

# 딥러닝 기반의 주행가능 영역 추출 모델에 관한 연구

## A Study on Model for Drivable Area Segmentation based on Deep Learning

전 효 진<sup>1</sup>                      조 수 선<sup>1\*</sup>  
Hyo-jin Jeon                      Soo-sun Cho

### 요 약

인공지능, 빅데이터, 자율주행 등 4차 산업혁명시대를 이끄는 핵심기술은 컴퓨팅 파워의 급속한 발전과 사물인터넷에 기반한 초연결 네트워크를 통해 구현되고 서비스된다. 본 논문에서는 자율주행을 위한 기본적인 기능으로 다양한 환경에서도 정확하게 주행가능한 영역을 인식하여 추출하는 인공지능 딥러닝 모델들을 구현하고, 그 결과를 비교, 분석한다. 주행가능한 영역을 추출하는 딥러닝 모델은 영상 분할 분야에서 성능이 우수하고 자율주행 연구에서 많이 사용하는 Deep Lab V3+와 Mask R-CNN을 활용하였다. 다양한 환경에서의 주행 정보를 위해 여러 가지 날씨 조건과 주-야간 환경에서의 주행 영상 및 이미지를 제공하는 BDD 데이터셋을 학습데이터로 사용하였다. 활용한 모델들의 실험 결과, DeepLab V3+는 48.97%의 IoU를 보였으며, Mask R-CNN은 68.33%의 IoU로 더 우수한 성능을 보였다. 또한, 구현한 모델로 추출된 주행가능 영역을 이미지에 표시하여 육안으로 검사한 결과, Mask R-CNN은 83%, Deep Lab V3+는 69% 정확도로 Mask R-CNN이 Deep Lab V3+ 보다 주행가능한 영역을 추출하는 분야에서는 더 성능이 높은 것으로 확인하였다.

☞ 주제어 : 주행가능 영역 추출, 영상 분할, 딥러닝, DeepLab V3+, Mask R-CNN, BDD 데이터셋

### ABSTRACT

Core technologies that lead the Fourth Industrial Revolution era, such as artificial intelligence, big data, and autonomous driving, are implemented and serviced through the rapid development of computing power and hyper-connected networks based on the Internet of Things. In this paper, we implement two different models for drivable area segmentation in various environment, and propose a better model by comparing the results. The models for drivable area segmentation are using DeepLab V3+ and Mask R-CNN, which have great performances in the field of image segmentation and are used in many studies in autonomous driving technology. For driving information in various environment, we use BDD dataset which provides driving videos and images in various weather conditions and day&night time. The result of two different models shows that Mask R-CNN has higher performance with 68.33% IoU than DeepLab V3+ with 48.97% IoU. In addition, the result of visual inspection of drivable area segmentation on driving image, the accuracy of Mask R-CNN is 83% and DeepLab V3+ is 69%. It indicates Mask R-CNN is more efficient than DeepLab V3+ in drivable area segmentation.

☞ keyword : Drivable area segmentation, Segmentation, Deep Learning, DeeLab V3+, Mask R-CNN, BDD Dataset

## 1. 서 론

4차 산업혁명시대의 대표적인 핵심기술인 인공지능 기술은 수십 년 동안 연구되어 왔지만 딥러닝의 등장으로 새로운 전기를 맞게 되었다. 인공지능 분야에서는 딥러닝 덕분에 과거에는 해결하지 못했던 많은 문제에 대

해 새로운 해결책이 제시되고 있으며 또한 이전에는 생각지도 못했던 전혀 새로운 문제에 도전하고 있다. 특히 딥러닝을 통한 이미지 인식 분야에서는 복잡한 분류 감지 기술이 크게 발전하면서 이미지 속 사물의 종류를 인식하는 것뿐만 아니라 영상 속의 다양한 상황까지 이해할 수 있게 되었다. 이를 바탕으로 4차 산업혁명시대를 이끄는 또 다른 핵심기술인 자율주행 시스템에 대한 연구가 활발하게 진행되고 있다.

자율주행의 핵심 기술은 크게 3가지로, 인식, 판단, 제어로 구성된다. 감지기 등을 통해서 자신의 위치 및 주변 장애물을 검출, 인식 등을 수행하는 인식 기술과 인식된 결과를 기본으로 다음 행동을 결정하는 판단기술, 그리고

<sup>1</sup> Department of Computer Science and Information Engineering, Korea National University of Transportation, Uiwang Gyeonggi, 16106, Korea.

\* Corresponding author (sscho@ut.ac.kr)

[Received 28 August 2019, Reviewed 23 September 2019, Accepted 12 October 2019]

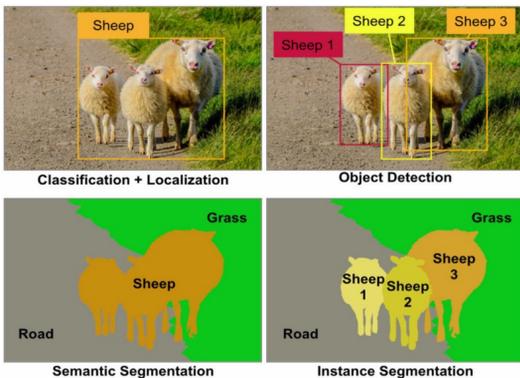
판단한 것을 토대로 빠르고 정확하게 수행하는 제어기술 이다[1]. 하지만 이러한 기술은 감지기의 고장 혹은 시계 나 악천후로 인한 도로 인식이 불가능한 상황에서 오작 동으로 인한 사고가 발생할 위험성이 크다.

본 논문에서는 다양한 환경에서도 정확하게 주행가능 한 영역을 인식하여 추출하는 딥러닝 모델들을 구현하 고, 그 결과를 비교하여 분석하고자 한다. 비교실험을 위 해 이미지에서 사물의 종류를 인식하고 위치를 표시하는 영상 분할 모델들 중에서 가장 성능이 우수하여 자율주 행 연구에서 많이 사용되는 모델들[2, 3]을 활용하였다. 또 여러 가지 복잡한 환경에서의 주행 정보를 위해 비 내 리는 날씨, 흐린 날씨, 맑은 날씨, 안개와 같은 다양한 날 씨 조건과 주·야간 환경에서의 주행 영상을 제공하는 데 이터셋[4]을 모델의 학습데이터로 사용하였다.

본 논문의 2장에서는 활용한 모델에 대하여 간략히 소 개하고, 3장에서 모델의 성능 비교를 위해 제안하는 방법 에 대해 알아본다. 이 후 4장에서 제안하는 방법으로 실험 한 결과를 보이고 5장으로 결론을 맺는다.

## 2. 관련 연구

주어진 이미지에 원하는 객체가 존재하는지, 존재한 다면 어디에 위치하는지를 보여주는 기능에는 크게 객체 검출(Object Detection)과 분할(Segmentation)로 나눌 수 있 다. 객체검출은 분류(Classification)를 수행하고 경계 상자 (Bounding Box)를 통해 객체의 위치를 나타낸다. 분할은 이미지를 픽셀 단위로 구분해 어떤 객체 클래스인지 판 단하므로 검출보다 더 자세하게 위치를 표시한다.



(그림 1) Object detection과 segmentation  
(Figure 1) Object detection and segmentation

그림 1은 객체검출과 분할을 설명하고 있다. 분할은 객체를 구분한 뒤 각각의 객체가 어떤 클래스인지 판단 하는 시맨틱 분할(Semantic Segmentation)과 같은 클래스 이더라도 다른 인스턴스라면 이를 구분하는 인스턴스 분 할(Instance Segmentation)로 나눌 수 있다. 시맨틱 분할은 서로 다른 종류의 객체들을 클래스별로 분할해내는 것이 목표로서 FCN, DeepLab 등이 대표적인 모델이며, 인스턴 스 분할은 동일한 클래스라도 서로 다른 인스턴스에 해 당하면 이들을 예측 마스크 상에 다른 색상으로 표시하 며 DeepMask, Mask R-CNN 등이 대표적인 모델이다[2, 3].

본 논문에서는 자율주행 연구에서 활발하게 사용되는 딥러닝 객체 인식 모델들 중 가장 최근에 발표된 DeepLab V3+[2] 모델과 Mask R-CNN[3] 모델을 활용하여 주행가능 영역을 추출하여 두 모델의 결과를 비교하고 분석하고자 한다.

### 2.1 DeepLab V3+

시맨틱 분할모델을 활용한 다양한 연구들이 활발히 진 행되고 있으며 최근에는 자율주행의 성능을 높이기 위해 여러 가지 DeepLab 모델들이 사용되고 있다. DeepLab V3+[2]은 시맨틱 분할에서 높은 성능을 보이는 방법 중 하나로 최근 오픈소스[5]로 공개가 되어 많은 사람들의 주목을 받은 모델이다. 초기 DeepLab 은 Deep Convolutional Nets와 Fully Connected CRFs를 사용하여 분할 을 하는 방법을 제안한 모델이다. 그러나 Feature Resolution 이 줄어드는 현상과 이미지의 분할을 정확히 얻지 못하는 문제점을 가지고 있었으며, 이를 해결한 DeepLab V2 가 발표되었다. 이후 기존 ResNet 구조에 Atrous Convolution을 활용해 좀 더 조밀한 특징 맵을 얻는 방법 을 제안한 Deep Lab V3가 발표되었다. 나아가 Atrous Separable Convolution을 활용한 Deep Lab V3+ 순서로 발 전하면서 DeepLab 모델은 시맨틱 분할 분야에서 가장 수 준 높은 성능을 보여주는 모델로 발전하였다[2].

### 2.2 Mask R-CNN

Mask R-CNN[3]은 객체검출과 분할, 두 기능을 모두 구현한 모델로 R-CNN의 계보를 잇는 모델이다. 초기 R-CNN은 이미지 분류를 수행하는 회전신경망(Convolutional Neural Network)과 이미지에서 물체가 존재할 영역 을 제안해주는 영역제안(Region Proposal)알고리즘을 연 결하여 높은 성능의 객체검출을 수행할 수 있음을 보여

준 모델이다. 그러나 초기 R-CNN은 시간이 오래 걸린다는 문제점을 가지고 있었으며, 이를 해결하기 위해 Region of Image Pooling 개념을 도입하고 모든 모델을 하나의 네트워크로 구성함으로써 시간과 비용 문제를 해결한 Fast R-CNN이 발표되었다. 또한 모델 내부에 Network 설계를 제안함으로써 더욱 빠르면서도 정확한 영역제안을 생성해 시간을 절약한 Faster R-CNN도 등장하였다. 이후, 픽셀 수준의 분할이 가능하도록 하기 위해 Faster R-CNN에서 Fully Convolutional Network를 CNN Feature 상단에 놓아 마스크를 생성하는 Branch를 추가하는 방법인 Mask R-CNN으로 발전하면서 객체검출 뿐만 아니라 분할까지 가능하게 되었다[3].

### 3. 제안하는 방법

본 연구에서는 실제 도로주행을 위한 판단에 도움을 주고자 다양한 기후조건을 비롯한 여러 가지 환경조건 하에서의 주행가능 영역 정보가 담겨있는 데이터셋을 활용하여 DeepLab V3+와 Mask R-CNN을 학습시킨 사례를 개발하고자 한다. 이를 위해 BDD 데이터셋[4]을 사용하였는데 DeepLab V3+와 Mask R-CNN의 논문에서는 BDD 데이터셋으로 실험한 사례가 없으므로 본 실험의 결과를 이들 논문의 결과와 비교하면서 동시에 두 모델의 성능 또한 서로 비교, 분석하고자 한다.

이 데이터셋은 BDD100K로 불리는 운전 데이터베이스로 UC 버클리 인공 지능 연구 실험실에서 공개했다[4]. BDD100K는 Berkeley Deep Drive의 약자로, 1280x720 크기의 이미지로 이루어져 있으며, 7만장의 학습 이미지와 1만장의 검증 이미지 및 2만장의 테스트 이미지로 구성되어 있다. 총 10만개의 이미지로 이루어진 이 데이터셋은 버스, 신호등, 교통 표지, 사람, 자동차 등을 2D 경계 박스로 표현하여 이미지에 주석으로 처리한 것과 주행 영역을 그림 2와 같이 영상분할 및 폴리곤 형태로 표현한 정보들이 포함되어 있다.



(그림 2) BDD 데이터셋 주행가능 영역  
(Figure 2) BDD dataset Drivable Areas

또 비 내리는 날씨, 흐린 날씨, 맑은 날씨, 안개와 같은 다양한 날씨 조건이 기록되어 있다. 데이터셋 및 영상 분할 라벨과 주석 정보는 BDD 홈페이지[6]에서 다운로드 가능하다.

본 실험에서는 주행가능 영역 정보가 담겨있는 BDD 데이터셋을 DeepLab V3+와 Mask R-CNN의 입력 이미지로 활용, 학습과 검증을 진행함으로써 주행가능한 영역을 추출해주는 딥러닝 모델을 구현하고, 각 모델의 인식 결과와 정확도를 비교하고자 한다. 또한 실험을 통해 얻은 가장 성능이 우수한 학습 가중치를 활용하여 테스트한 후, 주행가능 영역을 추출하여 그 결과를 육안으로 검사한다. 이를 통해 각 모델에서 잘못된 분류를 하는 영역을 구체적으로 확인해본다. 제안하는 방법의 대략적인 알고리즘은 표 1과 같다.

(표 1) 제안하는 주행가능 지역 추출 모델 알고리즘  
(Table 1) Proposed drivable area segmentation algorithm

1. 주행가능 영역 정보가 담긴 이미지(1280x720)를 input image로 활용해 convolution 및 pooling 수행
2. 학습을 통해 (배경, 주행가능 영역)으로 분류된 정보가 담긴 weights 획득
3. 획득한 weights로 IoU 측정
4. 측정된 IoU를 이전 IoU와 비교하여 더 높은 IoU를 얻도록 필터 값 수정
5. 1 ~ 4번 항목을 반복하여 가장 성능이 우수한 weights로 테스트

## 4. 실험

### 4.1 실험 환경

실제 학습과 테스트가 진행된 컴퓨터의 사양과 소프트웨어 환경은 표 2와 같다. 딥러닝에서 학습속도를 높이기 위한 아키텍처이자 소프트웨어인 CUDA 9.0과 cuDNN 7.5.0을 이용하여 GPU 연산을 수행하였다.

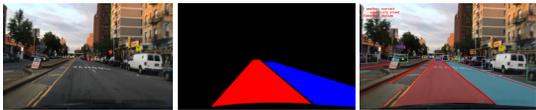
### 4.2 실험 과정

실험은 각 모델의 학습 가이드를 참고하여 진행하였으며, DeepLab V3+의 정답 정보는 그림 3의 <Segmentation>과 같이 주행가능 영역을 분할한 이미지를, Mask R-CNN의 정답 정보는 그림 3의 <Labeled image>와 같이 주행가능 영역의 위치를 이용해 라벨링한 이미지를 사용하여 학습하였다.

(표 2) 실험 환경

(Table 2) Experimental conditions

CPU	Intel(R) Core(TM) i7-6700 CPU @ 3.40 GHz
RAM	64 GB
SSD	250 GB
GPU	NVIDIA GeForce GTX 1080
OS	Ubuntu 16.04.1 LTS (4.4.0-71-generic)
SW	CUDA 9.0 & cuDNN 7.5.0
	TensorFlow 1.12.0
	Python 3.6



(Original Image) (Segmentation) (Labeled Image)  
 (그림 3) BDD 데이터셋: 원본 이미지, 영상 분할, 라벨링된 이미지  
 (Figure 3) BDD dataset: original image, segmentation, labeled image

#### 4.2.1 DeepLab V3+ 학습 방법

DeepLab V3+는 PASCAL VOC 2012 데이터셋과 Cityscapes 데이터셋, ADE20K 데이터셋을 사용해 학습할 수 있는 가이드를 제공한다[2]. 본 실험에서 사용하는 BDD 데이터셋에 대한 학습 가이드는 제공되지 않기 때문에 가이드를 제공하는 모델 중 하나를 선택하여 BDD 데이터셋의 구성을 그와 유사하게 변경해야 한다. 본 실험에서는 BDD 데이터셋을 주행가능 영역 정보가 없는 원본 이미지, 주행가능 영역 정보가 색상으로 분할된 이미지와 흑백으로 분할된 이미지, 학습 이미지 및 검증 이미지의 이름을 저장한 텍스트 파일로 PASCAL VOC 2012 데이터셋[7]의 형태처럼 재구성하여 학습 가능하도록 변경하였다. 또한 학습 데이터 7만장, 검증 데이터 1만장 총 8만장의 데이터로 배경과 주행가능 영역 두 클래스로 분류한 train 및 validation의 pre-trained weights를 만들어 효율적인 학습을 수행하였다.

학습 횟수는 DeepLab V3+로 다른 데이터셋을 사용했을 때[2]와 동일하게 총 3만번 진행하였으며, crop size는 데이터셋의 모든 이미지의 크기가 1280x720으로 동일하기 때문에 1281x721로 수정하여 학습을 진행하였다. 신경망은 Xception을 사용하였으며, 실험 환경의 자원을 고려하여 learning rate를 1/10 줄여 1e-5로 학습을 진행하였다. 이외의 학습 옵션은 다른 데이터셋의 경우와 동일하게 하였다.

#### 4.2.2 Mask R-CNN 학습 방법

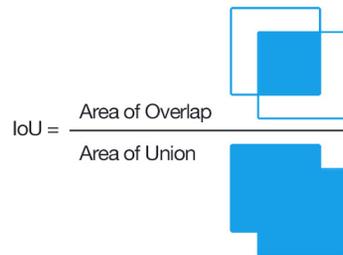
Mask R-CNN은 MS COCO 데이터셋을 사용해 학습할 수 있는 가이드를 제공한다[3]. 실험에서 사용하는 BDD 데이터셋에 대한 학습 가이드는 제공하지 않았으나 Mask R-CNN에서 MS COCO 데이터셋[8]을 활용하는 경우에 대한 가이드를 제공하기 때문에 그 정보를 참고하여 학습 구동 파일을 만들어 사용하였다.

BDD 데이터셋에서 각 이미지의 정보는 json 형태의 파일로 저장되어 있으며, 데이터셋에서 제공하는 검출 객체의 정보와 주행가능 영역에 대한 정보를 포함하고 있다. 원본 이미지와 정보를 포함한 json 파일을 Mask R-CNN의 입력 데이터로 사용할 수 있으며, 학습시 객체의 카테고리 json 파일의 “drivable area”로 표시되어 있는 정보만 추출하여 주행가능 영역만을 사용해 학습할 수 있도록 학습 파일을 수정하여 사용하였다.

학습 횟수는 1step 당 100번 씩 200epochs를 수행하여 총 2만번 진행하였으며, 모델의 신경망을 ResNet50과 ResNet101 두 가지 경우 모두 실험함으로써 신경망의 깊이 따른 성능을 비교하였다. 이외에도 주행가능 영역 추출 시 잘못된 결과가 미치는 영향의 위험성을 고려하여 90% 이상의 Confidence만 채택하도록 하였다.

#### 4.3 실험 결과 및 분석

실험에서 모델의 훈련 정확도를 검사할 때는 객체 검출에서 예측된 경계 상자의 정확도를 평가하는 지표 중 하나로 사용되는 Intersection over Union(IoU)[9]이라는 기준을 사용하였다. IoU는 예측된 경계 상자와 실제 경계 상자가 서로 겹치는 비율이며 이에 대한 개념은 그림 4로 설명된다. 그림에서 보는 것처럼 예측과 실제 경계 상자의 겹치는 면적의 비율이 높을수록 정확한 검출이 이루어진 것이 된다.



(그림 4) Intersection over Union 설명  
 (Figure 4) Explanation of Intersection over Union

실험의 성능 비교는 각 모델의 논문[2, 3]에서 가장 성능이 좋았던 결과의 IoU와 BDD 데이터셋을 이용해 학습한 결과의 IoU를 비교하여 확인하였으며, 이후 다양한 학습 방법을 통해 도출된 결과의 성능을 비교하여 가장 좋은 성능을 가진 학습 가중치를 획득하는 것을 목표로 하였다. 또한 획득한 학습 가중치를 이용하여 실제 차량이 주행 중인 이미지에서 주행가능 영역을 추출해 일반 데이터에서도 효과적인지 확인하였다.

#### 4.3.1 DeepLab V3+ 실험 결과

DeepLab V3+가 주행가능 영역을 추출을 하는데 얼마나 높은 성능을 보여주는지 확인하기 위해 모델의 논문 [2]에서 가장 성능이 좋았던 결과와 비교하였다. 실험 결과 비교는 표 3과 같다.

(표 3) DeepLab V3+: PASCAL VOC 2012와 BDD 데이터셋 실험 결과 비교

(Table 3) DeepLab V3+: Compare PASCAL VOC 2012 with BDD dataset

Datasets	Loss	Time	IoU
PASCAL VOC 2012	0.6577	349ms/step	0.8741
BDD Dataset	0.4382	1,118ms/step	0.4897

BDD 데이터셋으로 학습을 수행하였을 때, DeepLab V3+의 논문[2]에서 가장 성능이 좋았던 PASCAL VOD 2012 데이터셋[8]으로 학습한 결과와 비교하여 Loss는 0.2195 더 낮았으며, 1 step에 대한 소요 시간은 약 3배가량 더 오래 걸렸고, IoU가 38.44% 더 낮은 수치를 보여 정확도가 떨어진 것을 확인할 수 있었다. 이는 PASCAL 데이터셋의 클래스가 21개인 것에 비해 우리가 학습에 사용한 BDD 데이터셋의 클래스는 배경과 주행가능 영역 2개로 이루어져 손실이 낮게 나온 것으로 보이며, 데이터셋의 학습 및 검증 이미지의 합이 각각 11,530장과 80,000장으로 BDD 데이터셋의 이미지 개수가 많아 소요 시간이 오래 걸린 것으로 보인다. 또한 DeepLab V3+가 객체에 대한 영상 분할 분류는 잘 수행하나, 주행가능 영역을 분류하는 능력은 비교적 좋지 않은 것을 확인할 수 있었다.

#### 4.3.2 Mask R-CNN 실험 결과

Mask R-CNN의 실험은 모델의 기본 학습 설정인 이미

지 크기의 최소 사이즈를 800, 이미지 크기의 최대 사이즈를 1024로 변경하였고, Detection Confidence가 0.9 이상만 채택하도록 수치를 높여서 학습하였다. 또한, Mask R-CNN이 주행이 가능한 영역을 추출하는데 얼마나 높은 성능을 보여주는지 확인하기 위해 모델의 논문[3]에서 가장 성능이 좋았던 결과와 비교하였다. 실험 결과 비교는 표 4와 같다.

(표 4) Mask R-CNN: COCO와 BDD 데이터셋 실험 결과 비교

(Table 4) Mask R-CNN: Compare COCO with BDD dataset

Datasets	Layers	Loss	Time	IoU
COCO	heads	1.1820	536ms/step	0.6563
	4+	1.1020	696ms/step	
	all	0.6098	818ms/step	
BDD Dataset	heads	1.1796	498ms./step	0.6833
	4+	0.7375	661ms/step	
	all	0.4372	778ms/step	

BDD 데이터셋으로 학습을 수행하였을 때, Mask R-CNN의 논문[3]에서 가장 성능이 좋았던 MS COCO 데이터셋[9]으로 학습한 결과와 비교하여 Loss는 각 Layer 별로 0.0024, 0.3645, 0.1726 더 낮았으며, 1 step에 대한 소요 시간은 약 1.07배 가량 더 빨랐고, IoU가 2.7% 더 높은 수치를 보여 정확도가 더 우수한 것을 확인할 수 있었다. 이는 COCO 데이터셋의 클래스가 81개인 것에 비해 우리가 학습에 사용한 BDD 데이터셋의 클래스는 배경과 주행가능 영역 2개로 이루어져 손실이 낮게 나온 것으로 보이며, 데이터셋의 학습 및 검증 이미지의 합이 각각 123,287장과 80,000장으로 BDD 데이터셋의 이미지 개수가 COCO 데이터셋의 경우보다 더 적어 소요 시간이 적게 걸린 것으로 보인다. 또한 Mask R-CNN은 객체에 대한 검출 능력뿐만 아니라 주행가능한 영역을 검출하는 능력도 비교적 좋은 성능을 보인다는 것을 확인할 수 있었다.

#### 4.3.3 테스트 이미지를 이용한 육안 검사

각 모델에서 실험을 통해 성능이 가장 우수한 학습의 가중치를 얻은 후 BDD 데이터셋의 테스트 이미지를 활용하여 주행가능 영역을 추출한 결과를 육안으로 검사하였다. 추출된 영역에서 실제로 차량이 주행할 때 사고가 발생할 확률이 0인 경우에만 주행가능 영역이 올바르게

추출되었다고 판단하였다. 테스트셋은 BDD 데이터셋의 테스트 이미지에서 임의로 100장을 뽑아 구성하였으며 도시, 고속도로, 교차로, 커브길 및 주간, 야간, 우천, 눈 등 어려가지 주행 환경을 포함하도록 구성하였다.

Mask R-CNN 및 DeepLab V3+의 최선의 학습 가중치로 테스트셋 이미지를 테스트한 결과는 표 5와 같다.

(표 5) Mask R-CNN 및 DeepLab V3+ 테스트 결과  
(Table 5) Test result of Mask R-CNN: and DeepLab V3+

	Mask R-CNN	DeepLab V3+
True (TP)	83	69
False (F:P)	17	31
Precision	83%	69%

총 100개의 이미지 중에 Mask R-CNN은 83개의 테스트 이미지에서 올바르게 주행가능 영역을 추출하였고, 17개의 테스트 이미지에서 잘못된 주행가능 영역을 추출하였다. DeepLab V3+의 경우에는 69개의 테스트 이미지에서 올바르게 주행가능 영역을 추출하였으며, 31개의 테스트 이미지에서 잘못된 주행가능 영역을 추출하였다. 또한 각 모델의 테스트 결과에서 잘못 추출된 영역을 확인하여 각 모델의 취약점을 확인하였으며, 표 6에 그 결과를 정리하였다.

(표 6) 각 모델의 추출 실패 원인 정보  
(Table 6) Fault segmentation for drivable area

	Mask R-CNN	DeepLab V3+
Opposite Side	5	20
Shoulder	8	4
Parking Area	3	1
Center Line	1	1
No Segmentation	0	5
Total Fault	17	31

Mask R-CNN은 인식에 실패한 17개 중 반대편 차선을 인식한 이미지가 5개, 갓길을 인식한 이미지가 8개, 주차 지역을 인식한 이미지가 3개, 중앙선을 인식한 이미지가 1개로 나타났다. DeepLab V3+의 경우에는 실패한 총 31개의 이미지 중 반대편 차선을 인식한 이미지가 20개, 갓길을 인식한 이미지가 4개, 주차 지역을 인식한 이미지가 1개, 중앙선을 인식한 이미지가 1개, 주행가능 영역 추출에 실패한 이미지가 5개로 나타났다.

Mask R-CNN과 DeepLab V3+의 잘못 추출된 영역의 정보를 확인한 결과, Mask R-CNN은 위치 정보를 포함하여 학습하기 때문에 비교적 추출 실패가 적은 것으로 해석할 수 있으며, 갓길이나 주차 지역, 중앙선의 너비가 넓은 경우나 중앙선의 표시가 불확실한 경우에 모델이 취약한 것을 확인하였다. 이에 비해 DeepLab V3+는 하나의 클래스는 같은 색상으로 분할하는 시멘틱 분할의 특징 때문에 반대편 차선의 주행가능 영역을 함께 추출하는 것이 가장 큰 취약점으로 보이며, 이외의 추출 실패는 Mask R-CNN의 실패 이유와 비슷한 것으로 확인하였다.

## 5. 결론 및 향후 연구과제

본 논문에서는 영상 분할 모델 중에서 가장 유명하며 최근까지 연구가 진행되고 있고, 자율주행과 같이 이미지에서 객체의 위치 정보가 필요한 연구 분야에서 각광 받고 있는 DeepLab V3+ 및 Mask R-CNN 모델에 BDD 데이터셋을 사용해 주행이 가능한 영역을 추출하는 모델을 구현하였다.

BDD 데이터셋을 활용한 모델의 훈련 정확도를 검사하기 위해 IoU를 기준으로 각 모델의 정확도를 평가하였다. DeepLab V3+에서는 기존 데이터셋인 PASCAL VOC 2012로 훈련한 경우보다 BDD 데이터셋을 사용했을 때 정확도가 많이 떨어졌으며, Mask R-CNN에서는 기존 데이터셋인 MS COCO로 훈련한 경우보다 정확도가 올라간 것을 확인할 수 있었다.

학습한 모델이 잘 작동하는지 확인하기 위해 각 모델에서 가장 우수한 결과를 낸 학습 가중치를 이용해 테스트를 진행하였으며, 테스트 이미지 100장을 육안으로 확인한 결과 Deep Lab V3+의 정확도는 69%, Mask R-CNN의 정확도는 83%로 Mask R-CNN이 테스트에서도 우수한 성능을 보여주었다.

향후 연구 과제로는 Mask R-CNN을 통해 구현할 수 있는 객체 검출 기능을 활용하여 주행가능한 영역을 추출할 뿐만 아니라 BDD 데이터셋에서 제공하는 10가지 클래스에 대한 객체들을 검출하여 주행가능 영역과 함께 결과로 제공함으로써 주행 중인 환경에서 얻을 수 있는 정보의 질과 양을 확대할 계획이다. 또한, 실시간으로 객체 검출 기능을 처리함으로써 실시간 영상에서 검출된 객체가 어떤 물체인지 분류하고, 검출한 객체의 움직임과 속도, 방향 등에 대한 정보를 제공하는 모델을 구현함으로써 자율주행 연구 분야에 기여하고자 한다.

## 참고문헌(Reference)

- [1] G. B. Seeong, G. W. Min, J. D. Choi, "Autonomous car technology trends and key technologies", KICS Information & Communication Magazine - Open Lecture Series 35, 2018.5, 3-13(11 pages), 2018.  
<http://www.dbpia.co.kr/journal/articleDetail?nodeId=NO D E07456307>
- [2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation", arXiv: 1802.02611, 2018.  
<https://arxiv.org/abs/1802.02611>
- [3] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN", IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, pages 2980 - 2988, 2017.  
<https://arxiv.org/abs/1703.06870>
- [4] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "BDD100K: A Diverse Driving Video Database with Scalable Annotation Tooling", arXiv:1805.04687, 2018.  
<https://arxiv.org/abs/1805.04687>
- [5] <https://github.com/tensorflow/models/tree/master/research/deeplab>
- [6] <https://bdd-data.berkeley.edu/>
- [7] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserma, "The Pascal Visual Object Classes Challenge - A Retrospective", IJCV, January 2015, Volume 111, Issue 1, pp 98 - 136, 2014.  
<http://host.robots.ox.ac.uk/pascal/VOC/voc2012/>
- [8] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, Piotr Dollár, "Microsoft COCO: Common Objects in Context", arXiv:1405.0312, 2014.  
<http://cocodataset.org/>
- [9] Intersection over Union (IoU) for object detection  
<https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>

## ◎ 저 자 소개 ◎



### 전 호 진(Hyo-jin Jeon)

2018년 한국교통대학교 컴퓨터정보공학과(공학사)  
 2018년~현재 한국교통대학교 컴퓨터정보공학과 석사과정  
 관심분야 : 물체인식, 딥러닝, 빅데이터, 인공지능  
 E-mail : hyodg@naver.com



### 조 수 선(Soo-sun Cho)

1987년 서울대학교 계산통계학과(이학사)  
 1989년 서울대학교 대학원 계산통계학과(이학석사)  
 2004년 충남대학교 대학원 컴퓨터과학과(이학박사)  
 1994년~2004년 한국전자통신연구원 선임연구원  
 2006년~2007년 7월 미시간대학교(엔아버) 통계학과 방문연구원  
 2004년~현재 한국교통대학교 컴퓨터정보공학과 교수  
 관심분야 : 데이터 마이닝, 빅데이터 분석, 인공지능  
 E-mail : sscho@ut.ac.kr