
Counterpropagation 알고리즘에서 퍼지 제어 기법을 이용한 경쟁층 설정 방법

김광백*

Setting Method of Competitive Layer using Fuzzy Control Method for Enhanced Counterpropagation Algorithm

Kwang-baek Kim*

요 약

본 논문에서는 개선된 CP 알고리즘에서 경쟁층의 수를 효율적으로 설정하기 위해 퍼지 제어 기법을 이용하여 경쟁층의 수를 결정하는 방법을 제안한다. 제안된 방법은 CP 알고리즘에 입력되는 패턴의 정보를 이용하여 퍼지 소속 함수를 설계하고 입력에 대한 소속도를 계산한 후, 퍼지 제어 규칙을 적용하고, Mamdani의 Max_Min 추론 방법으로 추론한다. 퍼지 추론을 통해 최종적으로 얻어진 값을 무게 중심법으로 비퍼지화 하여 최종적으로 개선된 CP 알고리즘의 경쟁층의 수를 결정하는데 적용한다.

제안된 방법의 학습 및 인식 성능을 평가하기 위해, 영문과 같이 다양한 패턴을 실험에 적용한 결과, 제안된 방법이 경쟁층의 수를 결정하는데 효과적임을 확인할 수 있었다.

ABSTRACT

In this paper, we go one step further in that the number of competitive layers is not determined by experience but can be determined by fuzzy control rules based on input pattern information. In our method, we design a set of membership functions and corresponding rules and used Max-Min reasoning proposed by Mamdani. Also, we use centroid method as a defuzzification.

In experiment that has various patterns of English inputs, this new method works beautifully to determine the number of competitive layers and also efficient in overall accuracy as a result.

키워드

경쟁층, 퍼지 제어 기법, Max_Min 추론, 비퍼지화

Key word

Competitive Layer, Fuzzy Control Method, Max_Min Reasoning, Defuzzification

I. 서론

1987년 로버트 헤트-닐센(R. Hecht-Nielsen)에 의해 제안된 CP(Counterpropagation) 알고리즘은 서로 다른 구조의 네트워크가 조합된 혼합형 모델이다. 중간의 은닉층은 경쟁적인 뉴런들로 이루어진 Kohonen의 경쟁층을 사용하고, 상위층은 Grossberg의 아웃스타 층으로 이루어져 있다[1]. 결합된 두 네트워크로 인해 각 층마다 서로 다른 학습 방법을 사용함으로써 다른 신경망 모델보다 비교적 빠른 학습 시간을 가능하게 한다. CP 네트워크는 3개의 층으로 구성되어 있다.

첫번째 층은 입력을 위한 입력층이고, 두번째 층은 Kohonen의 경쟁층이며, 세번째 층은 Grossberg의 아웃스타구조의 출력층이다. CP 알고리즘은 입력층에서 입력 패턴이 입력되면 Kohonen의 경쟁층에서는 뉴런 중 출력이 최대인 뉴런만이 경쟁에서 승리하여 활성화되며 승자 뉴런과 연결된 연결강도를 조정한다. 반면에 Grossberg의 아웃스타 학습법은 승자 뉴런으로부터 출력층의 출력 값을 구하고 승자 뉴런과 출력층의 연결 강도를 조정한다.

그러나 CP 알고리즘은 경쟁층에서의 학습시 다양한 패턴으로 구성된 경우에는 패턴 분류의 정확성이 낮아지고, 입력 벡터와 승자 뉴런으로 선택된 대표 벡터간의 차이와 승자 뉴런의 빈도수를 반영하지 않고 경험적으로 학습률을 설정함으로써 Kohonen의 경쟁층에서 불안정한 학습 결과를 보인다[2]. 경쟁층에서의 불안정한 학습은 결과적으로 CP 학습에 영향을 주어 학습 및 인식을 저하시키는 원인이 된다. 따라서 이러한 문제점들을 개선한 CP 알고리즘이 제안되었다[3]. 개선된 CP 알고리즘에서는 다수의 경쟁층을 설정하여 패턴 분류의 정확성을 높이고 입력 벡터와 승자 뉴런의 대표 벡터간의 차이와 승자 빈도수를 반영하여 학습률을 동적으로 조정하여 경쟁층에서의 학습이 안정적으로 진행되도록 하였으나, 응용 분야에 따라 다수의 경쟁층을 경험적으로 설정해야하는 문제점이 있다[4].

따라서 본 논문에서는 [3]의 방법(개선된 CP알고리즘)에서 경쟁층의 수를 효율적으로 설정하기 위해 퍼지 제어 기법[5]을 이용하여 경쟁층의 수를 설정하는 방법을 제안한다. 제안된 경쟁층의 수 설정 방법은 입력 패턴 종류에 대한 소속 함수, 입력 패턴 수와 패턴의 노드

수에 대한 소속 함수를 각각 설계하고, 이들 소속 함수에 대해 소속도를 계산한 후, 퍼지 제어 규칙을 적용하고, Mamdani의 Max_Min 추론 방법으로 추론한다. 퍼지 추론을 통해 최종적으로 얻어진 값을 무게 중심법으로 비퍼지화 하여 최종적으로 개선된 CP 알고리즘의 경쟁층의 수를 결정한다.

II. 개선된 Counterpropagation 알고리즘

본 논문에서 제안하는 개선된 CP 구조는 그림 1과 같다.

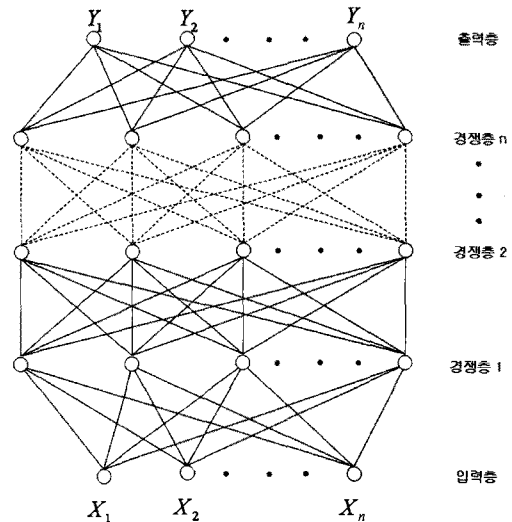


그림 1. 개선된 CP 알고리즘의 구조
Fig. 1 Improved CP Structure

개선된 CP 알고리즘의 학습 단계는 다음과 같다.

- 단계 1: 학습에 사용되는 인수를 초기화 한다.
- 단계 2: 입력층에서 입력 벡터를 제시한다.
- 단계 3: 입력 패턴의 정보를 이용한 퍼지 제어기법으로 경쟁층의 수를 설정한다.
- 단계 4: 입력벡터와 경쟁층의 노드간의 거리를 식(1)과 같이 계산한다.

$$s_j = \sum_{i=0}^{n-1} (x_i - w_{ij})^2 \quad (1)$$

식(1)에서 x_i 와 w_{ij} 는 각각 입력층의 입력 벡터와 경쟁층의 연결 강도이고, s_j 는 경쟁층의 출력 벡터이다. 그리고 입력 벡터와 경쟁층의 뉴런간의 거리는 유클리드 거리를 이용한다.

단계 5: 최소 거리에 있는 경쟁층의 뉴런을 승자 뉴런으로 선택하고, 승자 뉴런은 1, 나머지는 0으로 설정한다.

$$\begin{aligned} \text{if (Winner)} \quad z_j &= 1 \\ \text{else} \quad z_j &= 0 \end{aligned}$$

여기서 z_j 는 경쟁층의 승자 뉴런이다.

단계 6: 승자 뉴런과 연결된 연결 강도의 조정은 식(2)와 같이 계산한다.

$$w_{ij^*}(n+1) = w_{ij^*}(n) + \alpha(x_i - w_{ij^*}(n)) \quad (2)$$

식(2)에서 학습률 α 는 입력 벡터와 승자 뉴런으로 선정된 대표 벡터와의 차이와 승자 뉴런의 빈도수를 반영한 식(3)을 이용하여 학습률을 동적으로 조정한다.

$$\begin{aligned} \alpha &= f(e_{j^*}) + \frac{1}{f_{j^*}} \quad (3) \\ e_{j^*} &= \frac{1}{1 + e^{-(x_i - w_{ij^*})}} \end{aligned}$$

식(3)에서 j^* 는 j 번째 출력 노드가 승자로 선택된 뉴런을 의미하고 f_{j^*} 는 승자 뉴런으로 선택된 빈도수이고, $f(e_{j^*})$ 는 시그모이드 함수이다. 시그모이드 함수를 이용하여 입력 벡터와 승자로 선택된 대표 벡터간의 차이를 0에서 1사이의 값으로 정규화 하고, 승자 뉴런의 빈도수에 따라 최종 학습률을 구한다.

단계 7: 단계 3에서 설정된 경쟁층의 수만큼 단계 5와 단계 6를 반복한다.

단계 8: 출력층의 출력 벡터는 식(4)와 같이 계산한다.

$$y_k = \sum_{i=0}^{n-1} z_{j^*} \times v_{j^*k} \quad (4)$$

식(4)에서 z_{j^*} 와 v_{j^*k} 는 각각 최종 경쟁층의 출력 벡터와 출력층의 연결 강도이고 y_k 는 출력층의 출력 벡터이다.

단계 9: 출력층에서 연결 강도는 식(5)를 적용하여 조정한다. 그리고 모멘텀 학습법[6]을 이용하여 연결 강도의 변화량은 식(6)과 같이 계산한다.

$$v_{j^*k}(n+1) = v_{j^*k}(n) + \Delta v_{j^*k}(n) \quad (5)$$

$$\Delta v_{j^*k}(n) = \alpha \delta z_{j^*} + \beta \Delta v_{j^*k}(n-1) \quad (6)$$

$$\delta = e_k \times y_k \times (1 - y_k)$$

$$e_k = t_k - y_k$$

$$v_{j^*k}(n+1) = v_{j^*k}(n) + \alpha z_{j^*}(t_k - y_k) \quad (7)$$

식(6)에서 β 는 모멘텀 계수이고, $\Delta v_{j^*k}(n-1)$ 는 이전의 연결 강도 변화량이다. 기존의 CP알고리즘에서는 학습률을 경험적으로 설정하여 식(7)을 적용하여 학습에 적용하므로 많은 학습 시간이 요구되고, 지역 최소점(Local Minimum)에 위치할 가능성이 있다. 따라서 본 논문에서는 이전의 연결강도 변화량을 고려하여 연결 강도를 조정하기 위하여 모멘텀 학습법을 출력층의 연결 강도 조정에 적용한다.

단계 10: 오차 자승합이 오류 한계 보다 작거나 학습 반복 횟수가 최대 학습 횟수를 초과할 경우에는 학습을 종료하고, 그렇지 않으면 단계 8부터 학습을 반복한다.

III. 퍼지 제어 기법을 이용한 제안된 경쟁층 수 설정 방법

본 논문에서는 CP 알고리즘의 경쟁층의 수를 효과적으로 설정하기 위해 퍼지 제어 기법을 이용한다.

제안된 경쟁층의 수 설정 방법은 입력 패턴 종류에 대한 소속 함수, 입력 패턴 수와 패턴의 노드 수를 이용한 소속 함수를 각각 설계하고, 이들 소속 함수에 대해 소속도를 계산한 후, 퍼지 제어 규칙을 적용하고, Mamdani의 Max_Min 추론 방법으로 추론한다. 퍼지 추론을 통해 최종적으로 얻어진 값을 무게 중심법으로 비퍼지화 하여 최종적으로 제안된 CP 알고리즘의 경쟁층의 수를 결정한다.

퍼지 제어 기법을 이용하여 경쟁층의 수를 설정하는 과정에서 패턴 종류에 대한 소속 함수는 그림 2와 같다. 그림 3은 전체 패턴의 수와 입력 노드 수에 대한 소속 함수이다.

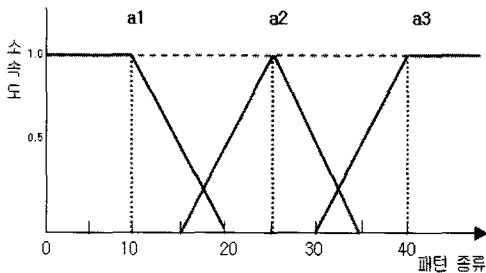


그림 2. 패턴의 종류에 대한 소속 함수
Fig. 2 Membership Function for Type of Input Patterns

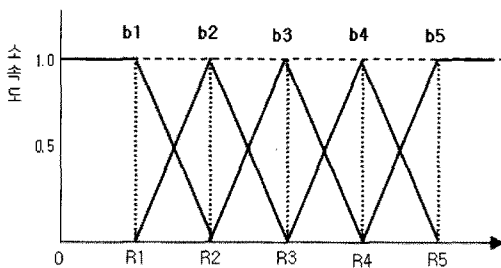


그림 3. 전체 패턴 수와 입력 노드 수에 대한 소속 함수
Fig. 3 Membership Function for Number of Input Nodes and Patterns

그림 3에서 소속 함수의 구간 $[R_i, R_{i+1}]$ 은 다음과 같이 계산한다.

$$R_i = \frac{i}{5} \times (\text{입력 노드 수} + \text{전체 패턴 수}) \quad (8)$$

식(8)에서 i 는 1부터 5까지의 값을 가진다. 그림 3의 소속 함수에 대한 입력 값은 식(9)와 같이 계산한다.

$$R = \frac{1}{1 + e^{-(t+9)}} \quad (9)$$

$$t = \log_2(\text{패턴 수} + \text{노드 수})$$

제안된 방법에서 퍼지 제어 규칙을 퍼지 관계로 표현하면 다음과 같다.

- R_1 : If A is a1, B is b1 Then C is c1
- R_2 : If A is a1, B is b2 Then C is c2
- R_3 : If A is a1, B is b3 Then C is c4
- R_4 : If A is a1, B is b4 Then C is c6
- R_5 : If A is a1, B is b5 Then C is c7
- R_6 : If A is a2, B is b1 Then C is c3
- R_7 : If A is a2, B is b2 Then C is c5
- R_8 : If A is a2, B is b3 Then C is c6
- R_9 : If A is a2, B is b4 Then C is c7
- R_{10} : If A is a2, B is b5 Then C is c8
- R_{11} : If A is a3, B is b1 Then C is c4
- R_{12} : If A is a3, B is b2 Then C is c6
- R_{13} : If A is a3, B is b3 Then C is c7
- R_{14} : If A is a3, B is b4 Then C is c8
- R_{15} : If A is a3, B is b5 Then C is c9

그림 2의 패턴 종류에 대한 소속 함수와 그림 3의 전체 패턴 수와 입력 노드 수에 대한 소속 함수를 적용하여 각각의 소속도를 구한다. 각각의 소속도를 퍼지 제어 규칙에 적용하고, Mamdani의 Max_Min 추론 방법으로 추

론한다. 퍼지 추론 후에는 무게 중심법으로 비퍼지화 하여 최종적으로 경쟁층의 수를 결정하는 그림 4와 같은 소속 함수를 적용하여 개선된 CP 알고리즘의 경쟁층의 수를 결정한다. 비퍼지화는 식 (10)과 같은 무게 중심법을 적용한다.

$$Y^* = \frac{\sum y(\mu(y))}{\sum \mu(y)} \quad (10)$$

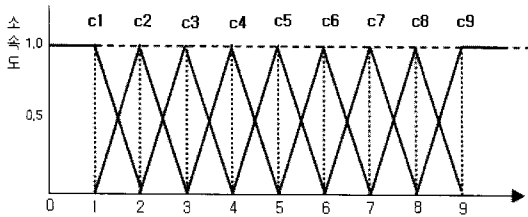


그림 4. 경쟁층 수에 대한 소속함수
Fig. 4 Membership Function for Number of Competitive Layers

그림 4의 소속 함수를 이용한 소속 구간별 경쟁층 수는 표 1과 같다.

표 1. 소속 구간별 경쟁층 수
Table 1. The Number of Competitive Layers for Membership Category

소속구간	c1	c2	c3	c4	c5	c6	c7	c8	c9
경쟁층수	2	3	5	8	10	12	15	18	20

IV. 실험 및 결과 분석

본 논문에서 제안된 CP 알고리즘의 학습 및 인식 성능을 평가하기 위하여 Intel Pentium-IV 3.0GHz CPU와 1GB RAM이 장착된 IBM 호환 PC상에서 Visual Studio 6.0으로 구현하였다. 실험에 적용된 데이터는 여권 영상에서 추출한 15×17 크기의 영문 패턴을 성능 평가 실험에 적용하였다.

표 2와 3은 영문 패턴에 대해 기존의 CP 알고리즘과 제안된 CP 알고리즘의 경쟁층에서 패턴을 분류 시에 각 패턴의 승자 뉴런과 빈도수를 나타내었다.

기존의 CP 알고리즘을 적용하여 패턴을 분류한 경우에는 동일한 패턴이 서로 다른 승자 뉴런으로 선정되어 같은 패턴이 서로 다른 패턴으로 분류되는 경우가 발생하였다. 제안된 CP 알고리즘은 기존의 CP 알고리즘 보다 같은 종류의 패턴들이 다른 승자 노드로 분류되지 않아 패턴 분류의 정확성이 개선되었다.

표 2. 기존의 CP 알고리즘에서 영문 패턴 분류
Table 2. English Character Pattern Classifications of Conventional CP Algorithm

입력 패턴	승자 뉴런	빈도 수	입력 패턴	승자 뉴런	빈도 수
A	45	2	I	81	10
	49	1	J	43	10
	132	1	K	49	9
	146	4		233	1
	184	1	L	89	10
B	257	1	M	146	1
	38	3		258	9
	128	5	N	38	1
	133	1		49	2
149	1	258	7		
C	107	10	O	133	9
D	209	6		193	1
	226	4	P	128	10
E	34	5	Q	193	10
	89	2	R	38	10
	128	3	S	226	10
F	34	34	T	76	10
G	154	154	U	133	10
	193	5	V	71	10
	226	3	W	35	10
	240	1	X	146	10
H	35	3	Y	186	10
	258	7	Z	81	10

기존의 CP 알고리즘과 참고문헌[3]의 방법, 제안된 CP 알고리즘에서 경쟁층에서 학습시 오류 한계와 최대 학습 반복 횟수를 표 4와 같이 설정하여 영문 패턴들을 학습하였다. 여기서 β는 모멘텀 계수이다.

표 3. 제안된 CP 알고리즘에서 영문 패턴 분류
Table 3. English Character Pattern Classifications of Proposed CP Algorithm

입력 패턴	승자 뉴런	빈도 수	입력 패턴	승자 뉴런	빈도 수
A	57	10	O	107	4
B	34	9		133	6
	209	1	P	34	1
C	116	10		52	5
D	209	10		110	3
E	5	10		180	1
F	110	10	Q	193	10
G	154	10	R	52	10
H	146	10	S	226	10
I	143	10	T	76	10
J	43	10	U	128	10
K	49	9	V	35	10
	233	1	W	38	10
L	89	10	X	132	10
M	258	3	Y	186	10
	146	7	Z	81	10
N	146	10			

표 4. 제안된 CP 알고리즘의 출력층의 파라미터
Table 4. Parameters for Output Layer in proposed Algorithm

	오류 한계	최대 학습 횟수	β
CP	0.01	20000	
참고문헌 [3]의 방법	0.01	20000	0.7
제안된 알고리즘	0.01	20000	0.7

제안된 CP 알고리즘을 영문 패턴에 적용하여 입력 벡터와 승자 뉴런으로 선택된 대표 벡터간의 차이와 승자 뉴런의 빈도수를 반영하여 학습률을 동적으로 조정하는 과정을 그림 5로 나타내었다.

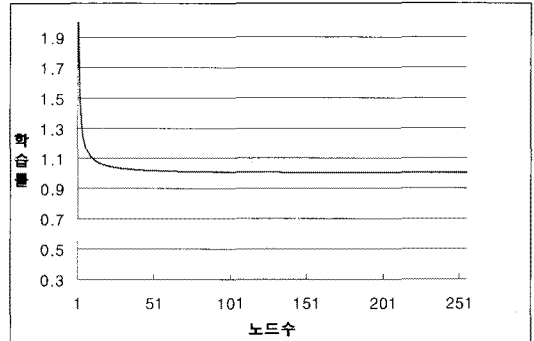


그림 5. 제안된 학습률 제어
Fig. 5 Proposed Dynamic Learning Rate Control

그림 5에서와 같이 경쟁층의 노드 수에 대한 누적된 승자 뉴런의 빈도수가 증가할수록 학습률이 감소하는 것을 알 수 있다. 이것은 현재 패턴을 학습시키는 과정에서 이전에 학습된 패턴의 정보를 최대한으로 반영하여 학습하므로 패턴의 분류의 정확성이 기존의 CP 알고리즘보다 개선되는 것을 의미한다. 표 5는 패턴 종류, 패턴 수, 입력 노드 수에 대하여 참고문헌[3]의 방법과 제안된 CP 알고리즘에서 경쟁층의 수에 따른 인식 결과를 나타내었다.

표 5. 기존의 CP와 참고문헌[3]의 방법, 제안된 방법간의 인식률 비교
Table 5. Comparison of Recognition Rate among Conventional CP, Reference [3], and Proposed Algorithm

패턴	패턴 종류	패턴 수	노드 수	방법	경쟁층 수	인식률 (%)
영문	26	260	15*17 (255)	CP	1	44
				[3]의 방법	1	47
					2	60
					3	77
					4	80
					5	83
					.	
					.	
					.	
					12	83
					제안된 방법	12

패턴	패턴 종류	패턴 수	노드 수	방법	경쟁층 수	인식률 (%)
영문	26	390	15*17 (255)	CP	1	14
				[3]의 방법	1	28
					2	67
					3	73
					4	79
					5	85
					6	82
					7	82
					8	79
					9	87
					10	84
					11	88
					12	81
제안된 방법	12	81				

표 5에서 확인할 수 있듯이 제안된 CP 알고리즘으로 경쟁층의 수를 설정하는 것과 참고문헌[3]의 방법을 적용해 경험적으로 경쟁층의 수를 설정하여 최적의 인식 성능을 보인 것과 비슷한 결과가 나타나는 것을 확인할 수 있다.

그러나 표 5의 영문 패턴 실험에서와 같이 제안된 CP 알고리즘으로 경쟁층의 수를 설정하여 패턴들을 학습하고 인식하는 것은 최적의 인식 성능을 보장하는 것은 아니지만 다양한 패턴들이 많은 경우에는 경험적으로 경쟁층의 수를 설정하는 방법보다 제안된 방법이 효율적인 것을 표 5에서 확인할 수 있다.

V. 결론

기존의 CP 알고리즘들에서는 입력되는 패턴의 정보에 따라서 경험적인 실험을 통해 가장 인식률이 높은 경쟁층의 수를 설정하였다. 그러나 이와 같은 방법은 패턴의 종류가 증가할수록 최적의 경쟁층의 수를 설정하기가 어렵고 경험적인 실험으로 인하여 많은 시간이 걸린다는 단점이 있다.

따라서 본 논문에서는 참고문헌[3]의 방법에서 경쟁층의 수를 경험적으로 설정하는 문제점을 개선하기 위

하여 퍼지 제어 기법을 이용하여 경쟁층 수를 설정하는 방법을 제안하였다. 그리고 다수의 경쟁층을 제안된 방법으로 설정하여 경쟁층에서 패턴 분류의 정확성을 높이고, 입력 벡터와 승자 뉴런의 대표 벡터간의 차이와 승자 빈도수를 반영하여 학습률을 동적으로 조정하여 경쟁층에서 안정적으로 학습이 되도록 하였다.

제안된 CP 알고리즘의 학습 및 인식 성능을 평가하기 위하여 영문자 패턴 분류 문제에 적용한 결과, 기존의 CP 알고리즘 보다는 제안된 방법이 총오차차승합이 현저히 감소하므로 패턴의 분류 성능이 개선된 것을 확인하였다. 또한 기존에 제안한 개선된 CP 알고리즘에서 경쟁층의 수를 경험적으로 설정하여 학습하는 것과 비교해서 제안된 경쟁층 설정 방법으로 경쟁층의 수를 설정하는 것이 효율적인 것을 확인하였다.

참고문헌

- [1] P. D. Wasserman, *Neural Computing Theory and Practice*, Van Nostrand Reinhold, 1989.
- [2] 조병래, 박경숙, 정환목, "다중연결 Belief Network에서의 동적 추론을 위한 카운터프로퍼게이션네트워크," *한국정보과학회논문지*, 제21권, 제2호(A), pp.776-779, 1994.
- [3] 김광백, "효과적인 패턴 인식을 위한 개선된 Counterpropagation 알고리즘," *한국해양정보통신학회논문지*, 제12권, 9호, pp.16822-1688, 2008.
- [4] K. B. Kim, Abhijit S. Pandya, "Color Image Vector Quantization Using an Enhanced Self-Organizing Neural Network," *Lecture Notes in Computer Science*, LNCS 3314, Springer, pp.1121-1126, 2004.
- [5] 김광백, 박충식, "퍼지 제어 시스템을 이용한 학습률 자동 조정 방법에 의한 개선된 역전파 알고리즘", *한국해양정보통신학회논문지*, 제8권, 2호, pp.464-470, 2004.
- [6] 정형환, 허동렬, 김상효, "신경회로망을 이용한 SVC 계통의 안정화에 관한 연구", *한국조명전기설비학회, 조명전기설비학회논문지*, 제14권, 제3호, pp. 49-58, 2000.

저자소개



김광백(Kwang-Baek Kim)

1999년 : 부산대학교
전자계산학과(이학박사)
1997년~현재 신라대학교
컴퓨터공학과 교수

2005년~현재 한국해양정보통신학회 학술상임이사

2007년~현재 Scientific Journals International(USA)
Editor

※관심분야: 퍼지 논리, 영상 처리, 유전자 알고리즘,
의료정보시스템, 생물정보학