

개선된 신경회로망을 이용한 반도체 Wafer ID 인식시스템

조영임
 평택대학교 컴퓨터학과
 yicho@ptuniv.ac.kr

Semiconductor Wafer ID Recognition System using an Improved Neural Network

Young Im Cho
 Dept. of Computer Science, Pyongtaek University

요 약

본 논문에서는 반도체의 Wafer ID 문자인식을 위해 기존의 오류 역전파 학습알고리즘을 개선하여 최적의 학습 학습 조건에 관해 연구하였다. 결과, 오류 역전파 학습알고리즘의 학습 최적 조건은 은닉층수는 1층, η 값은 0.6 이상, 은닉층 노드수는 10개일 때 99%의 높은 인식률을 보였다. 본 논문에서 제안하는 최적조건을 사용함으로써 기존의 오류역전파 학습 알고리즘이 가진 문제점을 해결할 수 있었다.

1. 서론

우리의 반도체 산업은 전반적인 경기침체에도 불구하고 높은 성장률을 보여왔다. 그래서 우리의 반도체 기술은 날로 발전하여 세계시장에서 우위를 점하였으며 각광을 받는 산업이 되었다. 문자인식 기술도 산업전반에 걸쳐 폭넓게 활용되어 산업전반에 걸쳐 파급되는 효과는 매우 크지만 아직도 정확성을 기하기에는 부족한 점이 많다[1].

반도체산업에서 적용되고 있는 주요 문자 인식기술 중 하나가 반도체 Wafer 표면에 새겨져 있는 일련번호인 Wafer ID를 인식하는 기술이다. 반도체 생산라인은 Wafer의 생산투입 시 각 Wafer마다 고유 번호인 ID Number를 부여하여 Wafer의 생산에 따른 진행 상태를 자동으로 검사할 수 있게 하고 있다. ID Number는 Wafer의 전면이나 후면에 SEMI 표준에 의거하여 OCR, 2차원 매트릭스(2D matrix), 바코드(bar code)등의 형태로 표시된다. 따라서 생산공장에서 Wafer를 제조하는 장비에 입출력될 때 Wafer의 ID Number를 읽어 들임으로써 Wafer마다 어떤 생산의 진행정도를 실시간으로 알 수 있게 해준다.

일반적으로 Wafer ID Number는 머신비전을 사용하여 인식하는 경우가 대부분인데 Wafer ID 문자의 일부가 손상되어 흐려지거나 없어져서 비전 카메라를 통하여 인식하는 경우 잘 보이지가 않을 수 있으며, 머신비전의 특성상 동일한 조명 조건 하에서 Wafer 표면의 성질이 달라짐에 따라 비전 카메라를 통해 보이는 Wafer ID 문자가 제대로 보이지 않는 경우가 종종 발생하게 된다. 기존의 Wafer ID 인식 시스템에서는 Wafer ID를 촬상한 이미지에 대해 개선패턴과 같은 직관적인 화상 처리를 통해 문자를 인식

하는 데 앞서 설명한 Wafer ID 문자의 손상이나 Wafer 표면의 성질변화에 대한 조명조건이 달라짐에 따른 문자인식의 문제점을 해결하기가 어려워져 문자인식이 제대로 되지 않는 경우가 때때로 발생하게 된다. 따라서 본 논문에서는 손상된 문자를 인식하기 위한 보다 강건하고 효율적인 문자인식 방법을 제안하고자 한다[2].

2. Wafer ID인식을 위한 신경회로망 알고리즘

문자인식을 위한 다층 신경망을 학습시키기 위해 역전파 알고리즘이 널리 사용되고 있다. 다층 퍼셉트론은 단층 퍼셉트론과 유사한 구조를 가지고 있지만 중간층과 각 입력노드의 입출력 특성을 비선형으로 함으로써 네트워크의 능력을 향상시키고 있다[3].

그러나 오류 역전파 학습 알고리즘에 다음과 같은 문제점이 있다. 첫째, 학습이 완료되기까지 많은 횟수의 반복 학습이 필요하다. 둘째, 응용분야에 따라 학습 파라미터의 조절이 필요하다. 셋째, 추가학습 시 전체적인 재학습이 필요하다. 위의 문제점을 해결하고 시간 속도와 인식률의 최적의 조건을 시뮬레이션을 통해 찾아보고자 한다.

본 논문에서 Wafer ID 문자인식을 위해 사용한 신경회로망 알고리즘은 다음과 같다.

- Step 1 : 모든 노드에 대한 가중치와 임계값을 임의의 작은 수로 초기화한다.
- Step 2 : 학습 패턴을 반복적으로 입력한다.
- Step 3 : 전방향 학습과정

1) 입력층에서 은닉층으로, 은닉층에서 출력층으로의 출력값을 구한다.

$$o_{pj} = \sum_i w_{ij} * i_p$$

② 구해진 출력값을 시그모이드 함수에 적용한다.

$$f(a) = \frac{1}{1 + e^{-a}}$$

③ 출력층의 델타 오차를 계산한다.

$$\delta_j = y_j (1 - y_j) (d_j - y_j)$$

④ 은닉층의 델타 오차를 계산한다.

$$\delta_i = x_i (1 - x_i) \sum_k w_{ki} \delta_k$$

● Step 4 : 역방향 학습과정

① 은닉층에서 출력층으로 가중치를 수정한다.

$$w_{ij}(t+1) = w_{ij}(t) + \Delta \delta x'_i$$

② 입력층에서 은닉층으로 가중치를 수정한다.

$$w_{ij}(t+1) = w_{ij}(t) + \Delta \delta x'_i$$

가중치가 거의 변하지 않을 때까지 거의 0에 접근할 때까지 반복한다.

● Step 5 : Step 2로 이동하여 반복한다.

3. 시뮬레이션 방법 및 대상

① OCR 폰트 시스템에 사용한 글자체로 시스템 메뉴 바에서 "EBP 메뉴" 에 "OCR Font"를 클릭하면 "Pattern" 다이얼로그 화면이 보여진다. 본 시스템의 입력패턴으로 숫자 10개와 알파벳 26개로 이루어져있다.

② Wafer ID 인식 시스템에 사용한 Wafer ID



<그림 1> Wafer ID 실제 예



<그림 2> 시스템에서 사용한 실제 예



<그림 3> 시스템에서 사용한 3배 확대된 ID

ID의 알파벳 순서나 숫자의 배열은 각 반도체마다 차이가 있는데 본 시스템에서는 총 9자로 구성되어 있고 "를 제외한 8자는 앞에 3자의 ID는 알파벳이고, 뒤의 5자리 ID는 숫자로 구성되어 있다.

기본 패턴을 이진비트로 검은색은 0으로 흰색은 1로 하여 배열을 사용하여 배열[11][16]으로 176의 기본 패턴을 입력 노드에 적용한다. 은닉층 노드수는 10개, 은닉층 수는 1-3 층, 출력층 노드수는 숫자인 경우 10개, 알파벳인 경우 26개이다.

본 시스템에서는 숫자패턴 10개와 알파벳 패턴 26개를 따로 프로그램화 하였다. Wafer ID 9개 중 앞의 글자 3개만 알파벳이기 때문에 3자에만 알파벳 패턴의 학습을 적용시켰고, 나머지 5자에는 숫자만을 학습시킴으로써 효율적으로 시간 낭비를 줄였다.

실제 데이터인 반도체에서 사용되는 10개의 손실 ID를 가지고 실험했다. bmp파일은 정상적인 ID와 손실이 있는 ID이다. <표 1>은 한 글자에 따른 이진비트 수로 손실률을 계산한 것이다. (비트맵의 검은색을 0으로 흰색을 1로 보았다.) 손실이 있는 ID의 이진비트를 정상적인 ID의 이진비트로 나눠서 손실률을 계산했다. 손실률이 높을수록 손실이 많은 것이다. 손실률은 (노이즈 비트수/전체 비트수)에 대한 백분율이다. (예 : A의 기본패턴의 수 : 54, A의 손실패턴의 수 : 13, 손실률 : (13/54)*100% = 24%)

1) AEIO16-13



id1.bmp



error1.bmp

<그림 4> 실험에 사용된 파일1의 예

<표 1> 글자체의 손실률

1	A	E	I	O	1	6	-	1	3
기본패턴	54	62	52	72	54	59	-	54	64
손실패턴	13	21	18	29	22	25	-	23	22
손실률	24	34	35	40	41	42	-	43	34
	%	%	%	%	%	%	-	%	%

2) AMI891-01



id2.bmp



error2.bmp

<그림 5> 실험에 사용된 파일2의 예

<표 2> 글자체의 손실률

2	A	M	I	8	9	1	-	0	1
기본패턴	54	72	52	74	59	54	-	72	54
손실패턴	20	19	17	26	24	16	-	29	15
손실률	37%	26%	33%	35%	41%	30%	-	40%	28%

4 시뮬레이션 결과

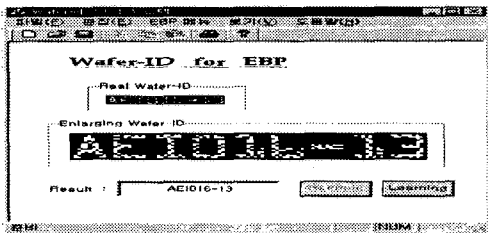
Bitmap으로 만들어진 파일을 가지고 학습 알고리즘을 사용하여 ID를 인식한다.

3 Wafer ID 프로그램의 실행 초기화면에서 파일-열기에 Wafer ID 비트맵으로 인식되기를 원하는 Wafer ID를 연다.

4 실제 Wafer ID Size 와 3배 확대된 Wafer ID Size를 보여준다. 시스템의 입력노드에 사용되는 것은 실제 Wafer ID를 사용한다.

5 Learning 버튼을 클릭하여 기본 패턴을 학습시킨다. Learning 이 실행되면 활성화 되어있는 두개의 버튼도 비활성화되는데, 비활성화된 버튼이 활성화 될 때까지 기다린 후 다음 단계로 넘어가야 한다.

6 활성화 된 Execute 버튼을 클릭하여 Wafer ID를 인식시킨다. Execute를 실행하면 실제 Wafer ID의 이진비트가 입력노드에 적용된다. 아래 <그림 6>에서처럼 A1D16-13에서 먼저 A의 손실된 이진비트를 실행시켜 인식하고 그 다음 글자인 E로 넘어가 이진비트를 사용하여 인식한 후 다음 글자로 넘어간다. 이런 방식으로 8글자를 인식한다.



<그림 6> Wafer ID 실행화면

한번의 학습으로 실험한 10개의 인식률 평균을 구한 뒤 10번의 학습을 가지고 나타낸 평균들을 가지고 다시 평균을 구했다. 한번의 학습 때 실험 ID 한 개만이 기본패턴과 다른 글자가 인식되었다면 10개의 실험 중 하나가 8개의 ID에서 1개만이 인식이 틀렸을 경우이므로

인식률 : 한 개의 실험대상/전체 실험대상 맞게 인식한 ID글자수/ID글자수

$$\frac{7}{10} \times \frac{7}{8} = \frac{7}{80}$$

한번의 실험에서 ID 8개가 모두 인식이 알맞게 되었다면

$$\frac{8}{10} \times \frac{8}{8} = \frac{8}{10} = \frac{1}{10}$$

이 10개의 ID 중 9개는 알맞게 인식이 되고 1개만이 1개의 글자만이 있을 때의 평균은

$$\left(\frac{7}{30} \times 1 \right) + \left(\frac{1}{10} \times 9 \right) \times \frac{1}{10} = \frac{79}{800}$$

이렇게 구한 평균 10개를 다시 평균을 구해서 인식률이 되도록 100을 곱해 계산한다.

① 은닉층 수(1층, 2층, 3층 이상)에 따른 η변화(0.1 ~ 0.9)에 따른 인식률

<표 3> 은닉층 수에 따른 η변화의 인식률

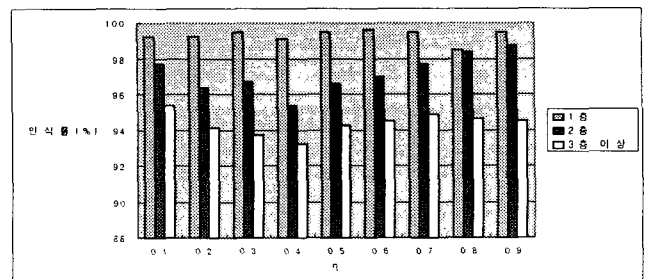
ETA 수	은닉층 수	1층	2층	3층이상
0.1		99.25	97.75	95.375
0.2		99.25	96.375	94.125
0.3		99.5	96.75	93.75
0.4		99.125	95.375	93.25
0.5		99.5	96.625	94.25
0.6		99.625	97	94.5
0.7		99.5	97.75	94.875
0.8		98.5	98.375	94.625
0.9		99.5	98.75	94.5

② 표 1의 인식률에 따른 η 값을 95% ~ 99%에서 찾은 결과

<표 4> η 값에 따른 인식률

은닉층	1층	2층
95%	0.1~0.9	0.4
96%	0.1~0.9	0.2, 0.3, 0.5
97%	0.1~0.9	0.1, 0.6, 0.7
98%	0.1~0.9	0.8, 0.9
99%	0.1~0.9	

인식률은 은닉층이 1층일 때는 η값이 0.1에서 0.9사이의 어떤 값을 넣더라도 99%를 유지한다. 98% 인식률인 경우엔 2층에서 0.8, 0.9의 큰 값이 인식률이 좋다. 2층의 0.1, 0.6, 0.7의 η값이 97%의 인식률을 보여준다. 3층에서는 0.1만이 95% 이상인 인식률을 보여준다.



<그림 7> 은닉층수에 따른 인식률 그래프

<그림 7>은 은닉층 수가 1인 경우 η값은 0.1~0.9사이 값은 모두 인식률이 99%이상이다. 은닉층 수가 2인 경우 η값이 클수록(0.6≤η≤0.9) 인식률이 98%에 근접하고 있

다. 은닉층 수가 3인 경우 η 값이 작은 값($\eta \leq 0.1$)이면 인식률이 95%이다. 따라서 은닉층 수는 20이하이고 η 값이 큰값이면 인식률이 높았다.

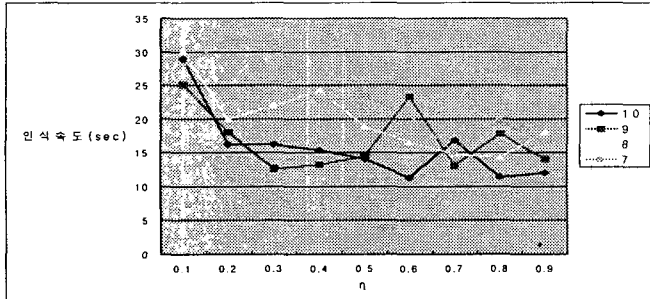


그림 23. 은닉층 노드 수에 따른 η 변화의 인식속도

<그림 8>은 은닉층 1층에 대한 은닉층 노드수를 7~10까지 변화시키면서 인식속도를 비교한 것이다. 그래프에 따르면 노드수가 많고 η 값이 클수록 인식속도가 빠르다. 은닉층이 10이고 노드수가 10개인 경우 인식을 99%와 빠른 속도를 보이려면 η 값은 클수록 좋다. 특히 η 값에 따른 인식률은 거의 동일하지만 0.6일 때 인식률과 인식속도가 가장 좋다.

본 논문에서는 오류 역전파 학습알고리즘이 가지는 문제점을 해결하고 최적의 학습조건을 찾아내고자 하였다. 먼저 모든 시스템에서 가장 중요시 되는 시스템 속도를 위해 여러 시뮬레이션을 통해 최적 방법을 제안했다.

본 시스템에서는 몇가지 문제점을 보완하는 해결책을 찾았다. 첫번째 문제점으로는 시스템이 실행되고 학습이 완료되기까지 반복학습이 필요함에 따라 그에 따른 시간이 많이 걸린다는 것이었다. 따라서 본 논문에서는 최적의 조건을 찾아 최적의 은닉층 노드 수에 따른 최적의 η 값 변화의 시간 속도를 사용하여 시간을 단축할 수 있었다. 두번째 문제점은 실험의 결과로 가장 적절한 η 값 중에서 구체적인 범위 내에 있는 값을 사용한다면 99%의 인식률을 보장할 수 있다. 세번째 문제점은 추가학습을 할 때마다 전체적인 재학습이 필요하다는 것이다. 본 논문에서는 실무적인 반도체 기술을 고려하여 한번의 전체적인 학습을 한 후 재학습할 필요 없이 여러 개의 ID를 한번의 학습으로 인식할 수 있도록 하여 시간을 줄일 수 있었다.

5. 결론

손상된 Wafer ID Number를 인식하기 위해 개선된 오류 역전파 알고리즘을 사용하여 알파벳과 숫자 ID 인식 시스템을 개발하였다. 이 시스템은 기존의 오류 역전파 학습

알고리즘이 문제영역에 대한 최대인식률을 갖는 최적의 구조를 찾는 문제를 해결하기 위해 다양한 실험을 통해 다중 퍼셉트론 구조의 상한선과 하한선을 결정함으로써 Wafer ID에 적합한 최적의 시스템 구조를 제안하였다. 인식률을 알아보기 위해 은닉층 수에 따른 η 값 변화의 인식률과 은닉층 노드수에 따른 η 값 변화의 시간에 대해 실험해 보았다.

결과적으로 오류역전파 학습알고리즘의 최적 조건은 은닉층 수는 1층이고, η 값은 0.6 이상, Hidden Node 수는 10개가 99%의 높은 인식률을 보였다. 본 논문에서 제안하는 최적 조건을 사용함으로써 기존의 알고리즘이 가진 문제점을 문제점을 해결할 수 있었다. 또한 높은 인식률과 빠른 속도로 실제 반도체 Wafer ID 문자 인식을 위한 알고리즘에도 사용 가능하다. 따라서 빠른 속도와 높은 인식률을 가진시스템으로 위 결과를 토대로 한 알고리즘을 개발한다면 실제 반도체 Wafer ID 문자인식 산업에도 많은 도움이 될 것이다.

향후 인식속도를 향상시키고, Wafer ID가 뒤집혀져 있는 경우도 인식할 수 있도록 시스템을 좀더 개선해야 할 것이다. 글자가 역으로 들어오면 이를 검사하여 뒤집어져 있는 경우임을 알고 해당 문자를 180° 회전시켜서 인식이 가능하도록 만들어야 할 것이다.

참고 문헌

- [1]이인동, "문자인식 기술", 정보처리 제 6권 제 4호, pp.11-16, 1999
- [2]www.nexteye.com , Wafer-ID Track & Read System (주) 넥스트아이
- [3] 조영임, 인공지능시스템, 홍릉과학출판사, 2003.10
- [4]신충호, 이종은, 김단환, 김형균, 김재석, 오무송, "문자인식의 전처리과정에서 영상향상", 한국정보처리학회 춘계 학술발표논문집, 제8권, 제1호, pp.139-142, 2001
- [5] 장승주, "적응 알고리즘을 이용한 자동차 번호판 인식 시스템 개발에 대한 연구", 한국정보처리학회 논문지 제 7권 제 10호, pp.3155-3163, 2000
- [6] 조원현, "인공신경망을 이용한 필기체 한글의 자동분류와 인식", Proceedings of the Spring Conference, Korean Statistical Society, pp.123-128, 1999
- [7] 이이표, 김병세, Microsoft Visual C++ Bible 6.0, 삼양 출판사, pp.555-565, 2001
- [8]김민수, 강은영, 김우성, 한선화, 김진형, "인쇄체 문자 인식기의 성능 평가에 관한 연구", 한국정보처리학회 논문집 제 7권 제 11호, pp.3584-3591, 2000