



자동차 주행환경에서 보행자 분류를 위한 딥러닝 모델의 전이학습 및 성능비교

변영현*, 객근창**

A Transfer Learning and Performance Comparison of Deep Learning Models for Pedestrian Classification under Automobile Driving Environment

Yeong-Hyeon Byeon*, Keun-Chang Kwak**

이 논문은 2015년도 정부(미래창조과학부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (NRF-2015R1D1A1A01060701)

요 약

본 논문은 자동차 주행환경에서 딥러닝 모델들에 따른 보행자 분류의 성능비교를 한다. 요즘 자동차의 경우 블랙박스는 대부분 설치하고, 운전자 보조 시스템도 카메라를 이용한 영상처리기반 기술들이 적용되고 있다. 보행자 분류는 영상에서 사람후보지역에 대해 사람인지 아닌지 최종적인 판단하는 중요역할을 한다. 본 논문에서는 딥러닝 모델로써 잘 알려진 AlexNet, GoogLeNet, ResNet 모델을 이용하여 전이학습(transfer learning)을 수행한다. 딥러닝 모델들의 성능 비교를 위해 INRIA 데이터베이스와 자동차 운전환경에 구축된 CU 데이터베이스를 사용하였다. INRIA 학습용 데이터는 전이학습을 위해 사용하고, 성능검증은 INRIA 검증용 데이터와 CU 데이터를 사용하였다. 실험결과 전이학습에 근거한 ResNet의 성능이 AlexNet과 GoogLeNet보다 더 높은 분류율을 보였다.

Abstract

In this paper, a performance comparison of deep-learning models for pedestrian classification under automobile driving environment is performed. Most automobiles nowadays are equipped with black boxes, and driver assistance systems are also applied to camera based image processing technologies. Pedestrian classification plays an important role in determining the final decision whether a candidate region is a person or not. We perform the transfer learning based on AlexNet, GoogLeNet, and ResNet that are well known as deep-learning models. For comparison experiments of the deep learning models, we used INRIA database and Chosun University (CU) database constructed under automobile driving environment. The INRIA training data set is used for transfer learning and performance validation is used with INRIA testing data set and CU database. The experimental results showed that the performance of ResNet based on transfer learning outperformed AlexNet and GoogLeNet.

Keywords

pedestrian classification, deep-learning, automobile driving environment, transfer learning, INRIA database

* 조선대학교 제어계측공학과

- ORCID: <https://orcid.org/0000-0002-7480-4971>

** 조선대학교 전자공학부 객근창교수(교신저자)

- ORCID: <https://orcid.org/0000-0002-3821-0711>

· Received: Sep. 27, 2018, Revised: Oct. 12, 2018, Accepted: Oct. 15 2018

· Corresponding Author: Keun-Chang Kwak

School of Electronics Engineering, Chosun University, 309, Pilmun-daero, Dong-gu, Gwangju, 61452 Dept. of , Korea

Tel.: +82-62-230-6086, Email: kwak@chosun.ac.kr

1. 서 론

우리 주변에는 가정의 IP카메라, 방범지역의 CCTV, 교통감시 카메라 등 수많은 카메라들이 설치되어 있고 더 싸고 더 좋은 카메라들이 개발되면서 그 양은 계속 증가하고 있다. 또한 사고를 기록하여 책임과실을 따지기 위해, 편리한 운전을 위해 자동차에도 카메라 장착이 보편화 되었다. 보통 카메라는 항상 촬영하기 때문에 혹시 모를 일들을 녹화할 수 있다. 그렇지만 사람이 카메라에 녹화된 내용을 항상 볼 수는 없기 때문에 또는 눈에 안 띄어 보지 못할 수 있기 때문에 특정 목적에 따라 영상에서 조건이 만족하면 사람에게 알려준다. 이러한 기능을 위해서 가장 기본이 되는 기술이 검출 및 인식, 추적이기 때문에 관련 학문들이 계속해서 연구되고 있다. 그중에서 사람분류는 영상에서 사람으로 예상되는 후보지역에 대해 최종적으로 사람인지 판단할 수 있다. 또한 사람을 대상으로 이벤트를 주는 것은 범죄, 안전, 보안[1], 인구통계[2] 등에 관련성이 크기 때문에 중요하다. 보안 또는 안전 등의 이유로 사람이 있어서는 안 될 장소에서 사람이 검출된다면 그것은 안전이든 보안이든 문제가 있음을 의미한다. 마찬가지로 자동차를 운전할 때 진행방향 쪽에 사람이 있는데 운전자가 확인을 못할 수도 있기 때문에 이를 영상처리를 이용해서 운전자에게 알림을 줌으로서 보행자를 사고로부터 예방할 수 있다[3][4].

몇 년 전만해도 영상처리는 사람에 의해 수동적으로 특징들이 추출되었고 형태적인 조건들을 쫓아서 영상에서 원하는 기능들을 구현하였다. 사람에 의해 제안된 경계선 특징추출 방법만 해도 Prewitt[5], Sobel[6], Robert[7], Canny[5] 등이 있다. 또한 통계적인 방법으로는 주성분 분석, 선형 판별 분석 등이 있고[8], 그 외에도 변형된 알고리즘들이 다수 존재한다[9]-[12]. 이런 방법들을 이용하여 실험자가 시스템을 만들더라도 실험환경, 데이터 특성에 따라 성능에 상이한 차이를 보여 제한적으로 활용될 수밖에 없었다. 그러다가 머신러닝이 등장하여 입력된 데이터를 바탕으로 분류기가 자동으로 그 데이터를 잘 분류할 수 있도록 학습되는 기법[13][14]들이 나타났다. 전보다 사람이 분석하기 어려웠던 부분들을

복잡한 계산을 통해 수치를 근거로 더 나은 성능을 얻을 수 있었다.

최근에는 딥러닝이 개발되어 기존의 많은 한계점들을 돌파하였고 여전히 개선된 방법들이 제시되고 있다. 딥러닝[15]은 기존 머신러닝의 다층퍼셉트론[14]을 시작으로 다양한 타입의 레이어들이 추가되고 그 레이어들이 쌓이고 쌓여 깊어진 방법으로 작은 층 수로는 해결되지 못한 문제들이 깊은 층으로 해결되고 있는 추세이다. 대표적으로 심층 신경망(Deep Neural Network)[16], 합성곱(Convolutional) 신경망[17], 순환(Recurrent) 신경망[18], 심층 신뢰 신경망(Deep Belief Network)[19], 심층 Q-네트워크(Deep Q-networks)[20] 등이 있다. 영상처리에서 가장 두드러지는 방법 중 하나가 합성곱 신경망으로 다양한 특징추출 필터들을 자동으로 학습하고 자동으로 분류하도록 설계된 신경망이다. 기존에 특징추출이 사람에 의해 설계되고 분류만 기계학습으로 했었다면 합성곱 신경망은 특징추출과 분류가 하나의 패키지로 되어 분류기만 아니라 좋은 특징도 학습한다[21]. 합성곱 신경망을 이용해 ILSVRC (ImageNet Large Scale Visual Recognition Challenge)에서 좋은 성능을 보인 딥러닝 모델들이 있는데 이미 좋은 성능을 검증받아서 많은 사람들이 그 모델을 연구에 활용하고 있다. 대개는 학습데이터가 적거나 하드웨어 성능이 부족할 경우 전이학습을 이용해 학습시간도 줄이고 충분히 잘 분류할 수 있게 되었다[22]. 대표적으로 AlexNet[23], GoogLeNet[24], ResNet[25] 등이 있다. AlexNet은 7개 레이어로 구성되어 있으며, 기존 LeNet[26]과 비슷한 구조로 2개의 GPU를 이용하여 병렬처리하였다[23]. GoogLeNet은 22개 레이어로 구성되어 있으며, 인셉션(Inception) 모듈을 고안하여 9개의 모듈을 적용하였고[24], ResNet은 152 레이어로 구성되어 있으며, 깊은 층의 망을 학습하는데 발생하는 문제를 나머지 학습(Residual Learning)을 통해 시간도 단축시키고, 성공적으로 학습을 가능하게 한다[25].

본 논문은 자동차 주행환경에서 딥러닝 모델에 따른 보행자인식의 성능비교를 한다. 2장은 딥러닝 모델에 대해 기술하고, 3장은 데이터베이스를 보여준다. 4장은 실험 및 결과를 나타내며, 5장에서 결론을 맺는다.

II. 딥러닝 모델

2.1 AlexNet

AlexNet은 2개의 GPU에 병렬적으로 적용한 딥러닝 구조로서 5개의 합성곱 레이어, 3개의 풀링 레이어, 2개의 전연결 레이어 그리고 출력 레이어를 가진다. 입력 이미지는 반드시 227×227 크기로 변환되어야 한다. 그림 1은 AlexNet의 구조를 보여주고 있다. $227 \times 227 \times 3$ 입력 이미지는 96개의 $11 \times 11 \times 3$ 필터들과 4칸 건너뛰기(Stride), 0칸 패딩(Padding)의 합성곱을 계산하는데 병렬계산되므로 $55 \times 55 \times 48$ 특징 큐브가 2개 생성된다.

그 다음 활성화함수인 ReLu (Rectified Linear Unit)를 계산하고, 이어서 정규화인 LRN(Local Response Normalization)을 수행한다. 큐브마다 $3 \times 3 \times 1$ 필터로 2칸 건너뛰기 최대값 풀링을 적용해 $27 \times 27 \times 48$ 로 차원이 축소된다. 다시 256개의 $5 \times 5 \times 48$ 필터로 1칸 건너뛰기, 2칸 패딩의 합성곱을 실행해서 $27 \times 27 \times$

128이 되고, ReLu와 LRN이 적용된다. $3 \times 3 \times 1$ 필터로 2칸 건너뛰기와 0칸 패딩의 최대값 풀링을 수행해서 $13 \times 13 \times 128$ 이 되고, 특징큐브를 합친 $13 \times 13 \times 256$ 상태에서 384개의 $3 \times 3 \times 256$ 필터를 192개씩 1칸 건너뛰기, 1칸 패딩의 합성곱을 적용하여 $13 \times 13 \times 192$ 가 되고 ReLu를 적용한다.

그 다음 $3 \times 3 \times 192$ 필터를 192개씩 1칸 건너뛰기, 1칸 패딩의 합성곱을 적용하여 $13 \times 13 \times 192$ 가 되고 ReLu를 적용한다. $3 \times 3 \times 192$ 필터를 128개씩 1칸 건너뛰기, 1칸 패딩의 합성곱을 통해 $13 \times 13 \times 128$ 이 된다.

그 다음 $3 \times 3 \times 1$ 필터로 2칸 건너뛰기와 0칸 패딩의 최대값 풀링을 통해 $6 \times 6 \times 128$ 이 되고 1차원으로 정렬되어 4608길이의 벡터가 된다. 그 후 전연결 레이어에서 ReLu와 드롭아웃(Dropout)을 통해 2048개 노드로 매핑된 후 다시 전연결 레이어에서 ReLu와 드롭아웃을 통해 2048개 노드로 매핑하고 끝에서 1000개의 노드로 매핑시킨 후 소프트맥스(Softmax)를 이용해 분류한다[23][27].

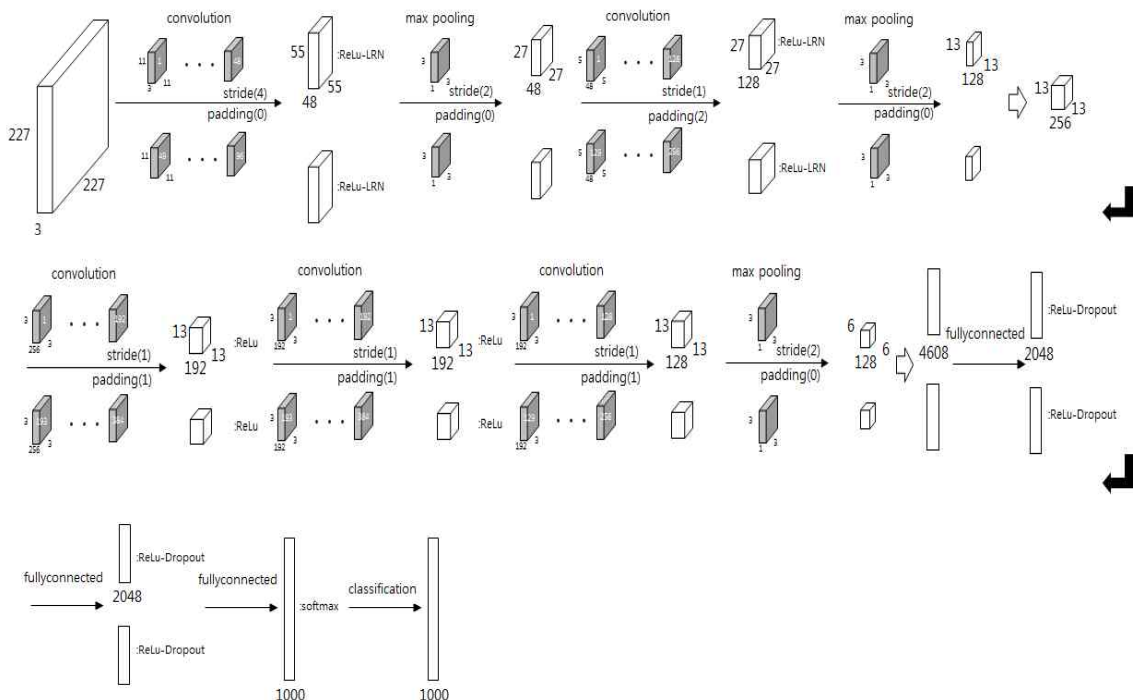


그림 1. AlexNet의 구조
Fig. 1. Structure of AlexNet

2.2 GoogLeNet

GoogLeNet은 9개의 인셉션 레이어로 구성되어 있다. 인셉션 레이어의 아이디어는 이미지에서 더 큰 영역을 커버하면서 작은 정보를 위해 해상도를 좋은 상태로 유지하는 것이다. 따라서 병렬로 가장 정확한 1x1 부터 더 큰 5x5 까지 다양한 크기들로 합성곱을 동시에 하는 것이다. 그림 2는 인셉션 모듈의 구조를 보여준다. 인셉션에서 1x1 합성곱은 특징맵의 수를 줄이기 때문에 연산량도 줄일 수 있다. 그래서 인셉션은 1x1 합성곱을 수행한 다음에 다른 크기의 합성곱을 수행한다. 또한 이전 레이어의 내용을 요약하기 위해 최대값 풀링이 추가되었고, 모든 결과는 결합되어 다음 레이어에 전달된다. 또한 보조 분류기(Auxiliary Classifier)를 2개 구성하여 학습동안 경사도 소멸(Vanishing Gradient) 문제를 회피하였다[24][27][28].

2.3 ResNet

신경망의 레이어가 깊어질수록 경사도 소멸, 경사도 폭발(Exploding Gradient), 저하(Degradation) 문제가 발생한다. 경사도 소멸은 역전파시 경사도가 너무 작아져서, 경사도 폭발은 경사도가 너무 커져서 학습이 제대로 이뤄지지 않는 경우를 의미한다. 저하는 과적합이 발생하지 않았음에도 깊은 신경망이 얇은 신경망보다 성능이 더 좋지 못한 것을 의미한다. ResNet은 이런 문제를 해결하기 위해 상위

레이어의 입력 특징을 재사용하여서 학습효율이 개선하였다. 그림 3은 나머지 학습의 원리를 보여준다. X가 입력되어 출력 Y가 구해지고, 그 Y에 입력 X가 다시 더해진다. 그리고 $\text{ReLu}(X*Y)$ 가 0이 되도록 학습을 하게 되면, Y는 X와 거의 같은 값을 출력하게 된다. 이는 중간에 위치하는 가중치들이 작아도 문제가 없게 되며, 입력의 작은 변화도 출력에 반영된다. 중간 가중치 레이어의 수는 마음대로 설정할 수 있으며, ResNet은 이 방법으로 구조를 깊게 쌓아서 만든 모델이다. 기본 틀은 VGGNet[29]을 참고하였으며, 합성곱 레이어는 3x3필터를 사용하고, 풀링(Pooling), 드롭아웃 등을 사용하지 않는다. 특징맵의 크기를 줄일 때는 풀링 대신에 합성곱에서 2칸 건너뛰기를 사용한다. 그리고 2개의 합성곱 레이어마다 상위 입력을 재사용하였다[25][27][30].

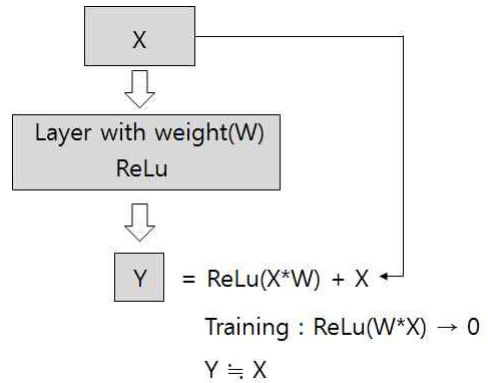


그림 3. 나머지 학습의 원리
Fig. 3. Principle of residual learning

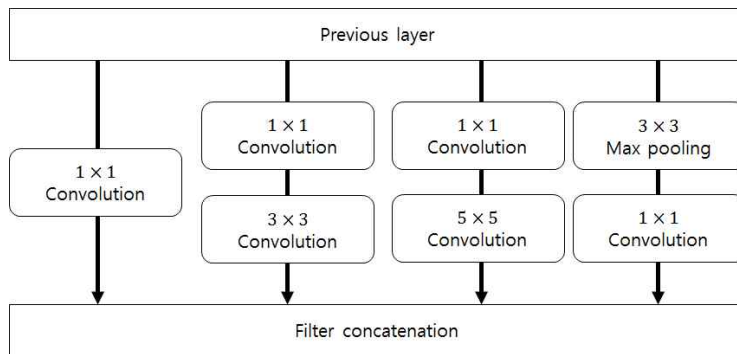


그림 2. 인셉션모듈의 구조
Fig. 2. Structure of inception module

III. 보행자 데이터베이스

벤치마킹 데이터베이스인 INRIA는 학습데이터와 검증데이터가 구분되어 있으며, 각각 사람이미지와 배경이미지로 구성되어 있다. 사람이미지에서 한 이미지에는 사람 한명이상 포함되어 있으며, 이미지에서 사람의 정보는 별도의 주석파일에 주어진다. 학습용 사람이미지는 614개 있으며, 주석에 포함된 사람의 수는 1237명이다. 학습용 배경이미지는 1218개 있다. 검증용 사람이미지는 288개 있으며, 주석파일에 포함된 사람의 수는 589명이다. 검증용 배경이미지는 453개 있다. 그림 4는 INRIA 데이터의 예시를 보여준다.

유튜브에서 수집한 데이터와 직접 카메라 장비로 획득한 데이터를 사용한다. 유튜브에서 수집한 동영상 수는 10개이며 그 비디오들은 119초, 119초, 40초, 119초, 119초, 60초, 59초, 3219초, 455초 및 2157초짜리로 구성되어 있다. 이 비디오들의 내용으로는 자동차가 주택이 밀집지역, 상가 밀집지역, 주유소에서 다양한 시간대, 다양한 날씨 등이 포함되어 있다. 도로에는 차가 천천히 서행하고 있으며, 길가에 자동차들이 주차된 곳도 여러 군데 있다. 또한 보행자들은 보도와 횡단보도에서 대기하거나 걸어가고 있다. 유튜브 데이터는 광각 뷰가 적용되어 모서리부분에 약간의 변형이 있으며, 비디오 해상도는 1280×720이고 비디오 프레임 속도는 30 프레임/초이다. 비디오는 비슷한 내용이 중복되기 때문에 특정 시간마다 한 이미지를 샘플링하여 사용한다.

동영상 별로 추출된 프레임의 수는 119, 119, 40, 119, 119, 60, 59, 161, 152 및 144개이다. 긴 영상은 긴 주기로 추출되고 짧은 영상은 작은 주기로 추출된다. 직접 카메라 장비로 수집한 데이터는 삼성 갤럭시 노트5와 블랙뷰 DR400G로 녹화하였다. 이 장비들은 자동차 실내 앞에 부착하였다. 촬영한 내용은 캠퍼스에서 자동차 주행 중에 두 개의 세션으로 녹화하였으며, 각 영상의 길이는 181초와 79초이다. 캠퍼스의 보도에는 나무들이 있어 그림자로 인한 어두운 이미지와 강한 빛에 의해 매우 밝은 이미지들이 포함되어 있다. 추출된 프레임 수는 59와 27개이며 해상도는 1280×720이다.



(a) 원본 이미지
(a) Raw image



(b) 추출된 사람 이미지
(b) Extracted person images

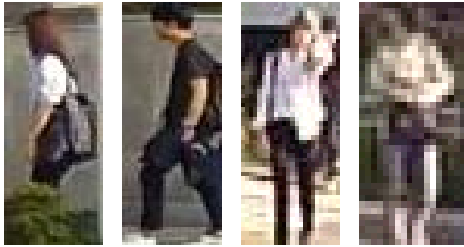


(c) 제공된 배경 이미지
(c) Provided background images

그림 4. INRIA 데이터 예시
Fig. 4. Example of INRIA data



(a) 원본 이미지
(a) Raw image



(b) 추출된 사람 이미지
(b) Extracted person images



(c) 추출된 배경 이미지
(c) Extracted background images
그림 5. 직접 구축한 CU 데이터 예시
Fig. 5. Example of directly built CU data

직접 카메라 장비로 수집한 데이터는 사진에서 직접 사람부분과 배경부분에 라벨링을 하여 잘라내었다. 그리고 이미지상에서 사람이 사람이라고 판단되면 확대를 해서라도 라벨을 설정하였기 때문에 아주 작은 사람부터 큰 사람까지 다양한 해상도의 사람이미지를 라벨링하였고, 또한 자동차나 건물 등에 가려서 신체일부분 나온 사람도 상반신이 어느 정도 노출된 경우 라벨링을 하였다. 배경은 사진에서 사람이 없는 부분들을 무작위로 선택하여 라벨링하였다. 마찬가지로 아주 작은 배경부터 큰 배경까지 다양한 해상도로 라벨링 하였으며, 높이와 너비의 비율도 다양하게 라벨링하였다. 직접 구축한 데이터베이스 크기는 사람이미지는 4787개이며, 배경이미지는 4426개이다[31].

IV. 실험 결과 및 분석

실험을 위한 환경은 CPU-Intel (R) Xeon (R) E5-160 v3 @ 3.50GHz, RAM-32GB, GPU-GTX Titan X 및 OS-windows7 64bit 및 MATLAB R2018a이다. 비교실험에 사용한 딥러닝 모델은 AlexNet, GoogLeNet, ResNet이고, 모델의 학습은 INRIA 데이터베이스에서 학습용 사람이미지 1237장과 학습용 배경이미지 1218장이다. 학습데이터 중에서 70%는 학습(Training)에 사용하고, 30%는 확인(Validation)에 사용한다. 검증(test)은 INRIA의 검증용 사람이미지 589장과 검증용 배경이미지 453장을 사용하였고, 추가로 직접 구축한 데이터베이스인 사람이미지 4787장과 배경이미지 4426장도 검증에 활용하여 성능을 비교하였다. 성능은 인식의 정확도로 식 (1)과 같이 정의한다.

$$\text{정확도} = \frac{\text{참인식수}}{\text{참인식수} + \text{오인식수}} \times 100\% \quad (1)$$

표 1은 AlexNet을 이용한 보행자인식의 정확도를 보여준다. 표 2는 GoogLeNet을 이용한 보행자인식의 정확도를 보여준다. 표 3은 ResNet을 이용한 보행자인식의 정확도를 보여준다. 학습방법으로 확률적 미분 강하법(SGDM; Stochastic Gradient Descent Method), RMSProp, 적응 모멘트 예측(Adam; Adaptive Moment Estimation)을 사용하였다. 기본적인 학습방법은 확률적 미분 강하법으로 무작위 샘플 그룹으로부터 미분방향과 반대로 이동하는 방법이다. 이로 부터 파생된 학습방법이 RMSProp, Adam 등이 있다. 모티브는 학습 초기에는 적당히 큰 학습률로 파라미터 업데이트를 하고 해에 다가갈수록 학습률을 작게 하는 기법들이다. Adam은 모멘텀(Momentum)과 RMSProp의 장점을 결합한 방법이다. 일반적으로 미니배치 미분강하 학습을 하면 손실함수는 줄었다가 커졌다가 반복하면서 전체적으로 줄어드는데, 모멘텀은 과거의 미분을 통해 현재의 미분이 너무 튀는 것을 방지한다[32]. 초기 학습률은 모두 0.0001로 진행을 하였고, 미니배치의 크기를 변경하면서 인식률을 확인하였다.

표 1. AlexNet을 이용한 보행자인식의 정확도

Table 1. Accuracy of pedestrian recognition using AlexNet

Training method	Mini-batch size	Multiplier of learning rate	Epoch	Transferred units (END:25)	Validation accuracy (INRIA)	Test accuracy (INRIA)	Test accuracy (BUILT DB)	Total test accuracy
sgdm	10	20	6	X	99.18	99.33	86.14	92.74
rmsprop	10	20	6	1-22	99.05	99.42	87.28	93.35
adam	10	20	6	X	96.88	96.93	91.09	94.01
adam	10	20	6	1-22	98.64	99.62	88.52	94.07
adam	20	20	6	1-22	98.78	99.71	84.95	92.33
adam	30	20	6	1-22	99.51	99.52	85.67	92.60
adam	40	20	6	1-22	98.37	99.90	86.04	92.97
adam	50	20	6	1-22	98.37	98.94	77.62	88.28
adam	10	10	6	1-22	98.51	99.62	85.23	92.43
adam	10	30	6	1-22	97.42	98.94	81.34	90.14
adam	20	30	6	1-22	98.78	99.42	90.16	94.79
adam	20	30	10	1-22	99.05	99.52	78.73	89.13

표 2. GoogLeNet을 이용한 보행자인식의 정확도

Table 2. Accuracy of pedestrian recognition using GoogLeNet

Training method	Mini-batch size	Multiplier of learning rate	Epoch	Transferred units (END:144)	Validation accuracy (INRIA)	Test accuracy (INRIA)	Test accuracy (BUILT DB)	Total test accuracy
sgdm	10	10	6	1-110	99.46	99.52	82.97	91.25
rmsprop	10	10	6	1-110	99.18	99.23	84.49	91.86
rmsprop	10	20	6	1-110	99.59	99.33	91.77	95.55
rmsprop	10	30	6	1-110	99.73	99.71	86.63	93.17
rmsprop	20	20	6	1-110	100.00	99.62	89.63	94.63
rmsprop	30	20	6	1-110	99.73	99.90	88.28	94.09
rmsprop	40	20	6	1-110	100.00	99.81	90.82	95.32
adam	10	10	6	1-110	100.00	99.81	89.99	94.90
adam	20	10	10	1-110	100.00	99.81	90.43	95.12
adam	30	10	10	1-110	99.73	99.62	87.86	93.74
adam	40	10	10	1-110	100.00	100.00	91.60	95.80
adam	50	10	10	1-110	99.73	99.90	91.36	95.63
adam	100	10	10	1-110	98.23	99.14	79.79	89.47
adam	20	20	6	1-110	100.00	100.00	90.57	95.29
adam	20	30	6	1-110	99.32	98.37	93.53	95.95

표 3. ResNet을 이용한 보행자인식의 정확도

Table 3. Accuracy of pedestrian recognition using ResNet

Training method	Mini-batch size	Multiplier of learning rate	Epoch	Transferred units (END:347)	Validation accuracy (INRIA)	Test accuracy (INRIA)	Test accuracy (BUILT DB)	Total test accuracy
sgdm	10	10	10	X	99.05	99.71	90.56	95.14
sgdm	100	10	10	1-310	99.32	99.42	91.60	95.51
rmsprop	50	20	6	1-310	99.86	99.81	92.29	96.05
adam	50	10	6	1-310	99.86	99.81	91.96	95.89
adam	50	20	6	1-310	100.00	99.90	92.68	96.29
adam	50	30	6	1-310	99.73	100.00	92.32	96.16
adam	50	10	10	1-310	99.86	100.00	92.41	96.21
adam	100	10	10	1-310	100.00	99.90	92.75	96.33
adam	200	10	10	1-310	99.73	99.90	92.07	95.99

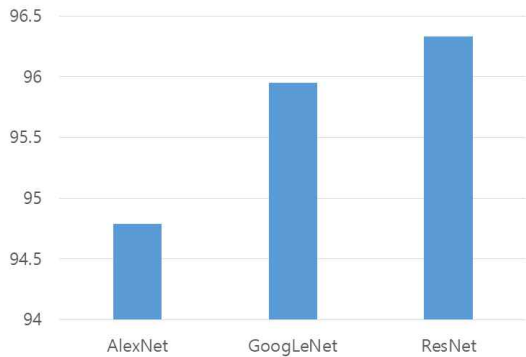


그림 6. 인식률 비교

Fig. 6. Comparison of recognition rate

학습률 승수(Multiplier for the Learning Rate)는 마지막 전연결 레이어의 가중치와 바이어스에 대한 학습률의 승수이고, 그 값을 변경하면서 인식률을 확인하였다. 학습횟수는 6회 짜리, 10회 짜리 진행하였고 전이구간(Transferred Units)은 이미 학습된 딥러닝 모델의 파라미터 값들을 그대로 가져올지 학습시킬지 구간을 명시하며 X표시는 전이없이 학습한 것이다. 전이구간의 끝은 모델별로 마지막 유닛의 번호를 나타낸다. 확인 인식률은 학습동안 학습상태를 확인하기 위한 데이터의 인식률이고, 검증 데이터는 학습이 완료된 후 새로운 데이터를 입력하였을 때의 인식결과이다. 검증은 INRIA데이터와 구축한 데이터(BUILT DB)로 실험하였다. AlexNet은 Adam으로 학습하고 미니배치가 20이고, 학습률 승수가 30이고, 전이학습을 6회 시켰을 때, 전체 검증 인식률이 94.79%로 가장 높았다. GoogLeNet은 Adam으로 학습하고 미니배치가 20이고, 학습률 승수가 30이고, 전이학습을 6회 시켰을 때, 전체 검증 인식률이 95.95%로 가장 높았다. ResNet은 101모델을 사용하였고, 그 모델은 Adam으로 학습하고 미니배치가 100이고, 학습률 승수가 10이고, 전이학습을 10회 시켰을 때, 전체 검증인식률이 96.33%로 가장 높았다.

V. 결 론

본 논문은 자동차 주행환경에서 딥러닝 모델에 따른 보행자인식의 성능비교를 한다. 자동차 주행환

경에서 보행자인식은 운전자의 안전과 보행자 안전을 위해서 중요하다. AlexNet, GoogLeNet, ResNet이 실험에 사용되었으며, 비교실험을 위해 INRIA 데이터베이스와 자동차 운전주행에 적용하기 위해 직접 구축한 데이터베이스를 사용하였다. 딥러닝 모델들은 INRIA 데이터를 학습하고 INRIA와 직접 구축한 데이터를 검증에 사용하였다. 그 결과 ResNet이 AlexNet과 GoogLeNet보다 각각 1.54%, 0.38% 더 높은 인식률을 보였다. 향후 보행자뿐만 아니라 운전 동안에 안전과 관련된 지능적인 영상처리 방법들을 연구할 계획이다.

References

- [1] G. G. Lee and W. Y. Kim, "Measuring pedestrian traffic using feature-based regression in the spatiotemporal domain", *International Journal of CAS*, Vol. 10, No. 2, pp. 328-340, Apr. 2012.
- [2] G. J. Kim, T. K. An, J. P. Kim, and Y. G. Cheong, and M. H. Kim, "Directional pedestrian counting with a hybrid map-based model", *International Journal of CAS*, Vol. 13, No. 1, pp. 201-211, Feb. 2015.
- [3] H. J. Choi, Y. S. Lee, D. S. Shim, C. G. Lee, and K. N. Choi, "Effective pedestrian detection using deformable part model based on human model", *International Journal of CAS*, Vol. 14, No. 6, pp. 1618-1625, Dec. 2016.
- [4] H. C. Song, G. H. Lee, D. S. Shim, and K. N. Choi, "Visual distinctiveness detection of pedestrian based on statistically weighting PLSA for intelligent systems", *International Journal of CAS*, Vol. 16, No. 2, pp. 815-822, Apr. 2018.
- [5] Nisha, R. Mehra, and L. Sharma, "Comparative analysis of Canny and Prewitt edge detection techniques used in image processing", *International Journal of ETT*, Vol. 28, No. 1, pp. 48-53, Oct. 2015.
- [6] W. Gao, X. Zhang, L. Yang, and H. Liu, "An improved Sobel edge detection", *International*

- Conference on CSIT, Vol. 5, pp. 67-71, Aug. 2010.
- [7] G. N. Chaple, R. D. Daruwala, and M. S. Gofane, "Comparisons of Robert, Prewitt, Sobel operator based edge detection methods for real time uses on FPGA", International Conference on TSD, pp. 1-4, Feb. 2015.
- [8] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, pp. 711-720, Jul. 1997.
- [9] Y. W. Jeong and H. S. Kim, "New speaker adaptation method using 2-D PCA", Signal Processing Letters, IEEE, Vol. 17, No. 2, pp. 193-196, Feb. 2010.
- [10] S. Z. Rizvi, J. Mohammadpour, R. Toth, N. Meskin, "A kernel-based PCA approach to model reduction of linear parameter-varying systems", Control Systems Technology, IEEE, Vol. 24, No. 5, pp. 1883-1891, Sep. 2016.
- [11] R. Saeidi, R. F. Astudillo, and D. Kolossa, "Uncertain LDA: Including observation uncertainties in discriminative transforms", Pattern Analysis and Machine Intelligence, IEEE, Vol. 38, No. 7, pp. 1479-1488, Jul. 2016.
- [12] M. Kyperountas, A. Tefas, and I. Pitas, "Weighted piecewise LDA for solving the small sample size problem in face verification", Neural Networks, IEEE, Vol. 18, No. 2, pp. 506-519, Apr. 2007.
- [13] J. Wu and H. Yang, "Linear regression-based efficient SVM learning for large-scale classification", Neural Networks and Learning Systems, IEEE, Vol. 26, No. 10, pp. 2357-2369, Oct. 2015.
- [14] S. Narayan, G. A. Tagliarini, and E. W. Page, "Enhancing MLP networks using a distributed data representation", Systems, Man, and Cybernetics Part B, IEEE, Vol. 26, No. 1, pp. 143-149, Feb. 1996.
- [15] M. S. Pak and S. H. Kim, "A review of deep learning in image recognition", International Conference on CAIPT, pp. 1-3, 2017.
- [16] C. H. Chang, "Deep and shallow architecture of multilayer neural networks", Neural Networks and Learning Systems, IEEE Vol. 26, No. 10, pp. 2477-2486, Jan. 2015.
- [17] P. L. Callet, C. Viard-Gaudin, and D. Barba, "A convolutional neural network approach for objective video quality assessment", Neural Networks, IEEE, Vol. 17, No. 5, pp. 1316-1327, Sep. 2006.
- [18] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification", IEEE Transactions on Geoscience and Remote Sensing, Vol. 55, No. 7, pp. 3639-3655, Jul. 2017.
- [19] W. Diao, X. Sun, X. Zheng, F. Dou, H. Wang, and K. Fu, "Efficient saliency-based object detection in remote sensing images using deep belief networks", Geoscience and remote sensing letters, IEEE, Vol. 13, No. 2, pp. 137-141, Feb. 2016.
- [20] H. Sasaki, T. Horiuchi, and S. Kato, "A study on vision-based mobile robot learning by deep Q-network", Annual Conference of SICE, pp. 799-804, Sep. 2017.
- [21] J. C. Hou, S. S. Wang, Y. H. Lai, Y. Tsao, H. W. Chang, and H. M. Wang, "Audio-visual speech enhancement using multimodal deep convolutional neural networks", Emerging Topics in Computational Intelligence, IEEE, Vol. 2, No. 2, pp. 117-128, Apr. 2018.
- [22] L. Shao, F. Zhu, and X. Li, "Transfer learning for visual categorization: A survey", Neural Networks and Learning Systems, IEEE, Vol. 26, No. 5, pp. 1019-1034, May 2015.
- [23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional

neural networks", International Conference on NIPS, Vol. 1, pp. 1097-1105, Dec. 2012.

[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", Conference on Computer Vision and Pattern Recognition, IEEE, pp. 1-9, Jun. 2015.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", Conference on Computer Vision and Pattern Recognition, IEEE, pp. 770-778, Dec. 2015.

[26] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition", Proceeding, IEEE, Vol. 86, No. 11, pp. 2278-2324, Nov. 1998.

[27] Raon People, <https://laonple.blog.me> [accessed: Sep. 18, 2018]

[28] GoogleNet · Artificial Intelligence, <https://leonardoaraujosantos.gitbooks.io> [accessed: Sep. 18, 2018]

[29] K. Simonyan and A. Zisserman, "Very deep convolutional neural networks for large-scale image recognition", International Conference on learning Representations, pp. 1-14, Apr. 2015.

[30] ResNet Study, <https://blog.naver.com/dbstjrwnek/> [accessed: Sep. 18, 2018]

[31] Y. H. Byeon and K. C. Kwak, "Comparative analysis of performance using faster RCNN and ACF in people detection", Journal of KIIT, Vol. 15, No. 6, pp. 11-21, Jun. 2017.

[32] S. Ruder, "An overview of gradient descent optimization algorithms", arXiv:1609.04747v2 [cs.LG], Jun. 2017.

저자소개

변 영 현 (Yeong-Hyeon Byeon)



바이오인식

2013년 2월 : 조선대학교
제어계측로봇공학과(학사)
2014년 8월 : 조선대학교
제어계측공학과(석사)
2014년 9월 ~ 현재 : 조선대학교
제어계측공학과 박사과정
관심분야 : 보행자 검출, 딥러닝,

곽 근 창 (Keun-Chang Kwak)



선임연구원

2002년 : 충북대학교 전기공학과
박사 졸업
2003년 ~ 2005년 : 캐나다
앨버타대학교 전기 및 컴퓨터
공학과, 박사후과정
2005년 ~ 2007년 : 한국전자통신
연구원 지능형로봇연구단
2014년 ~ 2015년 미국 캘리포니아 주립대학교 플러튼,
방문교수
2007년 ~ 현재 : 조선대학교 전자공학부 교수
관심분야 : 인간-로봇상호작용, 계산지능, 바이오인식