

VGG-13 기반의 경량화된 딥러닝 기법을 이용한 차선 이탈 경고 시스템 구현

강 현 우[†]

Implementation of Lane Departure Warning System using Lightweight Deep Learning based on VGG-13

Hyunwoo Kang[†]

ABSTRACT

Lane detection is important technology for implementing ADAS or autonomous driving. Although edge detection has been typically used for the lane detection however, false detections occur frequently. To improve this problem, a deep learning based lane detection algorithm is proposed in this paper. This algorithm is mounted on an ARM-based embedded system to implement a LDW(lane departure warning). Since the embedded environment lacks computing power, the VGG-11, a lightweight model based on VGG-13, has been proposed. In order to evaluate the performance of the LDW, the test was conducted according to the test scenario of NHTSA.

Key words: Lane Detection, Adas, Image Segmentation, Lane Departure Warning,

1. 서 론

최근 출시되는 대부분의 자동차에는 운전자에게 안전성과 편의성을 제공하기 위하여 능동형 운전자 보조 시스템(ADAS, Advance Driving Assist System)이 탑재되어 있다. 능동형 운전자 보조 시스템의 대표적인 기능으로는 차선이탈경보(LDW, Lane Departure Warning), 전방충돌경보(FCW, Forward Collision Warning), 주차보조시스템(PAS, Parking Assist System) 등이 있으며, 이러한 ADAS 기술을 바탕으로 자율 주행과 관련된 연구도 활발하게 진행되고 있다[1-3]. 자율 주행 및 능동형 운전자 보조 시스템을 구현하기 위해서 Radar, Lidar, Ultrasonic, 비전 센서 등 다양한 센서들이 활용되고 있으며[4], 특히 차량 주행에 있어서 중요한 정보인 차선을 검출

하기 위해서는 비전 센서가 반드시 필요하다[5].

비전 센서 기반의 차선 검출과 관련된 많은 연구에서 차선을 검출하기 위하여 영상처리 기법으로 에지(Edge) 영상을 사용하고 있다[6,7,18]. 에지 영상을 이용한 방법은 고전적인 방법으로 그동안 많이 사용되었지만, 도로 환경에서 주로 나타나는 가드레일, 연석 등은 차선과 유사하게 에지가 두드러지기 때문에 오검출이 많이 발생하는 단점이 있다. 이러한 오검출을 줄이기 위한 많은 시도가 있었으며, 최근 영상 내 객체 검출 및 인식 분야에서 딥러닝 알고리즘이 크게 성과를 냄에 따라, 차선 검출 분야에서도 딥러닝을 활용한 연구들이 활발하게 진행되고 있다[8, 9] 이들 연구에서는 영상 내에서 차선이 존재하는 영역, 또는 각각의 픽셀에 대하여 차선임을 표기(labeling)하여 딥러닝 알고리즘으로 학습을 하는 방

* Corresponding Author: Hyunwoo Kang, Address: (41765) 15, Gukchaebosang-ro 43-gil, Seo-gu, Daegu, Korea, TEL : +82-53-560-3188, FAX : +82-53-560-3253, E-mail : hwkang@kopo.ac.kr

Receipt date : Jun. 14, 2021, Revision date : Jun. 25, 2021
Approval date : Jul. 12, 2021

[†] Dept. of AI Engineering, Korea Polytechnics

식을 취하였다. 이러한 기법을 통하여 기존의 에지 기반의 차선 검출 방식에 비하여 오검출을 줄이는 성과를 얻었다.

영상을 통해 인식된 차선의 정보는 능동형 운전자 보조 시스템에서 차량이 차선을 유지하는데 활용되거나(LKA, Lane Keeping Assist), 차량이 차선을 이탈할 경우 운전자에게 경보를 주는 시스템(LDW) 등으로 활용이 가능하다. 본 논문에서는 딥러닝 기반의 차선 검출 알고리즘을 제안하여, 에지 기반 차선 검출 대비 오검출률을 약 9.4% 정도 감소시켰으며, 이를 활용하여 임베디드 환경에서 차선 이탈 경고 시스템을 구현하였다. 특히, 컴퓨팅 파워가 부족한 임베디드 시스템에 적용하기 위하여 딥러닝 모델의 weight parameter를 줄이는 sparsity 기법[10]을 적용하였다.

본 논문에서 제안한 차선 검출 시스템은 TI사(社)의 TDA2x 임베디드 보드에 구현하였으며, 실험 결과 전체 시스템의 평균 처리 속도는 약 12.83fps이었다. 성능 평가를 위해 미국도로교통안전국(NHTSA [11])의 테스트 시나리오를 기반으로 테스트를 진행하였으며 그 결과 약 96%의 정확도를 보였다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문에서 구현한 전체 시스템의 구성에 대해서 설명하고, 3장에서는 제안하는 네트워크의 구조와 차선 이탈의 판단 알고리즘에 대하여 설명한다. 그리고 4장에서는 구현된 시스템의 성능 평가를 위한 실험 결과를 기술하고 마지막으로 5장에서 결론을 맺는다.

2. 시스템 구성

본 논문에서 제안한 시스템은 TI사의 TDA2x에

구현하였으며, Fig. 1은 각각의 코어에서 담당하는 알고리즘의 구성과 처리 순서를 보여주고 있다. TDA2x는 영상 병렬처리를 위한 4개의 EVE(Embedded Vision Engine)와 신호처리를 위한 2개의 DSP, 전체 시스템을 담당하는 두 개의 코어(ARM Cortex-15), 카메라 및 CAN 입출력을 담당하는 2개의 코어(Cortex-M4)로 구성되어 있다. 2개의 EVE를 사용하여 차선 검출을 위한 세그멘테이션 딥러닝 모델이 실행되며, ARM 코어에서 주요 차선(Ego Lane) 결정이 이루어진다. 차선 검출에 사용되지 않는 2개의 EVE에서는 차량 및 보행자 검출을 위한 SSD(Single Shot multibox Detector)[12] 알고리즘 기반의 객체 검출 모델이 동작하며, DSP에서 검출한 객체를 추적하기 위한 알고리즘이 동작한다.

카메라로부터 영상이 획득되면 전처리를 통해 딥러닝 모듈에 적합한 형태로 변환되어 Lane Segment 모듈(Fig. 1의 EVE 3, 4 모듈)에 입력된다. 검출된 차선 정보가 Ego & Lane fitting 모듈(Fig. 1의 A15 1 모듈)로 전달되면, Ego & Lane fitting 모듈은 세그멘테이션 된 정보를 바탕으로 차선을 모델링하고 Ego-Lane을 판별하게 된다. 각 모듈은 큐(Queue)로 연결되어 있어, 한 모듈의 결과를 큐에 저장하는 형태로 처리하기 때문에 독립적인 운영이 가능하다.

3. 제안한 방법

3.1 Pre-trained 모델

본 논문에서 차선 세그멘테이션 학습을 위해 사용한 pre-trained 모델은 VGG-13[13]을 기반으로 한다. 1000개의 class를 분류하기 위해 사용된 VGG-13

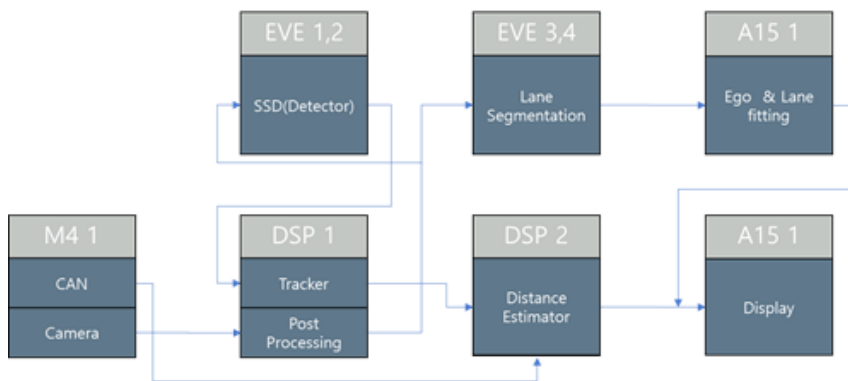


Fig. 1. Overview of Algorithms Assigned in Each Core.

의 경우 네트워크의 깊이가 깊어질 수 밖에 없으므로, 자원이 한정된 임베디드 보드 상에서 동작시키기에는 한계가 있다. 따라서, VGG-13의 구조를 Fig. 2와 같이 변형하여 사용하였다. VGG-13에 대하여 전체 레이어의 수를 2개 줄여서 학습하였으므로, 본 논문에서 사용한 pre-trained 모델은 VGG-11로 명명하였다.

VGG-13 모델은 ImageNet 데이터베이스를 사용하여 학습된 모델이므로, ImageNet 데이터 베이스에서 도로와 차량 환경에 적합한 175개의 class를 선정하여 별도의 데이터베이스를 제작하였다. 또한, 임베디드 환경에서 실시간 처리가 가능하도록 stride의 크기를 VGG-13 보다 크게 적용하였으며 모든 Layer에서 Convolution 필터의 채널 수를 줄여서 사용하였다. 제안한 데이터베이스를 사용하여 VGG-11 모델을 학습한 결과 VGG-13 모델에 비하여 학습 데이터에 대한 인식률에서 약 1.75%의 성능 하락이 있었으나, 속도 면에서 많은 성능 향상을 가져올 수 있었다.

3.2 차선 세그멘테이션

본 논문에서 제안한 차선 검출 알고리즘은 전처리 및 차선 검출 단계로 나누었으며, 전처리 단계는 영

| VGG-11 (ours) | VGG-13 |
|---------------------------------|-------------------------------|
| Input Image (224x224 RGB image) | |
| Conv3-16-s2 Conv3-16 | Conv3-64-s2 Conv3-64 |
| Max pool | |
| Conv3-32 Conv3-32 | Conv3-128 Conv3-128 |
| Max pool | |
| Conv3-64 Conv3-64 | Conv3-256 Conv3-256 |
| Max pool | |
| Conv3-128 Conv3-128 | Conv3-512 Conv3-512 |
| Max pool | |
| Conv3-256 Conv3-256 | Conv3-512 Conv3-512 |
| Average pool | Max pool |
| FC-175 | FC-4096 FC-4096 FC-1000 |
| Soft-max | |

Fig. 2. Difference between VGG-13 and proposed VGG-11,

상 Resize 모듈과 딥러닝을 이용한 세그멘테이션 영상 생성 모듈로 구성되어 있다. 그리고 차선 검출 단계는 Ego-lane 결정, 차선 Fitting 알고리즘으로 구성되어 있다. Fig. 3은 본 논문에서 제안한 차선 검출 알고리즘에 대한 개요도를 나타낸 것이다.

비전 센서를 이용한 차선 검출과 관련된 연구에서는 전처리 과정에서 canny edge detection와 같은 알고리즘으로 에지를 검출한다. 하지만 차선을 포함한 모든 에지 정보가 검출되므로 잡음에 취약한 단점이 있다. 이를 해결하기 위하여 본 논문에서는 전처리 단계로 딥러닝 기술을 이용한 세그멘테이션 영상을 사용하였다.

세그멘테이션 모듈의 결과는 영상 내에서 차선은 1, 나머지 부분은 0으로 표시되는 이진 영상을 만들어 내는 것이다. 따라서, 영상의 각 픽셀에 대하여 차선은 1, 나머지 부분은 0으로 표시된 7660장의 영상을 사용하였으며, VGG Image Annotation 도구를 활용하여 직접 데이터베이스를 구축하였다. 학습을 위하여 3.1 장에서 기술한 VGG-11 pre-trained 모델을 활용하여 세그멘테이션 모델을 구현하였다. 그런데, 학습 영상의 크기가 작을수록 원거리의 차선을 검출하는데 용이한 장점이 있으나, 근거리 차선에서 미검출 현상이 발생한다. 따라서, 본 논문에서는 pre-train된 VGG-11의 파라미터값을 초기값으로 사용하되, 입력 영상의 크기를 320x320로 변경하여 세그멘테이션 모듈을 재학습하였다. Adam optimizer를 사용하였으며, Learning rate는 10^{-4} 을 적용

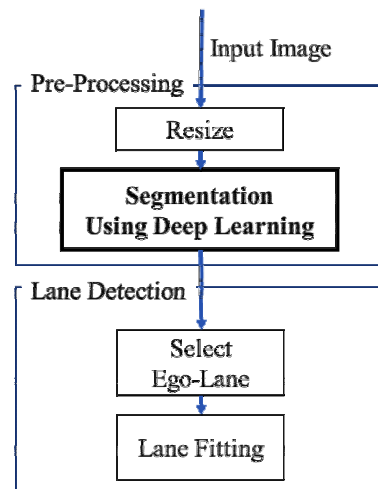


Fig. 3. Overview of Lane Detection Algorithm,

하였다.

Fig. 4는 사용한 세그멘테이션 모델의 전체 구조를 보여준다. VGG-11 모델을 통과하면 10x10 크기로 영상이 압축된다. 시그멘테이션은 입력 영상에서 차선과 차선이 아닌 부분을 표현해야 하므로 줄어든 크기만큼 업 샘플링을 해야 한다. 따라서 업 샘플링을 할 때마다 Convolution 필터들이 학습되어 최종 출력력이 320 × 320이 되도록 네트워크를 구성하도록 하였다. 마지막으로 softmax 계층을 추가하여 각 픽셀이 차선인지 아닌지를 확률값으로 출력하도록 하였다.

획득된 영상이 세그멘테이션 모듈을 거치게 되면 차선 부분이 1이고 다른 부분이 0인 이진 영상이 생성된다. Fig. 5는 Canny 에지 검출기를 사용한 결과와 제안한 세그멘테이션 모듈의 결과를 비교한 그림이다. Canny 에지 검출의 실행 결과는 노이즈를 포함하여 결과를 반환하지만 제안한 세그멘테이션 모듈을 사용한 결과는 차선의 정보만 존재하는 것을 확인할 수 있다. 또한 원거리의 차선이 보이지 않거나 점선 차선일 경우에도 제안한 방법의 성능이 좋은 것을 확인할 수 있다.

3.3 차선 검출

3.2장의 차선 세그멘테이션 과정을 통하여 이진화

영상이 생성되면, 이진화 영상으로부터 ego-lane을 결정한다. 우선 영상에서 연결된 픽셀들을 하나의 그룹으로 만들어주기 위해서 연결 요소 분석(Connected Component Analysis) 방법을 사용한다. 연결 요소 분석 알고리즘의 대표적인 방식으로는 Grass-fire 알고리즘[14]과 two-pass labeling 알고리즘[15] 등이 있다. 연결 요소 분석의 정확도는 중요한 요인이지만, 임베디드 환경에서 동작시킬 경우 메모리 관리, 처리 속도 면에서 문제가 발생한다. 본 논문에서 제안한 시스템에서는 적은 컴퓨팅 파워 환경에서 ego-lane을 빠르게 결정하는데 목적이 있으므로, 알려진 방법들을 변형하여 휴리스틱(heuristic) 방법을 제안한다. Fig. 6은 제안한 차선 데이터 labeling 방법이다.

Fig. 6(a)는 영상의 검색 방향을 나타내며 처음 검색 위치는 영상의 왼쪽 가장 윗부분이다. 검색 위치에서 Fig. 6(b)와 같이 검색 방향에 인접한 픽셀들을 우선적으로 선택한다. Fig. 6(c)는 선택된 픽셀을 기준으로 중복되지 않는 8방향 인접 픽셀들을 검사한다. 이 때 세그멘테이션 결과 중 labeling이 되지 않은 픽셀에 대해서 Fig. 6(d)와 같이 표시를 하며 만약 labeling 이 되어 있는 픽셀에 대해서는 labeling을 다시 표시하지 않는다. 기존의 Two pass labeling 알고리즘에서는 모든 픽셀에 대해서 8방향 검사를 하지

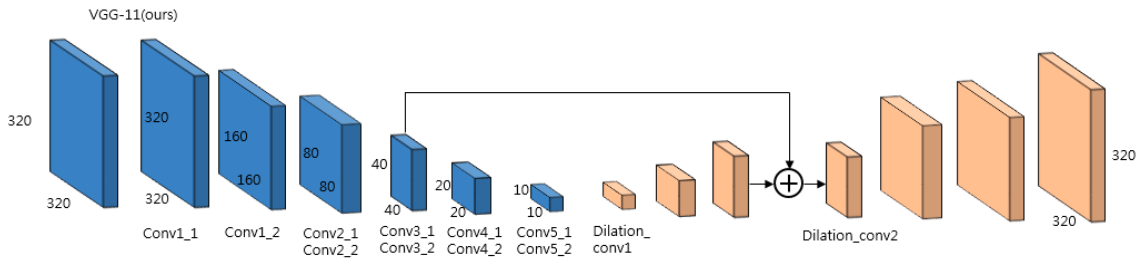


Fig. 4. Segmentation model.

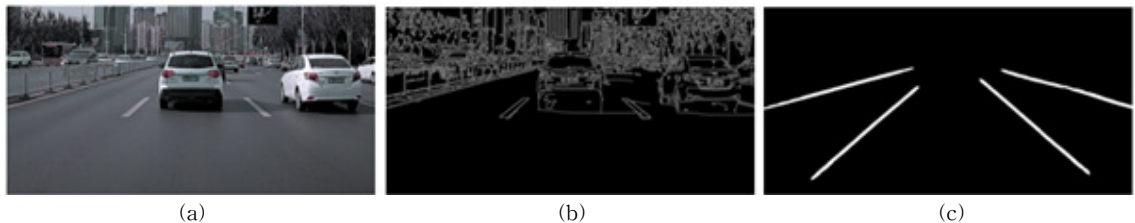


Fig. 5. Preprocessing of our LDW system. (a) Input image, (b) result of canny edge detection, and (c) result of deep learning based segmentation.

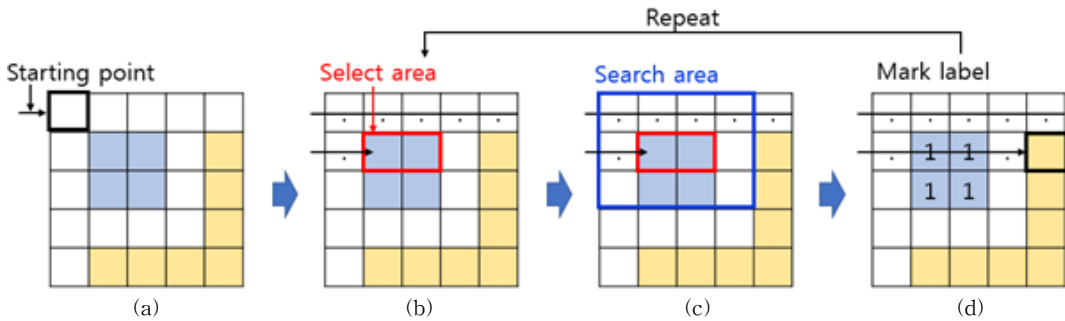


Fig. 6. First pass multi search labeling. (a) set starting point, (b) select area in segmentation result, (c) search pixels for marking label, and (d) mark label each pixel.

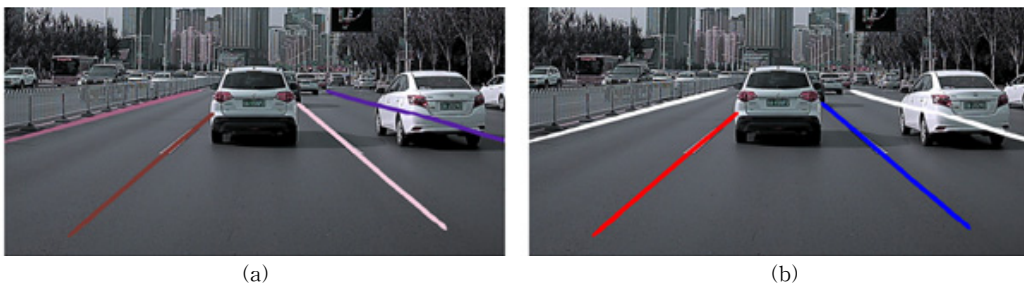


Fig. 7. Result of labeling. (a) each labeling result and (b) selected ego-lane.

만 본 논문에서 제안한 알고리즘은 Fig. 6(d)와 같이 선택된 픽셀을 기준으로 labeling을 표시해 줌으로써 중복된 처리를 생략하여 빠른 연산이 가능하다.

Labeling이 완료되면, 화면 중심을 기준으로 가장 양쪽 labeling을 선택하며, 선택된 라벨링을 기준으로 ego-lane을 결정한다. Fig. 7(a)는 세그멘테이션된 영상을 labeling 한 결과이고, Fig. 7(b)는 labeling된 차선 중 ego-lane을 선택한 결과 영상이다. ego-lane이 아닌 차선은 흰색으로 표시하고 ego-lane은 붉은색과 푸른색으로 표시하였다.

3.4 차선 이탈 검출

차선 이탈은 차량의 타이어가 차선을 넘을 때를 의미하며, TLC(Time to Line Crossing)를 기반으로 차선 이탈 여부를 판단한다.[16, 17] 그러나, TLC를 구하기 위해서는 차선의 곡률을 계산하는 모듈이 필요하며, 차량의 중속 및 횡속을 차량의 ECU로부터 CAN을 통하여 받아야만 TLC의 구현이 가능하다. 따라서 본 논문에서는 차량의 타이어로부터 차선까지의 거리를 lateral offset으로 정의하고, 이를 이용하여 차선 이탈 여부를 판단하였다. 차선의 폭(m)을

실제 환경에서 측정하고 영상 내에서 차선의 폭이 몇 개의 픽셀로 표현되는지 개수를 측정하여 한 개의 픽셀이 나타내는 실제 거리를 m/픽셀 로 계산하였다. 그 후 검출된 차선의 위치에 따라 Lateral offset의 값을 계산하여 차선 이탈 여부를 판단하였다.

최종적인 Lateral offset의 값은 아래 식 (1)을 통하여 얻을 수 있다. X는 장착된 카메라로부터 차선까지의 거리이며, 이 때, 카메라는 차량의 중앙에 붙어 있다고 가정하였다. tread는 테스트 차량의 제원 중 차량의 폭 값을 사용하였고, alpha 값은 타이어의 중심으로부터 타이어 바깥까지의 거리이다. 차선까지의 남은 거리를 기준으로 임계 값 이하가 되면 차선을 이탈했다고 판단하였으며, 임계값은 0.1m를 사용하였다.

$$Lateral\ offset = X - (tread/2) + \alpha \sin(x) \quad (1)$$

4. 실험 평가

4.1 세그멘테이션

차선 검출 성능 향상을 평가하기 위하여 제안한 방법의 세그멘테이션 결과와 예지 영상을 사용한 차

Table 1. Lane Detection Rate.

| | Algorithm | |
|----------------|---------------------------|-------------------------|
| | Edge Based Lane Detection | Proposed Lane Detection |
| True Positive | 2515 | 2779 |
| False Positive | 270 | 37 |
| False Negative | 328 | 64 |
| Recall | 88.46 | 97.74 |
| Precision | 90.30 | 98.68 |

선 검출 결과를 비교 평가하였다. 정량적인 성능 평가를 위하여 1,465장의 영상에서 2,843개의 ego-lane에 대한 GT(Ground Truth)를 작성하였다. 성능 비교는 Recall, Precision으로 진행하였으며 그 결과는 Table 1과 같다.

실험 결과, 오검출이 약 9.4% 줄어 들었는 것을 확인할 수 있었으며, 성능은 Recall, Precision에서 각각 약 9.28%, 8.4% 성능이 향상되었음을 확인하였다.

4.2 차선 이탈 검출

차선 이탈 검출은 NHTSA의 LDW Conformation Test 문서의 테스트 시나리오를 기반으로 테스트를 진행하였다. 테스트는 실제 차량을 사용하여 진행되었으며, 차선을 이탈할 경우 경고 알람이 발생하면 합격, 그렇지 않을 경우 불합격으로 판단하였다. 차선의 종류에 따라 차선의 색상 및 차선의 종류별로 구분하여 테스트를 진행하였으며 각각 50번씩 총 400회의 테스트를 진행하였다. 또한 좌/우측 차선의 종류가 다른 경우도 고려하여 테스트를 진행하였다. Table 2는 차선 이탈 경고 시스템의 테스트 결과를 나타낸다. 흰색과 황색의 직선 점선 환경에서 차선 이탈 경고 발생 여부를 통하여 성능을 평가하였다. 흰색 차선에서는 100% 알람이 발생하였으며, 황색 차선에서는 각각 96% 및 94%의 알람이 발생하였다. 학습 데이터 세트에 흰색 차선의 분포가 많은 것이 원인으로 판단된다.

Table 2. The Result of the Accuracy test for LDW.

| Lane color | white | | | | Yellow | | | |
|-----------------|-------|--------|--------|--------|--------|--------|--------|--------|
| | Solid | | Dashed | | Solid | | Dashed | |
| Left lane type | | | | | | | | |
| Right lane type | Solid | Dashed | Solid | Dashed | Solid | Dashed | Solid | Dashed |
| Accuracy | 100% | 100% | 100% | 100% | 98% | 96% | 97% | 93% |

Table 3. Comparison of VGG-11 and VGG-13.

| | VGG-11 (ours) | VGG-13 |
|-----------|---------------|-------------|
| Parameter | 3,381,071 | 133,047,848 |
| FPS | 12.83 | - |
| mAP | 85.95% | 87.70% |

4.3 모델 성능 비교

Table 3은 제안한 VGG-11 모델과 VGG-13 모델의 학습 파라미터수와 성능을 보여준다. VGG-11 모델은 약 330만개의 학습 파라미터 만을 사용하여 VGG-13 모델에 비하여 약 2% 수준의 파라미터를 사용하므로 속도면에서 많은 이익이 있었다. 다만, VGG-13 모델은 개발한 임베디드 시스템의 메모리 한계로 보드에 탑재하는 것이 불가능하여 임베디드 환경에서의 FPS 비교는 불가능하였다. pre-trained 모델을 개발하기 위하여 사용한 학습 데이터 셋에 대하여 mAP가 약 86%로 VGG-13 모델에 비하여 약 1.75% 정도의 성능 하락이 있었던 것을 확인하였다.

5. 결 론

본 논문에서는 딥러닝 모델 기반의 차선 이탈 경고 시스템을 임베디드 환경에서 구현하고 성능을 평가하였다. 임베디드 환경에서 실시간으로 동작할 수 있도록 네트워크의 크기를 줄이고, 줄어든 네트워크 크기에 맞도록 새로운 pre-trained 모델을 학습시켜 사용하였다. 네트워크의 크기를 줄이더라도 인식 성능에 큰 차이가 없었으며, 속도 면에서 많은 성능 향상이 있었기에 임베디드 환경에서 사용할 수 있었다.

차선 검출에 있어서 기존의 에지 기반의 알고리즘보다 오검출률이 9.4% 줄어들면서도 Recall과 Precision에서 각각 9.28%, 8.4%의 성능 향상을 확인하였다. 또한, 차선 이탈 검출의 성능 평가를 위하여 미국도로교통안전국의 테스트 시나리오를 사용하여 약 96% 정확도로 동작하는 것을 확인하였다.

최근 임베디드 환경에서 딥러닝을 이용하여 차선

검출에 적용하는 연구들이 나타나기 시작했다. 추후 연구에서는 제안한 알고리즘과 다른 연구들과의 비교를 통해 제안한 알고리즘의 완성도를 높이고, 차선 이탈 경보의 정확도를 높일 계획이다.

REFERENCE

- [1] Y. Kim and H. Kim, "Trends of Autonomous Vehicle Development," *Information and Communication Magazine*, Vol. 34, Issue 5, pp. 10-18, 2017.
- [2] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," *IEEE Access*, vol. 8, pp. 58443-58469, Mar. 2020.
- [3] A. Ziebinski, et al., "Review of advanced driver assistance systems (ADAS)," *AIP Conference Proceedings*. Vol. 1906. No. 1. AIP Publishing LLC, 2017.
- [4] L. Meng, W. Kees, and R.V. Der Heijden, "Technical Feasibility of Advanced Driver Assistance Systems for Road Traffic Safety," *Transportation Planning and Technology*, Vol. 28, No. 3, pp. 167-187, Mar. 2005.
- [5] J. Son, H. Yoo, S. Kim, and K. Sohn, "Real-Time Illumination Invariant Lane Detection for Lane Departure Warning System," *Expert Systems with Applications*, Vol. 42, No. 4, pp. 1816-1824, Oct. 2015.
- [6] J. Shin, J. Jung, and M. Kim, "Lane Detection Based on Vision Sensor Using a Robust Filter for Inner Edge Detection," *Journal of Sensor Science and Technology*, Vol. 28. No. 3, pp. 164-170, 2019.
- [7] K. Lee and C. Lin. "A New Efficient Detection Method in Lane Road Environment," *Journal of Korea Institute of ITS*, Vol. 17, No. 1, pp. 129-136, 2018.
- [8] D. Neven, et al., "Towards End-to-End Lane Detection: An Instance Segmentation Approach," *IEEE Intelligent Vehicles Symposium (IV)*, pp. 286-291, 2018.
- [9] Q. Zou, et al., "Robust Lane Detection from Continuous Driving Scenes Using Deep Neural Networks," *IEEE Transactions on Vehicular Technology*, Vol. 69, No. 1 pp. 41-54, 2019.
- [10] A. Yaguchi, et al., "Adam Induces Implicit Weight Sparsity in Rectifier Neural Networks," *17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. pp. 328-325, 2018.
- [11] NHTSA (National Highway Traffic Safety Administration), <https://www.nhtsa.gov/> (accessed June 10, 2021).
- [12] W. Liu, et al., "SSD: Single Shot Multibox Detector," *European conference on computer vision*. Springer, Cham, pp. 21-37, 2016.
- [13] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv Preprint*, arXiv: 1409.1556, 2014.
- [14] F. Leymarie and M.D. Levine. "Simulating the Grassfire Transform Using an Active Contour Model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14. No. 1 pp. 56-75, 1992.
- [15] L.G. Shapiro and G. Stockman, *Computer Vision*, 1st edition, Prentice Hall, pp. 69-73, 2002.
- [16] W. van Winsum, K.A. Brookhuis, and D. de Waard, "A Comparison of Different Ways to Approximate Time-to-Line Crossing (TLC) during Car Driving," *Accident Analysis & Prevention* Vol. 32. No. 1 pp. 47-56, 2000.
- [17] S. Mammari, S. Glaser, and M. Netto, "Time to Line Crossing for Lane Departure Avoidance: A Theoretical Study and an Experimental Setting," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 7. No. 2, pp. 226-241, 2006.
- [18] J. Lee and Y. Kang, "A Method of Lane Marker Detection Robust to Environmental Variation Using Lane Tracking," *Korea Multimedia Society*, Vol. 21, No. 12, pp. 1396-1406, 2018.



강 현 우

2005년 경북대학교 컴퓨터공학
공학사

2009년 경북대학교 컴퓨터공학
공학석사

2017년 경북대학교 컴퓨터공학
공학박사

2009년 2월 ~ 2017년 11월 한국전자통신연구원
선임연구원

2017년 12월 ~ 현재 한국폴리텍대학
AI엔지니어링과 조교수

관심분야: 컴퓨터비전, 머신러닝