

Out-layer를 제거한 End to End 자율주행 시스템

정승혁¹, 윤동호², 홍성훈^{3*}

¹전남대학교 ICT융합시스템공학과 석사과정, ²한국생산기술연구원 수석연구원, ³전남대학교 ICT융합시스템공학과 교수

End to End Autonomous Driving System using Out-layer Removal

Seung-Hyeok Jeong¹, Dong-Ho Yun², Sung-Hun Hong^{3*}

¹Master's Course, Department of ICT Convergence System Engineering, Chonnam University

²Principal Researcher, Korea Institute of Industrial Technology

³Professor, Department of ICT Convergence System Engineering, Chonnam University

요약 본 논문에서는 비전 센서 기반 시스템의 차선 이탈과 신호등 오인식 등을 개선하기 위해 End to End 모델을 활용한 자율주행 시스템을 제안한다. End to End 학습은 다양한 환경 조건에 대해 확장을 할 수 있다. 비전 센서 기반 모형 자동차를 이용하여 주행 데이터를 수집한다. 수집한 데이터를 이용하여 기존의 데이터와 아웃레이어를 제거한 데이터로 구성한다. 입력 데이터인 카메라 이미지 데이터, 출력 데이터인 속도와 조향 데이터로 클래스를 구성하고 End to End 모델을 활용하여 데이터 학습을 수행하였다. 학습된 모델의 신뢰성을 확인했다. 모형 자동차에 학습한 End to End 모델을 적용하여 이미지 데이터로 조향각을 예측한다. 모형 자동차의 학습 결과, 아웃레이어를 제거한 모델이 기존 모델보다 향상된 것을 볼 수 있다.

주제어 : 자율주행 자동차, 컴퓨터 비전, 딥러닝, 아웃레이어 제거, 자율주행

Abstract In this paper, we propose an autonomous driving system using an end-to-end model to improve lane departure and misrecognition of traffic lights in a vision sensor-based system. End-to-end learning can be extended to a variety of environmental conditions. Driving data is collected using a model car based on a vision sensor. Using the collected data, it is composed of existing data and data with outliers removed. A class was formed with camera image data as input data and speed and steering data as output data, and data learning was performed using an end-to-end model. The reliability of the trained model was verified. Apply the learned end-to-end model to the model car to predict the steering angle with image data. As a result of the learning of the model car, it can be seen that the model with the outlayer removed is improved than the existing model.

Key Words : Autonomous Vehicle, Computer Vision, Deep Learning, Out Layer Removal, Autonomous Driving

1. 서론

딥러닝(Deep Learning)[1]은 기계 학습(Machine-Learning)을 이용한 방법으로 사물이나 데이터를 군집

화하거나 분류하는 데 사용하는 기술이다. 딥러닝(Deep Learning)은 여러 층을 가진 인공신경망(Artificial Neural Network, ANN)을 사용하여 머신러닝 학습을 수행하는 것으로 심층학습이라고도 부른다. 인공신경망

본 논문은 2023년도 한국생산기술연구원 기관주요사업 "스마트 모빌리티 핵심 요소기술 개발(1/3)(kitech JA-23-0011)" 의 지원을 받아 수행된 것임

*교신저자 : 홍성훈(hsh@chonnam.ac.kr)

접수일 2022년 12월 21일

수정일 2023년 1월 20일

심사완료일 2023년 1월 23일

은 여러 뉴런이 서로 연결된 구조의 네트워크이며, 입력층(Input layer)을 통해 학습하고자 하는 데이터를 입력받게 된다. 이렇게 입력된 데이터들은 여러 단계의 은닉층(Hidden layer)을 지나면서 처리가 이루어져 출력층(Output layer)을 통해 최종 결과가 출력되게 된다. 이러한 신경망을 3개 이상 중첩한 구조를 깊은 신경망(Deep Neural Network, DNN)이라고 부르며, 이를 활용한 머신러닝 학습을 특별히 딥러닝이라고 부르는 것이다. 따라서 딥러닝이란 기계가 자동으로 대규모 데이터에서 중요한 패턴 및 규칙을 학습하고, 이를 토대로 의사결정이나 예측 등을 수행하는 기술로 정의할 수 있습니다. 초창기 딥러닝의 발전은 CNN(Convolutional Neural Network)[2]이 기반이었지만 요새는 다양한 분야에서 활용되고 CNN은 최근 몇 년 동안 많이 발전하였다. ILSVRC(Large Scale Visual Recognition Challenge) [3] 와 같은 레이블이 지정된 대규모 데이터 세트를 교육 및 검증에 사용할 수 있게 되었다[4]. CNN 학습 알고리즘은 학습 및 추론을 엄청나게 가속하는 대규모 병렬 그래픽 처리 장치(GPU)에 구현되었다.

비전 센서 기반 자율주행 시스템은 자율주행에 있어 가장 기본적인 시스템이다. 카메라 영상을 OpenCV 라이브러리를 이용하여 캐니 엣지(Canny Edge)[5]와 허프 변환(Hough Transform)[6], 색상(HSV, RGB) 추출 등을 활용하여 차선 검출하고 차선의 중앙점을 찾아 자율주행차의 조향 제어를 하여 주행할 수 있다. 이러한 비전 센서 기반 자율주행은 환경 조건에 따라 오인식 될 수 있다. 날씨 조건이 좋지 않아 카메라가 잘 보이지 않거나 카메라에 빛 반사가 심할 경우에 번짐이 일어날 수 있어 인식이 쉽지 않다.

자율주행 자동차(Autonomous Vehicle)는 운전자가 개입하지 않고 정밀지도, GPS, 카메라, 라이다 등으로부터 데이터를 취득해 주변을 인지, 판단하여 스스로 운행하는 자동차이다. 일반적으로 사람이 운전할 경우에 눈으로 차선, 신호등 보행자 등을 보고 인지한다. 자율주행 자동차에서는 이러한 기능을 하는 것이 센서이다. 카메라, 라이다, 레이더, GPS 등을 차량에 탑재하여 차량 주변을 인지하고, 어떻게 주행할 것인지 대한 정보를 제공한다. 인지 단계에서 차량 또는 보행자 등 주변 환경의 위치와 속도 정보를 파악하면 판단 알고리즘을 통해 각 물체의 향후 이동 방향을 예측하고, 정해진 경로를 어떻게 주행할지를 결정한다. 판단 단계에서 결정된 대로 차량을 움직이게 만드는 것이 바로 제어 단계이다. 판단으로 결정된 조향각과 가속도 값을 MDPS, EMS, ESC에

통신하여 제동, 엔진, 조향을 제어함으로써 자율주행 자동차가 주행하게 된다[7].

End-to-End는 스티어링 휠과 페달의 연속적인 작동 또는 가속 및 좌회전과 같은 별개로 동작할 수 있다. End-to-End에는 지도학습 딥러닝[8-11], 신경 진화[12, 13] 및 최근의 심층 강화 학습[14]의 세 가지 주요 접근 방식이 있다. 가장 기초의 End-to-End 기반의 자율주행[15]은 오프로드 주행을 위한 시스템에서 시작되었다[16]. 인공지능의 발전으로 딥 컨볼루션 네트워크로 자율주행이 실현할 수 있게 되었다. 이미지를 입력 및 출력 조정으로 사용하는 심층 컨볼루션 신경망이 제안되었다.

기존의 모형 차는 비전 센서 기반으로 작동하기 때문에 차선 이탈 등 사고가 발생하는 문제점과 빛 반사로 인해 오인식하는 문제점이 있다. 본 논문에서는 입력 데이터 카메라 이미지 데이터, 출력 데이터 속도와 조향 데이터를 이용하여 자율주행을 위한 조향 제어를 하고자 한다. 자율주행을 위해 모형 차로 데이터를 수집한 후 End to End로 학습시킨 후에 이미지를 통해 조향각을 예측하였으며 기존의 학습모델과 아웃레이어를 제거한 후에 학습모델과 비교하였다.

2. 시스템 구성

실험에 사용된 모형 자동차 시스템의 임베디드 보드는 NVIDIA TX2를 사용한다. 전체 임베디드 시스템은 Ubuntu 기반의 NVIDIA Jetpack으로 구성되어 있다.

모형 자동차 시스템은 구동 시스템과 비전 시스템으로 나눌 수 있다. 임베디드 시스템은 각각의 구동 시스템과 비전 시스템을 ROS(Robot Operation System)로 제어하도록 설계되어 있다. <그림 1>은 모형 자동차의 전체 시스템을 보인다.

구동 시스템은 바퀴 구동 모터와 조향 모터로 나누어져 있다. VESA(Vedder Electric Speed Controller)로 모터를 제어한다. 구동 모터는 전진, 정지를 제어하고 조향 모터는 모형 자동차의 바퀴의 방향을 결정한다.

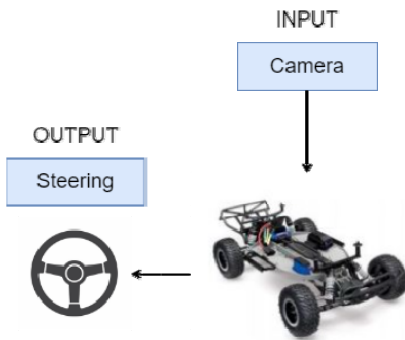
비전 시스템은 170° 어안렌즈와 2D 라이다로 구성되어 있다. 어안렌즈의 입력 영상 데이터는 MJPEG의 이미지 포맷을 사용하고 이 데이터를 YUV 에서 RGB로 변환하여 시스템에 사용하였다. 2D 라이다는 1-채널, 1° degree, 8,000 sampling도 구성되어 있다. 스캔 데이터로 받고 포인트 클라우드로 변환하여 사용한다.

시스템은 어안렌즈를 통해 받은 RGB 데이터를 이용하였다. OpenCV 기반의 영상처리를 통해 전체 이미지에서 차선의 위치를 파악한다. RGB 이미지 데이터를 Gray Scale로 변환하고 이진화 변환을 수행한다. 결과 이미지를 통하여 차선 추출을 한다. 정지선 인식은 화면 화면에 ROI(Region Of Interest) 영역을 설정한 후에 흰 픽셀이 많으면 정지선으로 인식하여 정지한다. 신호등 인식은 화면 상단에 ROI 영역을 설정한 후에 연속된 동그라미를 찾아 신호등을 인식하고 명도를 이용하여 신호등 상태정보를 획득한다.

2D 라이다로 장애물과의 각도와 거리 정보를 (X, Y) 좌표로 변환하고 극좌표계와 직교 좌표계를 이용하여 주위 장애물을 감지하고 정지한다.



[Fig. 2] Track



[Fig. 1] Vehicle system

3. 제안한 시스템

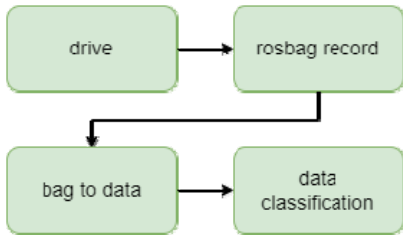
앞서 설명한 기존 비전 센서 기반의 시스템은 다양한 환경조건에서 주행이 어렵다. 이를 해결하기 위해 본 논문에서는 End to End 모델을 사용한 자율주행 시스템을 제안한다. End to End 자율주행 시스템은 차량의 탑재된 카메라 이미지를 입력데이터로 하고, 이를 통해 조향 데이터가 출력되는 시스템이다. 다양한 데이터로 학습시켜 복잡하고 다양한 환경조건에서 주행을 할 수 있기 때문에 비전 시스템의 한계를 개선하여 다양한 주행 환경에서의 자율주행 시스템 개발이 가능하다. 이에 본 논문에서는 기존 비전 센서 기반의 시스템이 주행 트랙[Fig. 2]으로부터 주행 데이터를 획득하여 End to End 자율주행 시스템을 구축하였다. 시스템 구축은 데이터 세트 구축 및 데이터 전처리, 딥러닝 학습을 진행하였다.

3.1 데이터 세트

모형차가 주행 트랙에서 주행 중에 수집하는 데이터에 대하여 자율주행할 수 있도록 주행 트랙을 사용하여 운전 모드로 차량을 주행하고 데이터 세트를 구축하였다. 입력 데이터 이미지와 출력 데이터 속도와 조향 데이터를 Rosbag으로 데이터를 수집한 후에 bag 파일로 저장한다. bag 파일의 이미지 데이터를 추출한다. 모든 데이터를 타임스탬프 순으로 정렬하여 데이터 세트 [Fig. 3]를 구성했다. 데이터 세트를 전체 트랙을 주행하는 데이터 세트와 주행 구간을 직진, 좌, 우 분리하여 주행하는 데이터 세트를 준비하였다. 데이터 세트는 트랙을 1바퀴 주행하여 기록한 것이고 총 4번의 직진, 7번의 좌우 구간이 있다. 총 2,801개의 이미지 데이터 세트가 있으며 train과 validation을 8:2로 나눠서 구성하고 학습 이후 test는 트랙 데이터 세트로 사용하였다. [Fig 4]는 데이터 수집의 순서도를 나타낸 것이다.

image_name	steering_angle	speed
1455209118.434021.png	0.034000002	1
1455209118.465142.png	0	1
1455209118.501359.png	0	1
1455209118.575235.png	0	1
1455209118.593880.png	0	1
1455209118.626458.png	0	1
1455209118.660148.png	0	1
1455209118.696134.png	0	1
1455209118.728781.png	0	1
1455209118.765582.png	0.017000001	1

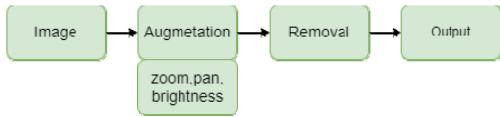
[Fig. 3] Example of dataset



[Fig. 4] Data flow chart

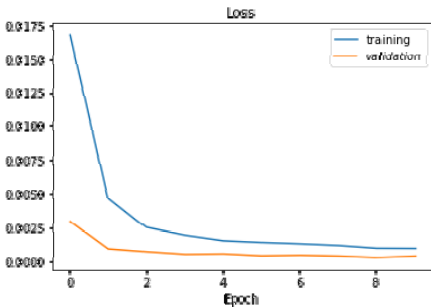
3.2 데이터 전처리

주행 데이터 세트에서 학습을 진행하기 전에 전처리하는 과정이 필요하다. 먼저 Data Augmentation을 진행하고 조향 데이터의 아웃 레이어를 제거한 후에 총 2,081개의 이미지 데이터를 구축하였다. Data Augmentation은 이미지에 zoom, pan, brightness를 랜덤하게 적용하였다. 아웃 레이어는 직진 구간에서 정면을 90도를 기준으로 하고 나머지 값을 제거하고 좌우 구간에서는 90도 기준으로 좌우로 20도 이후로 제거하고 학습을 진행하였다. 데이터 세트의 이미지는 640*480이므로 640*280으로 크기를 변경하였다.



[Fig. 5] Preprocessing

3.3 딥러닝 학습



[Fig. 6] Learning result

카메라 이미지와 조향, 속도 데이터를 학습시키기 위해 자율주행 시스템에 대한 End to End 학습을 하였다. Pilot net[16] 기반의 Convolution 5개, Max pooling 3개 등의 과정을 통해 조향각을 예측하게 된다. 자율주

행 시스템의 입력은 이미지 데이터였고, 출력은 조향 데이터다. 그리고 이미지 이름과 조향 데이터값으로 구성된 CSV 파일과 이미지 데이터 학습을 진행하였다. 학습은 총 10 epochs로 하였고 [Fig. 3]과 같이 일정 구간에서부터 수렴하는 것을 볼 수 있다.

4. 실험 및 결과

본 논문에서 제안하는 End to End 딥러닝 학습을 위해서 이미지와 조향 데이터가 필요하다. <Table 1>은 학습에 사용된 데이터의 각도에 따른 데이터 수량이고 2081개의 이미지 데이터이다.

<Table 1> Number of data

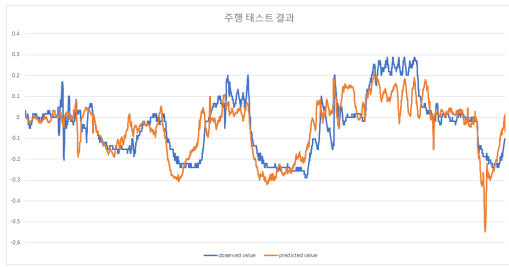
	Number of data
0 < data < 90	459
90	879
90 < data < 180	743

End to End로 학습을 진행하고 모델을 저장했다. 학습된 모델을 이용하여 차선 이미지를 넣어 예측하였다. 학습이 완료된 모델에 대해서 원래 값과 비교하여 RMSE 값을 확인하였으며, 원래 값과 아웃레이어 제거한 후에 학습결과값을 <Table 2>로 각각의 값을 비교하여 보여 준다. 아웃레이어 제거한 후 결과값 0.0508이고 제거 전의 결과값은 0.0864이다. steering angle이 radian 값이므로 degree로 변환하여 계산하였을 때 각각 2.914°, 4.958°인 것을 확인할 수 있다.

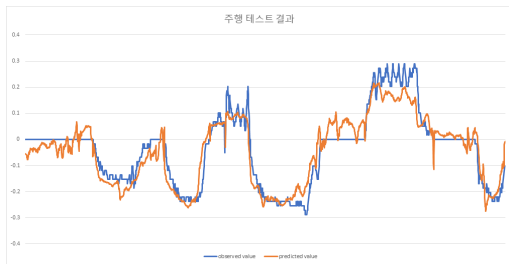
기존의 데이터로 학습한 결과와 아웃레이어 제거 후 학습한 결과를 비교하였을 때 아웃레이어 제거한 후에 결과가 더 향상된 것을 확인할 수 있다. 전체 구간의 조향 데이터와 예측한 전체 구간의 데이터와 비교한 것이다.

<Table 2> Result

	RMSE	Removed RMSE
steering angle(rad)	0.0864	0.0508
degree	4.958	2.914



[Fig. 7] Test result



[Fig. 8] Removal test result

5. 결론

본 연구는 모형 차로부터 취득한 카메라 이미지 데이터, 조향 데이터와 속도 데이터를 활용하여 딥러닝 모델을 기반으로 하는 자율주행 시스템을 개발했다. 자율주행 시스템은 기존 비전 센서 기반의 주행 시스템을 딥러닝 알고리즘을 통하여 구성하고자 했다. 특히 End to End를 이용하여 구성하고, 이미지 데이터, 조향과 속도에 대하여 데이터 수집을 했다. 이를 기반으로 딥러닝 학습을 진행했다. 이를 기반으로 딥러닝 학습을 진행 후 End to End의 모델 학습 과정을 수행하여 직선 구간의 경우엔 0.413rad, 좌우 곡선 주행 구간에서는 각각 0.0636rad, 0.0505rad를 확인하였다. radian을 degree로 환산 하면 직선 구간의 경우엔 2.36°, 좌우 곡선 주행 구간에서는 각각 3.65, 2.89°의 오차가 발생하는 것을 확인하였다. End to End의 모델은 기존 비전 기반의 시스템보다 다양한 주행환경에서 실험할 수 있어 범용성이 더 좋다고 할 수 있고 이를 기반으로 아웃레이어를 제거한 시스템은 제거 전보다 오차가 줄어들어 성능이 향상된 것을 확인할 수 있다.

향후 연구에서 데이터가 부족한 부분을 실패 사례와 시뮬레이션 환경에서의 데이터를 수집한다면 더욱 정확도를 향상시킬 수 있을 것이다. 그 이후에 실제 차량과

정밀 센서와 딥러닝 알고리즘으로 더욱 향상된 자율주행 시스템을 만들고자 한다.

REFERENCES

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, Vol.521, No.7553, pp.436-444, 2015.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, Vol 60, No.6, pp.84-90, 2017.
- [3] <http://www.image-net.org/challenges/LSVRC>
- [4] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, Vol.115, No.3, pp.211-252, 2015.
- [5] Rashmi, M. Kumar, and R. Saxena, "Algorithm and Technique on Various Edge Detection : A Survey," *Signal & Image Processing An International Journal*, Vol.4, No.3, pp.65-75, 2013.
- [6] P. Mukhopadhyay and B. B. Chaudhuri, "A survey of Hough Transform," *Pattern Recognition*, Vol.48, No.3, pp.993-1010, 2015.
- [7] <https://blog.naver.com/autoa2z/222069303400>
- [8] C. Chen, A. Seff, A. Kornhauer, and J. Xiao, "DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving," 2015 IEEE International Conference on Computer Vision (ICCV). IEEE, 2015.
- [9] Pomerleau, D. A. "Alvin: An autonomous land vehicle in a neural network." *Advances in neural information processing systems*, 1. 1988.
- [10] Y. LeCun, U. Muller, J. Ben, E. Cosatto, and B. Flepp. "Off-road obstacle avoidance through end-to-end learning," In *Proceedings of the 18th International Conference on Neural Information Processing Systems (NIPS'05)*, pp.739-746, 2015.
- [11] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-End Learning of Driving Models from Large-Scale Video Datasets," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.
- [12] S. Baluja, "Evolution of an artificial neural network based autonomous land vehicle controller," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, Vol.26, No.3, Institute of Electrical and Electronics Engineers (IEEE), pp.450-463, 1996.
- [13] J. Koutník, G. Cuccu, J. Schmidhuber, and F. Gomez, "Evolving large-scale neural networks for vision-based reinforcement learning," *Proceedings of the 15th annual conference on Genetic and evolutionary computation*. ACM, 2013.
- [14] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "Deep Reinforcement Learning framework for

Autonomous Driving," Electronic Imaging, Vol.29, No.19, pp.70-76, 2017.

- [15] M. Bojarski et al., "End to End Learning for Self-Driving Cars." arXiv, 2016.
- [16] A. Kendall et al., "Learning to Drive in a Day," 2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019.

정 승 혁(Seung-Hyeok Jeong) [준회원]



- 2018년 2월 : 조선대학교 전자공학(공학학사)
- 2020년 3월 ~ 현재 : 전남대학교 ICT융합시스템학과(공학석사)

<관심분야>

자율주행, 컴퓨터 비전

윤 동 호(Dong-Ho Yun) [정회원]



- 2017년 2월 : 광주과학기술원 기계공학부 (공학박사)
- 2017년 12월 ~ 현재 : 한국생산기술연구원 수석연구원
- 2021년 3월 ~ 현재 : UST KITECH스쿨 겸임조교수

<관심분야>

자율주행, 인공지능, TLS

홍 성 훈(Sung-Hoon Hong) [정회원]



- 1991년 2월 : 한국과학기술원 전기 및 전자공학과(공학석사)
- 1999년 8월 : 한국과학기술원 전기 및 전자공학과(공학박사)
- 1991년 ~ 2000년 : LG전자 DTV 연구소 책임연구원

- 2000년 ~ 현재 : 전남대학교 ICT융합시스템공학과 교수

<관심분야>

영상처리, 영상부호화, 멀티미디어시스템, 이종센서 융합 응용처리