분류 기법을 제안한다. 여기서 블록 다면체를 적응적으로 생성하기 위하여 퍼지 신경망 분류기를 구성하고, 이를 이용한 패턴 유형들을 생성한다.

또한, 본 논문에서는 FCSC와 FMMCNN를 비교 관찰할 수 있도록 모의실험을 수행한 바 서론에서 주장한 단일형태의 정형인 Hyperbox를 적용했을 때보다 블록다면체를 정형으로 하였을 때 분류판단오류를 상당히 줄일 수 있음을 입증하였다.

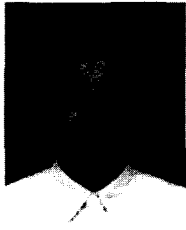
즉, 실험결과에서 동일한 표본자료에 대해 FCSC는 부유형 개수 37개에 99.93%라는 정확도를 얻었으며, FMMCNN은 부유형 개수 39개 임에도 불구하고 94.5%이라는 정확도를 얻었다. 추후 연구과제로서는 블록다면체 크기를 정확히 얻을 수 있는 방법이 개발되어야 하며, 적절한 파라미터 값들의 선택에 관한 분석이 이루어져야 할 것이다.

참 고 문 헌

- [1] P.Devijver and J.Kittler, "Pattern Recognition : A Statistical Approach.Englewood Cliffs," NJ : Prentice-Hall, 1982.
- [2] K.Fukunaga, "Introduction to Statistical Pattern Recognition," New York: Academic Press, 1972.
- [3] S. Abe and M. Lan, "A Method for Fuzzy Rules Extraction Directly from Numerical Data and its Application to Pattern Classification," *IEEE Trans. Fuzzy Syst.*, Vol. 3, pp.18-28, Feb. 1995.
- [4] I. H. Suh, J. H. Kim, and F. C. H. Rhee,

- "Convex-Set-Based Fuzzy Clustering," *IEEE Trans, Fuzzy System*, Vol. 7, pp.271-285, June. 1999.
- [5] P. Simpson, "Fuzzy Min-Max Neural Networks Part1 : Classification," *IEEE Trans. Neural Net-work*, Vol. 3, pp.766-786, Sept. 1992.
- [6] P. Simpson, "Fuzzy Min-Max Neural Networks Part 2 : Clustering," *IEEE Trans. Fuzzy Systems* Vol. 1, pp.32-45, Feb. 1993.
- [7] P. Simpson, "Fuzzy Min-Max Neural Networks," in *Proc. Int. Joint Conf. Neural Network*, pp.1658-1669, Nov. 18-21, 1991.
- [8] G.Carpenter, S.Grossberg, and D. Rosen, "Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by Adaptive Resonance System," *Neural Network*, Vol. 4, pp.759-771, 1991.

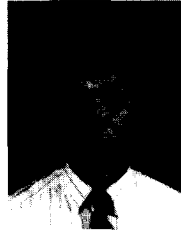
저 자 소 개



金榮俊 (正會員)

1986년 한양대학교 전자공학과 졸업. 1990년 한양대학교 대학원 전자공학과 공학석사. 1994년 한양대학교 대학원 전자공학과 박사과정 수료. 1996년~현재 해천대학 정보시스템 계열 조교수. 주관심 분야: 컴

퓨터 통신, 분산 시스템, 초고속 통신, 이동데이터 통신



朴容震(平生會員)

1969년 와세다 대학교 전자통신공학과 졸업. 1971년 와세다 대학원 공학석사. 1978년 와세다 대학원 공학박사. 1979년~현재 한양대학교 전기전자공학부 교수. 1983년~1984년 Univ. of Illinois, Urban 전산학과

방문 교수. 1991년~1992년 영국 Kent 대학 방문교수. 주관심분야: 컴퓨터 통신, 분산시스템, 초고속 통신, 이동데이터 통신