

<https://doi.org/10.7236/JIIBC.2022.22.4.171>

JIIBC 2022-4-24

테이터별 딥러닝 학습 모델의 정확도 향상을 위한 외곽선 특징 적용방안 연구

A Study on Application Method of Contour Image Learning to improve the Accuracy of CNN by Data

권용수*, 황승연**, 신동진***, 김정준****

Yong-Soo Kwon*, Seung-Yeon Hwang**, Dong-Jin Shin***, Jeong-Joon Kim****

요 약 CNN은 딥러닝의 한 종류로 이미지나 영상 데이터를 처리할 때 사용하는 신경망이다. 필터가 이미지를 순회하며 이미지의 특징을 추출하여 이미지를 구분한다. 딥러닝은 데이터가 많을수록 좋은 모델을 만들 수 있는 특징이 있고, CNN에서는 적은 데이터의 약점을 보완하기 위해 회전, 확대, 이동, 뒤집기 같은 방법의 데이터 증강이라는 기법으로 데이터의 양을 인위적으로 늘리는 방법을 사용한다. 외곽선 이미지 학습은 이미지 데이터에서 외곽선에 해당하는 영역을 추출하는 것이다. CNN 학습 시, 외곽선 이미지 학습이 기존의 데이터 증강기법과 비교하여 성능 향상의 도움이 되는지 확인하고자 한다.

Abstract CNN is a type of deep learning and is a neural network used to process images or image data. The filter traverses the image and extracts features of the image to distinguish the image. Deep learning has the characteristic that the more data, the better models can be made, and CNN uses a method of artificially increasing the amount of data by means of data augmentation such as rotation, zoom, shift, and flip to compensate for the weakness of less data. When learning CNN, we would like to check whether outline image learning is helpful in improving performance compared to conventional data augmentation techniques.

Key Words : CNN, Contour detection, Data augmentation

1. 서 론

딥러닝의 발전함에 따라 이미지나 영상 처리를 하는 신경망인 CNN도 VGGNet, GoogLeNet, ResNet 순서로 발전해왔다. 그 중 CNN에서 훈련데이터의 양이 부족

할 때 이를 늘리기 위한 방법으로 데이터 증강기법을 사용한다. 이는 이미지를 회전, 확대, 이동, 뒤집기와 같은 방법으로 기존 이미지에 변화를 주어 새로운 이미지 데이터를 만드는 기법이다.

본 논문은 기존의 방법과는 다르게 이미지의 외곽선을

*준회원, 안양대학교 ICT융합학부 소프트웨어전공

**준회원, 안양대학교 컴퓨터공학과

***준회원, 안양대학교 컴퓨터공학과

****정회원, 안양대학교 ICT융합학부 소프트웨어전공

접수일자 2022년 7월 14일, 수정완료 2022년 7월 30일

게재확정일자 2022년 8월 5일

Received: 14 July, 2022 / Revised: 30 July, 2022 /

Accepted: 5 August, 2022

****Corresponding Author: jkim@anyang.ac.kr

Dept. ICT Convergence Engineering at Anyang University, Korea

추출한 데이터를 학습에 사용하면 CNN 정확도 향상으로 이어지는지 확인해보는 것을 목적으로 한다.

본 논문은 다음과 같이 구성된다. 2장에서 CNN, 데이터 증강, 외곽선 추출과 관련된 연구를 살펴보고, 3장에서는 사용될 모델을 설명하고, 4장에서는 실험 결과를 분석하여 5장에서 결론을 내어 외곽선 추출 이미지가 CNN 학습에 미치는 영향을 확인한다.

II. 관련 연구

1. CNN(Convolutional Neural Networks)^[1]

CNN(Convolutional Neural Networks)은 딥러닝에서 주로 이미지나 영상 데이터를 처리할 때 쓰이는 신경망 모델이다.

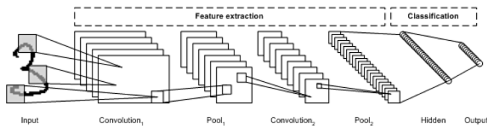


그림 1. CNN의 구조
Fig. 1. CNN architecture

CNN은 그림 1과 같이 이미지 데이터가 들어오면 입력데이터를 필터가 순회하며 합성곱을 계산하고, 그 계산 결과를 이용하여 Feature map을 만든다. Convolution Layer는 Filter 크기, Stride, Padding 적용 여부, Max Pooling 크기에 따라서 출력 데이터의 Shape이 변경된다.

2. 데이터 증강(data augmentation)^{[2][3]}

데이터 증강은 CNN에서 데이터 양이 적을 때 데이터의 양을 증가시키는 기법이다. 회전(Rotation), 크기(Scale), 반전(Flip) 등을 기존 이미지에 적용하여 새로운 데이터를 만들어내는 것으로 CNN 학습 시 데이터 부족의 문제를 해소할 수 있다.

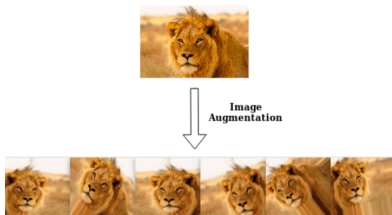


그림 2. 데이터 증강
Fig. 2. Data augmentation

그림 2에서 볼 수 있듯이 하나의 이미지로 여러 이미지를 만들어내는 것이 데이터 증강이다.

3. 외곽선 감지(contour detection)^[4]

외곽선 감지는 이미지 데이터에서 외곽선을 검출하는 기법이다. 본 논문에서 사용할 알고리즘은 Canny Edge Detection이다.



그림 3. 외곽선 감지
Fig. 3. contour detection

III. 모델 설계

1. CNN 모델

가. GoogLeNet^[5]

GoogLeNet은 2014년 이미지넷 인식 대회(ILSVRC)에서 VGGNet(VGG19)을 이기고 우승을 차지한 알고리즘이다. GoogLeNet은 3가지 특징을 가지고 있다. 인셉션(Inception) 모듈, Global average pooling, Auxiliary classifier가 3가지 특징이다. 인셉션 모듈은 다양한 필터를 사용하여 기존 CNN보다 다양한 특성을 추출할 수 있다는 것이 특징이고, Global average pooling은 기존 Fully connected layer 대신 신경망 후반에 사용하는 것으로 가중치 개수를 줄여 신경망을 가볍게 해준다. Auxiliary classifier는 GoogLeNet의 vanishing gradient 문제를 해결하는 것으로 학습하는 과정에서만 쓰이고 학습이 끝난 뒤에는 신경망에서 삭제된다.

장단점) 장점으로는 Global average pooling를 사용하여 모델이 가볍다는 것과 22층으로 구성된 층으로 정확도가 높다는 것이다.

단점은 인셉션 모듈의 사용, Auxiliary classifier의 사용으로 구조가 복잡하다는 점이다.

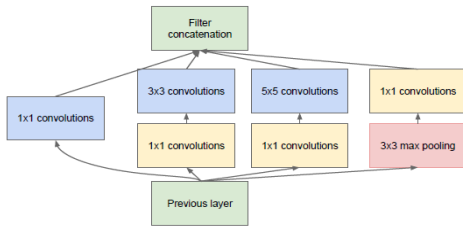


그림 4. 인셉션 모듈
 Fig. 4. Inception module

나. VGGNet^[6]

VGGNet은 옥스퍼드 대학의 연구팀 VGG에 의해 개발된 모델로써, 2014년 이미지넷 인식 대회(ILSVRC)에서 준우승을 한 모델이다.

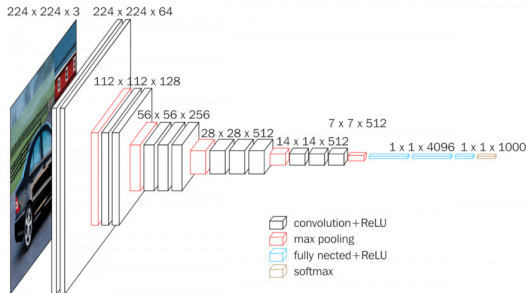


그림 5. VGG16의 구조
 Fig. 5. VGG16 architecture

VGGNet은 모든 컨볼루션 층에서 3x3 필터, padding은 동일, stride는 1, 활성화 함수는 ReLU 함수를 사용한다. 본 논문에서 사용할 모델은 VGG16이다.

그림 5는 VGG16의 구조로 합성곱과 max pooling의 반복하는 과정과 신경망 후반에 Fully connected layer 층을 쌓고 soft-max 함수로 활성화하는 것으로 구성된다.

장단점) 16층으로 깊게 구성된 층으로 정확도가 높고, 구조가 쉽다는 점이 장점이다.

단점은 매우 많은 메모리를 사용하며 많은 연산을 한다는 것이다.

다. ResNet^[7]

마이크로소프트에서 개발한 알고리즘인 ResNet은 2015년 이미지넷 인식 대회(ILSVRC)에서 우승을 차지한 모델이다. ResNet은 신경망이 깊어짐에 따라 발생하는 문제인 vanishing gradient 문제를 해결하기 위해 residual block을 사용하였다.

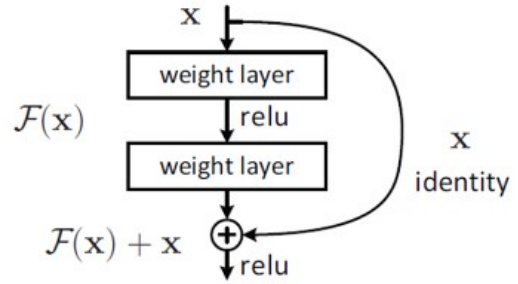


그림 6. residual block
 Fig. 6. residual block

residual block은 기존 네트워크에서 그림 6처럼 입력값을 출력값에 더해줄 수 있도록 지름길(shortcut)을 하나 만들어준 것이다.

장단점) 50층으로 구성된 깊은 신경망으로 정확도가 높다는 점이 장점이며, 단점은 입력과 출력의 dimension을 통일해줘야 한다는 점이다.

2. 데이터 증강

데이터 증강은 keras의 ImageDataGenerator를 사용하여 진행한다. ImageDataGenerator는 다음과 같은 인자를 통해 데이터 증강을 진행할 수 있다.

- rotation_range : 이미지 회전 범위 (degrees)
- width_shift, height_shift : 그림을 수평 또는 수직으로 랜덤하게 평행 이동시키는 범위 (원본 가로, 세로 길이에 대한 비율 값)
- rescale : 데이터 픽셀의 범위를 0과 1사이로 scaling 하기 위한 인자
- zoom_range : 임의 확대/축소 범위
- horizontal_flip : True로 설정할 경우, 50% 확률로 이미지를 수평으로 뒤집는다.
- fill_mode : 이미지를 회전, 이동하거나 축소할 때 생기는 공간을 채우는 방식

본 논문에서는 수평, 수직 이동 범위는 0.1, 확대 축소 범위는 0.2, 수평 반전은 True, 이미지 채우기 방식은 근처와 비슷하게 하여 진행하였다.

```
rescale=1./255,
width_shift_range=0.1,
height_shift_range=0.1,
zoom_range=0.2,
```

horizontal_flip=True,
fill_mode='nearest'

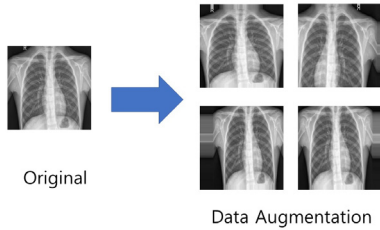


그림 7. 데이터 증강 예시
Fig. 7. Example of data augmentation

그림 7은 위와 같은 설정을 실제 데이터에 적용한 것이다.

3. 외곽선 추출

외곽선 추출은 Canny Edge Detection을 이용하여 이미지의 외곽선을 추출하였다.

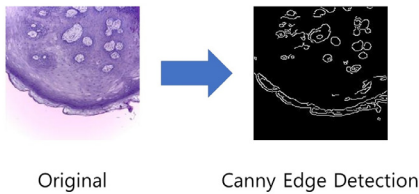


그림 8. Canny Edge Detection 예시
Fig. 8. Example of Canny Edge Detection

그림 8은 Canny Edge Detection을 실제 데이터에 적용한 것이다. 이렇게 추출된 데이터는 기존 데이터와 함께 학습에 사용한다.

IV. 실험 및 결과

1. 데이터셋

실험 데이터셋은 폐렴, 구강암, 피부암 데이터셋을 사용하였다. 폐렴 데이터셋^[8]은 흑백의 X-Ray 이미지 데이터로 정상 데이터, 폐렴 데이터로 나뉜다. 구강암 데이터셋^[9]은 정상 데이터와 OSCC라는 구강 편평세포암종 데이터로 나뉜다. 피부암 데이터셋^[10]은 9가지의 피부암으로 나뉜다.

2. 실험 방법^{[11][12]}

먼저 모든 데이터를 224 x 224 크기의 해상도를 갖도록 resize한다. 처리된 데이터를 갖고 다음과 같은 기준을 통해 학습을 진행한다.

- 가. CNN 모델별 성능 비교
- 나. 신경망을 처음부터 학습시키는 것과 사전훈련(pretrained) 모델과의 성능 비교
- 다. 데이터셋 종류에 따른 성능 비교
- 라. 데이터 증강과 외곽선 여부에 따른 4가지 상황별 성능 비교(둘 다 적용 안 함, 데이터 증강만 적용, 외곽선 추출만 적용, 둘 다 적용)
- 마. 각 케이스 별 훈련 데이터 양에 따른 성능 비교(전체 훈련 데이터에서 100%, 70%, 40% 데이터만 사용하여 훈련)

하이퍼 파라미터는 다음과 같이 설정하여 학습을 진행하였다.

Epoch : 100

Learning Rate : 0.001

옵티마이저는 Adam을 사용하였고, 학습 후 최종적으로 저장되는 모델은 학습 중 validation accuracy가 가장 높은 모델이다.

표 1. 구강암 데이터셋 실험결과 1

Table 1. Oral Cancer Dataset Experiment Result 1

구강암 데이터셋	적용 x	데이터 증강만	외곽선 데이터 추가	모두 적용
GoogLeNet	0.5123/	0.6139/	0.5102/	0.5350/
	0.5215/	0.5183/	0.5100/	0.5100/
	0.5038	0.5688	0.5194	0.5201
VGGNet	0.5190 /	0.5231/	0.5259/	0.5253/
	0.5190/	0.5228/	0.5219/	0.5313/
	0.4987	0.5038	0.5038	0.5101
ResNet	0.8191/	0.8075/	0.6388/	0.6350/
	0.8413/	0.8584/	0.6913/	0.6637/
	0.9361	0.8537	0.6787	0.6350

표 2. 구강암 데이터셋 실험결과 2 (사전 훈련된 모델)

Table 2. Oral Cancer Dataset Experiment Result 2 (Pretrained model)

구강암 데이터셋	적용 x	데이터 증강만	외곽선 데이터 추가	모두 적용
GoogLeNet	0.8246/	0.9054/	0.7188/	0.8250/
	0.8944/	0.9437/	0.8444/	0.8498/
	0.9028	0.9041	0.8083	0.8810
VGGNet	0.5230/	0.5142/	0.5263/	0.5131/
	0.5297/	0.5228/	0.5269/	0.5256/
	0.5102	0.5230	0.5225	0.5213
ResNet	0.9800/	0.9508/	0.9638/	0.8062/
	0.9931/	0.9810/	0.9794/	0.8398/
	0.9885	0.9438	0.9862	0.8396

표 1과 2는 구강암 데이터셋의 실험결과다. 한 칸 안에 들어있는 정확도는 각각 훈련 데이터의 양을 100%, 70%, 40% 사용했을 때의 모델의 정확도를 나타낸다.

표 3. 피부암 데이터셋 실험결과 1
Table 3. Skin Cancer Dataset Experiment Result 1

피부암 데이터셋	적용 x	데이터 증강만	외곽선 데이터 추가	모두 적용
GoogLeNet	0.2029/ 0.2261/ 0.2188	0.4614/ 0.5151/ 0.2281	0.6625/ 0.3363/ 0.2018	0.3844/ 0.3615/ 0.2612
VGGNet	0.2029/ 0.2028/ 0.2000	0.1839/ 0.2050/ 0.2344	0.2044/ 0.1882/ 0.2147	0.2119/ 0.2240/ 0.1656
ResNet	0.6441/ 0.5325/ 0.4375	0.5664/ 0.5516/ 0.4062	0.3850/ 0.2450/ 0.3675	0.3375/ 0.3675/ 0.3200

표 4. 피부암 데이터셋 실험결과 2 (사전 훈련된 모델)
Table 4. Skin Cancer Dataset Experiment Result 2 (Pretrained model)

피부암 데이터셋	적용 x	데이터 증강만	외곽선 데이터 추가	모두 적용
GoogLeNet	0.4987/ 0.5938/ 0.6209	0.8331/ 0.6965/ 0.7500	0.5781/ 0.7067/ 0.6925	0.5044/ 0.6060/ 0.6137
VGGNet	0.2035/ 0.1675/ 0.2094	0.1915/ 0.2103/ 0.2062	0.1887/ 0.1775/ 0.2005	0.1912/ 0.1950/ 0.2096
ResNet	0.9469/ 0.9559/ 0.9902	0.8995/ 0.8938/ 0.9125	0.9306/ 0.9036/ 0.9715	0.5650/ 0.5537/ 0.6488

표 3과 4는 피부암 데이터셋의 실험결과다.

표 5. 폐렴 데이터셋 실험결과 1
Table 5. Pneumonia Dataset Experiment Result 1

폐렴 데이터셋	적용 x	데이터 증강만	외곽선 데이터 추가	모두 적용
GoogLeNet	0.9881/ 0.9760/ 0.9762	0.9663/ 0.9501/ 0.9287	0.9296/ 0.8619/ 0.7400	0.9097/ 0.9038/ 0.7175
VGGNet	0.7400/ 0.7330/ 0.9944	0.7212/ 0.7156/ 0.7285	0.8926/ 0.7325/ 0.9585	0.7319/ 0.7331/ 0.7244
ResNet	0.9525/ 0.9638/ 0.9937	0.9625/ 0.9500/ 0.9375	0.9388/ 0.9275/ 0.9572	0.8737/ 0.8813/ 0.8438

표 6. 폐렴 데이터셋 실험결과 2 (사전 훈련된 모델)
Table 6. Pneumonia Dataset Experiment Result 2 (Pretrained model)

폐렴 데이터셋	적용 x	데이터 증강만	외곽선 데이터 추가	모두 적용
GoogLeNet	0.9518/ 0.9539/ 0.9542	0.9319/ 0.9375/ 0.9377	0.9223/ 0.9044/ 0.9256	0.8981/ 0.8819/ 0.9005
VGGNet	0.7339/ 0.7355/ 0.9580	0.7332/ 0.7323/ 0.7266	0.7394/ 0.7274/ 0.7294	0.7331/ 0.7350/ 0.7288
ResNet	0.9981/ 1.0000/ 1.0000	0.9725/ 0.9640/ 0.9599	0.9819/ 0.9783/ 0.9994	0.9310/ 0.9000/ 0.9246

표 5와 6은 폐렴 데이터셋의 실험결과다.

V. 결 론

실험의 결과를 기준으로 살펴보면, 먼저 CNN 모델별 성능은 ResNet, GoogLeNet, VGGNet 순서로 성능이 좋았다.

사전 훈련 여부를 기준으로 본다면, VGGNet은 사전 훈련된 모델이 특별히 뛰어난 성능을 보여주진 않았지만, GoogLeNet과 ResNet은 사전 훈련된 모델이 훨씬 높은 정확도를 갖는 것을 보여줬다. 두 신경망 모두 구조가 단순하지 않은 만큼 사전 훈련 여부가 정확도에 많은 영향을 준 것 같다.

데이터셋을 기준으로 비교해 보면 폐렴, 구강암, 피부암 데이터셋 순서로 정확도가 높았다.

데이터 증강과 외곽선 여부에 따른 4가지 상황별 성능 비교를 해보자면, 데이터 증강으로 정확도 향상을 이룬 모델은 구강암과 피부암에서 GoogLeNet이 효과를 봤고, 이것을 제외하면 비슷하거나 오히려 정확도가 낮아지는 경우가 많았다. 외곽선 데이터 추가도 피부암에서 GoogLeNet이 효과를 본 것 말고는 정확도가 떨어졌고, 모두 적용한 경우는 VGGNet은 크게 영향을 받지 않았고, GoogLeNet과 ResNet은 정확도가 꽤 낮아진 경우가 대부분이다. GoogLeNet이 구강암에서 외곽선 데이터 추가의 효과를 본 것 말고는 외곽선 데이터 추가는 정확도 향상에 도움이 되지 못했고, 외곽선 데이터 추가와 데이터 증강을 모두 적용한 경우는 성능이 오히려 낮아졌다.

각 케이스별 훈련 데이터 양에 따른 성능 비교는 데이터의 양이 많았을 때 학습이 잘 되는 경우와 적었을 때

오히려 잘 되는 경우가 있어서 정확한 판단이 어렵다.

궁극적으로, 본 논문에서 알아보고자 했던 새로운 데이터 증강 기법으로 외곽선 이미지가 신경망 성능 향상에 도움이 되는지에 대한 것의 결론은 외곽선 이미지는 성능 향상에 도움이 되는 경우는 거의 없다는 것이다.

References

- [1] "CNN, Convolutional Neural Network Summary", Jan 2018.
- [2] "Building powerful image classification models using very little data", 2018.
- [3] "Data Augmentation", Aug 2021.
- [4] "Canny Edge Detection Step by Step in Python — Computer Vision", Jan 2019.
- [5] "Going Deeper with Convolutions", Sep 2014.
DOI: <https://arxiv.org/abs/1409.4842>
- [6] "Very Deep Convolutional Networks for Large-Scale Image Recognition", Apr 2015.
DOI: <https://arxiv.org/abs/1409.1556>
- [7] "Deep Residual Learning for Image Recognition", Dec 2015.
DOI: <https://arxiv.org/abs/1512.03385>
- [8] kaggle "Chest X-Ray Images", 2018.
- [9] kaggle "Histopathologic Oral Cancer Detection using CNNs", 2021.
- [10] kaggle "Skin Cancer ISIC", 2019.
- [11] Seok-Jin Kwon, Min-Soo Kim, "Flaw Evaluation of Bogie connected Part for Railway Vehicle Based on Convolutional Neural Network", Journal of the Korea Academia-Industrial cooperation Society(JKAIS), Vol. 21, No. 11, pp. 53-60. 2020.
DOI: <https://doi.org/10.5762/KAIS.2020.21.11.53>
- [12] Byoung-Woo Oh, "A Method of Spatial Data Classification using CNN", The Journal of KIIT, Vol. 20, No. 4, pp. 25-30. 2022.
DOI : <https://doi.org/10.14801/kiit.2022.20.4.25>

저 자 소 개

권 용 수(준회원)

- Yong-Soo Kwon is currently studying software at Anyang University. His research interests include Deep Learning, Database, Big data, etc.

황 승 연(준회원)

- Seung-Yeon Hwang received his BS in Computer Engineering at Korea Polytechnic University in 2019. He is currently a Master's course in the department of Smart Manufacturing Engineering at Korea Polytechnic University. His research interests include Big Data, Internet of Things(IoT), Machine Learning, etc.

신 동 진(준회원)

- Dong-Jin Shin received his BS in Engineering at Korea Polytechnic University in 2018. He is currently a Master's course in the department of Smart Manufacturing Engineering at Korea Polytechnic University. His research interests include Big Data, Internet of Things(IoT), Network Security, etc.

김 정 준(정회원)

- Jeong Joon Kim received his BS and MS in Computer Science at Konkuk University in 2003 and 2005, respectively. In 2010, he received his PhD in at Konkuk University. He is currently a professor at the department of Computer Science at Korea Polytechnic University. His research interests include Database Systems, BigData, Semantic Web, Geographic Information Systems (GIS) and Ubiquitous Sensor Network (USN), etc.