

유전 목 지도의 동적 확장 (Dynamic Extension of Genetic Tree Maps)

하성욱^{*} 권기항^{**} 강대성^{***}
(Seong wook Ha) (Kee hang Kwon) (Dae seong kang)

요약 본 논문에서는, 인식될 데이터에서 최적 특징을 구성할 수 있는 새로운 신경망 구조인 동적 유전 트리맵(DGTM)을 제안한다. DGTM은 기존의 신경망(neural networks)에서 고려되지 못한 데이터의 특징(feature)에 대한 중요도를 유전 알고리즘(genetic algorithm)으로 구성하고, 특징의 우선순위에 따라 트리 구조를 도입한 GTM(genetic tree-map)을 적용한다. 데이터의 유사성에 따라서 신경망의 뉴런이 동적으로 분리되고 병합될 수 있도록 동적인 기능을 갖는 DGTM(dynamic GTM)으로 확장한 방식을 제안한다.

키워드 : 패턴 인식, 동적, 트리맵, 유전 알고리즘

Abstract In this paper, we suggest dynamic genetic tree-maps(DGTM) using optimal features on recognizing data. The DGTM uses the genetic algorithm about the importance of features rarely considerable on conventional neural networks and introduces GTM(genetic tree-maps) using tree structure according to the priority of features. Hence, we propose the extended formula, DGTM(dynamic GTM) has dynamic functions to separate and merge the neuron of neural network along the similarity of features.

Keyword : Pattern recognition, Dynamic, tree-maps, genetic algorithm

1. 서론

일반적으로 음성, 문자, 숫자, 지문 등의 패턴 인식은 신경망을 사용하여, 입력된 데이터에 대한 고유한 특징(feature)을 분류하는 방식으로 인식한다. 신경망에서는 입력된 데이터를 분류하기 위해 효율적인 특징을 요구한다. 대부분의 신경망은 추출된 특징을 우선순위나 중요도에 관계없이 뉴런의 구성요소로 사용하거나 실험을 통하여 얻어진 가중치(weight vector)를 사용하며, 얻어진 특징은 모두가 동일하게 훈련과 인식에 사용된다. 따라서, 입력 데이터에서 N개의 특징을 추출한다면, N개에 대응하는 신경망의 뉴런 구조가 갖추어 진다. 본 논

문에서는 이러한 특징을 효율적으로 이용하기 위해, 이들의 중요도를 유전 알고리즘(GA)으로 구한다. 유전 알고리즘은 어떠한 문제에 대해 전역탐색(global search) 기법을 사용하여 최적의 해를 구하는 방식으로, 본 연구에서는 입력된 데이터들에 대해 최적의 출력을 얻을 수 있도록 최적화된 가중치를 구한다. 유전 알고리즘을 사용한다면, 가중치를 사용하지 않거나 실험에 의해서 정해졌던 최적화되지 않은 기존의 신경망의 구조를 개선할 수 있다. 기존의 신경망에서는 N개의 특징 모두를 사용하여, 몇 개의 특징만으로 구분될 수 있는 데이터의 특성을 무시하게 된다. 예를 들어, 숫자 1이나 7은 N개의 특징에서 첫 번째와 두 번째 특징만으로 분류가 가능하고, 6과 9는 세 번째 특징까지 이용하여 분류가 가능하며, 나머지 숫자들은 그 이하의 특징을 이용하여 분류가 가능할 경우, 기존의 신경망 구조에서는 이러한 특성이 무시된다. 이를 해결하기 위해, 본 논문에서는 N개의 특징에서 최소 특징으로 분류될 수 있는 데이터로부터 최대 개수의 특징을 갖는 데이터까지 분류될 수 있는 구조를 갖도록 새로운 트리맵(tree-map)을 제안한다. 기존의 신경망에서는 같은 출력이 요구되는 데이터

· 본 연구는 정보통신부의 2000년도(계속과제) 대학기초연구지원사업의 지원으로 수행되었음.

* 정 회 원 : (주)이지하모니 기술연구소 연구원
president@easyharmony.com

** 비 회 원 : 동아대학교 컴퓨터공학과 교수
keekke0000@hotmail.com

*** 비 회 원 : 동아대학교 전기전자컴퓨터공학부 교수
dskang@daumnet.donga.ac.kr

논문접수 : 2001년 10월 9일

심사완료 : 2002년 3월 21일

가 확실히 다른 특징을 갖고 있음에도 불구하고 $N \times N$ 의 한정된 구조 때문에 특징이 서로 다른 데이터의 질층을 이용하게 된다. 예를 들어, 필기체의 3과 출판물 형식의 3은 동일한 출력을 갖지만, 특징에서는 굵기, 기울기, 넓이 등이 서로 다를 수 있다. 여기서, 질층은 훈련에 의하여 많은 입력을 갖는 데이터 부류로 조정되거나, 평균값이나 통계 수치에 의해 조정되어, 입력 데이터에 대한 오인식률이 증가된다. 또한, 최소 M 개의 뉴런들로 분류가 가능한 경우, M 보다 큰 $N \times N$ 의 신경망이 구성되면 공간 낭비가 심각할지라도 구조적인 특성으로 사용할 수밖에 없게 된다. 예를 들어, 19×19 에서 분류가 안되어 20×20 으로 인식이 될 경우, 최소 $19 \times 19 + 1$ 개에서 분류가 가능하다면, $20 \times 20 - 19 \times 19 + 1$ 의 공간이 낭비된다. 이는 $N \times N$ 의 크기가 증가될수록 공간 낭비가 심각해질 것이다. $N \times N$ 구조의 또 다른 심각한 문제는 구조가 고정되어 있기 때문에, 인식률이 낮은 경우 N 의 수치를 증가시켜 재훈련이 요구되므로 인식률이 만족되기까지 시간 낭비가 많다는 점이다. 본 논문에서 제안한 트리맵 구조는 오인식이 일어나는 단계에서 동적으로 뉴런의 개수를 증가시키며, 뉴런 사이의 벡터의 유사성으로 일정 문턱치(threshold)를 만족하면 병합할 수 있어 동적인 능력이 우수하다. 패턴인식의 인식률에 영향을 미치는 인자 중에서 가장 중요한 특징은 연구자의 선택에 의하여 사용되거나 무시될 수 있으며, 반복적인 실험을 통하여 적절한 특징을 사용할 수 있으나 최적의 특징은 될 수 없다. DGTМ을 이용한다면, 어떠한 데이터를 인식하는 경우 블랙박스로 구성된 시스템을 이용하여 원하는 출력을 얻을 수 있게 된다. II장에서는 유전 알고리즘을 이용한 특징 우선순위를 구성하는 방법을 설명할 것이고, III 장에서는 특징의 우선순위에 따라 트리맵을 구성하는 방법을 설명한다. IV장에서는 기존의 신경망과 본 논문에서 제안하는 방식의 실험결과를 비교할 것이며, 마지막으로 V장에서 결론을 짓겠다.

2. 유전 알고리즘을 이용한 특징 우선순위 부여

본 장에서 제안하는 유전 알고리즘[1][2]은 주어진 문제에 대해서 국부탐색(local search)이 아닌 전역탐색을 이용하여 최적의 해를 찾아내는 방식을 신경망에도 입하여 특징의 우선순위를 구하는데 사용한다. 일반적으로, 패턴인식에 사용되는 신경망은 순수한 입력 데이터를 사용하지 않고, 효율적인 인식을 위하여 입력 데이터의 특성을 이용하게 된다. 이때에 추출되는 특징은 인식률에 직접적인 영향을 미칠 수도 있으나, 대부분은 신경망 알고리즘의 훈련 과정에서 사용되는 반복 횟수, 뉴런

매트릭스(matrix) 크기, 연결 강도 벡터 등의 인자들에 따라 인식 결과가 달라진다. 여기서, 사용되는 인자들은 주로 계속적인 실험에 의하여 인위적으로 계산되거나, 비효율적으로 수동적인 방식으로 결정된다. 본 연구에서 제안하는 유전 알고리즘을 이용한 특징의 우선순위 결정은 인식에 영향을 미치는 여러 인자 중 연결 강도를 사용한다. 연결 강도벡터의 최적화를 바탕으로, 최적화된 벡터에 따라 우선순위를 부여한다. 여기서 사용된 연결 강도 벡터의 최적화 방식은 여러 신경망 알고리즘의 벡터 설정, MPEG-7의 색인 기술자(descriptor) 생성에 필요한 샷(shot) 검출, 입력 파형에서 원하는 파형을 얻어낼 때 새로운 여과기(filter) 계수 생성 등의 여러 분야에 적용 가능하다.

신경망 알고리즘의 연결 강도 벡터 최적화에서, 입력 벡터 $I_{f1}, I_{f2}, \dots, I_{fn}$ 는 N 개의 특징이며, 신경망의 뉴런은 $N_{f1}, N_{f2}, \dots, N_{fn}$ 으로 N 개의 벡터로 구성되고, 연결강도는 W_1, W_2, \dots, W_n 으로 N 개의 벡터로 구성된다. 여기서 사용되는 N 개의 연결 강도 벡터를 최적화하기 위해, 이들을 유전 알고리즘의 개체군으로 사용하여 유전 연산을 통해 최적화한다.

MPEG 스트리밍 비디오 데이터에서 특징을 추출한 후, 연결 강도를 적용하여 문턱치 이상이 되는 영상을 샷으로 검출하며 이때에 사용된 특징을 기반으로 MPEG-7 기술자로 생성할 수 있다. 스트림(stream)에서 한 프레임의 특징과 연결강도 최적화에 본 장에서 제안하는 방법을 적용할 수 있다.

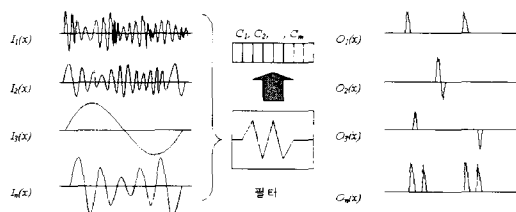


그림 1 목적 파형을 위한 최적 필터 구성

그림 1은 입력되는 파형에 대하여 목적 파형을 구성하는 필터를 나타낸 것이다. 그림에서 $I_1(x), I_2(x), \dots, I_n(x)$ 은 입력 파형, $O_1(x), O_2(x), \dots, O_n(x)$ 은 목적 파형이며, 필터는 C_1, C_2, \dots, C_m 의 M 개의 계수로 구성된다. M 개의 계수로 구성되는 필터는 수학적인 모델로서 구성되거나 실험에 의하여 적용할 수 있으나 새로운 필터의 생성은 여러 공학자와 수학자에게 매우 어려운 문제이다. 그러나, 본 장에서 제안하는 연결 강도의 최적화

에서 연결 강도 벡터 대신 필터의 계수를 적용한다면 입력 파형과 목적 파형만으로 최적의 필터 계수를 구성할 수 있게 된다.

유전 알고리즘은 자연 생태계에서 우성 인자가 열성 인자보다 생존확률이 높다는 기본 규칙을 이용한 컴퓨팅 알고리즘으로, 공학 문제에서는 주어진 문제에 대해 해로 근접시키는 적합도 함수에 따라 우성과 열성으로 구분하여, 우성의 해가 발생 빈도가 높도록 만든 알고리즘이다. 유전 알고리즘을 사용하는 데 있어서 가장 중요한 문제는 염색체의 정의와 적합도 함수 정의이다.

신경망 알고리즘을 사용할 때, 가장 중요한 것은 입력 패턴 선정, 특징점 선정, 강도 벡터 선정, 매트릭스 개수 등의 여러 구성요소가 존재한다. 본 연구에서는 특징점의 우선순위로 강도 벡터의 값을 사용한다. 따라서, 신경망의 구성요소 중 강도 벡터를 유전 알고리즘을 통하여 최적화하였다. 초기화 과정은 연결강도에 대한 개체군을 무작위(random) 값으로 구성한다.

개체의 구성요소는 연결 강도 벡터 W_1, W_2, \dots, W_n 에 대응되는 실수 값으로 구성되며, 개체는 M개의 개체군으로 초기에 무작위 값을 갖는다. 초기에 무작위 값을 갖도록 하는 것은 주어진 문제에 대하여 전구간에 대한 해를 찾기 위한 과정으로, 자연 생태계에서 무수한 유전 인자를 갖는 생물이 발생하는 것에 대응된다. 적합도 계산 과정은 모든 개체를 주어진 문제에 대해서 적합한지를 판정하는 과정으로 적합도 함수 정의가 요구된다.

일반적인 신경망 알고리즘에서는 연결 강도 벡터를 수작업을 통하여 수정하면서, 좋은 결과를 산출하는 벡터를 최적의 강도 벡터로 선정한다. 수작업을 진행하는 과정에서 벡터 W_n 값을 수정하여 대입한 인식이 좋아진다면 좋은 강도 벡터가 되며, 반대로 인식이 나빠진다면 수정하기 전의 벡터가 더 좋은 강도 벡터가 된다. 이를 근거로 유전 알고리즘에서 무작위로 생성된 벡터 개체군들의 적합도는 인식률로서 판단할 수 있다. 생성된 강도 벡터 개체군 P_a 의 인식률보다 P_b 의 인식률이 더 좋다면, P_b 가 주어진 입력에 대한 신경망의 강도 벡터로서 더욱 적합한 것으로 판단할 수 있다. 입력 데이터에 대한 출력력을 알 수 있다고 가정하고, 입력 데이터의 각 분류별로 10%의 데이터들의 각 특징 평균을 뉴런으로 구성한다. 신경망의 인식 알고리즘으로 전체 입력 데이터의 특징과 연결강도를 이용하여 뉴런을 통해 인식된 인식률을 적합도 함수로 사용한다. 다음과 같은 연결 강도 적합도 함수가 정의된다.

$$f(x) = \frac{R(x)}{M} \quad (\text{식 1})$$

위의 수식에서, M은 입력한 전체 데이터이며, R(x)는 인식된 개수를 나타낸다. R(x)는 M개의 입력 데이터에서 분류별로 10%의 데이터들을 무작위로 추출하여 이들에 대한 평균값으로 뉴런을 구성한 뒤, 한 개체의 연결강도를 적용하여 인식된 결과가 된다. 전체 개체군에 대해서 무작위로 구성되었던 뉴런은 동일하게 적용되어야 한다. 모든 개체군에 대해 적합도가 구해지면, 적합도 순위에 따라 정렬한다. 교배 과정에서는 실수형 데이터를 다루므로 유전 알고리즘의 최적 재교배 알고리즘을 선택하며, 생존 규칙은 룰렛 휠 방식을 사용한다. 돌연변이 과정은 최적 해가 아닌 범위에서 수렴되어지는 경우, 이를 극복하기 위한 유전 연산으로 유전 알고리즘의 전체 반복회수의 10% 주기마다 반복 적용한다. 재생산 연산에서는 교배되기 전의 개체군(N)과 교배·돌연변이 과정 이후 생성된 개체군(N)으로 구성된 전체($2 \times N$)에서 적합도에 준하여 최상위 해 집단 N개를 선택한다. 유전 연산 교배, 돌연변이, 재생산 과정은 유전 알고리즘에서 해를 구하는데 적합한 알고리즘으로 본 논문에서 선택한 방법 외에 다른 연산으로 대체될 수 있다. 최적 조건은 수렴되거나, 필요한 인식을 이상의 해가 나온 경우이다. 여기서, 필요 인식률은 인식할 데이터에서 요청하는 값으로, 예를 들어 99%의 값을 사용할 수 있다.

이상의 알고리즘을 통해 구해진 최적화된 연결강도를 오름차순으로 정렬한 값은 인식에 사용되는 중요도와 같다. 연결강도의 값이 클수록 데이터를 분류하는데 높은 기여를 하는 값이 된다. 따라서, 정렬된 연결강도를 특징의 우선순위로 사용한다면, 이들 값으로 다음 장에서 설명하는 트리(tree) 형태의 신경망 구조를 구성할 수 있다.

3. DGTM(dynamic genetic tree-map)

기존 연구에서 패턴 인식은 특징 추출에 단순한 방법을 사용하거나 기존의 방법을 개선시켜 사용하고 분류 및 인식을 위한 새로운 알고리즘을 제안하고 있다. 분류 및 인식을 위한 알고리즘에는 학습을 통한 특징점 분류 방식을 사용하는 신경회로망이 주로 이용되는데 그 중에서도 자가 분류 특성이 있는 SOM(Self-Organizing Maps)이 패턴 인식에 널리 사용되고 있다. N. V. Subba[3]는 패턴 인식을 위하여 MSOM(Modified Self-Organizing Map)과 LVQ(Learning Vector Quantization)를 적절히 조합하여 3차원 신경망(Three-dimensional neural network)을 제안하였다. 제안한 알고리즘은 기존 신경망 알고리즘을 단독 적용할 때 보다 학습시간, 인식률, 오인식률이 모두 개선되었고, SOM과

LVQ를 조합한 알고리즘은 동일한 인식률로 비교할 때 학습시간이 줄어들었으며, 오인식률이 감소하였다. SOM에 퍼지 영역을 접목한 FRSOM(Fuzzy Region Self-Organizing Map)[4]과 SOM의 구조를 대폭 개선한 DSNN(Doubly Self-Organizing Neural Net work)[5] 등에서도 기존의 SOM을 개선하여 시간과 인식률이 모두 개선되었다. CME(Combination of Multiple Experts)[4]에 의한 방법은 여러 가지 다른 알고리즘을 조합해서 인식 성능을 향상시키는 방법을 사용하였다. 투표(voting), Bayesian, Dempster-Shaper에 의한 방법이 주로 이용되는데, 투표에 의한 방법은 여러 가지의 알고리즘을 이용해 인식하고 최종 결과는 투표에 의하여 결정하는 비교적 단순한 방법인데 비하여 Bayesian 방법이나 Dempster-Shaper 방법에서는 분류를 위한 식이나 조건 등을 추가하여 구조가 더 복잡해진 반면 인식의 성능이 더욱 개선되었다.

일반적으로 신경망을 이용한 패턴 인식에서는 입력 데이터에 대한 모든 특징을 이용한다. 데이터의 특성에 관계없이 모든 특징이 인식에 사용되므로 비효율적이다. 이를 극복하기 위해 II 장에서 제안했던 유전 알고리즘을 이용하여 특징의 우선순위를 구하여, 데이터 분류를 효율적으로 개선할 수 있다. 예를 들어, 첫 번째 우선순위의 특징이 "최대 수직선 길이"인 경우, '1' '4' '7'이 갖는 숫자의 특징과 '0' '8'이 갖는 특징은 확실히 구분되며, 이후, 이들로 파생된 노드에서 다음 우선 순위의 특징을 사용하여 구분할 수 있으므로 인식의 정확도는 더욱 높아질 것이다.

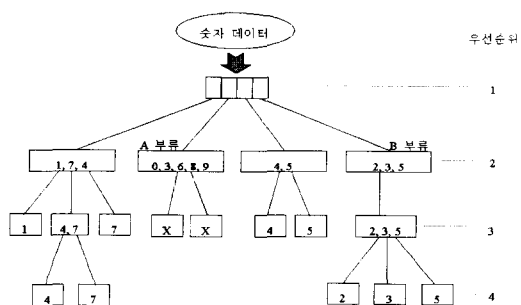


그림 2 우선순위에 따른 트리 구조

그림 2는 이를 도식화한 것으로, 첫 번째 우선순위의 특징을 통해 분류된 데이터들은 두 번째 우선순위의 특징에 따라 분류된다. 그림에서 우선순위는 트리 구조의 레벨을 나타낸다. 레벨 2에서 같은 숫자가 여러 부류에

포함된 것은 숫자들은 필기체나 출판물에 따라 각기 다른 특징을 갖게 되어 다른 부류에 속하는 것으로, 예를 들어 A 부류의 '3'과 B 부류의 '3'은 인식 후 결과는 같겠지만 특징이 다르므로 다른 부류에 속한 것이다. 이를 간과한 기존의 신경망에서는 데이터에 대한 정확한 분석 없이 무작위(random) 노드들을 구성하여 무수한 훈련과 실험을 통하여 노드들을 조정하지만, 특징이 다른 데이터를 같은 결과로 유도하는 식에 따라 노드가 최적화되지 못하는 경우 매트릭스 크기를 늘린다거나 사용되는 파라미터를 조정하여 최적의 결과가 나타날 때까지 무한히 반복되는 실험이 요구된다. 따라서, 훈련 과정이 끝나서 인식 결과가 나오기 전에 중단하는 경우, 입력 데이터에 대해서 노드들의 맵이 변경되는 추이에 대한 통계를 분석할 수 없다.

신경망 훈련은 맵의 변화가 수시로 변경되어 다음 단계에서 변경되는 맵을 예측할 수 없으며, 인식을 변화량은 수렴되지 못하는 상황으로 파라미터를 조정하여 재 실험이 요구된다. 인식을 변화량이 대체적으로 수렴되는 경우도 최적이라 볼 수 없다. 보통의 경우, 훈련과정에서 인식률이 최소가 되었던 지점의 노드들을 저장할 수 있으나 최적이라 할 수 없으며, 데이터의 특성을 정확히 분석하지 못한다. 본 장에서 제안하는 트리맵은 이러한 불합리한 인식의 훈련 과정을 개선하여 부류에 속하였던 노드의 특징이 변하지 않으며, 오 인식이 발생하는 위치에서는 다음 레벨의 노드를 분리하여 인식률에 변화는 주지 않는다. 일반 신경망의 훈련 과정에서는 최선으로 적용된 맵을 사용하여 노드의 값은 변동되지 않으며, 맵에 존재하지 않는 값이나 잘못된 출력은 오인식으로 분리된다. 반면에, 본 알고리즘의 인식과정에서는 맵에 존재하지 않는 값이나 잘못된 출력이 발생하는 부분은 동적으로 맵을 확장할 수 있다. 또한 새로운 특징이 필요한 경우 마지막 노드에 추가할 수 있다. 이와 같은 동적인 확장과 병합 기능, 유전 알고리즘을 이용한 우선 순위 선정, 트리맵 구조를 갖는 DGTM의 자세한 알고리즘은 다음의 1절 훈련 알고리즘과 2절 인식 알고리즘에서 설명하겠다.

3.1 DGTM의 훈련 알고리즘

DGTM의 훈련 알고리즘은 인식을 위해 사용될 표본 데이터의 특성을 트리맵 구조로 형성하는 단계와 형성된 트리맵에서 출력 결과가 같은 데이터의 특징을 병합하는 과정을 갖고 있다. 병합에 사용되는 문턱치(threshold)는 트리맵의 레벨에 따라 차등 적용되므로 인식에 유용성을 갖게 된다. DGTM의 훈련 알고리즘은 반복 회수를 단 1회 요구하며, 병합 단계를 거치면 최적

화된 DGTM 트리맵이 형성된다. 알고리즘에 사용되는 구조체는 다음과 같다.

```

TDGTNode = record
  Min, Max : float;
  Result : integer;
  Next : PDGTM;
end;
PPDGTMNode = ^^TDGTNode;

TDGTM = class
  Node : PPDGTMNode;
  No : integer;
  Order: integer;
  procedure Attach(Node: TDGTNode);
  procedure Detach(Node: TDGTNode);
end;
PDGTM = ^TDGTM;
    
```

TDGTNode는 DGTM의 맵의 각 노드들을 정의한 레코드이며, 각 구분 범위의 최소(Min)·최대(Max)값, 마지막 결과(Result) 값, 다음 맵의 포인터로 구성된다. TDGTM은 레벨에서의 한 맵을 구성하며, 각 레벨에서 사용되는 범위에 따라 동적인 노드, 노드 개수, 특징 순서와 노드를 추가 및 삭제하는 함수를 갖는 클래스이다.

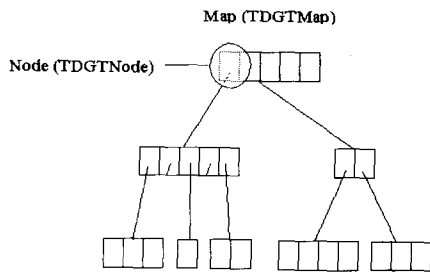


그림 3 DGTM 구조

그림 3은 DGTM의 구조를 나타낸 것으로, 첫 번째 Map이라는 TDGTM형 변수이며, 각각의 TDGTNode 형 노드를 갖고 있다. 아래에는 DGTM의 훈련 알고리즘이다.

```

procedure SGTMTaining;
begin
  repeat
    if i = 0 then
      CreateAllLevelMap;
      SettingNode;
    
```

```

else begin
  if FinalLevel and NotEqualOutput then begin
    if NoMoreFeatures then Break
    else RequiredNewFeature;
  end;

  if OverScope then begin
    SettingNode;
    CreateNewLevelMap;
  end else begin
    UpdateNode;
    MergingNode;
  end;
end;
until i >= MAX_INPUT
FinalizingTreeMap;
end;
    
```

훈련 알고리즘은 처음에 입력되는 데이터의 특징 개수에 따른 레벨별 맵을 구성(CreateAllLevelMap)하고, 레벨내의 첫 번째 노드를 설정(SettingNode)한다. 입력되는 데이터들은 마지막 레벨의 특징을 사용하여 같은 출력을 내지 못하는 경우(FinalLevel and NotEqual Output), 구분을 위한 새로운 특징이 요구되므로 프로세스를 중단(RequiredNewFeature)한다. 그러나, 더 이상의 특징 추출이 어려운 경우, 오인식으로 판단하여 다음 프로세스로 넘어가도록 한다. 오인식이 발생하였을 경우, 현재의 특징 추출 알고리즘으로는 주어진 인식문제에 대한 해결 능력이 부족한 것으로 판단되는 것이다. 해당 레벨의 노드 범위를 벗어나는 경우(OverScope)는 노드를 설정(SettingNode)하고 새로운 맵을 구성(CreateNewLevelMap)한다. 입력되는 특징의 해당 레벨에서 범위의 교집합이 발생되면, 노드의 범위를 업데이트(UpdateNode)하고, 현재 레벨이하의 특징의 범위가 같은 경우는 서로의 노드를 병합(MergingNode)시킨다.

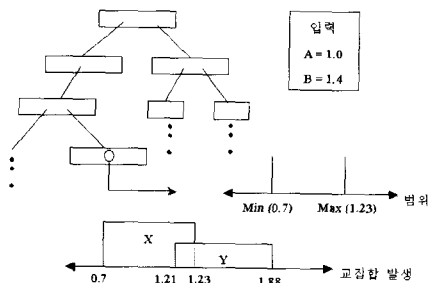


그림 4 노드의 범위 조정

그림 4에서 A 입력이 1.0으로 현재 노드에 속하는 경우 업데이트를 수행하며, B와 같이 범위를 벗어나는 경우, 새로운 맵과 노드를 설정하게 된다. 또한, 그림에서 X와 Y로 표현된 노드 범위의 교집합이 발생되면, 현재 레벨 이하의 맵 노드들이 서로 범위 내로 포함되며 동일한 출력이 발생하는 경우 하나의 노드로 병합한다. 본 알고리즘은 훈련 수행시간이 단 1회로 끝나며, 마지막으로 각 트리맵의 구조를 최적화하기 위한 프로세스(FinalizingTreeMap)를 수행한다. 최적화 프로세스에서는 현재 레벨 이하의 트리맵이 하나의 노드로 구성되는 경우, 인식의 속도를 위하여 마지막 노드를 현재 레벨로 구성하는 것이다.

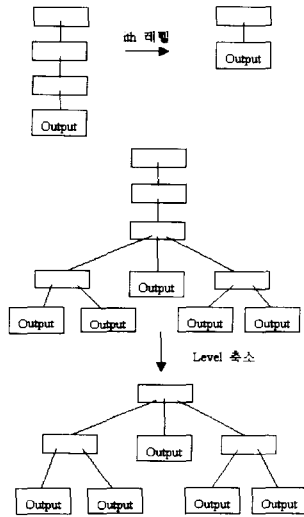


그림 5 DGTM의 최적화 프로세스

그림 5에서는 최적화 프로세스를 표현한 것으로, 상단의 그림은 현재 레벨이하에 하나의 노드만 갖는 맵을 줄여서 인식 과정에서 i 개의 특징만 사용할 수 있도록 노드를 줄이는 과정이다. 또한, 하단의 그림은 현재 레벨이 하나의 노드만을 가지는 경우, 이하의 특징을 상단으로 편입시켜 다음 레벨이 인식에 사용될 수 있도록 하는 병합하는 과정이다.

3.2 DGTM의 인식 알고리즘

DGTM의 인식 알고리즘은 훈련된 노드 맵을 이용하여 최적 노드 개수로 인식하는 알고리즘이다. 기존의 신경망에서는 훈련에 사용하였던 모든 특징 개수를 모두 동일하게 사용하여 인식하므로 프로세스가 비효율적이며, 클래스를 구분할 때 필요한 특징 외에도 불필요한

특징이 사용될 수 있다. 예를 들어, A, B라는 특징이 '가' 클래스 구분에 효율적으로 사용되고 A, C라는 특징은 '나' 클래스 구분에 효율적으로 사용되는 경우, 기존 알고리즘은 C란 특징이 '가' 클래스 구분에 비효율적일 지라도 사용할 수밖에 없으며, B라는 특징 역시 '나' 클래스 구분에 사용된다. 따라서, 모든 특징이 데이터 구분에 사용된다는 것이다. 반면에, DGTM은 현재 데이터 클래스에 분류에 필요한 특징만을 사용한다.

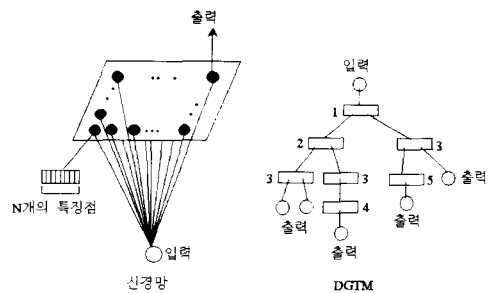


그림 6 신경망과 DGTM 구조

그림 6에서는 기존 신경망과 DGTM에서 사용되는 특징에 대한 비교이다. 좌측 그림의 기존 신경망에서 각 노드는 N개 특징 모두를 갖고 있으며, 대체적으로 입력된 특징과의 최소 거리 방식이 자주 사용된다. 우측 그림은 DGTM에 사용되는 구조로서 각 레벨에서는 동일한 순서의 특징이 사용되지 않으며, 맵의 좌우에 표시된 순번에 따르는 특징이 구분에 사용되므로, 훈련에서 분석되었던 입력 데이터의 특징에 따라 분류에 적절한 특징으로 효율적인 인식이 가능하다. 자세한 알고리즘은 다음과 같다.

```
function DGTMRognition(InputFeature: array of
feature): integer;
var
    Search: PDGTM;
    i: integer;
begin
    Search := root; // root는 훈련에서 생성된
DGTM의 루트 맵
while true do begin
    for i := 0 to Search.No-1 do begin
        if Search.Node[i].Min <=
            InputFeature[Search.Order] and
            Search.Node[i].Max >=
                InputFeature[Search.Order] then begin
```

```

if Search.Node[i].Result > 0 then begin
  DGTMRognition :=
    Search.Node[i].Result;
  exit;
end else Search := Search.Node[i].Next;
end;
end;
end
end;
알고리즘은 위와 같이 단순하다. 루트에서 해당 순서
특징의 범위를 찾아가는 방식으로 출력을 얻어낸다.
    
```

4. 실험

본 논문에서는 여러 패턴 인식 중 숫자 인식을 실험한다. 필기 숫자 인식에서는 입력 영상을 이진 영상으로 변환한 다음 1의 값을 갖거나 혹은 0의 값을 갖는 정보의 특징을 추출하는 방법이나 이진 영상에 대한 좌우 히스토그램 분포를 특징으로 추출하는 방법이 널리 사용된다. 히스토그램법은 구현이 간단하고 처리시간이 빠른 반면 모든 영상에 대하여 일률적인 특징을 추출하므로 많은 저장 공간이 필요한 단점이 있다. 이를 개선하는 특징 추출에 관한 연구로서 Gabor 특징에 의한 방법[11], 숫자의 윤곽 정보를 이용한 방법[12], 숫자의 구조적 정보(structural information)를 이용한 방법[13] 등 많은 방법이 제안되었다. Gabor 특징을 이용한 방법은 공간 영역과 공간-주파수 영역에 대한 최적 지점 영역화가 이루어지는 Gabor 필터를 이용한 인간의 시각 시스템과 유사한 구조로 특징을 추출하여 필기 숫자뿐만 아니라 인쇄체 한자 인식에도 적용되어 높은 인식률을 보였다. 필기 숫자의 구조적 정보를 이용한 방법에서는 입력 필기 숫자의 벡터 변환, 라인 변환, 라인 구조 해석, 그래프 구조 생성, 원형 매칭의 순서로 특징 벡터를 추출하며, 생성된 특징 벡터를 그래프 매칭법과 신경 회로망을 이용하여 인식하였다. 윤곽선 정보를 이용한 방법에서는 필기 숫자에 대하여 윤곽선 평탄화, CFAB(Contour Following Algorithm with Back tracking), 푸리에 색인 등을 사용하여 특징을 추출하고 기하학적 성질, 출발점의 위치, 통계학적 해석에 기반하여 인식하였다. 본 실험에는 제안한 인식 알고리즘을 숫자 인식에 사용한다. 사용된 숫자 데이터는 44000개를 무작위로 추출하여, 훈련과 인식에 사용하였다.

그림 7은 실험을 위해 작성한 시뮬레이션 프로그램으로 윈도 2000 환경 하에서 델파이 5.0으로 코딩되었다. 그림에서 첫 번째 유형은 해당 숫자 번호마다 200개의

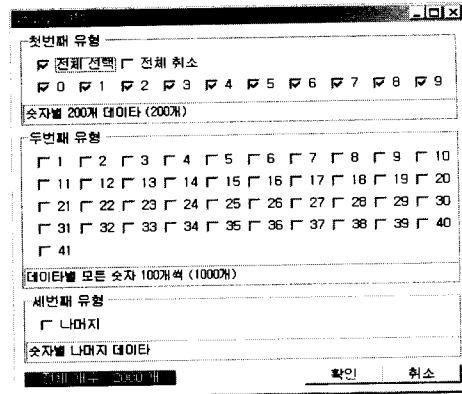


그림 7 시뮬레이션 프로그램의 데이터 로딩부

데이터가 저장된 텍스트형식의 파일이고, 두 번째 유형은 0~9의 숫자가 100개씩 1000개로 팩된 데이터(packed data)를 갖는 텍스트 파일이며, 세 번째 유형은 위의 두 가지 유형에 속하지 않는 나머지 데이터 1000개를 갖는 텍스트 파일이다. 훈련과 실험에 사용할 데이터를 체크 박스로 선택하여 사용할 수 있다. 사용된 데이터에서 모양에 대한 특징을 보존하기 위하여, 크기에 대한 평준화는 하지 않았다. 예를 들어, 가로에 대한 세로비를 특징으로 하는 경우, 모든 숫자를 동일한 가로 세로 크기로 평준화하는 경우 특징이 사라진다. 또한, 데이터의 크기가 큰 경우는 특징이 소멸될 수 있으며, 크기가 작은 경우는 불필요한 점들이 보충되므로 예상치 못한 에러를 생성할 수 있다. 실험에서 전처리 과정으로 입력된 데이터의 화질 열화나 해상도에 따른 데이터 소실을 보상하기 위하여, 모든 점에 대한 확장을 한번 수행하여 중심점을 추출하였다.

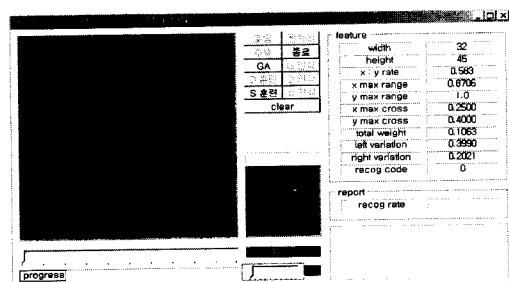


그림 8 시뮬레이션 프로그램의 전처리 과정

그림 8은 윈 데이터에서 중심점을 추출한 데이터이다. 전처리가 완료된 데이터에서 인식에 사용할 특징을 추

출한다. 실험에서는 넓이에 대한 높이 대비, 최대 가로 부피율, 최대 세로 부피율, 최대 가로 교차점 수, 최대 세로 교차점 수, 전체 데이터 분포율, 우측 변화율, 좌측 변화율로 8개의 특징을 추출하여 인식에 사용한다. 각 특징은 프로세싱의 효율을 위해 모든 특징을 0~1 사이로 평준화(normalization)하였다. 전처리 과정을 거친 데이터의 특징을 실험에서는 DGTМ과 신경망의 입력으로 사용한다. 본 논문에서 제안한 DGTМ 알고리즘의 검증에 위해 신경망의 한 부류인 SOFM(self-organizing feature maps)를 이용한다. DGTМ에 사용되는 특징의 우선순위 결정과 SOFM에서 입력 데이터와 히든 계층(hidden layer)의 노드사이의 가중치(weight vector) 결정에 유전 알고리즘을 사용한다. 유전 알고리즘으로 최적화된 가중치는 SOFM의 가중치로 바로 적용하며, 최적화된 가중치를 정렬하여 정렬된 수치에 따라 특징의 우선순위로 적용한다. 입력된 데이터에 대한 DGTМ 훈련은 루프가 필요하지 않으므로, 단 1회의 훈련으로 맵이 생성되었으며 입력 데이터에 대한 인식률은 100%의 결과를 나타내었다. SOFM은 루프가 완료된 이후, 훈련시의 인식 결과에 따라 루프 회수, SOFM의 맵 크기, 노드 조정에 필요한 α 값에 대한 수동으로 조절이 필요하다. 여기서의 α 값은 다음과 같은 인자 조정식에서 사용되는 인자이다.

$$f(x) = \frac{1}{\frac{s_r}{s_r - r}} \times \alpha \quad (\text{식 2})$$

식2에서 r 은 현재 반복 회수이며, s_r 은 전체 반복 회수이다. 조절된 인자에 따라 다시 루프 회수만큼의 반복이 요구되며, 연구원이 원하는 결과가 나올 때까지 훈련이 계속적으로 요구된다. 다음에 설명되는 표 I, II, III 은 SOFM의 반복 회수 = 1000, SOFM의 맵 크기 = 10×10 , α 값 = 0.911의 기본값을 근거로 인자를 조정한 인식 결과와 훈련 시간을 표현한 것이다.

표 1 SOFM 크기에 따른 인식 결과

SOFM 크기	10×10	20×20	30×30	40×40	50×50	60×60
인식률 (%)	33.50	56.15	73.30	88.45	92.00	95.75
인식 시간 (sec)	00:05:57	00:15:22	00:38:22	01:01:44	01:36:59	05:38:06

표 I은 기본 인자에서 SOFM의 맵크기를 10×10 에서 60×60 까지 10단위로 조정하면서 훈련한 결과이다. 10×10 에서 인식률은 33.5이며, 훈련에 필요한 시간은 5

분 57초이지만, 60×60 에서는 인식률이 95.75에 이르지 만 시간이 5시간 38분 6초로 맵크기를 크게 늘릴수록 인식 결과가 좋아지지만 훈련 속도가 현저히 많이 걸리게 된다.

표 2 SOFM 반복 회수에 따른 인식 결과

SOFM 반복회수	1000	2000	5000	10000
인식률 (%)	92.00	91.45	92.50	92.35
인식 시간 (sec)	01:36:59	06:23:28	10:37:09	19:36:21

다음으로 표 2는 기본 인자에서 SOFM의 반복 회수를 늘려가며 실험한 것으로 5000번에서의 인식률이 92.499인 반면, 10000번에서의 인식률은 92.3499로 더 떨어져 반복회수는 시간만 늘어날 뿐 인식 결과에는 영향을 미치지 않았다.

표 3 SOFM의 α 값 조정에 따른 인식 결과

SOFM의 α 값	0.0911	0.511	0.911	1.000
인식률 (%)	88.00	91.75	92.00	91.85

표 3는 기본 인자에서 각 노드 조정에 필요한 인자 조정식의 α 값을 일정 비율로 조정하며 실험한 것으로, 값이 높을수록 초기에는 인식률의 변화율이 높지만 올바른 해로 수렴하였으며, 값이 적을수록 인식률 변화율은 낮지만 최적 노드를 찾지 못하는 문제점을 나타내었다.

표 4 기존 신경망 알고리즘과 비교 평가

Methods	Recog. [%]	Error [%]	Reject [%]	Reliability [%]
Legault [14]	93.90	1.60	4.50	98.32
Krzyzak [15]	94.85	5.15	0.00	94.85
Suen [16]	93.05	0.00	6.95	100.0
Mai [17]	92.95	2.15	4.90	97.74
SOFM	92.35	3.15	3.50	96.70
DGTМ	98.95	0.00	1.05	100.0

표 4는 기존 인식 알고리즘 성능과 DGTМ을 비교한 결과이다. Suen 은 일반적인 특징과 인식 알고리즘을 사용하였고, Legault는 패턴 정보에서 윤곽선 정보와 파라미터 특징을 이용하여 인식에 사용하였으며, Krzyzak 은 패턴 분류를 위해 기존의 오류 역전파 알고리즘을

개선하였다. Mai 는 인식 단계별로 지식 기반 시스템을 이용하여 인식하였다. 본 논문에서는 특징점 추출 방식 개선이나 신경망 알고리즘 개선에 의한 방법이 아닌 새로운 구조를 갖는 DGTМ은 인식률이 높으며, 다른 알고리즘에서 필요한 인식 노드들의 수정 방식이 아니므로 학습 회수로 1회만 필요하다. 여기서, 신뢰도 (Reliability)는 다음 식과 같다.

$$f(x) = \frac{r(x)}{r(x) + e} \times 100 [\%] \quad (\text{식 3})$$

식3에서, $r(x)$ 는 인식률이며, e 는 에러율을 나타낸다. 제안된 DGTМ의 평균 인식률은 올바른 인식된 테스트 데이터 값을 전체 테스트 데이터 개수로 나눈 값에 대한 백분율이다. 이와 같은 결과에서, 여러 신경망 알고리즘은 인자 조정이 무수히 요구되며, 실험에 많은 시간이 소모되었으나, DGTМ은 실시간으로 동적 트리맵을 구성하여 시간 소모가 없으며 메모리 낭비도 줄일 수 있었다.

5. 결론

결론적으로 기존의 신경망은 다음과 같은 문제점을 갖게 된다.

- 훈련과정에서 입력되는 데이터 순서에 따라 인식에 영향을 미친다.
- 입력 데이터 분류시, 특징의 중요도가 고려되지 못한다.
- 최적의 노드 생성을 위해, 실험 완료 후 인식 결과에 따른 인자 조정이 요구된다
- 새로운 데이터가 추가될 경우, 반복되는 인자조정과 재훈련이 요구된다.
- 신경망의 초기맵 생성에 값에 따라 인식에 영향을 미친다.
- 실제 인식에서 사용되지 않는 노드를 발견하기 쉽지 않으며, 이로 인한 메모리 낭비가 심하다.
- 최적의 노트 생성 후, 인식 에러율이 다시 증가되므로 많은 시간이 소모된다.

본 논문에서 제안한 DGTМ은 이와 같은 문제를 해결하는 것으로, 입력되는 데이터의 순서에 관계없이 동적인 맵을 1회의 훈련과정으로 구성이 가능하여 시간을 절약할 수 있으며, 새로운 입력 데이터가 추가되는 경우 맵을 동적으로 확장 가능하다. 또한, 초기맵을 생성하지 않고, 입력되는 데이터에 따라 동적으로 맵이 확장되므로, 초기맵 생성에 따른 인식의 영향을 받지 않으며, 사용되지 않는 노드는 확장되지 않으므로 메모리 낭비가 없다. 입력되는 데이터의 특성에 따라 입력이 되어, 훈

련 과정에서는 인식률이 100%에 이른다. 그리고, 데이터의 특징에 대한 중요도를 GA를 통한 특징 우선순위로 고려한다. 일반적인 신경망은 입력 데이터에 대해 많은 시간의 실험과 인자 조정이 필요하지만, DGTМ은 입력 데이터에 따른 실시간 동적맵 구성이 가능하므로 일반적인 인식 문제가 발생하는 경우 최적의 특징을 갖는 맵을 구성한 인식 시스템을 구성할 수 있게 된다.

향후 연구 과제로는, 논리적으로 제안된 DGTМ을 이용한 패턴 네트워크 모델을 구성할 수 있도록 음성, 지문, 홍채, 숫자, 문자 등의 여러 인식 문제에 사용되는 특징을 기반으로 인터넷상으로 자동 인식 시스템을 구성하는 것이다

참고 문헌

- [1] D. E. Goldberg, *Genetic Algorithm in Search Optimization and Machine Learning*, Addison Wesley, 1989.
- [2] L. D. Davis, *The Handbook of Genetic Algorithms*, Van Nostrand Reinhold, 1991.
- [3] N. V. Subba Reddy and P. Nagabhushan, "A Three-dimensional Neural Network Model for Unconstrained Handwritten Numeral Recognition," *Pattern Recognition*, Vol. 31, No. 5, pp. 511-516, 1998.
- [4] Zheru Chi, Jing Wu and Hong Yan, "Handwritten Numeral Recognition Using Self-Organizing Maps and Fuzzy Rules," *Pattern Recognition*, Vol. 28, No. 1, pp. 59-66, 1995.
- [5] Sung-Bae Cho, "Recognition of Unconstrained Handwritten Numerals by Self-Organizing Neural Network," *Proceedings of ICPR'96*, pp. 426-430, 1996.
- [6] Y. S. Huang and C. Y. Suen, "A Methode Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 1, pp. 90-94, January 1995.
- [7] L. Xu, A. Krzyzak, and C. Y. Suen, "Method of Combining Multiple Classifiers and Their Application to Handwritten Numeral Recognition," *IEEE Transaction on Systems, Man and Cybernetics*, Vol. SMC-22, No. 3, pp. 418-435 1992.
- [8] T. K. Suen, R. Legault, C. Nadal, M. Cheriet, and L. Lam, "Building a New Generation of Handwriting Recognition Systems," *Pattern Recognition Letters*, Vol. 14, No.4, pp. 305-315, 1993.

- [9] J. Pearl, Probabilistic Reasoning in Intelligent System: Network of Plausible Inference, Morgan Kaufmann Publishers Inc., San Mateo, California, 1998.
- [10] M. D. McLeish, P. Yao, and T. Stirtzinger, "A Study on the Use of Belief Functions for Medical Expert Systems," Journal of Applied Statistics, Vol. 18, No.1, pp155-174, 1991.
- [11] Yoshihiko Hamamoto, Shunji Uchimura, Masanori Watanabe, Tetsuya Yasuda and Shingo Tomita, "Recognition of Handwritten Numerals Using Gabor Features," Proceedings of ICPR'96, pp. 250-253, 1996.
- [12] Dahai Cheng and Hong Yan, "Recognition of Handwritten Numeral Base on Contour Information," Pattern Recognition, Vol. 3, No. 1, pp. 235-255, 1998.
- [13] Sven Behnke and Marcus Pfister, "Recognition of Handwritten Digits Using Structural Information," Proceedings of ICNN'97, Vol. 4, pp. 1391-1396, 1997.
- [14] R. Legault and C. Y. Suen, "Contour Tracking and Parametric Approximations for the Digitized Patterns," Computer Vision and Shape Recognition : Singapore, pp. 225-240, 1989.
- [15] A. Krzyzak, W. Dai and C. Y. Suen, "Unconstrained Handwritten Character Classification using Modified Backpropagation Model," In Proc. 1st Int. Workshop on Frontiers in Handwriting Recognition, pp. 155-166, 1990.
- [16] C. Y. Seun, C. Nadal, R. Legault, T. A. Mai and L. Lam, "Computer Recognition of Unconstrained Handwritten Numerals," Proceeding of the IEEE, Vol. 80, No. 7, pp. 1162-1180, 1992.
- [17] T. Mai and C. Y. Seun, "A General Knowledge-based System for the Recognition of Unconstrained Handwritten Numerals," IEEE Trans. on Systems, Man and Cybernetics, Vol. 20, No. 4, pp. 835-848, 1990.



권 기 향
 서울대학교 컴퓨터공학과 학사. 미국 Georgia Tech Computer Science 석사. 미국 Duke University Computer Science 박사. 현재 동아대학교 컴퓨터공학과 조교수. 관심분야는 소프트웨어 공학, 인공 지능



강 대 성
 경북대학교 전자공학과 학사. 미국 Texas A&M 대학교 전기공학과 석사. 미국 Texas A&M 대학교 전기공학과 박사. 1984년 3월 ~ 1989년 6월 국방과학연구소 연구원. 1994년 7월 ~ 1995년 8월 한국전자통신연구원 선임연구원. 1995년 9월 ~ 현재 동아대학교 전기전자컴퓨터공학부 조교수. 관심 분야는 영상 처리, 영상 인식, MPEG



하 성 옥
 1997년 동서대학교 컴퓨터공학과 학사. 1999년 동아대학교 전자공학과 석사. 현재 동아대학교 컴퓨터공학과 박사과정. (주)이지하모니 기술연구소 연구원. 관심 분야는 인식 알고리즘, 멀티미디어