

망 분리를 이용한 딥러닝 학습시간 단축에 대한 연구

A Study on Reducing Learning Time of Deep-Learning using Network Separation

이 희 열*, 이 승 호*

Hee-Yeol Lee*, Seung-Ho Lee*

Abstract

In this paper, we propose an algorithm that shortens the learning time by performing individual learning using partitioning the deep learning structure. The proposed algorithm consists of four processes: network classification origin setting process, feature vector extraction process, feature noise removal process, and class classification process. First, in the process of setting the network classification starting point, the division starting point of the network structure for effective feature vector extraction is set. Second, in the feature vector extraction process, feature vectors are extracted without additional learning using the weights previously learned. Third, in the feature noise removal process, the extracted feature vector is received and the output value of each class is learned to remove noise from the data. Fourth, in the class classification process, the noise-removed feature vector is input to the multi-layer perceptron structure, and the result is output and learned. To evaluate the performance of the proposed algorithm, we experimented with the Extended Yale B face database. As a result of the experiment, in the case of the time required for one-time learning, the proposed algorithm reduced 40.7% based on the existing algorithm. In addition, the number of learning up to the target recognition rate was shortened compared with the existing algorithm. Through the experimental results, it was confirmed that the one-time learning time and the total learning time were reduced and improved over the existing algorithm.

요 약

본 논문에서는 딥러닝 구조를 분할을 이용한 개별 학습을 수행하여 학습시간을 단축하는 알고리즘을 제안한다. 제안하는 알고리즘은 망 분류 기점 설정 과정, 특징 벡터 추출 과정, 특징 노이즈 제거 과정, 클래스 분류 과정 등의 4가지 과정으로 구성된다. 첫 번째로 망 분류 기점 설정 과정에서는 효과적인 특징 벡터 추출을 위한 망 구조의 분할 기점을 설정한다. 두 번째로 특징 벡터 추출 과정에서는 기존에 학습한 가중치를 사용하여 추가 학습 없이 특징 벡터를 추출한다. 세 번째로 특징 노이즈 제거 과정에서는 추출된 특징 벡터를 입력받아 각 클래스의 출력값을 학습하여 데이터의 노이즈를 제거한다. 네 번째로 클래스 분류 과정에서는 노이즈가 제거된 특징 벡터를 입력받아 다층 퍼셉트론 구조에 입력하고 이를 출력하고 학습한다. 제안된 알고리즘의 성능을 평가하기 위하여 Extended Yale B 얼굴 데이터베이스를 사용하여 실험 하였다. 실험 결과, 1회 학습에 소요되는 시간의 경우 제안하는 알고리즘이 기존 알고리즘 기준 40.7% 단축하였다. 또한 목표 인식을까지 학습 횟수가 기존 알고리즘과 비교하여 단축하였다. 실험결과를 통해 1회 학습시간과 전체 학습시간을 감소시켜 기존의 알고리즘보다 향상됨을 확인하였다.

Key words : Deep Learning, Machine Learning, Convolution Neural Networks, Multi Layer Perceptrons, Learning Time

* Dept. Electronic Engineering, Hanbat National University

★ Corresponding author

E-mail : shlee@cad.hanbat.ac.kr, Tel : +82-42-821-1137

Manuscript received May. 17, 2021; revised May. 31, 2021; accepted Jun. 1, 2021.

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.

I. 서론

현재 딥러닝 알고리즘은 검색엔진 및 음성 인식, 그림 인식 등 여러 분야에서 사용되고 있으며 효과적인 인식률을 통해 성능을 입증하고 있다. 초기 딥러닝의 경우 1990년 Yann LeCun의 손글씨 인식에서 convolution 연산을 이용하여 인식률을 향상시키는 결과를 보였지만 당시의 하드웨어 성능 및 많은 연산량으로 인해 실용적인 문제가 있었다[1]. 하지만 이후 1995년 Yann LeCun에 의해 convolution 연산을 이용하여 재정립한 CNN(Convolution Neural Network) 알고리즘이 발표되면서 보다 높은 인식률과 효과적인 데이터 처리가 가능해졌다[2]. 이후 하드웨어의 발달과 함께 병렬처리 기법으로 처리 시간을 대폭 줄이는 등 실용성을 향상시켰다[3-5]. 하지만 여전히 문제점이 남아있다. 첫 번째로 많은 메모리와 연산량이 요구된다. 많은 메모리와 연산량을 처리하기 위해 고속으로 연산이 가능한 고사양 하드웨어를 사용해야 하는 등의 추가적인 문제도 발생한다. 두 번째로 병렬처리를 하더라도 학습에 많은 시간을 소요한다. 학습과정에서 딥러닝 전체 구조를 통해 오차를 전파해야 하기 때문에 많은 시간이 요구되어 사용에 번거로움이 있다.

따라서 딥러닝 구조를 분할을 이용한 개별 학습을 수행하여 학습시간을 단축하는 알고리즘을 제안한다. 그림 1은 본 논문에서 제안한 알고리즘의 전체 순서도이다. 제안하는 알고리즘은 망 분류 기점 설정 과정, 특징 벡터 추출 과정, 특징 노이즈 제거 과정, 클래스 분류 과정 등의 4가지 과정으로 구성된다. 첫 번째로 분류 기점 설정 과정에서는 딥러닝을 통과하여 생성되는 특징 벡터가 어느 지점까지 도달해야 효과적인 특징 벡터를 추출할 것인가 유추하여 구조의 분할 기점을 설정한다. 두 번째로 특징 벡터 추출 과정에서는 기존에 다른 데이터를 학습한 가중치를 사용하여 학습과정을 따로 거치지 않고 특징 벡터를 추출하여 학습시간을 단축한다. 세 번째로 특징 노이즈 제거 과정에서는 이전 단계에서 추출된 특징 벡터를 입력받아 각 클래스에 따라 생성되는 노이즈를 필터링하여 뒷단의 클래스 분류에서 사용될 특징 벡터를 생성한다. 네 번째로 클래스 분류 과정에서는 최종적으로 추출된 특징 벡터를 다층 퍼셉트론 구조를 통해 클래스를 학습하고 분류한다.

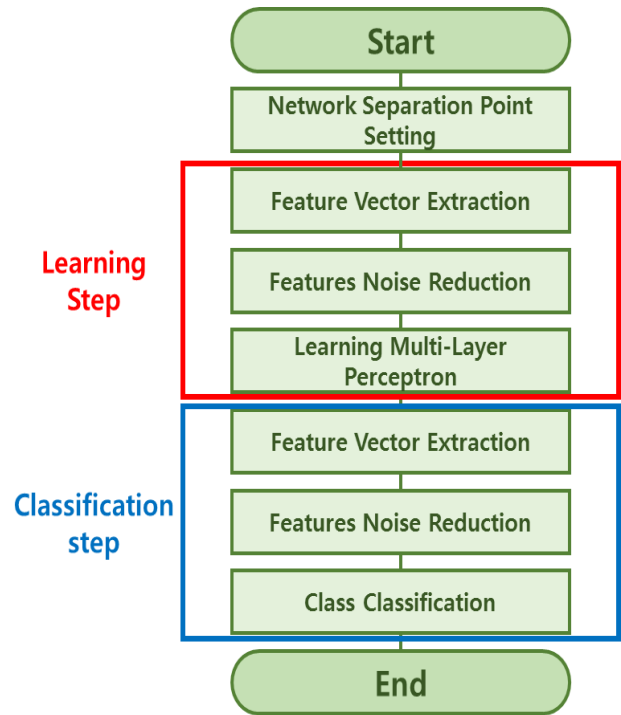


Fig. 1. The overall flowchart of proposed algorithm.
그림 1. 제안하는 알고리즘의 전체 순서도

II. 본론

1. 제안하는 알고리즘의 개요

기존의 딥러닝 알고리즘에서 영상처리에 사용되는 기존의 CNN 알고리즘의 개요도는 그림 2와 같이 입력받은 이미지에서 최종 벡터가 추출되는 지점까지 convolution 연산을 통해 특징 이미지를 생성하고 이를 특징 벡터로 전환하여 다층 퍼셉트론 구조에 입력하여 분류한다. 또한 전체 구조를 역전파 학습하여 많은 연산량과 시간이 필요로 한다. 그러나 그림 3에서와 같이 본 논문에서 제안하는 알고리즘의 경우는 단계에 맞추어 구조적으로 특징 벡터 추출 과정, 특징 노이즈 제거 과정, 클래스 분류 과정 등 3개의 구조로 분할한다. 또한 특징 벡터 추출 과정에서는 임의로 학습되어 있는 가중치를 로드하여 추가 학습이 없이 사용하여 학습시간을 단축한다. 특징 노이즈 제거 과정에서는 출력되는 특징벡터에 대한 평균값을 학습 목표로 하여 학습시킨다. 클래스 분류 과정에서는 앞서 생성한 데이터를 입력받아 클래스를 학습시킨다. 각각 구조에 따라 개별적으로 역전파 학습하기 때문에 역전파 되는 깊이가 줄어들어 연산량과 시간이 줄어들게 되는 장점이 있다.

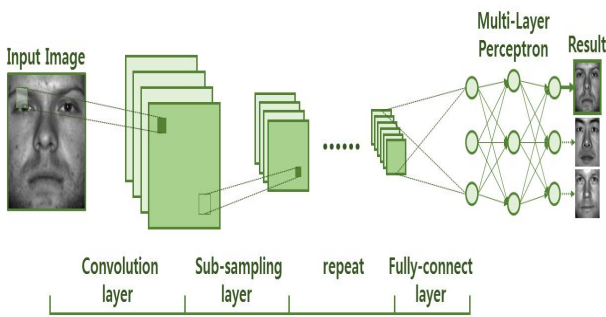


Fig. 2. Outline of CNN algorithm.
그림 2. CNN 알고리즘의 개요도

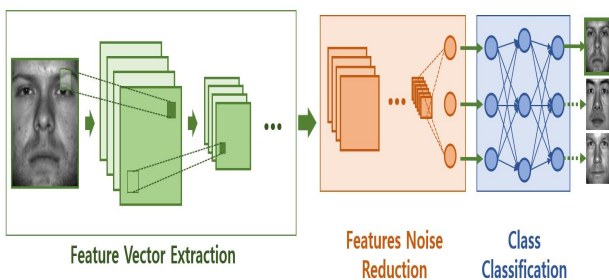


Fig. 3. Outline of proposed algorithm.
그림 3. 제안하는 알고리즘에 대한 개요도

2. 망 분리를 이용한 딥러닝 학습시간 단축 알고리즘
가. 망 분리 기점 설정

망 분리의 기점은 convolution 연산 시 픽셀간의 영향 범위를 기준으로 지정한다. 제안 알고리즘의 전체구조는 VGGNet 모델을 기반으로 구성한다[6]. VGGNet은 작은 convolution 연산 필터를 사용하는 대신 많은 레이어를 가진다. 이를 통해 높은 인식률과 오랜 학습시간이라는 대표적인 CNN 알고리즘의 특징을 가지고 있다. VGGNet은 convolution 레이어를 2회 연산한 뒤 Sub-sampling 레이어를 1회 연산하는 방식이 반복되는 구조를 사용하며 후반단에서 Fully-connect 레이어를 사용하여 신경망의 기본 구조인 MLP(Multi-Layer Perceptron)로 구성되어있다.

Convolution 레이어에서 convolution 연산시 3×3 필터의 지역연산범위로 인해 주변 1픽셀까지 영향을 준다. convolution 연산을 추가로 하면 기존보다 1픽셀 더 영향을 주는데 이는 원본영상 기준으로 첫 번째 픽셀이 총 2칸의 픽셀에 영향을 주는 것과 같다. Sub-sampling 단계에서는 영상의 너비와 높이가 반으로 줄어들고 이를 통해 픽셀간 영향을 주는 범위가 2배로 된다. 이는 원본영상 기준으로 첫 번째 픽셀이 총 4칸의 픽셀에 영향을 주는 것과 같

다. 이와 같은 과정을 반복하면 그림 4와 같이 픽셀간 영향의 범위가 증가한다. 입력이미지가 224pixel 일 경우 Convolution 레이어 8개와 Sub-sampling 레이어 4개를 사용하면 모든 픽셀에 영향을 줄 수 있다. 이를 특징 벡터 추출에 사용되는 레이어 범위로 설정한다.

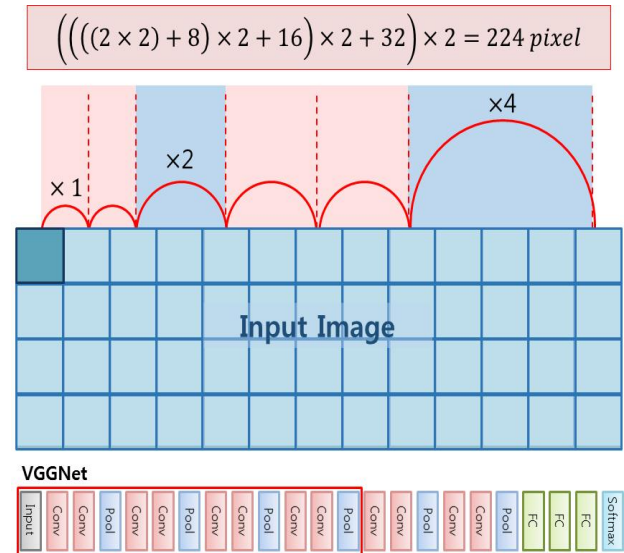


Fig. 4. Feature vector extraction layer range setting.
그림 4. 특징 벡터 추출 레이어 범위 설정

나. 특징 벡터 추출

특징 벡터 추출 과정은 그림 5와 같이 Convolution 레이어와 Sub-sampling 레이어의 반복하여 연산된다. Convolution 레이어에서는 주변 픽셀을 포함한 지역 연산 방법인 convolution 연산을 수행하기 때문에 노이즈의 간섭에 강인하고 지역적인 특징 추출이 가능하다. Sub-sampling은 알고리즘의 연산량이 많아 연산에 방해되는 요소를 줄여 데이터를 줄이는 단계이다. 주로 주변 픽셀값 중 최대값을 출력하는 max-pooling 방식을 사용한다.

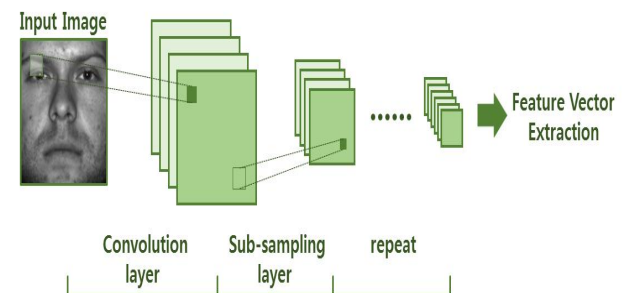


Fig. 5. Feature vector extraction process.
그림 5. 특징 벡터 추출 과정

기존의 CNN 알고리즘은 전체구조를 통과하여 출력된 결과와 목표값 사이의 오차를 역전파하여 학습한다. 하지만 이를 초기값으로부터 학습을 수행하는 경우 학습에 매우 많은 시간이 걸리기 때문에 최근에는 이미 다른 데이터들을 통해 학습된 가중치를 로드하여 학습을 시작한다[7]. 이러한 기법의 원리는 어떠한 데이터를 학습하던 상관없이 초기 필터들이 영상에서 대상을 표현하기 위한 기본 요소를 이루는 특징 데이터를 추출하기 때문이다. 본 논문에서는 이러한 원리를 이용하여 이미 다른 데이터가 학습되어 있는 가중치를 로드하여 사용과 동시에 망 분리 기점을 통해 구분한 특징 벡터 추출 레이어를 따로 학습 수행하지 않는다. 이를 통해 학습을 수행하는 레이어의 수를 감소시키고 역전파 학습하는 깊이가 줄어들어 연산량과 학습 시간을 감소시킨다.

다. 특징 노이즈 제거

특징 노이즈 제거 단계에서는 그림 6과 같이 앞단의 레이어를 통해 추출된 특징 벡터를 노이즈 제거 레이어에 입력하고 출력값을 저장한다. 이때 같은 클래스에 대한 출력값은 클래스의 특징 벡터에 노이즈가 더해진 것으로 추정한다. 이후 같은 클래스의 출력값들에 대한 평균 값을 계산한다.

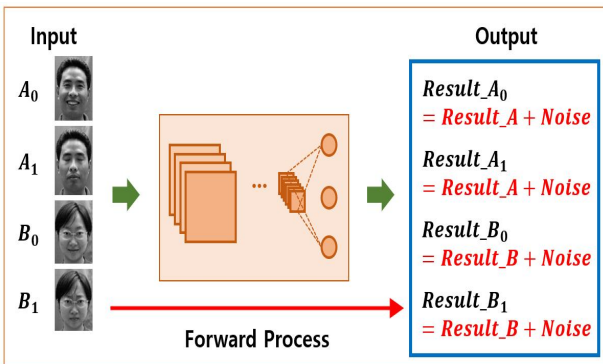


Fig. 6. Forward process of noise reduction network.
그림 6. 노이즈 제거망의 forward 처리 과정

이때 학습에 사용되는 데이터의 수가 적으면 평균값에 노이즈의 영향이 커질 확률이 높다. 하지만 일반적으로 딥러닝에서 학습에 사용되는 데이터의 수가 매우 많아 노이즈의 영향이 줄어든다. 이후 그림 7과 같이 생성한 데이터를 학습목표로 지정하고 이를 학습한다.

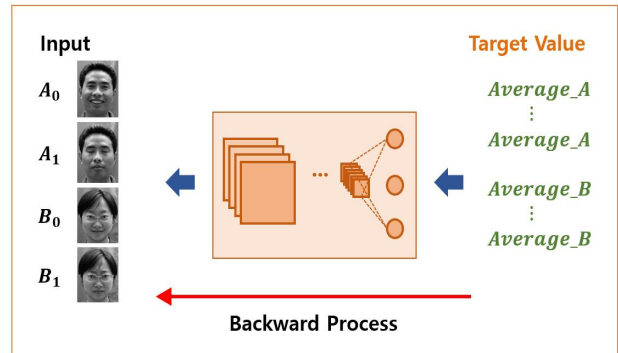


Fig. 7. Backward process of noise reduction network.
그림 7. 노이즈 제거망의 backward 처리 과정

라. 클래스 분류

클래스 분류 단계에서는 그림 8과 같이 Fully-connect 레이어 구조에서 특징 노이즈 제거 단계를 통해 정의한 각 클래스별 평균값을 입력값으로 지정한다. 이후 클래스 값에 따라 개별 학습한다.

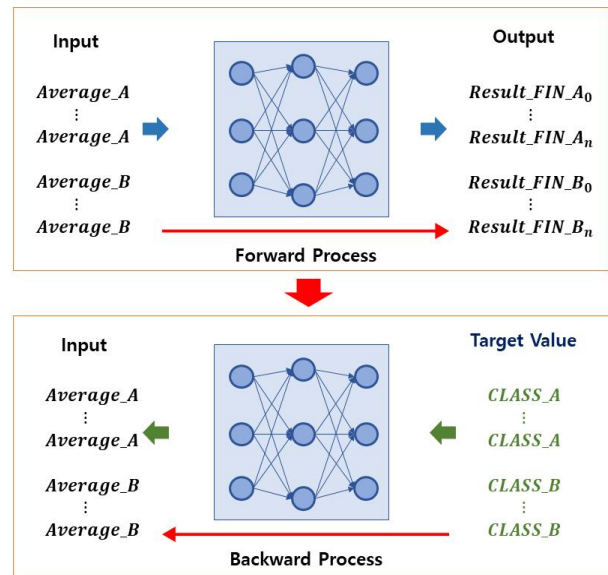


Fig. 8. Class classification process.
그림 8. 클래스 분류 과정

제안하는 알고리즘의 경우에 특징 벡터 추출 단계에서는 학습을 수행하지 않는다. 또한 특징 노이즈 제거단계와 클래스 분류 단계 등 학습을 수행하는 구조를 나누어 학습을 수행한다. 이를 통해 학습을 수행하는 깊이를 줄여 역전파 알고리즘의 연산량을 감소시키고 이를 통해 학습시간을 단축시킨다. 이때 쓰레드 등을 통한 병렬처리 기법을 사용하면 2개의 구조를 동시에 학습 가능하여 추가로 학습시간을 단축시킬 수 있다.

3. 실험

가. 실험 환경

본 실험에 사용된 하드웨어는 Intel(R) Core(TM) i5-3570 3.40GHz CPU 프로세서, RAM 16GB, NVIDIA GeForce GTX 1080(V-RAM 8GB) GPU 등으로 구성있다. 또한 Windows 10 Pro 64비트 운영체제와 Visual Studio 2015 개발도구 환경에서 수행하였다. 또한 Opencv 3.0 및 CUDA 8.0, cuDNN 4 라이브러리를 사용하여 실험하였다.

본 논문에서 제안하는 알고리즘의 검증에 사용된 데이터베이스로 Extended Yale B 데이터베이스를 사용하였다. Extended Yale B 데이터베이스는 Yale 대학교에서 다양한 조명 환경을 가지는 데이터가 특징이다. 총 38개의 클래스로 구성되어 있으며 각 클래스마다 64가지의 조명환경의 얼굴데이터이다. 실험에서는 데이터베이스를 8비트의 색구성으로 사용하였다. Extended Yale B 데이터베이스는 그림 9와 같이 조명의 정도에 따라 5개의 subset을 가지며 실험에서는 sub1 데이터를 학습하여 sub2~sub5 데이터로 인식률 및 학습 시간 검증을 수행하였다. 비교 알고리즘으로는 기존 VGGNet 구조를 사용한 일반 CNN 알고리즘으로 선택하였다.



Fig. 9. Extended Yale B database.
그림 9. Extended Yale B 데이터베이스

나. 실험 결과

비교 실험결과 표 1과 같이 VGGNet의 경우에 1회 학습에 약 65.3초 소요되었고 제안하는 알고리즘은 약 38.7초가 소요되었다. 1회 학습시 소요되는 시간이 VGGNet에 비교하여 26.6초 단축하여 학습시간이 약 40.7% 단축되었다. 표 2와 그림 10은 VGGNet의 학습경과에 따른 데이터의 에러율을 나타내고, 표 3과 그림 11은 제안하는 알고리즘의 학습경과에 따른 데이터의 에러율을 나타낸다. VGGNet은 학습 6회만에 0.0001 이하의 에러값을 보였으며 제안하는 알고리즘은 학습 5회만에 0.0001 이하의 에러값을 보였다. 또한 VGGNet의 학습경과에 비교하여 제안하는 알고리즘의 에러율이 비교적 안정적으로 감소하는 것을 확인하였다. 표 4와 그림 12는 같은 시간동안 학습 후 두 알고리즘의 인식률을 비교하였다. sub2 데이터의 경우 모두 100%의 인식률을 보였고 sub3에서부터 sub5까지 데이터의 인식률을 테스트 하였을 때 제안 알고리즘의 인식률이 더 우수한 결과를 보였다.

Table 1. Comparison of time required for learning once.

표 1. 1회 학습 소요시간 비교

Model	VGGNet	Proposed algorithm
Time required for learning once	65.3 sec	38.7 sec

Table 2. Error rate of data according to learning progress (VGGNet).

표 2. 학습 경과에 따른 데이터의 에러율(VGGNet)

Number of learning	Error rate
1	3.6635600000
2	0.0065643100
3	0.0094469200
4	0.0007384290
5	0.0177962000
6	0.0000290543
7	0.0020319200
8	0.0010056600
9	0.0002395420
10	0.0002582300
11	0.0001311950



Fig. 10. Graph of error rate of data according to learning progress(VGGNet).

그림 10. 학습 경과에 따른 데이터의 에러율 그래프(VGGNet)

Table 3. Error rate of data according to learning progress (proposed algorithm).

표 3. 학습 경과에 따른 데이터의 에러율(제안된 알고리즘)

Number of learning	Error rate
1	3.6635600000
2	0.0460282000
3	0.0063222300
4	0.0001487780
5	0.0000535993
6	0.0000610219
7	0.0000592422
8	0.0000868286
9	0.0000506458
10	0.0000628287
11	0.0000362526

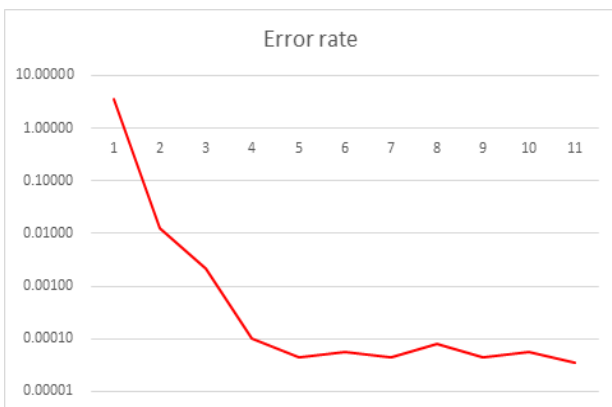


Fig. 11. Graph of error rate of data according to learning progress(proposed algorithm).

그림 11. 학습 경과에 따른 데이터의 에러율 그래프 (제안된 알고리즘)

Table 4. Comparison of recognition rate of proposed algorithm with VGGNet.

표 4. VGGNet과 제안된 알고리즘의 인식률 비교

Model (learning time)	VGGNet (700seconds)	proposed algorithm (700seconds)
sub2	100.00%	100.00%
sub3	88.99%	90.53%
sub4	54.36%	56.92%
sub5	16.53%	19.47%

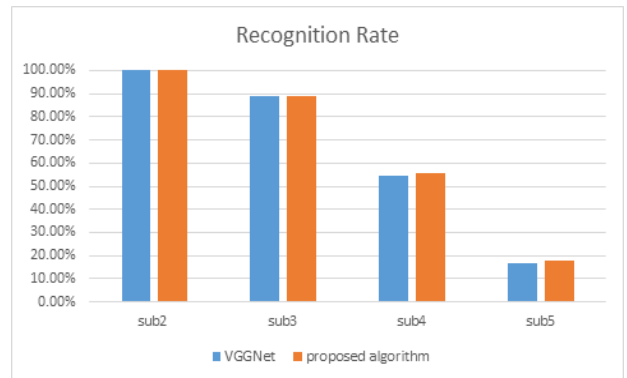


Fig. 12. Comparison of recognition rate of VGGNet and proposed algorithm.

그림 12. VGGNet과 제안된 알고리즘의 인식률 비교 그래프

III. 결론

본 논문에서는 망 분리를 이용하여 딥러닝 학습 시간을 단축시키는 알고리즘을 제안하였다. 제안하는 알고리즘은 1회 학습 소요시간을 기존 알고리즘과 비교하여 40.7% 감소시켜 학습시간을 단축하는 성능을 확인하였다. 또한 기존 알고리즘과 비교하여 목표 인식률에 더 빨리 도달함을 통해 성능을 입증하였다.

향후 연구 방향으로 정교한 인식률을 위한 구조 및 학습방법에 대한 연구가 필요하다고 사료된다.

References

[1] LeCun, Yann, et al. "Handwritten digit recognition with a back-propagation network,," *Advances in neural information processing systems*. 1990. DOI: 10.5555/109230.109279

[2] LeCun, Yann, et al. "Learning algorithms for classification: A comparison on handwritten digit

recognition,” *Neural networks: the statistical mechanics perspective*, pp.261-276, 2015.

[3] H. K. Jeon, K. Y. Lee, C. Y. Kim, “An Implementation of a Convolutional Accelerator based on a GPGPU for a Deep Learning,” *j.inst.Korean.electr.electron.eng*, vol.20, no.3, pp.303-306, 2016. DOI: 10.7471/ikeee.2016.20.3.303

[4] J. W. Bang, Y. J. Jeong, “A Real-Time Hardware Design of CNN for Vehicle Detection,” *j.inst.Korean.electr.electron.eng*, vol.20, no.4, pp.351-360, 2016. DOI: 10.7471/ikeee.2016.20.4.351

[5] H. Y. Lee, S. H. Lee, “A Study On Memory Optimization for Applying Deep Learning to PC,” *j.inst.Korean.electr.electron.eng*, vol.21, no.2, pp. 136-141, 2017. DOI: 10.7471/ikeee.2017.21.2.136

[6] Simonyan, Karen, Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556 2014.

[7] LeCun, Yann, Yoshua Bengio, Geoffrey Hinton, “Deep learning,” *nature* vol.521, pp.436-444, 2015.

BIOGRAPHY

Hee-Yeol Lee (Member)



2016 : BS degree in Electronic Engineering, Hanbat National University
 2018 : MS degree in Electronic Engineering, Hanbat National University
 2018~current : Ph. D degree course of Electronic Engineering, Hanbat National University

Seung-Ho Lee (Member)



1986 : BS degree in Electronic Engineering, Hanyang University
 1989 : MS degree in Electronic Engineering, Hanyang University
 1994 : Ph. D degree in Electronic Engineering, Hanyang University
 1994~current : Professor, Department of Electronic Engineering, Hanbat National University