

**A Study on Rotating Object Classification using
Deep Neural Networks**

이용규* · 이일병**†
Yong-Kyu Lee and Yill-Byung Lee†

*연세대학교 컴퓨터과학과, **연세대학교 컴퓨터과학과
*Department of Computer Science, Yonsei University
**Department of Computer Science, Yonsei University

요 약

본 논문은 딥러닝 알고리즘을 적용한 깊은신경망을 이용하여 회전 객체의 분류 효율성을 높이기 위한 연구이다. 회전객체의 분류 실험을 위하여 데이터는 COIL-20을 사용하며 객체의 2/3영역을 학습시키고 1/3영역을 유추하여 분류한다. 연구에 이용된 3가지 분류기는 주성분 분석법을 이용해 데이터의 차원을 축소하면서 특징값을 추출하고 유클리디안 거리를 이용하여 분류하는 PCA분류기와 오류역전파 알고리즘을 이용하여 오류 에너지를 줄여가는 방식의 MLP분류기, 마지막으로 pre-training을 통하여 학습데이터의 관찰될 확률을 높여주고 fine-tuning으로 오류에너지를 줄여가는 방식의 딥러닝을 적용한 DBN분류기이다. 깊은신경망의 구조별 오류율을 확인하기 위하여 은닉층의 개수와 은닉뉴런의 개수를 변경해가며 실험하고 실제로 가장 낮은 오류율을 나타내는 구조를 기술한다. 가장 낮은 오류율을 보였던 분류기는 DBN을 이용한 분류기이다. 은닉층을 2개 갖는 깊은신경망의 구조로 매개 변수들을 인식에 도움이 되는 곳으로 이동 시켜 높은 인식률을 보여줬다.

키워드 : PCA, MLP, DBN, Pre-training, 딥러닝

Abstract

This paper is a study to improve the classification efficiency of rotating objects by using deep neural networks to which a deep learning algorithm was applied. For the classification experiment of rotating objects, COIL-20 is used as data and total 3 types of classifiers are compared and analyzed. 3 types of classifiers used in the study include PCA classifier to derive a feature value while reducing the dimension of data by using Principal Component Analysis and classify by using euclidean distance, MLP classifier of the way of reducing the error energy by using error back-propagation algorithm and finally, deep learning applied DBN classifier of the way of increasing the probability of observing learning data through pre-training and reducing the error energy through fine-tuning. In order to identify the structure-specific error rate of the deep neural networks, the experiment is carried out while changing the number of hidden layers and number of hidden neurons. The classifier using DBN showed the lowest error rate. Its structure of deep neural networks with 2 hidden layers showed a high recognition rate by moving parameters to a location helpful for recognition.

Key Words : PCA, MLP, DBN, Pre-training, Deep learning

Received: Jun, 15, 2015
Revised : Sep, 16, 2015
Accepted: Sep, 16, 2015
†Corresponding author
yillbyunglee@yonsei.ac.kr

1. 서 론

객체인식 기술은 그림 1과 같이 데이터의 특징이나 속성을 찾아 동일한 객체를 찾아내는 방법으로 효율적인 인간-컴퓨터 상호작용을 위해서 시도되는 인터페이스 테마중 하나이다[1]. 그러나 최근 연구 결과에서 노이즈가 많은 데이터나 객체의 다양한 회전 각도에 의한 변형이 상당한 데이터에 대해 취약한 모습을 보이고 있는 것이 현실이다. 따라서 영상의 다양한 각도에서 동일한 객체를 인식하는 기술의 필요성이 대두되고 있다. 객체인식을 위한 연구는 단순한 템플릿 매칭 기법으로 시작되어 주성분 분석법을 이용한 방법과 신경 회로망을 이용한 방법 등 꾸준히 연구되고 있다.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

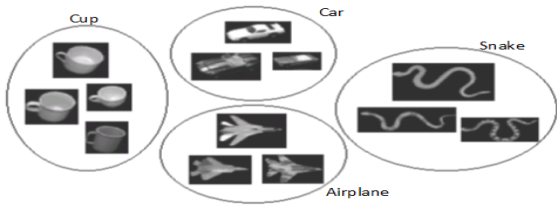


그림 1. 객체 인식

Fig. 1. Object recognition

그러나 주성분 분석법을 이용한 방법은 변형이 많은 데이터에 대해서는 비효율적이며 신경회로망을 이용한 다층퍼셉트론의 경우 학습데이터가 적을 경우 과적합 문제가 발생하고 지역 최소값에 빠지는 문제 등이 있다[1][2].

따라서 본 논문에서는 회전객체의 2/3영역을 학습시켜 나머지 1/3영역을 얼마나 잘 유추하여 분류하는지에 대한 실험을 한다. 기존 분류기 방식의 문제점을 찾고 딥러닝 알고리즘을 이용한 DBN(Deep Belief Networks)의 학습방법을 접목해 기존의 분류기 방식보다 좀 더 효율적인 깊은신경망 구조의 분류기를 제안한다.

2. 관련연구

2.1. 딥러닝

패턴인식에서 분류의 문제는 크게 2가지 단계로 구분된다. 바로 전처리를 통한 특징추출과 분류를 위한 분류기의 설계이다. 그러나 빅데이터로 인하여 매우 방대한 양의 데이터가 쏟아지고 최근에는 이를 처리하기 위해 특징추출과 분류기가 결합한 형태의 새로운 모형이 개발되고 있다[3].

신경망을 이용한 패턴인식의 경우 은닉층의 개수를 확장하는 시도가 있다. 이는 각종 특징을 수동으로 추출하여 학습시키던 기존의 방식에 비해 자동적이고 계층적으로 특징을 추출하고 학습하므로 매우 큰 장점이 있다. 2개 이상의 은닉층을 가진 신경망을 깊은신경망이라고 하며 이를 학습하기 위한 기법이 딥러닝이다[3][4].

2.2. 제한된 볼츠만 머신

제한된 볼츠만 머신은 패턴인식에서 특징을 추출하기 위하여 사용되는 알고리즘의 하나이다. RBM은 그림 2와 같이 기본적으로 가시뉴런과 은닉뉴런으로 구성된 2개의 층을 가지며 0또는 1이나 -1또는 1의 이진값을 갖는 뉴런으로 구성된 신경망이다[5].

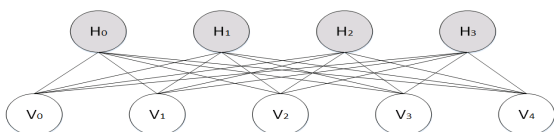


그림 2. RBM의 구조

Fig. 2. Structure of RBM

기본적인 볼츠만 머신과 달리 같은 층의 뉴런간의 연결을 제거한 구조로 가시뉴런과 은닉뉴런에 대한 에너지함수는 식 (1)과 같이 정의할 수 있다[5]. v 는 가시뉴런, h 는 은닉뉴런, b 는 가시뉴런의 bias, c 는 은닉뉴런의 bias, W 는 가시뉴런과 은닉뉴런의 연결강도를 나타낸다.

$$E(v,h) = -b^T v - c^T h - v^T W h \tag{1}$$

RBM은 에너지 베이스 모델이므로 에너지 함수가 볼츠만 분포와 같아야 한다. 볼츠만 분포에 따라 식 (2)와 같이 정리할 수 있다.

$$p(v,h) = \frac{1}{Z} \exp(-E(v,h)) \quad \text{단, } Z = \sum_v \sum_h \exp(-E(v,h)) \tag{2}$$

가시뉴런과 은닉뉴런에 대한 조건부 활성화 확률을 정리하면 식 (3)과 같이 나타낼 수 있다.

$$p(h_j = 1|v) = \text{sigm}(c_j + \sum_{i=1}^n v_i W_{ij})$$

은닉 뉴런의 조건부 활성화 확률

$$p(v_i = 1|h) = \text{sigm}(b_i + \sum_{j=1}^m h_j W_{ij})$$

가시 뉴런의 조건부 활성화 확률

RBM은 학습데이터를 통해서 구해진 확률의 로그-우도를 계산해서 그값을 최대한 크게하는 오르막 경사법을 사용한다. 이는 에너지가 낮을수록 데이터의 존재할 확률이 커지기 때문이다. 이를 정리하면 매개변수의 변화량은 식 (4)와 같이 쓸 수 있다. η 은 학습계수를 나타낸다.

$$\begin{aligned} \Delta W_{ij} &= \eta(\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}) \\ \Delta b_i &= \eta(\langle v_i \rangle_{data} - \langle v_i \rangle_{model}) \\ \Delta c_j &= \eta(\langle h_j \rangle_{data} - \langle h_j \rangle_{model}) \end{aligned} \tag{4}$$

data에 대한 기댓값은 학습데이터에 대한 은닉뉴런의 확률 분포에 대한 기댓값이므로 쉽게 구할 수 있다. 그러나 model에 대한 기댓값은 현재 연결강도에 대한 모든 가시뉴런과 은닉뉴런의 확률분포에 대한 기댓값이므로 분할상수를 알아야 한다. 하지만 이는 뉴런의 개수에 따른 상태가 지수승으로 늘어나기 때문에 실제로 구하는 것이 불가능하다[6].

이를 해결하기 위하여 Hilton은 새로운 알고리즘인 Contrastive Divergence를 제안하였다. 이 방법은 학습데이터로 n 번의 Gibbs sampling을 통해 만들어지는 분포로 대체하여 확률분포를 추정하는 방식이다. Hilton은 실제로 Gibbs step을 한 번만 사용해도 학습이 잘 된다는 사실을 밝혀냈다[7].

3. 시스템 설계

3.1. 데이터셋

본 논문의 실험을 위하여 데이터는 그림 3과 같은 COIL-20 dataset을 이용한다[8]. 총 20개의 객체를 가지고 있으며 객체 당 5°씩 회전하는 72개의 그레이스케일 이미지를 가진다. 총 1,440장이 존재하며 이미지의 크기는 128*128의 pixel이다. 실험을 위하여 32*32 pixel로 축소하여 사용하였다.

실험을 위하여 객체의 0° - 235°에 해당하는 960장(객체의 2/3 영역)은 학습데이터로 사용하고 240° - 355°에 해당하는 480장(객체의 1/3 영역)은 테스트데이터로 사용하였다.

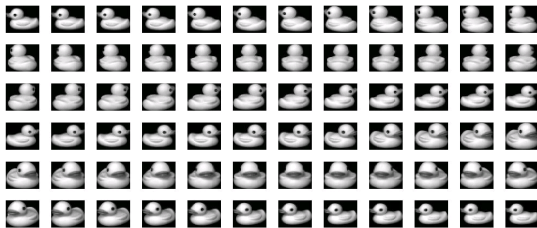


그림 3. COIL-20 데이터셋
Fig. 3. COIL-20 dataset

3.2. 분류기의 구조

딥러닝 알고리즘을 적용한 분류기의 구조는 그림 4와 같다. 데이터의 수가 많지 않아 drop-out기능과 ReLU함수는 사용하지 않는다.

본 실험의 데이터는 32*32 pixel 값을 가지므로 입력층은 총 1,024개의 뉴런, 출력층은 20개의 객체를 분류하기 위하여 총 20개의 뉴런이 필요하다. 초기 연결강도와 bias는 RBM의 학습을 통해 정해지게 된다.

학습을 위하여 학습데이터가 입력층으로 들어가고 이는 첫 번째 은닉층과의 pre-training 후 이를 고정하고 첫 번째 은닉층과 두 번째 은닉층과의 pre-training을 한다. 이를 반복하여 마지막 은닉층까지의 pre-training이 끝나고 나면 학습데이터의 관찰확률이 높아지게 된다. 이 과정이 끝나면 오류역전파 알고리즘을 이용하여 오류에너지를 줄여주는 fine-tuning을 하게 되고 분류기가 완성된다.

학습이 끝나고 나면 테스트데이터는 가중합을 구하여 활성화 함수를 통과하게 되는 forward pass를 진행한다. 출력층

에서 가장 높은 값이 나온 뉴런만 활성화 되어 클래스를 할당 받게 된다.

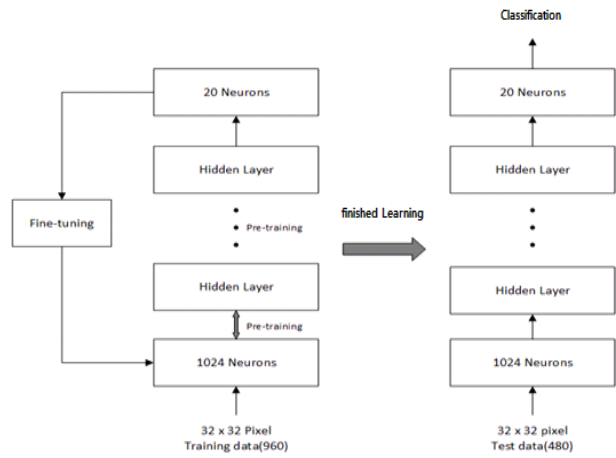


그림 4. 분류기의 구조
Fig. 4. Structure of Classifier

4. 실험 및 비교분석

4.1. PCA를 이용한 분류 실험

PCA를 이용한 분류기의 경우 주성분 벡터의 수를 정하는 것이 중요하다. 그림 5는 COIL-20 학습데이터에서 필요한 주성분 벡터의 개수를 정하기 위하여 고유치 합 비율의 변화를 나타낸 것이다.

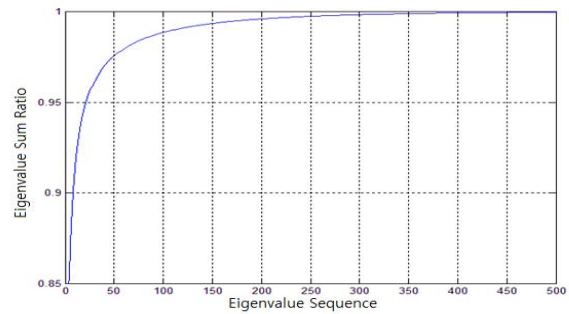


그림 5. 고유치 합 비율의 변화

Fig. 5. Change in the summation rate of the eigenvalue

주성분 벡터를 61개 사용할 때에 역치값이 0.9800으로 정보 손실량이 전체의 2% 이하임을 알 수 있고 23개를 사용할 때에 역치값이 0.9527로 정보 손실량이 5% 이하임을 알 수 있다. 분류를 위하여 주성분 벡터를 23개와 61개를 사용하였으며 내적을 통하여 학습데이터와 테스트데이터의 특징을 추출하고 유클리디안 거리를 이용하여 가장 가까운 값의 클래스에 할당하여 분류하였다.

그림 6은 테스트데이터 중 임의의 한 개를 61개의 주성분 벡터에 사영시켜 얻은 특징값과 학습데이터의 특징값에 대한 유클리디안 거리를 나타낸다.

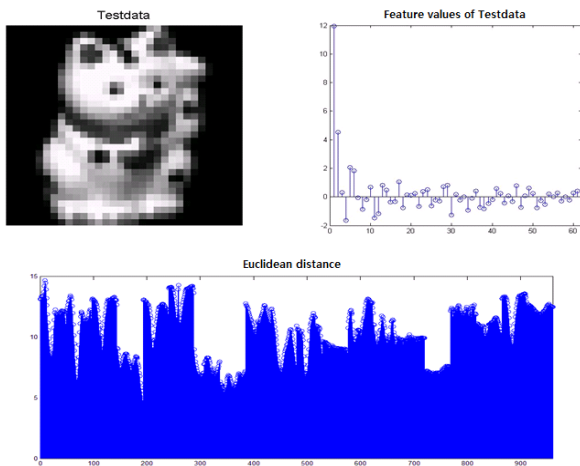


그림 6. 임의의 테스트데이터에 대한 특징값 및 유클리디안 거리
Fig. 6. Feature value and Euclidean distance for random test data

480개의 테스트데이터에 대하여 23개의 주성분 벡터를 이용한 경우 11.67%의 오류율을 보였고 61개의 주성분 벡터를 이용하여 분류한 경우 11.46%의 오류율을 보였다.

4.2. MLP를 이용한 분류 실험

표 1은 은닉 뉴런의 개수를 변경해가며 실험한 인식률이다. 반복학습 횟수는 500, 학습계수는 0.05로 고정하여 실험하였다.

표 1. 은닉층 1개의 MLP 분류기 오류율

Table 1. Error rate for MLP classifier with 1 hidden layer

Hidden neurons	Training Data	Test Data	Error Rate (%)	Time (seconds)
	MSE	MSE		
16	0.0329047	0.116946	18.9583	31.97
32	0.0274691	0.1028	14.3750	38.69
64	0.0202897	0.098573	12.7083	53.36
128	0.0188361	0.100966	13.9583	90.10
256	0.0183183	0.100985	14.5833	182.18
512	0.0177071	0.103716	15.2083	338.25
768	0.0181538	0.10273	12.9167	535.58
1024	0.0170347	0.103381	13.7500	662.94

실험결과 은닉뉴런의 개수가 많다고 해서 더 좋은 성과를 발휘하는 것은 아니며 초기값을 랜덤으로 설정하기 때문에 좋은 매개변수를 초기값으로 가진 경우 분류가 더 잘되는 것을 확인할 수 있었다. 테스트데이터에 대하여 가장 낮은 오류율 보인 구조는 1024 - 64 - 20 이었고 약 12.71%의 오류율을 보여준다.

표 2는 은닉층 2개로 늘려보았을 때의 오류율을 보여준다. 실험결과 은닉층을 늘려도 인식률 향상에 도움이 되지 않는 것을 확인할 수 있었다.

표 2. 은닉층 2개의 MLP 분류기 오류율

Table 2. Error rate for MLP classifier with 2 hidden layers

2-Hidden Layers		Training Data	Test Data	Error Rate (%)	Time (seconds)
		MSE	MSE		
16	16	0.116636	0.144806	27.5000	36.49
32	32	0.0426734	0.108072	16.2500	44.04
64	64	0.0296814	0.109456	17.5000	60.85
128	128	0.0236917	0.10104	14.3750	103.86
256	256	0.0176611	0.0991215	13.3333	218.10
512	512	0.0170893	0.101707	14.7917	520.32
768	768	0.0163216	0.0990717	13.1250	936.51
1024	1024	0.0159903	0.0997238	13.9583	1342.50

4.3. DBN을 이용한 분류 실험

표 3은 은닉뉴런의 개수를 변경해가며 실험한 인식률을 나타낸다. MLP 분류기와의 비교실험을 위하여 반복학습 횟수는 500, 학습계수는 0.05로 동일하게 고정하였다. 연결강도와 bias의 값을 구하기 위한 pre-training의 층별 반복학습 횟수는 100으로 고정하였다.

표 3. 은닉층 2개의 DBN 분류기 오류율

Table 3. Error rate for DBN classifier with 2 hidden layers

Hidden neurons		Training Data	Test Data	Error Rate (%)	Time (seconds)
		MSE	MSE		
16	16	0.0801634	0.141144	27.2917	45.88
32	32	0.0438276	0.119081	17.5000	59.21
64	64	0.0266393	0.10881	14.5833	75.10
128	128	0.0124702	0.0960501	12.5000	128.85
256	256	0.00893593	0.0902018	11.2500	264.27
512	512	0.00700639	0.0861054	7.7083	628.01
768	768	0.00552051	0.085764	8.5417	1061.37
1024	1024	0.00430644	0.0888654	10.2083	1656.17

가장 낮은 오류율을 나타내는 구조는 1024 - 512 - 512 - 20의 구조이고 약 7.71%이다. 이 구조의 각 층별 pre-training의 epoch에 따른 MSE는 그림 7과 같다.

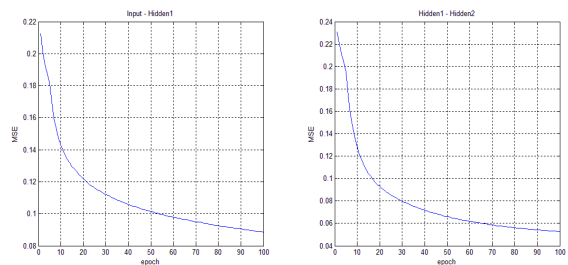


그림 7. Pre-training의 epoch에 따른 MSE 변화

Fig. 7. Change in MSE according to the epoch of pre-training

처음에는 MSE가 급격히 줄어들지만, 조금의 반복학습만으로도 입력값과 유사한 데이터를 추출하는 것을 알 수 있다.

실제로 pre-training의 학습 횟수에 따른 데이터의 추출 정도를 실험해 보았다. 그림 8에서 보듯 100번의 학습만으로도 입력데이터와 상당히 유사한 데이터를 추출해 낸다.

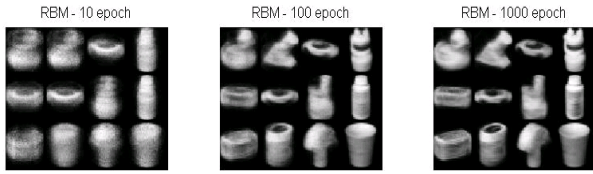


그림 8. RBM의 학습을 통한 데이터의 추출 정도
Fig. 8. Degree of data extraction through RBM training

표 4는 2개의 은닉층을 가진 구조에서 오류율이 낮았던 구조에 은닉층을 한층 더 올린 구조로 실험을 한 결과이다. 인식을 향상에는 도움이 되지 않았다.

표 4. 은닉층 3개의 DBN 분류기 오류율
Table 4. Error rate for DBN classifier with 3 hidden layers

3-Hidden neurons			Training Data	Test Data	Error Rate (%)	Time (seconds)
			MSE	MSE		
512	512	100	0,0189625	0,0938663	11,2500	692,17
512	512	200	0,00949723	0,0907514	10,8333	704,76
512	512	400	0,00668744	0,0919681	11,8750	821,09
512	512	600	0,00522963	0,0883271	10,8333	910,40
512	512	800	0,0147553	0,0972258	11,6667	971,03

4.4. 분류실험 비교 분석

표 5는 테스트데이터 480개에 대한 각 분류기의 오류율을 나타낸다. PCA와 MLP를 이용한 분류기보다 딥러닝 알고리즘을 적용한 분류기가 더 낮은 오류율을 보이는 것을 확인할 수 있다.

표 5. 분류기별 오류율 비교
Table 5. Comparison of the error rate for each classifier

Classifier	Structure	Error rate
PCA - 61 Dimension	61차원 - Euclidean distance	11,46%
MLP - 1 Hidden Layer	1024 - 64 - 20	12,71%
MLP - 2 Hidden Layers	1024 - 256 - 256 - 20	13,33%
DBN - 2 Hidden Layers	1024 - 512 - 512 - 20	7,71%
DBN - 3 Hidden Layers	1024 - 512 - 512 - 600 - 20	10,83%

그림 9는 표 5에 표기된 5개의 분류기에 대한 오류역전과 알고리즘의 학습 횟수에 따른 오류율을 나타낸다.

딥러닝 알고리즘을 적용한 DBN 분류기의 경우 pre-training만으로도 충분히 낮은 오류율을 보여준다.

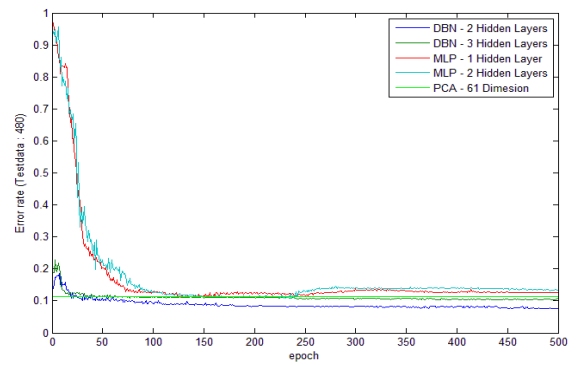


그림 9. 분류기별 학습 횟수에 따른 오류율
Fig. 9. Error rate according to the number of training sessions for each classifier

그림 10은 pre-training의 효과를 보여준다. 동일한 구조의 MLP와 DBN의 오류역전과 알고리즘의 학습 횟수에 따른 학습데이터에 대한 MSE를 나타낸다.

DBN을 이용하는 경우 pre-training의 학습시간이 필요하지만, 이를 통하여 데이터를 생성하는 데에 필요한 특징들을 잘 잡아내고 인식에 도움이 되는 곳으로 매개변수들이 이동했다고 볼 수 있다.

그에 비하여 MLP를 이용한 경우 더 좋은 매개변수가 있음에도 불구하고 지역해로 수렴하는 것을 확인할 수 있다.

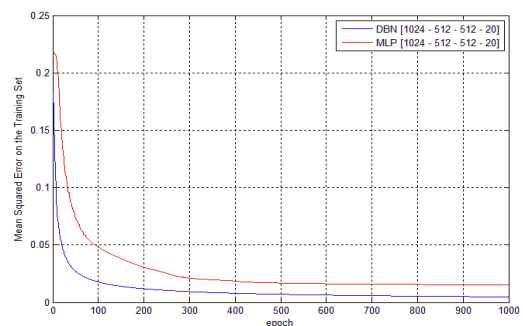


그림 10. 동일한 구조의 MLP 분류기와 DBN 분류기의 비교실험
Fig. 10. A comparison test between MLP classifier and DBN classifier of identical structures

본 실험은 총 1,440개의 데이터를 이용해 실험하였으므로 좀 더 일반화 오차에 근접한 오차값을 얻을 필요가 있다. 이 때문에 그림 11과 같은 5분절 교차 검증법을 이용하였다.

이는 데이터의 양이 충분하지 않을 때 분류기 성능의 통계적 신뢰도를 높이기 위한 방법으로 동일한 데이터에 대하여 여러 개의 부분집합으로 나누어 학습데이터와 테스트데이터를 변경해가며 실험한 후 평균 오류율을 구하는 방법이다[1].

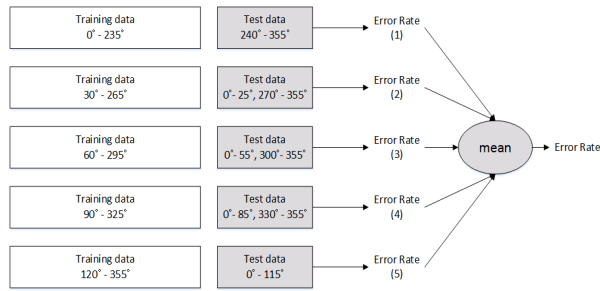


그림 11. 5분절-교차 검증법
Fig. 11. 5-fold cross validation

총 5개의 평균 오류율을 구하면 표 6과 같이 정리할 수 있다. 딥러닝 알고리즘을 적용한 DBN 분류기중 2개의 은닉층을 가진 1024 - 512 - 512 - 20의 구조가 회전 객체에 대하여 8.13%의 오류율로 가장 낮은 오류율을 보인다.

표 6. 교차 검증법을 이용한 평균 오류율
Table 6. Average error rate using cross validation

Classifier	Structure	Error rate
PCA	Euclidean distance	11.46%
MLP - 1 Hidden Layer	1024 - 64 - 20	12.75%
MLP - 2 Hidden Layers	1024 - 256 - 256 - 20	12.67%
DBN - 2 Hidden Layers	1024 - 512 - 512 - 20	8.13%
DBN - 3 Hidden Layers	1024 - 512 - 512 - 600 - 20	10.71%

5. 결론 및 향후 연구

본 논문에서는 회전객체에 대한 분류를 위하여 총 3가지의 분류기를 구현하고 실험하였다.

가장 우수한 성능을 보인 분류기는 딥러닝 알고리즘을 적용한 DBN분류기였다. PCA를 이용한 분류기에 비하여 약 3.33%의 인식률 상승이 있었고 MLP를 이용한 분류기에 비하여 약 4.54%의 인식률 상승을 보였다. 또한, 같은 구조의 단순한 깊은신경망 구조를 가진 MLP에 비하여 약 7%의 인식률 상승을 보였다. 이처럼 pre-training을 통하여 결정되는 신경망의 구조가 초기의 랜덤값을 가지는 신경망의 구조에 비하여 인식에 도움이 되는 곳으로 매개변수들을 이동시켜 훨씬 상승한 인식률을 보여주었다.

딥러닝 알고리즘은 본 논문에서 실험한 회전객체에 대한 분류뿐 아니라 다양한 분야에서 효과적으로 사용될 수 있는 알고리즘임에 틀림없다.

그러나 불분명한 값들이 많이 존재한다는 것이 문제점이다. 은닉층의 개수와 은닉뉴런의 개수, 학습계수, 반복학습 횟수 등 어떤 구조가 가장 효율적인지는 모두 실험으로 구하는 방법일 뿐 사용자가 임의로 결정해야 할 가변적인 값들이다. 이러한 가변적인 값들의 변화에 따라 학습의 성능은 더욱 뛰어난 효율을 보일 가능성이 있다. 그러므로 데이터에 따른 가변적인 변화를 최적화할 수 있는 연구가 필요하다.

References

- [1] Hye-Young Park and Kwan-Yong Lee, *Pattern recognition and Machine learning*, EHAN, 2011.
- [2] Il-Seok Oh, *Pattern recognition*, KYOBO, 2008.
- [3] Bengio, Yoshua, "Learning Deep Architectures for AI," *Foundations and Trends in Machine Learning 2*, 2009.
- [4] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," 2014.
- [5] Geoffrey Hinton, "A Practical Guide to Training Restricted Boltzmann Machines," UTML TR 2010-003, University of Toronto, 2010.
- [6] Min-Kyu Park, "Training Restricted Boltzmann Machines using MCMC-based Particle filter," University of Seoul, 2013.
- [7] Hinton, Geoffrey E., "Training products of experts by minimizing contrastive divergence," *Neural computation* 14,8, 1771-1800, 2002.
- [8] Sameer A.Nene, Shree K.Nayar, Hiroshi Murase, "Columbia Object Image Library (COIL-20)," No. CUCS-005-96, 1996.

저 자 소 개



이용규(Yong-Kyu Lee)

2012년 : 서울과학기술대학교 컴퓨터공학 학사

2015년 : 연세대학교 대학원 컴퓨터과학 석사

관심분야 : AI, Pattern Recognition, Neural Networks
Phone : +82-10-6649-5317
E-mail : uto4247@yonsei.ac.kr



이일병(Yill-Byung Lee)

1976년 : 연세대학교 전기전자 학사

1980년 : MS in Computer Science, University of Illinois

1985년 : PhD in Computer & Information Science, University of Massachusetts

2014년 : 연세대학교 컴퓨터과학과 교수

관심분야 : Pattern Recognition, AI, Data Mining
Phone : +82-2-2123-2713
E-mail : yillbyunglee@yonsei.ac.kr