

차량 내·외부 데이터 및 딥러닝 기반 차량 위기 감지 시스템 설계

손수락*, 정이나**

A Design of the Vehicle Crisis Detection System(VCDs) based on vehicle internal and external data and deep learning

Su-Rak Son*, Yi-Na Jeong**

요약 현재 자율주행차량 시장은 3레벨 자율주행차량을 상용화하고 있으나, 안정성의 문제로 완전 자율주행 중에도 사고가 발생할 가능성이 있다. 실제로 자율주행차량은 81건의 사고를 기록하고 있다. 3레벨과 다르게 4레벨 이후의 자율주행차량은 긴급상황을 스스로 판단하고 대처해야 하기 때문이다. 따라서 본 논문에서는 CNN을 통하여 차량 외부의 정보를 수집하여 저장하고, 저장된 정보와 차량 센서 데이터를 이용하여 차량이 처한 위기 상황을 0~1 사이의 수치로 출력하는 차량 내·외부 데이터 및 딥러닝 기반 차량 위기 감지 시스템을 제안한다. 차량 위기 감지 시스템은 CNN기반 신경망 모델을 사용하여 주변 차량과 보행자 데이터를 수집하는 차량 외부 상황 수집 모듈과 차량 외부 상황 수집 모듈의 출력과 차량 내부 센서 데이터를 이용하여 차량이 처한 위기 상황을 수치화하는 차량 위기 상황 판단 모듈로 구성된다. 실험 결과, VESCM의 평균 연산 시간은 55ms 였고, R-CNN은 74ms, CNN은 101ms였다. 특히, R-CNN은 보행자 수가 적을 때 VESCM과 비슷한 연산 시간을 보이지만, 보행자 수가 많아 질수록 VESCM보다 많은 연산 시간을 소요했다. 평균적으로 VESCM는 R-CNN보다 25.68%, CNN보다 45.54% 더 빠른 연산 시간을 가졌고, 세 모델의 정확도는 모두 80% 이하므로 감소하지 않으며 높은 정확도를 보였다.

Abstract Currently, autonomous vehicle markets are commercializing a third-level autonomous vehicle, but there is a possibility that an accident may occur even during fully autonomous driving due to stability issues. In fact, autonomous vehicles have recorded 81 accidents. This is because, unlike level 3, autonomous vehicles after level 4 have to judge and respond to emergency situations by themselves. Therefore, this paper proposes a vehicle crisis detection system(VCDs) that collects and stores information outside the vehicle through CNN, and uses the stored information and vehicle sensor data to output the crisis situation of the vehicle as a number between 0 and 1. The VCDs consists of two modules. The vehicle external situation collection module collects surrounding vehicle and pedestrian data using a CNN-based neural network model. The vehicle crisis situation determination module detects a crisis situation in the vehicle by using the output of the vehicle external situation collection module and the vehicle internal sensor data. As a result of the experiment, the average operation time of VESCM was 55ms, R-CNN was 74ms, and CNN was 101ms. In particular, R-CNN shows similar computation time to VESCM when the number of pedestrians is small, but it takes more computation time than VESCM as the number of pedestrians increases. On average, VESCM had 25.68% faster computation time than R-CNN and 45.54% faster than CNN, and the accuracy of all three models did not decrease below 80% and showed high accuracy.

Key Words : Autonomous vehicles, Convolutional Neural Networks, You Only Look Once, Crisis Detection, Pedestrian recognition

*Catholic Kwandong University, Department of Computer Engineering

**Catholic Kwandong University, Department of Software

Received March 02, 2021

Revised March 06, 2021

Accepted March 23, 2021

1. 서론

자율주행차량은 이제 상용화를 앞두고 있다. 해외의 자율주행차량은 이미 상용화되어 판매되고 있으나, 안정성의 문제로 완전 자율주행 중에도 사고가 발생할 가능성이 있다. 실제로 자율주행차량은 81건의 사고를 기록하고 있다[1]. 따라서 본 논문에서는 CNN을 통하여 차량 외부의 정보를 수집하여 저장하고, 저장된 정보와 차량 센서 데이터를 이용하여 차량이 처한 위기 상황을 0~1 사이의 수치로 출력하는 차량 내·외부 데이터 및 딥 러닝 기반 차량 위기 감지 시스템을 제안한다. 본 논문에서 제안하는 차량 위기 감지 시스템은 CNN을 기반으로 차량 외부의 주행 중인 차량과 보행자의 데이터를 수집하는 차량 외부 상황 수집 모듈과 차량 외부 상황 수집 모듈의 출력과 차량 내부 센서 데이터를 이용하여 차량이 현재 처한 위기 상황을 0~1 사이의 값으로 출력하는 차량 위기 상황 판단 모듈로 구성된다. 본 논문의 2절은 해당 연구가 직, 간접적으로 영향을 받은 자율차량 안전에 대한 연구를 소개하고, 3절은 본 논문에서 제안하는 차량 위기 감지 시스템에 대해 상세히 기술한다. 4절은 차량 위기 감지 시스템의 유효성을 검증하기 위한 실험을 진행하고 5절에서는 해당 논문을 전체적으로 정리하고 향후 연구에 관해 설명한다.

2. 관련 연구

상명대학교에서는 차량 밀도가 높아지고 채널 접근 권한 획득 확률이 낮아지는 IEEE 802.11 DCF 기반 Flooding 기법의 문제점을 개선하기 위하여 각 차량이 현재 주행상태에 따른 위험도를 감지하고 이에 근거하여 메시지 전송 빈도를 적응적으로 조절하게 하여 메시지 교환 효율을 증대시키는 방안을 제안했다. 해당 연구는 차량의 위험도를 계산하고 근거리 차량 사이의 통신을 연결하여 메시지를 교환하게 했다[2].

군산 대학교에서는 ad-hoc 네트워크를 이용하여 차량 간 통신을 통한 차 사고에 대한 위험인지와 회피방안에 대한 설계를 제안했다. 해당 방안은 RSE(Road Side Equipment)와 차량의 OBE(On Board Equipment)를 이용하여 차량 사이의 네트워크

를 구성하고 각 차량의 이동 특성을 고려하여 차량의 라우팅을 설계했다[3].

해군 사관학교에서는 자율주행차량 사이의 V2V 기법을 이용하여 상대 속도를 계산하고 전방 차량의 제동 거리를 예측하여 안전거리를 확보하는 시스템을 제안했다. 해당 시스템은 레이더와 V2V 통신을 동시에 사용하여 차량의 안정성을 높이면서, 긴급상황을 과대평가하지 않도록 했다[4].

진국대학교에서는 급제동 이외의 위험 상황을 판단하기 위한 외연 확장성에 한계가 존재하는 급정거 위험 상황인지 시스템의 문제를 해결하기 위하여 모든 감속 구간의 주행데이터를 대상으로 기계 학습한 검지 모델에 의해 위험 상황을 판별했다. 해당 연구는 실시간 주행데이터를 바탕으로 감속 구간에 대한 기계학습을 통해서 위험 상황을 자동으로 검지하는 모델을 제안했으며 지도학습 기반의 신경망과 랜덤포레스트 및 비 지도학습 기반의 클러스터링을 병행하여 검지 결과의 신뢰성을 확보하고, 다음 학습에 반영함으로써 운전자의 주행 형태에 맞는 자동 위험 검지 모델을 제안했다[5].

충남대학교에서는 차량 충돌로 인한 교량 피해에 대하여 위험도 분석을 시행하였다. 위험도 분석 단계는 세 단계로 구분하였고, 예비위험분석 단계에서는 충돌의 발생 가능성을 확인하며, 기본위험분석 단계에서는 발생 가능성, 취약성, 중요도에 대한 위험도 점수 평가를 통한 위험도 등급을 산정하였다. 마지막 상세 위험분석 단계에서는 위험도가 높은 등급에 대하여 상세분석을 시행했다. 본 연구에서는 예비위험분석과 기본위험 단계에 집중하여 위험도 등급 구분을 위한 네 가지의 급간 분류법을 적용하였다[6].

계명대학교에서는 자동차에 설치된 열영상 카메라를 이용하여 운전자 보조 시스템(ADAS: Advanced Driver Assistance System)의 핵심기술인 위험 보행자를 검출하는 알고리즘을 제안했다. 먼저 보행자 검출 시스템의 처리시간 단축을 위하여 영상 스케일링 비율과 보행자 검출을 위한 ROI(Region-of-Interest)를 자동으로 설정함으로써 실시간 보행자 검출의 문제점인 처리시간을 단축했다. 또한 위험 보행자 여부를 판단하기 위해 자동차의

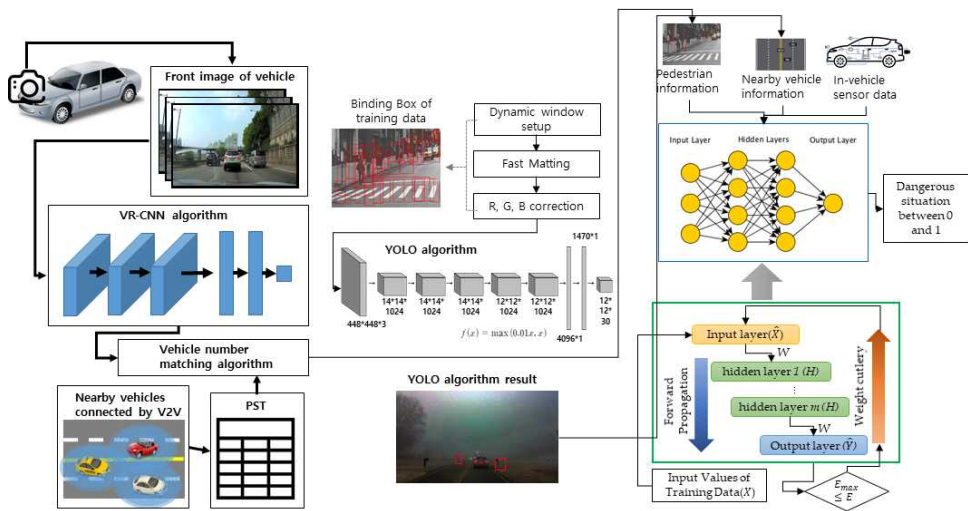


그림 1. 차량 위기 감지 시스템의 전체 구성도

Fig. 1. Overall configuration diagram of the Emergency-notification and Driver-response Confirmation System(EDCS)

진행 방향에 따라서 Reference Line을 설정하고, ROI 내에서 검출된 보행자가 Reference Line에 포함될 때 후보 위험 보행자로 판단했다. 이후 단계에서, 후보 보행자 윈도에서 Reference Line과 후보 보행자 윈도의 겹쳐진 비율, 보행자 윈도 내의 보행자 움직임 방향과 크기 값을 이용하여 3개의 위험 요소 벡터를 추출했다. 앞서 추출된 위험 요소 벡터들을 정규 함수에 적용하여 각 요소의 위험도를 추정하고, 이 위험도들을 결합하여 최종적인 위험 보행자 윈도와 위험도 스코어를 결정했다[7].

이용하여 차량이 현재 처한 상황의 위험도를 0~1 사이로 출력한다. 그림 1은 차량 위기 감지 시스템의 구성을 나타낸다.

3.2 차량 외부 상황 수집 모듈

차량 외부 상황 수집 모듈은 두 가지 CNN (convolutional neural network) 모델을 사용하여 차량 외부의 데이터를 수집한다.

3. 차량 위기 감지 시스템 설계

3.1 Overview

본 논문에서 제안하는 차량 위기 감지 시스템은 차량 외부의 상황과 차량 내부의 센서 데이터를 이용하여 차량이 현재 처한 위기 상황을 0~1 사이의 값으로 출력한다. 차량 위기 감지 시스템은 두 개의 모듈로 구성된다. 첫 번째 모듈인 차량 외부 상황 수집 모듈은 차량의 전방 카메라와 CNN을 이용하여 현재 주행 중인 차량의 주변 차량 정보와 보행자 정보를 수집한다. 두 번째 모듈인 차량 위기 상황 판단 모듈은 차량 외부 상황 수집 모듈의 결과와 차량 내부의 센서 데이터를

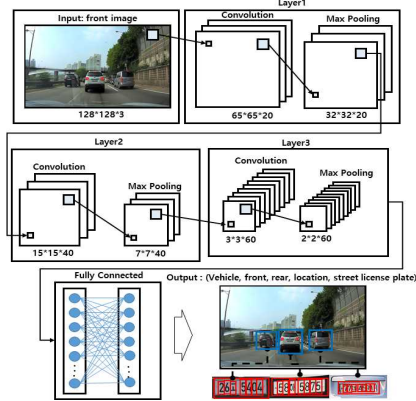


그림 2. VR-CNN을 통한 전방 차량 인식 과정

Fig. 2. Forward vehicle recognition process though VR-CNN

우선 차량 외부 상황 수집 모듈은 V2V 통신을 통해 주변 차량의 정보를 저장하고, 차량의 전면 카메라를 통해 현재 주행 중인 차량과 직접적으로 인접한 차량의 번호판을 인식한다. 그림 2는 전방 이미지를 통해 주변 차량을 인식하는 VR-CNN (Vehicle Recognition CNN) 모델의 구성을 나타낸다. VR-CNN은 1개의 Convolution과 1개의 Max Polling으로 구성된 3개의 레이어와 2개의 Fully Connected를 이용하여 결과를 출력한다. 차량 외부 상황 수집 모듈은 VR-CNN을 사용함과 동시에 CNN 기반 알고리즘인 YOLO(You Only Look Once)를 사용하여 주행 중 보행자를 탐지한다. 차량 외부 상황 수집 모듈은 두 개의 CNN 모델을 신속하게 연산하기 위하여 YOLO에 입력되는 이미지의 노이즈를 제거하고, YOLO에 사용되는 Binding Box의 크기를 제한하여 연산 속도를 높인다. YOLO는 12*12 Grid와 크기가 각각 14*14*1024, 12*12*1024인 5개의 Convolution Layer, 크기가 각각 4096, 1470인 2개의 Fully Connected를 사용한다. YOLO는 신속한 연산을 위해 Fully Connected 연산의 활성화 함수로 Leaky ReLU를 사용한다. 수식 1은 Leaky ReLU 함수의 계산 방식을 나타낸다.

$$f(x) = \max(0.01x, x) \quad (1)$$

3.3 차량 위기 상황 판단 모듈

차량 외부 상황 수집 모듈이 보행자 정보와 주변 차량 정보를 수집하면, 차량 위기 상황 판단 모듈은 DNN(Deep Neural Network) 모델을 사용하여 차량의 위기 상황을 판단한다. 표 1은 차량 위기 상황 판단 모듈의 입력을 나타낸다. 차량 위기 상황 판단 모듈은 13개의 입력을 사용하며, 입력값은 모두 0~1로 정규화된다. 차량 위기 상황 판단 모듈의 DNN은 단 하나의 출력 노드를 가지며, 출력 노드의 값은 0~1사이이다. 차량 위기 상황 판단 모듈은 딥 러닝 연산을 위하여 활성화 함수로 Sigmoid 함수를 사용한다.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

Sigmoid 함수는 항상 0~1 사이의 값을 가지기 때

문에 입력값과 출력값 모두 0~1 사이의 값을 갖는다. 수식 2는 Sigmoid 함수의 계산 방식을 나타낸다.

표 1. 차량 위기 상황 판단 모듈의 입력
Table 1. Virtual environment specification

node	meaning
x ₁	The number of vehicles ahead
x ₂	The distance to nearest vehicle
x ₃	The speed of nearest vehicle
x ₄	The distance to the farthest vehicle
x ₅	The number of pedestrians ahead
x ₆	The distance to nearest pedestrian
x ₇	The speed of the nearest pedestrian
x ₈	The distance to the farthest pedestrian
x ₉	Tire pressure
x ₁₀	Engine RPM
x ₁₁	Current speed
x ₁₂	Average speed over 5 seconds
x ₁₃	Headlight condition

차량의 긴급상황을 정확하게 판단하기 위하여 차량 위기 상황 판단 모듈은 사전에 DNN 모델을 학습시켜야만 한다. 차량 위기 상황 판단 모듈은 DNN을 Back-propagation 방식으로 학습시킨다. DNN의 학습을 위해 사용되는 트레이닝 데이터는 직접 수집된 정상 주행데이터와 교통사고 통계 분석 데이터를 이용한 차량의 사고 데이터를 기반으로 생성된다. 생성된 트레이닝 데이터 중 정상 주행데이터의 출력은 모두 0으로 설정되며, 사고 데이터는 차량의 수리 기록을 통하여 0~1 사이의 출력을 지정한다. 수식 3은 트레이닝 데이터를 사용할 때 손실함수를 나타낸다.

$$E = \frac{1}{2} \sum_{i=1}^k (d - y_i)^2 \quad (3)$$

4. 실험

본 논문은 차량 위기 감지 시스템의 유효성을 판단하기 위하여 가상환경의 주행데이터를 사용하여 차량 외부 상황 수집 모듈에 대한 실험을 진행한다. 차량 위기 감지 시스템의 차량 외부 상황 수집 모듈에서 사용된 보행자 탐지의 실시간성과 정확성을 입증하기 위하

여 차량 외부 상황 수집 모듈(VESCM)과 같은 조건으로 CNN, R-CNN을 구축하여 정확성과 연산 속도를 비교하였다.

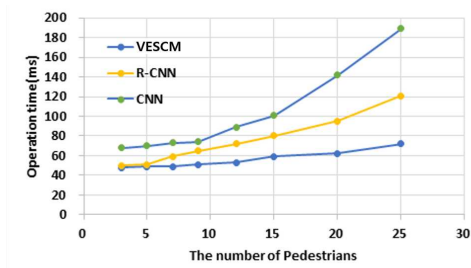


그림 4. VESCM, R-CNN, CNN의 연산 시간
Fig. 4. Calculation time of the VESCM, R-CNN and CNN

그림 4는 테스트 데이터 세트에서 보행자의 수가 늘어남에 따른 VESCM, R-CNN, CNN의 연산 시간을 나타낸다. 보행자의 수가 3명부터 25명까지 증가하면서 각 모델의 연산 시간이 측정되었다. 보행자가 3명일 때 VESCM의 연산 시간은 48ms였고, R-CNN은 50ms, CNN은 68ms였다. 보행자가 7명일 때 VESCM의 연산 시간은 49ms였고, R-CNN은 59ms, CNN은 73ms였다. 보행자가 12명일 때 VESCM의 연산 시간은 53ms였고, R-CNN은 72ms, CNN은 89ms였다. 보행자가 25명일 때 VESCM의 연산 시간은 72ms였고, R-CNN은 121ms, CNN은 189ms였다. VESCM의 평균 연산 시간은 55ms였고, R-CNN은 74ms, CNN은 101ms였다. 특히, R-CNN은 보행자 수가 적을 때 VESCM과 비슷한 연산 시간을 보이지만, 보행자 수가 많아질수록 VESCM보다 많은 연산 시간을 소요했다. 평균적으로 VESCM은 R-CNN보다 25.68%, CNN보다 45.54% 더 빠른 연산 시간을 가졌다. 따라서 VESCM은 차량 외부 상황 수집 모듈의 실시간성을 보장한다.

그림 5는 테스트 데이터 세트에서 보행자의 수가 늘어남에 따른 VESCM, R-CNN, CNN의 정확도를 나타낸다. 보행자의 수가 3명부터 25명까지 증가하면서 각 모델의 정확도가 측정되었다. 보행자만을 인식하기 때문에 각 모델의 정확도는 80% 이하로 감소하

지 않았다. 따라서 비슷한 정확도를 가지면서 연산 시간이 가장 빠른 VESCM을 사용하는 것은 실시간성과 정확성을 모두 만족할 수 있다.

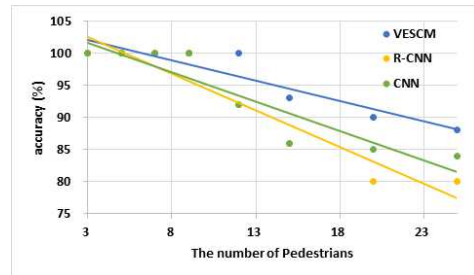


그림 5. VESCM, R-CNN, CNN의 정확도
Fig. 5. Accuracy of the VESCM, R-CNN and CNN

5. 결론

본 논문에서 차량 외부의 상황과 차량 내부의 센서 데이터를 이용하여 차량이 현재 처한 위기 상황을 0~1 사이의 값으로 출력하는 차량 위기 감지 시스템을 제안했다. 차량 위기 감지 시스템은 두 개의 모듈로 구성된다. 첫 번째 모듈인 차량 외부 상황 수집 모듈은 차량의 전방 카메라와 CNN을 이용하여 현재 주행 중인 차량의 주변 차량 정보와 보행자 정보를 수집한다. 두 번째 모듈인 차량 위기 상황 판단 모듈은 차량 외부 상황 수집 모듈의 결과와 차량 내부의 센서 데이터를 이용하여 차량이 현재 처한 상황의 위험도를 0~1 사이로 출력한다. 본 논문은 차량 위기 감지 시스템의 유효성을 판단하기 위하여 차량 외부 상황 수집 모듈에 대한 실험을 진행했다. 차량 위기 감지 시스템에 사용된 VESCM의 실시간성과 정확성을 입증하기 위하여 VESCM과 같은 조건으로 CNN, R-CNN을 구축하여 정확성과 연산 속도를 비교하였다. R-CNN은 보행자 수가 적을 때 VESCM과 비슷한 연산 시간을 보이지만, 보행자 수가 많아질수록 VESCM보다 많은 연산 시간을 소요했다. 평균적으로 VESCM은 R-CNN보다 25.68%, CNN보다 45.54% 더 빠른 연산 시간을 가졌고, 세 모델의 정확도는 모두 80% 이하로 감소하지 않으며 높은 정확도를 보였다. 그러나 해당 연구는 모두 가상 데이터를 통해 실험을 진행했으며, 긴급상

황 탐지 이후 차량의 주행을 결정하지 못했다. 해당 연구 이후 실제 차량에서 긴급상황을 탐지하여 유효성을 정확하게 입증해야 한다.

REFERENCES

- [1] Self-driving car accident, there were many collisions with general cars that followed, Young-joo Kim, Available online: <https://news.joins.com/article/23718449>
- [2] Ji Hoon Lee, Dae Youb Kim, "A Study on Low-Overhead Collision Warning Scheme using Vehicle-to-Vehicle Communications", *Journal of Korea Multimedia Society*, Korea Multimedia Society, Vol. 15, No.10, pp. 1221-1227, 2012
- [3] Sang Jun Park, Kwan Joong Kim, "A study of design mechanism for the alerting car accident", *Journal of Korea Academy Industrial Cooperation Society*, Korea Academy Industrial Cooperation Society, Vol.12, No.11, pp.5272-5277, 2011
- [4] Donghoon Shin, Kyongsu Yi, Yeonhwan Jeong, "Human Factor Considered Risk assessment of Automated Vehicle through Vehicular Communication", *Proceedings of the 2017 Fall Conference of the Korean Society of Mechanical Engineers*, The Korean Society of Mechanical Engineers, pp. 1563-1568, 2017
- [5] Kim Byeong Su, Noh Jun Ho, Park So Young, "Design of Automatic Risky Situation Detection Model based on Real-Time Driving Information", *Proceedings of Symposium of the Korean Institute of communications and Information Sciences*, Korea Institute Of Communication Sciences, pp. 783-784, 2017
- [6] Geun Hyung Min, Woo Seok Kim, Jun Sang Cho, Heung Bae Gil, "Vehicular Collision Risk Assessment on the Highway Bridges in South Korea", *Journal of the Korea Institute for Structural Maintenance and Inspection*, The Korea Institute for Structural Maintenance and Inspection, Vol.20, No.5, pp.009-017, 2016

- [7] MiRa Jeong, Byoung Chul Ko, Jae Yeal Nam, "Detection of sudden pedestrian crossings using the thermal camera installed in the vehicle", *Proceedings of the 2014 Spring Conference of the Korean Institute of Information Scientists and Engineers*, The Korean Institute of Information Scientists and Engineers, pp.668-670, 2014

저자약력

손 수 략(Su-Rak Son)

[정회원]



- 2018년 2월 : 가톨릭관동대학교 컴퓨터학과(공학사)
- 2019년 8월 : 가톨릭관동대학교 컴퓨터학과(공학석사)
- 2019년 8월 ~ 현재 : 가톨릭관동대학교 컴퓨터학과(공학박사 재학)

〈관심분야〉 빅데이터, 네트워크, 프로그래밍 언어

정 이 나(Yi-na Jeong)

[정회원]



- 2011년 2월 : 가톨릭관동대학교 컴퓨터학과(공학사)
- 2013년 8월 : 가톨릭관동대학교 컴퓨터학과(공학석사)
- 2018년 8월 : 가톨릭관동대학교 컴퓨터학과(공학박사)
- 2017년 3월 ~ 현재 : 가톨릭관동대학교 컴퓨터학과 초빙교수

〈관심분야〉 빅데이터, AI, 프로그래밍 언어