

CNN의 컨볼루션 레이어, 커널과 정확도의 연관관계 분석

공준배* · 장민석**

Association Analysis of Convolution Layer, Kernel and Accuracy in CNN

Jun-Bea Kong* · Min-Seok Jang**

요약

본 논문에서는 CNN의 컨볼루션 레이어 개수 및 커널의 크기와 개수가 CNN에 어떠한 영향을 끼치는지 실험을 통해 알아보기 위해 진행하였다. 또한 분석을 위해 일반적인 CNN도 실험하여 실험에 사용된 CNN과 비교하였다. 분석에 사용될 신경망들은 CNN을 기반으로 하며 각각의 실험모델들은 레이어 개수, 커널의 크기 및 개수를 일정한 값으로 고정해 실험을 진행하였다. 모든 실험에는 2계층의 완전연결계층을 고정으로 사용하였다. 다른 변수들은 모두 동일한 값을 주어 실험하였다. 분석결과 레이어의 수가 작을 경우 커널의 크기 및 개수와 상관없이 데이터의 분산 값이 작아 견고한 정확도를 보여주었다. 레이어의 수가 커질수록 정확도도 증가했으나 일정 수치 이상부터는 오히려 정확도가 내려갔으며 분산 값도 커져 정확도 편차가 크게 나타났다. 커널의 개수는 다른 변수보다 학습속도에 큰 영향을 끼쳤다.

ABSTRACT

In this paper, we experimented to find out how the number of convolution layers, the size, and the number of kernels affect the CNN. In addition, the general CNN was also tested for analysis and compared with the CNN used in the experiment. The neural networks used for the analysis are based on CNN, and each experimental model is experimented with the number of layers, the size, and the number of kernels at a constant value. All experiments were conducted using two layers of fully connected layers as a fixed. All other variables were tested with the same value. As the result of the analysis, when the number of layers is small, the data variance value is small regardless of the size and number of kernels, showing a solid accuracy. As the number of layers increases, the accuracy increases, but from above a certain number, the accuracy decreases, and the variance value also increases, resulting in a large accuracy deviation. The number of kernels had a greater effect on learning speed than other variables.

키워드

Deep Learning, Convolution Neural Network, Kernel, Layer, Accuracy, Learning Time
딥러닝, CNN, 커널, 레이어, 정확도, 학습 시간

* 군산대학교 석사과정(rhdwnsqo12@kunsan.ac.kr)

** 교신저자 : 군산대학교 컴퓨터정보통신공학부

• 접수일 : 2019. 09. 11

• 수정완료일 : 2019. 10. 28

• 게재확정일 : 2019. 12. 15

• Received : Sep. 11, 2019, Revised : Oct. 28, 2019, Accepted : Dec. 15, 2019

• Corresponding Author : Min-Seok Jang

School of Computer & Information & Communication Engineering, College of Engineering, Kunsan National University,

Email : msjang@kunsan.ac.kr

I. 서론

최근 딥러닝은 다양한 분야에서 활용되고 우수한 성능을 보이고 있다. 컴퓨터비전, 자연어처리, 지능형 로봇등과 같은 분야에서 두각을 나타내며 실제로 딥러닝이 적용된 다양한 제품들이 나오고 있는 추세이다. 앞서 언급한 분야들은 CNN(Convolution Neural Network), RNN(Recurrent Neural Network), DQN(Deep Q-Network)을 대표적인 학습 신경망 구조로 갖는다. 최근 딥러닝 연구 동향을 살펴보면 새로운 인공신경망구조제안과 DeepMind의 AlphaGo를 필두로 한 강화학습 분야 연구가 가장 활발히 이루어지고 있으며 최근 활발하게 연구되고 있는 주제로 딥러닝의 이론적인 이해, 해석, 탐구를 위한 기초이론연구 및 최적화가 있다. 또한 기존의 방식들과 딥러닝을 적용한 방식을 비교하며 분석하기도 한다[1].

CNN은 딥러닝의 대표적인 이미지인식 인공 신경망 구조로 CNN이 소개된 후로 최근까지 제안된 새로운 이미지인식 인공 신경망들은 CNN을 기본 틀로 사용한다. AlexNet, VGGNet, ZFNet, GoogLeNet, ResNet, DenseNet, ShuffleNet[2-8] 등등 수많은 신경망들이 파생되었고 해당 신경망들은 모두 CNN에 부가적인 아이디어를 섞은 신경망들이다. 컨볼루션 레이어의 활성화 함수, 커널의 크기 및 개수, 입력영상의 사용횟수, 스킵커넥션이 추가되거나 배치 정규화의 순서변경 등등이 구조마다 다르기 때문에 성능 및 용도 또한 신경망마다 다른 경향이 있다. 이러한 이유로 커널과 같은 한 가지 특성을 위주로 연구해 비교분석하기도 한다[9]. 본 논문에서는 CNN의 구성요소인 컨볼루션 레이어, 커널의 크기 및 개수를 통해 CNN의 기본구조를 파악하고 정확도와 학습시간에 어떠한 영향을 끼치는지 알아보하고자 한다.

따라서 본 논문에서는 선행 연구로써 레이어, 커널, dropout 및 상이한 영상에서도 동일한 구조가 성능이 우수한지 확인했던 성능실험 중[10] 일부를 제외하고 3개의 CNN의 기초구성요소인 레이어와 커널의 크기 및 사이즈에 중점을 두고 실험하였다. 레이어와 커널은 단일크기와 단일개수로 고정시켜 CNN의 기본 구조를 통일된 방법으로 재구성해 실험을 진행하였다. 또한 기존 CNN과 비교를 통해 해당 구성요소가 정확도에 어떠한 영향을 주는지 파악하였다.

II. 실험 환경 구성

2.1 관련 연구

CNN은 컨볼루션 레이어와 풀리커넥티드 레이어로 구성된다. 세부적으로 들어가면 컨볼루션 레이어의 경우 대부분의 논문에서 컨볼루션연산, 배치정규화, 활성화함수(ReLU), 풀링으로 정의한다[11]. 나열한 순서대로 연산이 진행되며 컨볼루션을 통해 영상의 특징을 강조하며 커널의 크기 및 개수에 따라 커널이 추출하는 특징도 조금씩 다르다. 특히 커널 크기의 경우 3x3 커널을 중첩해서 사용하면 5x5, 7x7, 9x9와 비슷한 효과를 볼 수 있으며 오히려 활성화함수를 더 많이 사용해 성능이 더 좋은 경우도 있으며 더 적은 파라미터를 요구하므로 3x3 커널을 중첩해서 사용하는 것이 고착화 되었다[3]. 그러나 3x3커널들로 완전히 동일한 다른 사이즈의 커널 결과값을 도출하려면 필요 이상의 3x3 커널을 요구하게 되는 경우도 있는데 하위계층의 3x3커널에서 여러 개의 특징맵을 수집하고 상위계층의 3x3커널을 활성화해야하기 때문이다. 이 경우 기존 5x5, 7x7 등의 커널에 비해 성능이 떨어질 수 있다. 본 논문에서는 해당 문제를 실험을 통해 확인하도록 한다.

배치정규화를 통해 전 컨볼루션 레이어의 출력값을 정규화시켜 다음 컨볼루션 레이어에 들어갈 값의 변동폭을 줄여 가중치를 완화하고 최종단계에서 완전히 다른 결과값이 나오는 것을 방지할 수 있다. 이는 활성화 함수인 ReLU와 동일하게 Vanishing Gradient 현상을 해결하고 올바른 값이 나오도록 유도하는 역할도 한다. 컨볼루션 연산을 통해 나온 특징맵에서 찾고자하는 강한 자극을 주는 영상을 활성화 함수를 통해 도출하고 출력값을 조절한다.

활성화 함수는 대표적으로 3개의 활성화 함수가 있으며 스텝(Step), 시그모이드(Sigmoid), 렐루(ReLU)가 있다. 이 중 최근 Vanishing gradient를 해결한 ReLU 함수를 일반적으로 가장 많이 사용하고 있다. 본 논문에서도 활성화함수로 ReLU를 사용하였다.

풀링(Pooling)은 일반적으로 특징맵의 크기를 줄이고 영상의 회전 및 이동에 좀 더 견고한 성질을 갖는 특징을 추출할 수 있다. 풀링의 경우 최대(Max), 평균(Mean), 최소(Min) 풀링이 있으나 최소 풀링은 제대로 된 특징을 뽑기도 힘들고 활성화함수에서 음수

는 0이 되기 때문에 최소값으로 영상인식과 같은 문제를 해결할 수 없으므로 사용되지 않는다. 평균 풀링의 경우 풀링영역안의 값에서 평균값을 도출해 사용한다. 평균을 사용하다보니 영상에서 찾아야하는 특징 맵의 자극점을 쉽게 찾지 못할 수 있다. 최대 풀링의 경우 평균과 달리 최대값 하나만 추출해 사용함으로 평균 풀링보다 특징맵에서 추출할 영역을 쉽게 찾을 수 있다. 본 논문에서는 최대 풀링을 사용한다.

풀리커넥티드 레이어에서는 마지막 컨볼루션 레이어를 통해 나온 결과값을 모두 1차원배열로 변환한 후 이전 레이어의 모든 노드들을 다음 레이어의 모든 노드와 연결한 레이어이며 최종적인 분류를 위한 작업을 수행한다. 본 논문에서는 2개의 풀리커넥티드 레이어를 사용한다.

2.2 Data 및 실험 환경

본 논문에서는 ImageNet을 통해 5종류의 차량 관련 영상데이터를 사용하였으며 약 7천장의 영상을 사용하여 학습을 진행하였다. 영상의 첫 크기는 224X224 픽셀로 학습을 진행해 실험CNN의 컨볼루션 레이어를 거치면 레이어의 깊이마다 최소 1X1 픽셀부터 56X56 픽셀까지의 각각 다른 결과물이 도출된다. 실험에 사용한 컴퓨터 성능은 다음과 같다.

OS : windows 10 pro 64 bit
 CPU : i7 - 7700
 RAM : 16G
 GPU : NVIDIA TITAN Xp

2.3 실험 CNN 구조

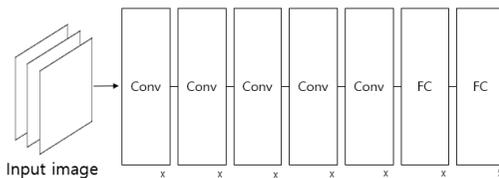


그림 1. 실험 CNN모델 구조
 Fig. 1 CNN structure for the experiments

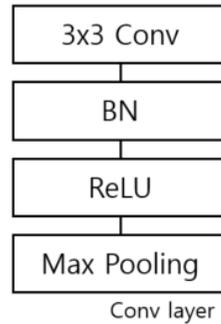


그림 2. 실험 컨볼루션레이어 구조
 Fig. 2 Convolution layer structure for the experiments

본 논문에서는 컨볼루션레이어, 커널 개수 및 크기를 변경하며 실험했으며 그 외의 하이퍼파라미터 조건들은 모두 동일한 값들을 부여하였다. 실험에서 각 컨볼루션 단계의 모든 커널의 개수 및 크기를 같은 개수 및 크기로 통일하여 진행하였다.

실험에 사용한 컨볼루션레이어는 컨볼루션, 배치 정규화, 활성화 함수, 풀링의 순서로 진행하며 최소 3개의 레이어부터 홀수개수로 13레이어까지 진행한다. 컨볼루션레이어는 단계가 많아질수록 영상의 특징을 더더욱 견고하게 검출하지만 장시간의 학습시간이 요구된다. 하지만 일정단계 이상을 넘어가면 영상이 과하게 함축되므로 제대로 된 특징을 잡지 못하고 성능이 떨어질 수 있다. 반대로 컨볼루션레이어의 단계가 적으면 특징 도출에 한계가 있어 정확도가 떨어진다. 따라서 최고 정확도의 컨볼루션레이어의 수를 도출시켜야하며 학습시간도 고려해 시간대비 정확도가 높은 컨볼루션 레이어의 수를 도출한다.

커널은 영상의 특징을 도출하기 위해 사용되며, 이는 학습된 특징을 바탕으로 해당영역에 검출하고자 하는 특징이 존재하는지 확인하는 역할을 수행한다. 크게 커널의 크기와 개수에 따라 상이한 정확도 값을 보인다. 커널의 크기가 클수록 한 번에 학습하는 특징이 넓어지며 넓어진 크기만큼 연산해야할 값이 많아진다. 커널은 짝수로 크기를 설정하게 될 경우 출력 영상이 입력 영상보다 커지게 되며 출력 영상 픽셀과 매핑해주기가 어려우므로 일반적으로 홀수 값의 크기를 준다. 본 논문에서는 3x3부터 11x11까지의 커널을 사용해 테스트하였다. 또한 커널의 개수를 늘릴수록 여러 종류의 특징을 도출 할 수 있다. 커널의 개수가

많아질수록 자연스럽게 정확도도 늘어나지만 학습시간도 늘어나게 된다. 반대로 일정 커널 개수 이후부터 오히려 정확도가 낮아지거나 비슷한 정확도를 유지하지만 학습시간만 오래 걸리는 경우도 있다. 이는 너무 많은 커널 개수로 인해 실질적으로 학습에 참여하지 않는 잉여 커널이 생기게 되는 경우이며 오히려 학습에 방해가 되고 시간만 잡아먹게 된다[4]. 따라서 커널의 개수 또한 최고 정확도의 커널 개수를 도출시키며 학습시간까지 고려한 시간대비 정확도가 높은 커널의 개수를 도출한다. 컨볼루션 연산을 거친 결과 값의 정확도를 높이기 위해서는 적절한 수준의 크기와 개수의 커널이 요구된다. 특히 3x3 커널은 컨볼루션 레이어를 깊게 쌓게 되면 5x5, 7x7 등의 커널과 같은 결과 값을 도출하거나 더 우수한 정확도를 보이는 것으로 알려져 있다[3].

폴리커넥티드레이어의 경우 본 논문에서 2계층을 사용하였다. 첫 폴리커넥티드레이어의 경우 1차원 데이터로 평탄화 한 후 커널의 개수와 동일한 개수로 연결해준다. 이후 두 번째 레이어에서 실험에 사용된 이미지 종류와 동일한 5개의 레이어로 연결해 영상을 분류한다.

III. 실험

표 1은 각 레이어별로 커널의 개수 및 크기에 따른 정확도 및 학습시간 표이다. 해당표의 행은 커널의 개수, 열은 커널의 크기이며 각 레이어의 개수마다 구역별로 작성하였다. 정확도(%)는 상단, 학습시간(초 단위)은 하단에 표기했다.

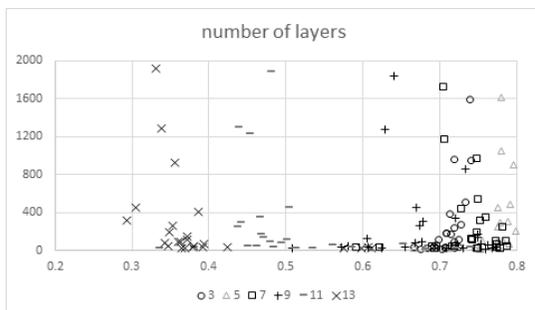


그림 3. 컨볼루션레이어 개수에 따른 학습시간/정확도 그래프

Fig. 3 Learning time/accuracy graph by number of convolution layers

그림 3은 레이어 수에 따른 시간/정확도 그래프이다. x축이 정확도, y축이 시간(초)이며 표기된 각각의 범례들은 동일한 범주에 속해도 표 1에서와 같이 커널의 개수와 크기가 모두 다르며 그에 따른 시간/정확도 좌표이다. 다양한 커널 개수와 크기가 섞여 있어도 레이어의 수가 동일한 경우 특정 정확도에 몰려있는 현상이 발견되었다. 예를 들면 3개의 레이어를 사용한 경우 원모양의 표기가 정확도 0.70.72부근에 다수 포진되어 있으며 5개의 레이어는 0.78~0.8근처에 다수 포진되어 있음을 볼 수 있다. 비록 같은 레이어여도 학습시간에 차이가 있으나 정확도는 큰 차이를 보이지 않는다. 3,5,7 레이어 모두 70%이상의 높은 정확도를 보여주었으나 그 중 5레이어가 가장 높은 정확도를 보여주었다.

이후 레이어를 깊게 쌓아도 오히려 정확도가 조금씩 내려가는 현상을 보여주었는데 이는 7레이어도 마찬가지다. 79% 부근의 정확도를 보이던 5레이어에 비해 7레이어는 77%의 정확도를 보인다. 9레이어에서는 70%이하의 성능을 보이며 11과 13레이어에서는 50%이하의 정확도를 보이면서 레이어가 깊어질수록 성능은 내려간다. 이는 Vanishing Gradient 현상으로 보일 수 있으나 본 논문에서는 해당 현상을 해결하기 위한 방법인 ReLU 활성화함수와 배치정규화를 채택해 사용하였으므로 Vanishing Gradient 현상과는 큰 연관이 없다. 이런 현상이 발생하는 이유는 최대풀링을 거치는 영상에서 발생하는 오류로 보인다.

1레이어를 거칠 때 마다 최대풀링을 통해 영상의 크기가 절반으로 작아지게 되는데 224x224 픽셀의 영상을 초기값으로 사용했으므로 약 7레이어부터 2x2 픽셀로 압축되며 9레이어 이후부터는 1개의 픽셀로 압축된다. 1개의 픽셀로는 더 이상 연산이 불가능하며 남은 레이어에서도 연산을 수행해야하는 9레이어 이상의 레이어부터는 결과값이 낮아질 수밖에 없다.

이러한 이유로 9레이어 이후 11, 13 더 나아가 다음 15, 17등등의 레이어들 모두 정확도 50%를 넘기지 못하며 못할 것이다. 13레이어의 경우 커널의 개수와 크기에 맞춰 일정한 결과가 나오던 다른 레이어와 달리 일정하지 않은 정확도 값이 나온다. 이를 개선하기 위해서는 초기 영상의 크기를 조절하거나 언풀링과 같은 방법으로 영상의 크기를 중간 중간 확대해 주는 방법이 있으나 컴퓨팅 자원의 한계로 비효율적인 시간이 소요되거나 구현이 복잡한 문제가 따른다.

표 1. 레이어 개수, 커널의 크기 및 개수에 따른 정확도 및 학습시간 표
 Table 1. Table of accuracy and learning time by number of layers, kernel size and number

size num	3X3	5X5	7X7	9X9	11X11	size num	3X3	5X5	7X7	9X9	11X11
	13 Layer						11 Layer				
32	59.86	57.46	36.89	36.40	37.95	accuracy	68.06	53.36	51.31	33.57	38.09
	20.16	21.73	25.89	27.20	32.49		seconds	16.86	19.03	23.12	26.60
64	61.13	39.15	42.40	38.02	34.70		72.01	58.52	48.41	45.02	46.15
	25.17	32.09	38.68	42.45	48.89		22.28	29.67	37.71	40.38	48.63
128	60.64	39.36	36.25	35.97	37.17		73.92	56.04	49.40	47.92	50.11
	40.24	62.90	88.34	95.41	116.90		35.88	58.84	84.19	94.53	115.49
256	34.28	37.12	35.34	29.33	30.53		65.16	47.00	43.75	44.17	50.39
	77.59	144.64	261.57	313.00	452.20		69.52	138.11	250.16	299.85	448.03
512	34.84*	38.66	35.62	33.85	33.14		46.71	46.64	43.89	45.23	47.99
	195.73	402.55	924.19	1286.67	1917.05		174.08	350.84	1294.27	1229.26	1881.38
	9 Layer						7 Layer				
32	75.97	70.88	62.40	50.74	57.24	accuracy	74.84	74.70	69.19	59.08	62.19
	17.15	18.40	19.63	26.12	31.78		seconds	19.49	23.60	22.76	29.07
64	76.82	72.93	65.51	60.71	58.09		77.74	77.39	71.87	71.59	68.90
	22.45	28.58	35.88	39.59	49.36		24.41	36.10	42.50	43.90	48.45
128	77.24	76.18	66.71	67.63	60.57		78.73	77.31	77.17	74.13	73.99
	35.02	57.09	84.31	94.22	128.40		43.07	68.42	104.13	115.26	129.04
256	75.27	74.91	67.35	67.84	66.86		78.52	74.77	75.19	75.97	72.65
	68.50	132.68	259.39	310.29	457.68		102.62	192.74	316.76	347.11	444.41
512	74.84	72.01	73.22	62.90	64.03		78.02	74.91	74.77	70.60	70.32
	171.38	334.82	858.13	1273.33	1838.58		254.61	547.31	970.46	1175.15	1725.42
	5 Layer						3 Layer				
32	75.19	75.12	71.24	68.13	66.64	accuracy	67.42	70.25	69.47	68.34	66.64
	17.57	21.41	18.91	32.14	27.33		seconds	15.98	16.62	20.12	28.02
64	77.60	77.03	76.96	74.42	74.77		71.80	70.67	72.37	70.74	69.47
	24.51	36.75	33.40	33.90	39.62		22.75	28.19	39.10	36.86	59.37
128	77.46	79.01	77.39	77.46	75.48		69.26	71.94	72.44	71.80	71.38
	43.26	68.51	84.51	91.02	99.68		50.21	60.00	113.46	90.35	168.93
256	78.23	79.86	78.80	77.81	77.46		69.75	70.81	71.87	72.65	73.22
	91.62	202.92	304.47	293.39	451.19		117.53	176.73	241.50	270.18	504.08
512	77.53	79.15	79.58	77.88	77.95		70.81	71.24	71.87	74.06	73.92
	248.14	482.38	906.24	1045.85	1620.57		176.76	384.39	963.86	947.18	1598.48

비교대상인 레이어들은 모두 학습시간적인 측면에서 큰 영향을 보이지 않는다. 대부분의 범주들이 y 축을 따라 포진되어 있으므로 정확도에만 영향을 줄 수 있다. 본 논문에서는 일반적으로 사용하는 CNN의 경우 5개의 레이어를 사용하는 것이 가장 정확도가 높게 도출되었으므로 5개의 레이어를 사용하는 것을 추천한다.

커널은 정확도와 학습시간에 레이어보다 큰 영향을

주었다. 특히 커널의 개수가 두 가지 모두 영향을 주었는데 커널의 개수가 적을수록 적은 학습시간을 요구했다. 그림 4에서 256개와 512개의 레이어는 대부분 600초이상을 소요해 비효율적이므로 그래프에서 생략했다. 정확도의 경우 그림 4에서와 같이 128개의 커널에서 평균적으로 가장 높은 정확도를 보였다. 256개의 커널도 근접한 정확도를 보였으나 2배 이상의 학습시간을 요구해 효율적이지 않다. 64개의 커널

이 뒤를 이어 128개와 근접한 정확도를 보여주었으나 표 1에서 9레이어의 5x5커널 64개와 128개 차이처럼 큰 경우 4%정도의 정확도차이를 보여 불안정한 정확도를 보인다. 하지만 학습시간은 128개에 비해 2배 이상 빠르므로 가장 효율적인 학습은 64개의 커널이다. 다만 둘 다 100초미만의 적은 학습시간을 요구하므로 정확도에 중점을 둘 경우 128개의 커널도 좋은 선택이다.

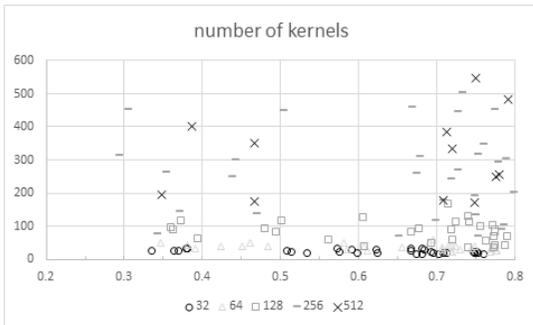


그림 4. 커널 개수에 따른 학습시간/정확도 그래프
Fig. 4 Learning time/accuracy graph by number of kernels

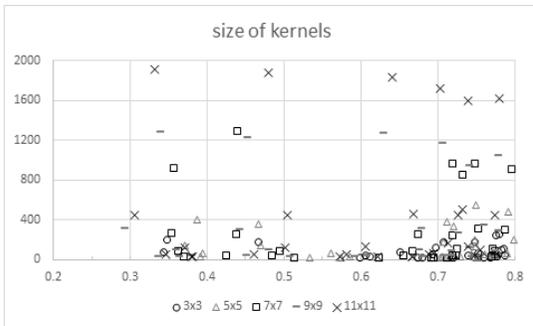


그림 5. 커널 크기에 따른 학습시간/정확도 그래프
Fig. 5 Learning time/accuracy graph by kernel size

커널의 크기는 모든 종류에서 정확도적인 측면에서는 큰 영향을 끼치지 못하나 커널의 크기가 클수록 학습시간이 전체적으로 상승하는 그래프를 보인다. 그림 5에서 그래프의 모든 x축(정확도)에서 종류별로 범주를 찾아볼 수 있으며 이는 정확도에 큰 영향을 주고 있지 않음을 뜻한다. 반면 3x3커널의 경우 400 초 미만의 학습시간에만 몰려있으나 11x11커널의 경우 최대 2000초까지 학습시간을 소요한다. 물론 400

초 미만에서도 11x11커널을 찾을 수 있으나 이는 커널의 크기와 레이어가 작은 경우에 해당되며 동일한 조건의 다른 크기의 커널보다 최대2배의 더 많은 시간을 소모한다. 커널의 경우 가장 작은 크기인 3x3 커널이 가장 적은 학습시간을 요구하므로 최적의 값으로 도출된다.

3x3커널은 2중첩으로 5x5커널과 동일하거나 그 이상의 효과를 볼 수 있다. 표 1의 9레이어 3x3커널과 5레이어 5x5커널의 학습시간과 정확도를 비교해보면 정확도는 비슷하거나 3x3커널이 조금 떨어진다. 하지만 학습속도면으로는 3x3커널이 우세한데 이는 더 적은 파라미터를 사용해 18C(C는 커널 개수)를 요구하나 5x5커널은 25C를 요구하기 때문이다[3].

표 2. 레이어 개수에 따른 일반적인 CNN의 정확도 및 학습시간 표

Table 2. Accuracy and learning time table of general CNN by layer number

layer	3	5	7	9	11	13
accuracy	71.94	80.14	76.82	71.02	46.15	32.65
seconds	26.29	34.82	33.85	34.88	37.65	35.30

표 2는 3x3커널을 사용한 일반적인 CNN의 학습시간/정확도 표이다. 비교를 위해 동일한 레이어들로 실험을 진행하였으며 32개 커널을 사용한 첫 레이어를 시작으로 다음 레이어마다 2배수 증가한 커널을 사용하며 512개 이후로는 레이어가 증가해도 동일하게 512개의 커널을 사용하였다. 본 논문에서 실험했던 CNN구조들과 비교해보기 위해 그림 6과 그림 7을 추가했다. 사용된 비교대상 구조는 3레이어부터 13레이어까지 비교하였으며 동일하게 3x3커널을 사용하고 64개와 128개의 커널 개수를 가진 CNN을 일반적인 CNN과 비교했다. 그림 6은 정확도만, 그림 7은 학습시간만 비교한 그래프이다.

해당 그림과 표에서 알 수 있듯 레이어의 개수를 다양하게 사용하는 방식이 통일된 하나의 레이어(64, 128개)를 사용하는 것 보다 성능이 떨어지거나 시간이 더 걸릴 수 있다는 것이다. 하지만 5개의 레이어에서는 일반적인 CNN이 3% 높은 정확도를 보였으나 학습시간이 통일된 레이어 대비 40% 더 많은 시간을 요구했다.

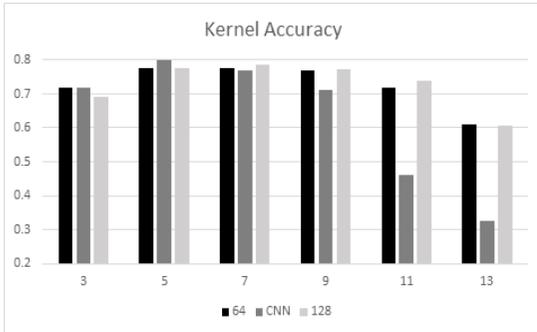


그림 6. 일반CNN과 64,128개 커널 CNN 정확도 비교
Fig. 6 Comparison of accuracy between general CNN and 64,128 kernel CNN

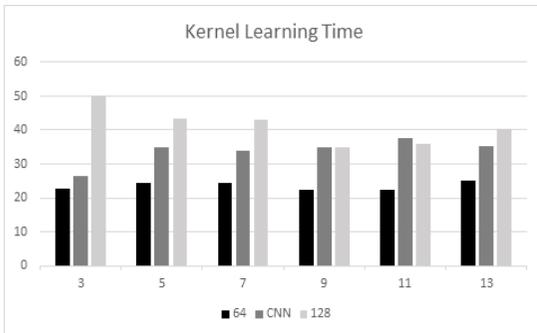


그림 7. 일반CNN과 64,128개 커널 CNN 학습시간 비교
Fig. 7 Comparison of learning time between general CNN and 64,128 kernel CNN

IV. 결론

본 논문은 CNN의 특정 변수가 정확도와 학습시간에 미치는 영향을 알아보기 위해 진행하였다. 일반적으로 사용되는 CNN도 비교대상으로 넣어 실험을 진행하였다. 컨볼루션 레이어는 정확도에 큰 영향을 끼치며 무조건 깊을수록 좋은 정확도가 도출되진 않았다. 따라서 적당한 임계값을 찾아야 하는데 5, 7개의 컨볼루션 레이어가 다른 3, 9, 11, 13 레이어에 비해 우수한 정확도를 보였다.

커널의 경우 3x3 커널을 사용할 때 다른 커널크기보다 정확도/학습시간이 우수했으며 128개의 커널만 사용할 때 가장 높은 정확도를 보였다. 64개의 커널은 128개보다 약간 낮은 정확도를 보였으나 학습에 걸리는 시간은 2배 이상 적게 걸려 시간대비 가장 좋은

정확도를 보여주었다. 이는 일반적인 CNN과도 비교했을 때 상당히 준수한 성능을 보여주었는데 일반적인 CNN을 사용하기보다 64개의 통일된 커널 개수를 사용한 CNN이 전반적으로 좋은 정확도와 학습에 필요한 시간이 적게 걸린다. 128개의 커널 또한 비슷한 성능을 보인다. 따라서 CNN의 커널 개수를 정할 때 2의 배수로 늘려가며 사용하는 것 보다 64개나 128개의 커널을 고정시켜 레이어 수를 늘리는 것이 가장 최적의 성능을 찾을 수 있는 방법이다.

예외적으로 일반적인 CNN을 사용하면 레이어가 깊어질수록 실험에 사용했던 통일된 512개의 커널 개수를 가진 CNN과 비슷한 정확도를 보이는데 13레이어, 3x3, 512개의 커널과 비교시에 각각 32%*와 34%*로 상당히 낮은 정확도를 보여준다. 하지만 학습 시간은 일반적인 CNN이 실험에 사용된 CNN보다 약 5배 이상 빠르며 만약 초기 영상 값을 고화질 영상으로 사용하면 더 깊은 레이어에서는 일반적인 CNN이 더 높은 정확도를 빠르게 도출할 것으로 예상된다.

감사의 글

이 논문은 2019년 정부(산업통상자원부)의 재원으로 한국산업기술진흥원의 지원과 2018년 한국국토정보공사 공간정보연구원 산학협력 R&D 지원 사업(자유과제)의 지원을 받아 수행된 연구임 (N0002428, 2019년 산업전문인력역량강화사업)

References

- [1] S. Chung and Y. Chung "Comparison of Audio Event Detection Performance using DNN," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 13, no. 3, June 2018, pp. 571-578.
- [2] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, South Lake Tahoe, USA, 2012.
- [3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *The International Conference on Learning*

g Representations, San Diego, USA, 2015.

- [4] M. Zeiler and R. Fergus. "Visualizing and understanding convolutional neural networks," *European Conference on Computer Vision*, vol. 13, no. 1, Sept. 2014, pp. 818-833.
- [5] C. Szegedy, W. Lium, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Computer Vision and Pattern Recognition*, Boston, USA, 2015.
- [6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Computer Vision and Pattern Recognition*, Las Vegas, USA, 2016.
- [7] G. Huang, Z. Liu, K. Weinberger, and L. Maaten, "Densely connected convolutional networks," *Computer Vision and Pattern Recognition*, Honolulu, USA, 2017.
- [8] X. Zhang, X. Zhou, M. Lin, and J. Sun. "Shufflenet: An extremely efficient convolutional neural network for mobile devices," *Computer Vision and Pattern Recognition*, Salt Lake City, USA, 2018.
- [9] G. Son and J. Park "A Study of Kernel Characteristics of CNN Deep Learning for Effective Fire Detection Based on Video," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 13, no. 6, Dec. 2018, pp. 1257-1262.
- [10] J. Kong, H. Jang, M. Jang, and Y. Lee, "Analysis of CNN Performance According to various parameters," *Information and Control Symp.*, Kunsan, Korea, 2019.
- [11] S. Chung and Y. Chung "Sound Event Detection based on Deep Neural Networks," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 14, no. 2, Apr. 2019, pp. 571-578.

저자 소개

공준배(Jun-Bea Kong)



2017년 군산대학교 컴퓨터정보공학과 졸업(공학사)
2018년 군산대학교 대학원 컴퓨터정보공학과 재학(공학석사과정)

※ 관심분야 : 딥러닝

장민석(Min-Seok Jang)



1989년 연세대학교 전자공학과 졸업(공학사)
1991년 연세대학교 대학원 전자공학과 졸업(공학석사)

1997년 연세대학교 대학원 전자공학과 졸업(공학박사)
1997년~현재 군산대학교 컴퓨터정보통신공학부 교수
※ 관심분야 : IoT 시스템, 드론 응용, 딥러닝