



실제 도로환경에 적합한 Faster R-CNN 기반 차종 분류

Classification of vehicles based on Faster R-CNN suitable for use in actual road environments

안호연 · 이종택[†]
Ho-Yeon Ahn and Jong Taek Lee[†]

한국전자통신연구원
Electronics and Telecommunications Research Institute

요약

본 논문에서는 딥러닝의 객체 검출 알고리즘인 Faster R-CNN을 기반한 차종 분류 방법을 제안한다. Faster R-CNN 방법은 검출과 분류를 동시에 수행한다. 하지만 데이터의 양과 하드웨어의 제약으로 검출과 분류를 동시에 수행하는데 어려움이 있다. 본 연구에서는 Faster R-CNN 알고리즘을 차량 검출 네트워크로 사용하고 추가 네트워크를 구성하여 검출된 영역에 대해 분류를 수행한다. 기존의 Faster R-CNN은 검출과 분류를 동시에 수행하지만, 제안 방법은 검출과 분류를 나누어 수행한다. 제안한 방법 검증을 위해 획득된 8종의 차량 영상 총 15,400장을 training 데이터로 사용하였으며, 3,800장을 Validation 데이터로 사용하였다. 실험 결과, Faster R-CNN을 사용한 차량 검출 성능은 mAP 87.4%, 차종 분류율은 85.4%였고, 제안한 방법의 평균 차종 분류율은 89.5%의 결과를 보였다. 또한, 제안한 방법으로 분류를 수행하였을 때, 차종별 분류 성능의 편차를 줄일 수 있었다.

키워드 : 객체 분류, 객체 검출, 딥러닝, 기계학습

Abstract

In this paper, we propose a vehicle classification method based on Faster R-CNN, an object detection algorithm using deep learning. The Faster R-CNN method performs detection and classification simultaneously. However, it is difficult to perform detection and classification simultaneously due to the amount of data and hardware constraints. In this study, the Faster R-CNN algorithm is used as a vehicle detection network and an additional network is constructed to classify the detected areas. Faster R-CNN performs detection and classification at the same time, but the proposed method performs detection and classification separately. For the proposed method verification, 15,400 of 8 types of vehicle images obtained were used as training data and 3,800 images were used as validation data. As a result, Faster R-CNN achieved mAP of 87.4%(detection) and a mean accuracy of 85.4%(classification). The proposed method achieved a mean accuracy of 89.5%(classification). Also, the proposed method reduced the variation in the classification accuracy of each vehicle type.

Key Words : Object Classification, Object Detection, Deep Learning, Machine Learning

Received: May, 28, 2018
Revised: Jun, 19, 2018
Accepted: Jun, 20, 2018
[†]Corresponding authors
jongtaeklee@etri.re.kr

1. 서론

CCTV가 빠르게 보급됨에 따라, 가정용 IoT(Internet of Things)를 비롯한 다양한 CCTV 제품들을 시장에서 볼 수 있다. 기존의 CCTV의 역할인 감시뿐 아니라 다양한 용도에 사용될 수 있는 범용 알고리즘을 장착하여 시제품으로 제작된다. CCTV는 가정용, 생활감시용, 교통 분야에 활용되고 있으며, 생활감시용 CCTV는 범죄예방에 매우 효과적인 것으로 알려져 있다. 하지만 기존의 감시방식은 영상 녹화와 저장 기능만 존재하였다. 최근에는 기존의 영상을 녹화하고 저장하는 역할 뿐 아니라, 다양한 알고리즘 탑재가 가능하다. 알고리즘을 통해서 이벤트 발생 시 초기에 인지할 수 있으며, 또한 이벤트 발생 사후처리에 도움을 준다.

이벤트 발생 인지 알고리즘은 경찰 수사에 중요한 초기 수사를 진행하는 데 가장 중요한 단서가 될 수 있으며, 시민의 안전을 도모할 수 있다.

특히 교통 분야에서는 CCTV보급에 따라 지능형 교통시스템(Intelligence Transportation System)이 보편화 되었으며, 보행자와 도로를 주행하는 운전자 모두에게 안전하고 편리한 환경을 조성하는데 기여하고 있다.

본 논문은 2018년 대경권 지역산업기반 ICT 융합기술 고도화 지원사업의 지원을 받아 연구하였음.

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.

초기 지능형 교통시스템은 도로변에 설치된 CCTV에 어떤 알고리즘도 장착되지 않았으며, 교통류를 관찰하는 역할을 해왔다. 교통수단과 교통 시설물들이 분리된 상태에서 교통관리 또는 교통 소통 중심의 정보수집 제공시스템이기 때문에 영상검지기로부터 수집된 데이터를 활용하기 위해서는 많은 비용과 인력이 소모되었다.

새롭게 소개될 차세대 지능형 교통시스템에서는 교통안전중심의 교통체계라고 말할 수 있다. 차량과 차량 간(Vehicle to Vehicle), 차량과 도로시설물(Vehicle to Infrastructure)간의 지속적인 데이터 공유를 가능하게하며, 개별차량에 대한 실시간 정보를 제공 가능토록 하는 것이 목적이다. 기존 목시 검시자로 돌발 상황에 대응하는 방법과 달리 차량 간, 차량과 시설물관 정보공유를 통해 신속하고 능동적인 돌발 상황 대응이 가능하게 한다.

기존의 지능형 교통시스템의 비용 상의 문제와 비효율성을 극복하기 위해 딥러닝(Deep Learning) 알고리즘 및 통신기술을 사용한다면, 차세대 지능형 교통시스템 환경을 구축하는데 큰 도움을 줄 것으로 기대된다. 차세대 지능형 교통시스템에서 사용되는 다양한 정보들을 딥러닝 알고리즘을 통해 원활하게 획득할 수 있다. 대표적으로 획득된 영상에서 정보를 자동으로 추출할 수 있는 방법이 활발하게 연구되고 있다.

기계학습의 알고리즘 중 다층 퍼셉트론(Multi-layer perceptron)은 현재 뜨겁게 관심 받고 있는 딥러닝의 시조라고 할 수 있으며, 과거 퍼셉트론의 성능은 우수하지만 제한된 하드웨어 스펙에 의해 많은 층을 학습할 수 없었다. 하지만 고성능 연산장치를 비롯한 하드웨어의 성능의 비약적인 발전으로 인해 딥 뉴럴 네트워크(Deep Neural Network)를 사용가능 하다. 딥 뉴럴 네트워크는 매년 향상된 성능을 내는 알고리즘을 내놓고 있다. 딥러닝의 한 분야라 할 수 있는 강화학습(Reinforcement Learning)을 통해 학습된 알고리즘 ‘알파고’를 통해 해당 분야의 매우 급진적인 발전을 확인할 수 있다. 딥러닝은 기존의 기계학습 알고리즘의 성능을 앞섰으며 학습된 데이터 내에서만 강인한 것이 아닌, Generality를 통해 컴퓨터 비전분야 분류문제, 객체 검출과 추적 알고리즘 모든 분야에 우수한 성능으로 적용이 가능하다. 시계열 형태의 데이터도 딥러닝을 이용하면 쉽게 다룰 수 있다. 자연어처리에 매우 효과적인 것으로 알려져 있으며, 시제품으로 볼 수 있는 인공지능 스피커, 휴대폰에 탑재된 인공지능 비서는 모두 해당분야의 결과물이라고 할 수 있다.

본 연구에서는 2장을 통해 기존연구를 소개하고, 3장을 통해 제안 알고리즘을 소개한다. 4장에서는 제안한 방법의 실험결과를 보여주고, 5장을 통해 결론을 짓는다.

2 기존 연구

딥러닝의 검출기법과 달리 기존 기계학습 검출 알고리즘은 획득된 영상으로부터 특징을 추출한다. 대표적인 방법으로 HOG-SVM(Surppot Vector Machine)이 있으며[2], 영상의 특성에 따라 최적의 descriptor를 선택하여 이를 분류기의 입력으로 넣어 검출을 수행한다.[1,3] 반면, 딥러닝 알고리즘은 영상 최적의 descriptor를 선택하는 것이 아닌 추론과정과 역전파 알고리즘을 통해 학습하게 된다. 그림 1을 통해 특징기반 기계학습과 딥러닝의 차이를 볼 수 있다.

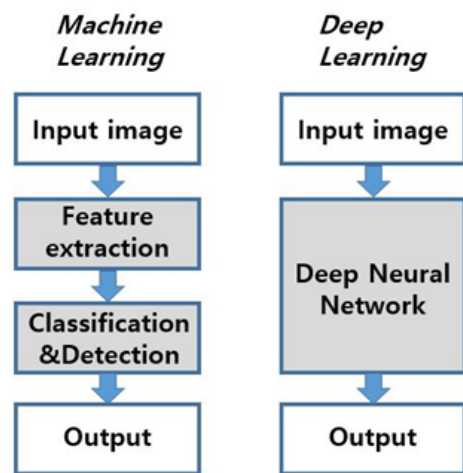


그림 1. 기계학습과 딥러닝의 학습 차이

Fig. 1. Difference between machine learning and deep learning

IISVRC (Large Scale ImageNet Challenge) 2012에서 큰 주목을 받은 AlexNet[4]을 시점으로 해마다 놀라운 결과의 성능을 가진 네트워크들이 제안되었다.[5,6]

Inception model[7]은 22의 층을 가진 구조로 깊은 네트워크 구조의 가능성을 보여 주었다. 이후 제안된 연구들은 gradient vanishing을 극복하면서 100층 이상의 깊은 구조를 갖고 있다.[8,9]

딥러닝 모델들을 바탕으로 다양한 컴퓨터 비전 Task에 적용이 가능하며, 특히 기존 특징추출에 기반한 객체 검출분야에도 많은 변화를 일으켰다. 딥러닝의 대표적인 객체 검출을 위한 알고리즘은 두 계통으로 나눌 수 있다. 첫 번째는 R-CNN[10] 계통이다. 기존 R-CNN이 지니던 region proposal의 bottle neck현상을 현저히 저감시켜 성능을 향상시킨 검출 알고리즘들이 개발되었다.[11, 12]

두 번째는 픽셀단위의 정답을 부여하는 semantic segmentation[13, 14] 방법으로 객체의 대상 영역을 명확히 부여할 수 있으며, 자율주행 자동차 플랫폼 등에 응용이 가능하다. 언급한 알고리즘들은 모두 객체검출에 최적화 되어있으며, 범용성이 매우 높다. 따라서 범용성이 학습된 데이터에 fine-tuning을 통해

다양한 분야에 사용되고 있다. Fine-tuning이란 기존에 학습된 모델을 기반으로 용도에 맞게 네트워크의 구조를 변형시키고 이미 학습된 가중치로부터 학습을 진행하는 것을 의미한다. 즉, 모델의 파라미터를 미세하게 조정하는 방법으로 사전에 학습된 네트워크에 데이터를 입력하여 가중치를 갱신한다.

3. Faster R-CNN 이용한 차종분류

딥러닝을 이용한 객체 검출 알고리즘은 분류 알고리즘 보다 복잡한 구조를 갖고 있다. 객체의 위치를 추정하는 bounding box를 예측하기 위한 학습이 요구된다. 대표적인 객체 검출 알고리즘인 R-CNN은 Faster R-CNN의 근간이 되는 연구라 할 수 있다.

Faster R-CNN은 선행 연구인 R-CNN의 성능을 개선하며 제안된 대표적인 딥러닝 기반 객체 검출 알고리즘이다. 기존 R-CNN 알고리즘이 객체 검출을 수행하는 단계는 다음과 같다.

- 단계 1: 객체가 존재하는 입력영상을 넣는다.
- 단계 2 : Selective search[15] 알고리즘을 통해 후보영역을 추출한다.
- 단계 3 : 후보영역을 사용되는 CNN 구조의 입력에 넣기 위해 크기를 조절한다. 크기 조절된 영상을 Warped region이라 한다.
- 단계 4 : Warped region에 대해 CNN의 입력으로 넣고 추출된 특징을 SVM을 통해 분류한다.
- 단계 5 : 객체의 Bounding Box 좌표를 알맞게 만들기 위해 선형 회귀를 수행한다.



그림 2. Selective search 알고리즘 수행결과
Fig. 2. Result of selective search algorithm

그림 2는 R-CNN 알고리즘 수행에 앞서 후보군을 찾기 위한

selective search 알고리즘의 예시를 보여준다.

Selective search에서 사용된 내부 문턱치 값에 따라 후보 영역의 파란색 Bounding Box를 조절할 수 있다. Bounding Box의 개수가 적을 경우 객체가 Box내부에 존재하지 않을 가능성이 높으며 반대로 개수가 많을 경우 알고리즘의 수행시간이 길어지기 때문에 적절한 개수의 Bounding Box가 생성되는 문턱치를 사용해야 한다.

R-CNN 알고리즘은 selective search 방법으로 획득된 후보군 영역에 대해 모두 CNN의 입력으로 넣어야 한다. 또한 CNN를 거치며 나온 결과 특징에 SVM로 분류를 하였고, 마지막으로 Bounding Box의 좌표 값을 학습시키기 위해 선형회귀 까지 총 3 단계의 학습과정을 거쳐야 한다.

여러 단계로 학습과정을 거치는 문제점을 Fast R-CNN에서는 RoIPool을 사용하였다. RoIPool 방법은 selective search 알고리즘을 사용하여 추출된 후보영역의 Bounding Box 정보를 CNN의 결과 특징 맵에 적용시키는 것이다. 해당 방법을 통해 모든 후보영역에 대해 CNN 특징을 추출하여 소모시간 문제를 해결 하였다.

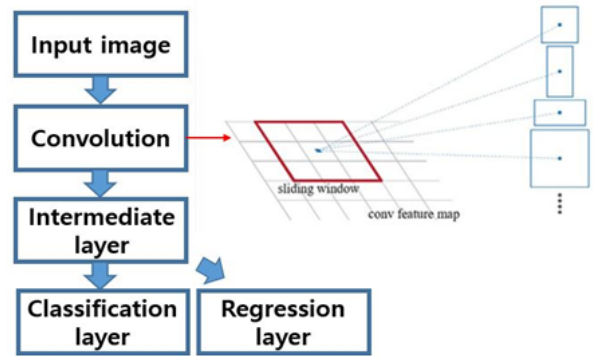


그림 3. Faster R-CNN 알고리즘 수행 절차

Fig. 3. Faster R-CNN algorithm performing procedure

Fast R-CNN이 영상에서 객체라고 할 수 있는 후보영역을 추출하는 것을 selective search 방법이 아닌 RPN(Region Proposal Network)이라 불리는 CNN 내부에서 해결할 수 있도록 설계한 것을 위의 그림 3을 통해 확인할 수 있다.

RPN은 영상을 입력받고 Box형태의 객체의 후보영역을 예측하는 역할을 한다. 사용된 anchor Box는 슬라이딩 윈도우를 하며 Bounding Box의 후보로 사용되는 사전에 정의된 상자이며, 크기와 비율을 달리하며 후보영역을 획득한다. 따라서 객체의 클래스를 분류하는 손실함수와 Bounding Box를 위한 손실함수를 더한 손실함수는 아래 식 (1)과 같이 정의할 수 있다.

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum L_{reg}(t_i, t_i^*) \quad (1)$$

여기서 p_i , t_i 는 각각 anchor Box의 예측 값과 bounding box의

예측 값이다. 해당 예측 값을 Ground-truth 라벨과 박스와의 손실을 계산하여 학습을 진행한다. Bounding Box regression 과정에서는 객체의 위치와 가로, 세로 총 4개의 좌표들에 대해서 다음 식(2)와 같은 연산을 취하게 되며 각 anchor box마다 4개 좌표에 대한 회귀를 수행한다. Intermediate layer를 통해 분류를 위한 네트워크와 box regression을 위한 네트워크로 이어져 하나의 단계로 학습이 가능한 장점이 있다.

$$\begin{aligned}
 t_x &= (x - x_a) / w_a, & t_y &= (y - y_a) / h_a \\
 t_x^* &= (x^* - x_a) / w_a, & t_y^* &= (y^* - y_a) / h_a \\
 t_w &= \log(w / w_a), & t_h &= \log(h / h_a) \\
 t_w^* &= \log(w^* / w_a), & t_h^* &= \log(h^* / h_a)
 \end{aligned}
 \tag{2}$$

기존의 R-CNN 알고리즘이 총 3단계의 학습과정을 사용하는 것과 달리 역전파 알고리즘으로 검출과 분류를 위한 네트워크를 동시에 학습이 가능하며 수행속도 또한 단축시켰다. bounding box의 회귀와 분류를 수행하는 네트워크가 한 네트워크로 수행되기 때문에 학습과정에서 적절히 균형을 이루도록 학습하여야한다.

하지만 분류와 bounding box 회귀 두 가지를 동시에 수행하는 구조를 사용할 경우 차량의 세부 분류의 정확성을 확보하기 어렵다. 차종분류는 차종 간 분산이 크지않기 때문에 세밀한 특징의 차이를 학습 중 네트워크에 잘 반영하여야한다. 이를 fine-grained라 하며 기존 분류 네트워크의 학습보다 어려운 문제이다.

따라서 본 연구에서는 그림 4와 같이 Faster R-CNN을 통해 차량을 대분류한 후, 추가의 네트워크 학습을 통해 검출된 차량에 대해 세부 분류를 수행하는 알고리즘을 제안한다.

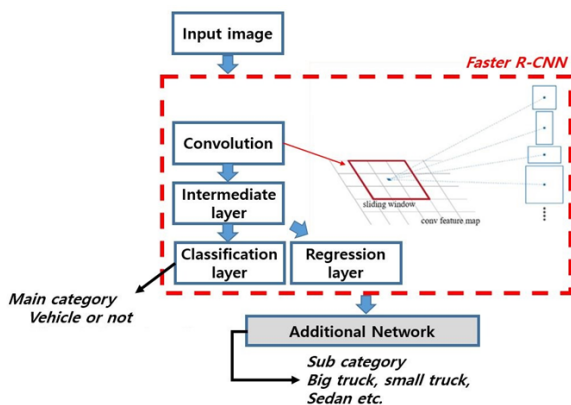


그림 4. Faster R-CNN 기반 차종 분류 제안 방법

Fig. 4. Faster R-CNN based vehicle classification proposal method

제안한 방법은 세부 분류를 수행하기 전 Faster R-CNN의 네트워크를 통해 차량을 검출하고 추가 네트워크 학습을 통해 세부 분류를 수행하였다. Faster R-CNN의 네트워크는 차량의 존재 여부를 분류하기 위해 사용하고, 차량이 검출된 영상에서 검출된

영역을 분할하여 추가네트워크를 학습시켰다. 총 8가지 차종에 대해 세부 분류를 수행하였다.

4. 실험

본 논문에서 제안한 알고리즘의 성능을 평가하기 위해 사용된 PC는 Intel core i7 CPU, 16G Byte이다. 개발 환경은 Visual studio 2017 compiler, Tensorflow r1.7을 이용하여 구현하였다. 영상은 주간 오전9시~오후 5시에 획득하였으며 실험에 사용된 영상은 도로변에 설치된 2메가 픽셀 해상도 카메라로 그림 5와 같이 차량의 전면과 후면 영상을 동시에 획득한다. 영상의 크기는 1920 픽셀이며, 동영상이 아닌 정지 영상을 사용하였다.

사용된 학습 데이터는 그림 6과 같이 라벨링을 하였다.라벨링은 모든 차량에 대해서 사각형 박스 정보와 차량의 종류로 구성되어 있다. 차량의 전면과 후면 모두를 학습에 사용하였으며, big truck, middle truck, small truck, sedan, compact car, suv, van 8종의 차량으로 구성된 데이터 셋이다. 획득된 영상은 모두 같은 위치에서 획득하였다. 하지만 시간에 따라 그림자와 광량의 차이로 인한 약간의 배경의 차이가 있다.

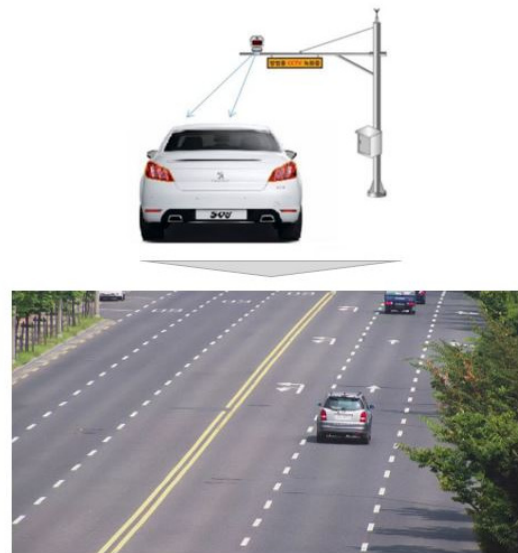


그림 5. 실험에 사용된 영상 획득 방법

Fig. 5. Image acquisition method used in experiment

사용된 Faster R-CNN의 내부 네트워크는 Inception v2를 사용하였으며, 실험에 사용되는 모든 네트워크는 ImageNet 데이터 셋으로 사전에 학습된 가중치를 사용하고, 준비된 데이터 셋으로 fine-tuning하였다. 또한 세부분류를 위한 추가 네트워크도 마찬가지로 검출에 사용된 Inception v2를 사용하였다.

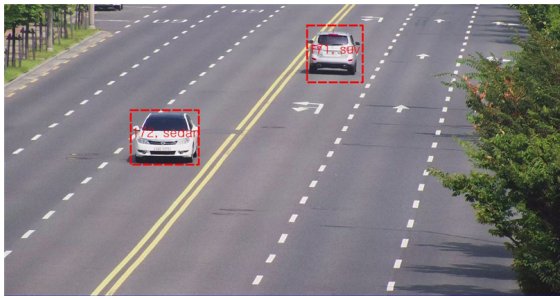


그림 6. 학습에 사용된 데이터 예제
Fig. 6. Example of training data sets

실험은 총 2가지 방법으로 수행하였다.

첫 번째는 Faster R-CNN의 네트워크 자체로 세부분류를 수행하는 한 단계로 학습이 가능한 방법이다. 즉 추가 네트워크의 학습 없이 수행하는 방법이다. 학습데이터는 training 데이터 6,400장 validation 데이터 1,600장을 사용하였다.

두 번째는 본 연구의 제안 방법으로 Faster R-CNN의 네트워크를 통해 차량의 영역을 검출하고 세부 분류는 추가의 네트워크 학습을 통해 수행하는 방법이다. 따라서 검출을 위한 학습과 분류를 위한 학습이 두 단계로 수행된다. 해당 방법의 학습데이터는 총 19,200 장으로 이중 15,400장을 training에 사용하고 3,800장을 validation 데이터로 사용하였다. mAP(mean Average Precision)는 딥러닝 객체 검출 알고리즘에서 주로 사용되는 성능지표이다. IoU(Intersection of Union)은 Ground truth bounding box와 예측된 box사이의 겹치는 영역의 비율을 말하며 해당 연구에서는 0.5로 설정하였다. 즉, Ground truth와 영역이 0.5 이상 겹칠 경우 참인 box로 간주한다.

표 1. 차량 종류별 mAP

Table 1. Calculated mAP on each vehicle types

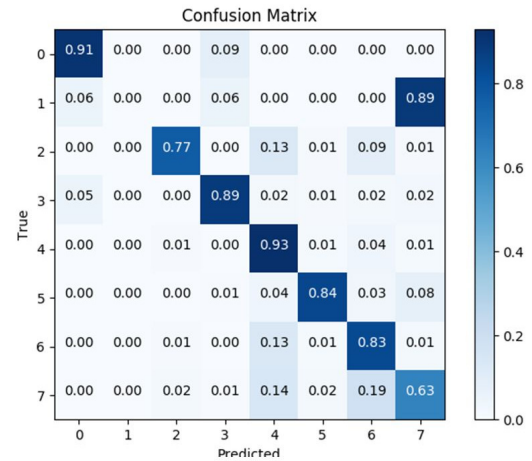
data	AP(%)
big_truck	91.4
middle_truck	98.3
small_truck	99.2
suv	95.2
compact_car	94.0
sedan	98.7
van	76.0
bus	87.1
mAP	87.4

4.1 Faster R-CNN을 이용한 차종 분류

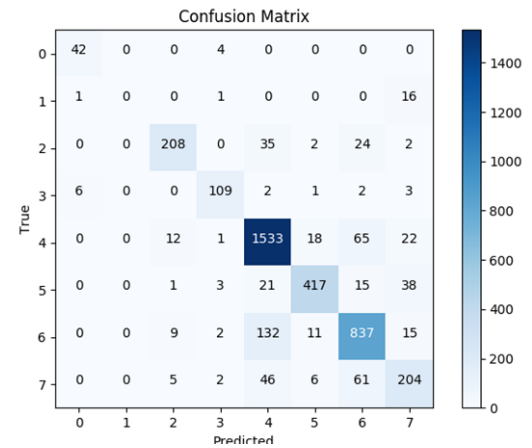
Faster R-CNN의 네트워크를 통해 차종을 분류할 경우 검출과 분류를 한 단계의 학습과정으로 수행할 수 있는 장점이 있다. Faster R-CNN 내부에서 Intermediate layer로부터 이어지는 분류를 위한 층과 bounding box 회귀를 위한 층이 균형을 이루며 학습할 수

있도록 손실함수의 균형을 갖도록 학습한다. RPN(Region Proposal Network)는 3개의 비율을 갖는(1:1, 1:2, 2:1) anchor box들을 사용하였다. 우선 ImageNet 데이터를 이용하여 네트워크에 사전에 학습된가중치를 불러온다. 획득한 학습용 데이터를 사용하여 Fine-tuning한 후 학습을 진행한 결과를 표 1과 같이 혼동행렬로 나타냈다.

big_truck=0 bus=1 compact_car=2 middle_truck=3 sedan=4 small_truck=5 suv=6 van=7



(a)



(b)

그림 7. Faster R-CNN을 이용한 분류 혼동행렬:

(a) 정규화된 혼동행렬, (b) 데이터 수에 따른 혼동행렬

Fig. 7. Confusion matrix of classification using Faster R-CNN: (a) Normalized confusion matrix, (b) Confusion matrix based on the number of data

Fine-tuning을 사용하면 초기 수렴속도를 앞당길 수 있으며, ImageNet 데이터의 차량 클래스 검출 특징을 초기 값으로 활용할 수 있는 장점이 있다. 또한 특정 분류 문제에서 초기 랜덤 가중치를 사용하는 것보다 효과적인 것으로 알려져 있다.

그림 7의 혼동행렬로 나타낸 Faster R-CNN의 네트워크 차종 분류 성능과 비교했을 때, 표 1의 AP는 다소 높은 성능을 보인다. 이러한 현상은 과 검출이 많은 것으로 예상할 수 있으며, IoU 설정에 따라서

분류 성능에 영향을 미칠 수 있다. 낮은 IoU를 허용할 경우, 차량의 일부만 Ground Truth와 겹치게 되도 차량으로 검출하고 분할된 영상으로 분류 네트워크가 예측을 했을 때 잘못 분류할 가능성이 높다.

Faster R-CNN을 기반으로 차종분류 시 그림 7 (a)의 정규화 된 혼동행렬을 살펴보면 클래스 별 분류율의 편차가 큰 것을 알 수 있다. 특히 그림 7 (b) 에서 bus의 경우 클래스의 데이터 수가 적으며, 성능이 매우 낮다. van으로 분류되는 경우가 가장 많다. 반면에 클래스 데이터수가 많은 sedan은 분류율이 가장 높게 나타났다.



그림 8. Faster R-CNN을 사용한 분류:

(a) 올바른 예, (b) 잘못된 예

Fig. 8. Classification using Faster R-CNN

(a) True, (b) False

Faster R-CNN의 자체 분류 성능을 살펴보면 충분한 데이터 수가 확보된 경우 우수한 성능을 보인다.

반면, bus와 같이 데이터수가 적은 경우 좋지 못한 성능을 나타내는 경향을 보인다. 이는 차량의 검출과 분류 네트워크가 동시에 학습되기 때문에 클래스 데이터 수의 편차를 극복할 만큼 충분한 학습이 되지않은 것이다. 그림 8은 Faster R-CNN을 사용하여 차종 분류 수행결과를 보여준다. 그림 8의 (b)에서 compact car는 sedan으로 van을 suv로 분류하였다.

4.2 Faster R-CNN을 이용한 차량 검출 후 추가 네트워크를 통한 차량세부분류

차량의 여부를 Faster R-CNN을 사용하여 대분류하고 개별 학습된 추가의 네트워크를 구성하여 8종의 차량을 세부분류를 한다. Faster R-CNN 알고리즘을 사용하면 차량 검출부터 분류까지 한

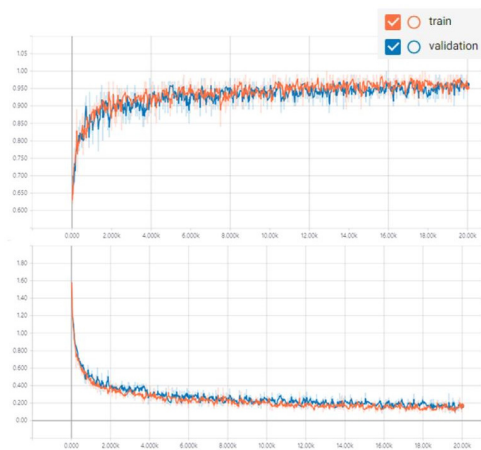


그림 9. 추가 네트워크의 training loss와 validation error

Fig. 9. Training loss and validation error of additional network

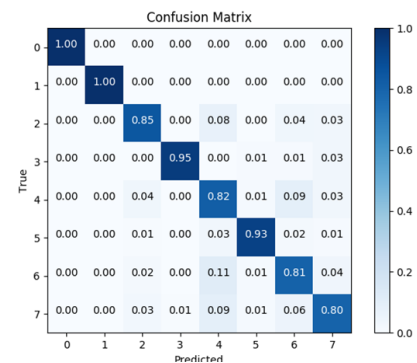


그림 10. 제안한 방법을 이용한 분류 혼동행렬

Fig. 10. Classification confusion matrix using the proposed method



그림 11. 제안한 방법을 사용한 분류:

(a) 올바른 예, (b) 잘못된 예

Fig. 11. Classification using the proposed method

(a) True, (b) False

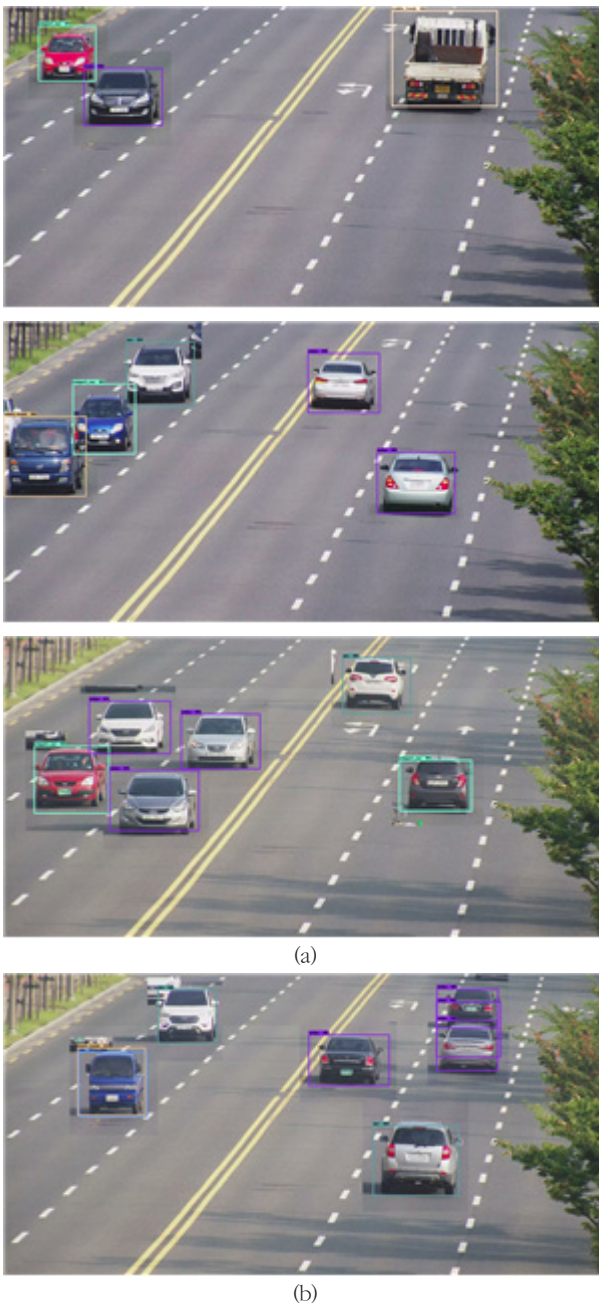


그림 12. 차량 검출 및 분류 수행결과:
(a) 올바른 예, (b) 잘못된 예

Fig. 12. Vehicle detection and classification result:
(a) True, (b) False

번에 학습이 가능하지만, 제안한 방법은 추가 네트워크를 통해 더욱 정확한 차량의 세부분류를 수행한다. 하지만 차종 분류를 수행하는 시간이 소모 증가되며, 추가 네트워크 학습을 수행해야하는 단점이 있다. 추가 네트워크 학습을 통해 사용된 모델은 Inception v2으로 학습과정의 training loss와 validation error의 추세 그래프는 그림 9와 같다.

Faster R-CNN는 클래스 별 데이터 수가 충분하고 검출네트워크와

분류를 위한 네트워크를 균형 있게 학습할 수 있다면 우수한 성능을 낼 수 있다. 하지만 제한된 하드웨어와 데이터가 부족한 경우 성능의 한계가 있다.

반면, 제안한 방법은 차량 검출을 위한 Faster R-CNN과 차량 분류를 위한 추가 네트워크를 따로 학습해야 하는 단점이 있지만 Faster R-CNN를 잘 학습할 수 없는 제한된 하드웨어와 부족한 데이터 셋에서 좋은 결과를 보여준다. 즉, Faster R-CNN이 검출과 분류를 모두 수행해야하는 부담을 추가 네트워크가 분담하여 원활한 성능 향상을 가능하게 한다.

검출된 차량과 차량에 대한 세부분류 결과를 그림 12를 통해 보여준다. 그림 12의 (a)는 모두 정답과 일치하게 검출과 분류가 수행되었다. 반면 그림 12 (b)의 경우 한 차량이 두 개의 차량으로 검출되며, 같은 방향으로 진행하는 근접한 차량에 추가의 box가 그려진 것을 확인할 수 있다. 인접한 차량이 겹쳐져 있을 때 검출 오류가 발생하는 것을 확인할 수 있다. 두 차량이 겹친 경우 앞서가는 차량이 후속 차량에게 가려져 오류가 발생한다.

5. 결론 및 향후 연구

본 논문에서는 딥러닝의 검출 알고리즘인 Faster R-CNN을 기반으로 영상에서 차량을 검출하고 차종을 분류한다. 제안한 알고리즘은 공공 데이터 셋을 통해 학습된 네트워크를 fine-tuning 하여 실제 환경에 적합하도록 학습하였다. fine-tuning을 통해 총 8종의 차량의 특징들을 학습하여 차종분류를 수행하였다. Faster R-CNN 또한 검출과 분류를 동시에 수행하면서 우수한 성능을 내는 것을 알 수 있다. 하지만 데이터수가 적은 클래스에 대해 올바른 학습이 어렵다.

반면 제안한 방법으로 네트워크를 학습시켰을 때, 적은수의 클래스 데이터에 대해 분류성능이 우수하며 전체 분류 성능의 향상을 확인할 수 있었다. 하지만 이로 인해 차종분류의 수행속도는 느려지는 단점이 있다. 따라서 차종별 특징이 잘 반영될 수 있는 일부 차종에 편향되지 않는 좋은 데이터 셋을 구축하는 것이 필요하다. 실험에서 사용된 데이터는 sedan이 가장 많았으며, bus 클래스가 가장 적은 데이터 수를 지니고 있다.

Faster R-CNN의 학습만으로 fine-grained 문제라 할 수 있는 차종분류 수행이 가능하다. 하지만 데이터가 적고 편향적인 경우, 데이터 수에 덜 영향을 받는 검출 및 분류 알고리즘 개발에 대한 고려가 요구된다. 또한 하드웨어의 제약 극복하며 검출과 분류를 동시에 수행하는 네트워크에 관한 추가 연구가 필요하다.

본 연구를 통해 차량을 검출하고 분류함으로써 차세대 지능형 교통체계를 구축하는데 필요한 차량번호판, 색상정보 외 차량 종류

데이터를 추가할 수 있으며, 실제 환경에서 활용도가 높을 것으로 기대된다.

실험 결과, Faster R-CNN을 사용한 차량 검출 성능은 mAP 87.4%, 차종 분류율은 85.4%였고, 제안한 방법의 평균 차종 분류율은 89.5%의 결과를 보였다. 또한, 제안한 방법으로 분류를 수행하였을 때, 차종별 분류 성능의 편차를 줄일 수 있었다.

검출 성능을 더 올리기 위해 Faster R-CNN보다 수행 속도가 빠른 딥러닝 기반 검출 알고리즘을 사용하여 차종분류를 진행할 예정이며, 다양한 환경에서 획득한 학습 데이터를 활용한 추후 연구가 필요하다. 즉, 기존의 환경과 달리 같은 배경이라도 야간, 기상변화 눈, 비가 내리는 경우의 영상을 추가로 확보하고 강인한 차량 검출 및 분류 알고리즘 연구가 필요하다. 또한 도로의 중앙부가 아닌 신호 교차로에서 차량들의 대기 열이 긴 경우 차량이 겹쳐지는 영상에 대한 검출 방법론을 추후 연구할 예정이다.

References

[1] P. Viola, M.J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance", *International Journal of Computer Vision*, Vol. 63, No. 2, pp. 153-161, Jul. 2005.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", *Computer Vision and Pattern Recognition, IEEE Computer Society Conference*, pp. 886-893, Jun. 2005.

[3] X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling", *Computer Vision, 12th International Conference*, pp. 32-39, Oct. 2009.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", In *Advances in neural information processing systems*, pages 1097-1105, 2012.

[5] Zeiler, M.D. and Fergus, R, "Visualizing and understanding convolutional networks", *European Conference on Computer Vision(ECCV)*, 2014, pp. 818-833.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv:1409.1556, 2014.

[7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1-9, 2015.

[8] K. He , X. Zhang , S. Ren , J. Sun , "Deep residual learning for image recognition", *in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778.

[9] G. Huang, Z. Liu, K.Q. Weinberger, "Densely connected convolutional networks", *in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4700-4708.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", In *Computer Vision and Pattern Recognition(CVPR), 2014 IEEE Conference on*, pages 580-587.

[11] R. B. Girshick, "Fast R-CNN", CoRR abs/1504.08083, 2015.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks. arXiv preprint arXiv:1506.01497, 2015.

[13] V. Badrinarayanan, A. Handa, and R. Cipolla, "SegNet: a deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. arXiv preprint arXiv:1505.07293, 2015.

[14] J. Long, E. Shelhamer, and T. Darrel, "Fully convolutional networks for semantic segmentation", *In Computer Vision and Pattern Recognition(CVPR)*, 2015.

[15] J. Uijlings, K. Van de Sande, T Gevers, and A. Smeulders, "Selective search for object recognition", *IJCV*, 2013.

저자소개



안호연(Ho-Yeon Ahn)

2016년 : 아주대학교 산업공학과 공학사

2018년 : 경북대학교 전자공학부 공학석사

2017년~현재 : 한국전자통신연구원

대경권연구센터 위촉연구원

관심분야 : 컴퓨터비전, 객체인식, 기계학습

※ 개인정보 표시 제한

E-mail : ahy60209@etri.re.kr



이종택(Jong Taek Lee)

2005년 : 한국과학기술원 전기 및 전자공학과
공학사

2007년 : 텍사스 주립대(University of Texas at
Austin) 전자컴퓨터공학 공학석사

2012년 : 텍사스 주립대(University of Texas at
Austin) 전자컴퓨터공학 공학박사

2012년~현재 : 한국전자통신연구원 대경원연구센터 선임연구원

관심분야 : 컴퓨터비전, 객체인식, 기계학습

※개인정보 표시 제한

E-mail : jongtaeklee@etri.re.kr