

교통소외지역 양방향 단일차선에서 인프라 엣지를 이용한 자율주행 차량 상황 인지 기술 구현

¹김성중, ^{2*}송석일

Implementation of Autonomous Vehicle Situational Awareness Technology using Infrastructure Edge on a Two-way Single Lane in Traffic-isolated Area

¹Seongjong Kim, ^{2*}Seokil Song

요 약

이 논문에서는 교통소외지역의 양방향 단일 차선에서 자율주행 차량의 안전하고 원활한 운영을 위한 센서 데이터 공유 시스템을 제안하고 핵심 모듈인 상황인지 기술을 구현한다. 양방향 단일 차선 도로는 주차된 차량이나 마주오는 차량으로 인한 자율주행 차량의 후진 문제를 야기한다. 이 논문에서는 인프라 카메라를 사용하여 양방향 단일 차선 도로에 대한 차량의 접근, 진입 진출 상황을 실시간으로 인지하고 이 정보를 V2N 통신을 통해 자율주행 차량에 전송하여 자율주행 차량의 센싱 범위를 확장하는 시스템을 제안한다. 제안하는 시스템의 핵심은 인프라 카메라를 통해 양방향 단일 차선의 상황을 인지하는 것이며 이 논문에서는 이를 객체인식 및 객체 추적기술을 이용하여 구현한다. 마지막으로, 구현한 상황인지 기술을 실제 양방향 단일 차선에서 수집한 데이터를 이용하여 검증한다.

Abstract

In this paper, we propose a sensor data sharing system for the safe and smooth operation of autonomous vehicles on two-way single lanes in traffic-isolated areas and implement the core module, the situational awareness technology. Two-way single lanes pose challenges for autonomous vehicles, particularly when encountering parked vehicles or oncoming traffic, leading to reversing issues. We introduce a system using infrastructure cameras to detect vehicles' approach, enter, and leave on two-way single lanes in real-time, transmitting this information to autonomous vehicles via V2N communication, thereby expanding the sensing range of the autonomous vehicles. The core part of the proposed system is the situational awareness of the two-way single lane using infrastructure cameras. In this paper, we implement this using object detection and tracking technology. Finally, we validate the implemented situational awareness technology using data collected from actual two-way single lanes.

Keywords: Two-way single lane, Situational awareness, Object detection, Object tracking, Autonomous vehicle

¹ 한국교통대학교 컴퓨터공학과 석사과정 (ksj78561@u.ac.kr)

^{2*} 교신저자 한국교통대학교 컴퓨터공학과 교수 (sisong@u.ac.kr)

I. 서론

C-ITS (Cooperative Intelligent Transportation System)는 차세대 지능형 교통 플랫폼으로 V2X 통신 기술을 기반으로 한다. C-ITS의 목표는 차량과 인프라 사이에서 양방향으로 데이터를 교환해서 다양한 도로 문제에 신속하게 대응하고 예방하는 것이다. 자율주행 기술이 발전하면서 자율주행 차량의 인지 및 판단을 도와서 보다 안전한 운행을 지원하는 새로운 C-ITS 서비스에 대한 수요가 증가하고 있다[1]. 3GPP(3rd Generation Partnership Project)에서는 자율주행과 관련된 C-ITS 서비스를 군집 주행, 고도 주행, 센서 데이터 공유, 원격 주行的 네 가지 유형으로 분류하고 있다. 또한, 각 유형에 대한 요구사항을 같이 정의하고 있다[2]. 이 중 센서 데이터 공유는 자율주행 차량과 도로 인프라가 각자의 센서로 수집한 데이터와 영상을 공유해서 차량의 상황 인지 가능 범위를 확장하는 것이다. 센서 데이터 공유는 자율주행 차량이 보다 넓은 주변 환경을 정확하게 인식할 수 있도록 하여 주행 안전성 향상 및 교통 흐름의 효율성 향상이 가능하다.

최근, 교통 소외 지역의 이동권 확보를 위한 수단으로 자율주행 서비스가 부각되고 있으며 이를 위한 다양한 연구가 진행되고 있다[3][4][5][6]. [3]에서는 교통소외지역에 자율주행 서비스를 제공하기 위해 운행설계영역(ODD) 확장을 위한 기술적 고려 사항들을 분석하였다. [4]에서는 교통소외지역의 자율주행기반 이동지원 서비스 이용자의 요구사항을 도출하는 연구를 진행하였다. [5]에서는 교통소외지역에서 교통약자를 대상으로 하는 자율주행기반 이동지원 서비스 구현 및 실용화에 필요한 정책 요구사항을 연구하였다. [6]에서는 교통소외지역에서 자율주행 4 단계 기술을 기반으로 수요자 응답형 모빌리티 서비스 제공을 위해 필요한 요구사항을 연구하였다. 이 연구에서는 교통소외지역의 특징을 비정형도로, 좁은 차선, 도로 표식의 노후화, 도심과 다른 이동객체 형태 및 도로 환경을 들고 있으며 도출한 교통소외지역의 특징을 고려하는 자율주행 시스템 아키텍처를 제안하고 있다.

이상 기술한 바와 같이 교통소외지역을 위한 자율주행 서비스가 연구 개발되고 있고 실용화를 앞두고 있지만 교통소외지역의 도로 특성상 자율주행 차량 단독으로 해결할 수 없는 문제가 존재한다. 특히, 그림 1 과 같은 교통소외지역에서 흔히 존재하는 양방향 통행이 가능한 비정형의 좁은 도로(양방향 단일 차선)에서 자율주행 차량이 도로에 주차된 차량이나 농기계, 마주오는 차량을 마주하면 원활한 교통이 어려워진다. 최근에 자율주행 기술 발전으로 후진이 가능한 차량도 있지만, 여전히 후진이 어려운 차량도 있기 때문에 이 경우에는 원격제어 등 사람이 직접 개입해서 회피해야 한다. 이 논문에서는 이러한 문제를 보완하기 위해서 교통소외지역의 양방향 단일차선에서 주차나 차량 진출입 상황을 실시간으로 인지하고 이를 해당 도로에 진입하려는 자율주행 차량에 공유하는 센서 데이터 공유 시스템을 제안하고 핵심기술인 양방향 단일차선에 대한 상황인지 기술을 구현한다.



Figure 1. Example of a two-way single lane in a traffic-isolated area.

이 논문에서 제안하는 양방향 단일 차선 센서 데이터 공유 시스템은 카메라 등의 인프라 센서를 사용해서 자율주행 차량이 양방향 단일 차선에 진입하기 전에 반대쪽에서 접근하거나

이미 진입한 차량을 식별하여 상황을 인지한다. 이 상황 인지 정보는 자율주행 차량에 전송되고, 차량은 자기 상태를 기반으로 적절한 판단을 한다. 그림 2 는 이 논문에서 개발하는 양방향 단일차선 센서 공유 시스템의 개념도를 보여준다. 이 그림 에서처럼 이 논문에서는 양방향 단일차선에 지오펜스(Geo-Fence)[7]를 설정하고, 이 지오펜스의 양쪽 진출입 지점에 엣지(Edge) 장치가 결합된 인프라 카메라를 설치한다. 이 인프라 카메라는 감지 영역 내의 모든 이동 객체를 추적한다. 추적된 정보를 바탕으로 지오펜스에 접근, 진입, 진출하는 차량 또는 객체의 상황을 파악하며, 이 정보는 V2N(Vehicle-to-Network) 통신을 통해 자율주행 차량에 근 실시간(Near Realtime)으로 전송된다. 이를 통해 자율주행 차량은 해당 양방향 단일차선의 교통 상황을 더 정확하고 신속하게 파악 할 수 있다.

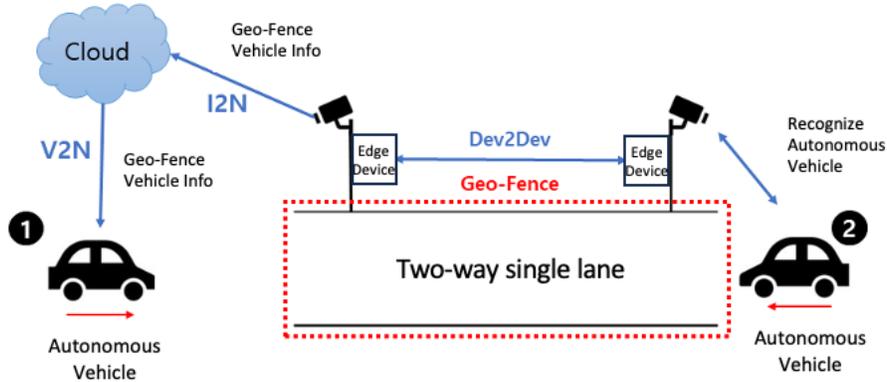


Figure 2. Conceptual diagram of the developed sensor sharing technology

II. 제안하는 양방향 단일차선 센서 데이터 공유 시스템

이 논문에서는 양방향 단일차선에서 운행되는 자율주행차량을 위한 센서 데이터 공유 시스템을 제안하고 핵심 기능인 상황인지 모듈을 구현한다. 제안하는 센서 데이터 공유 시스템은 지오펜스를 활용하여 양방향 단일차선을 포함하는 모니터링 영역을 정의한다. 그리고, Yolov4[8], DeepSort[9]등의 객체인식 및 객체 추적 기술을 이용하여 모니터링 영역 내의 이동 객체의 위치, 방향, 그리고 속도를 지속적으로 분석하여 지오펜스에 대한 차량들의 접근, 진입, 그리고 진출 상황을 인지한다.

양방향 단일차선 센서 데이터 공유 시스템의 전체적인 아키텍처는 그림 3 과 같다. 노변에 설치된 엣지장치는 연결된 카메라에서 영상을 수집하고 이를 분석하여 상황인지를 수행하고 결과를 다른 엣지장치나 클라우드로 전송하기 위한 V2X 모듈이 포함된다. 상황인지를 위해서는 먼저 영상에서 객체 인식 모델을 이용하여 각 이미지 프레임의 객체를 인식하고 인식한 객체들을 이용하여 객체 추적을 수행한다. 이 논문에서는 객체 인식 모델로 Yolov4 를 사용하며 객체 추적을 위해서는 DeepSort 를 이용한다. 상황인지 모듈은 DeepSort 에 의해서 생성된 궤적 데이터 (연속적으로 추적된 객체 정보 시퀀스)를 이용하여 지오펜스에 해당 객체가 접근하고 있는지(Approach), 진입 했는지(Enter), 진출 했는지(Leave)를 판단한다.

엣지장치는 20ms 주기로 상황인지 정보를 생성하고 이를 인근 엣지장치나 클라우드로 전송한다. 또한, 다른 엣지장치로부터 상황인지 메시지를 수신한 경우, 해당 정보와 자체 상황인지 정보를 통합하여 양방향 단일차선 내에 존재하는 차량의 총 수를 추가적으로 파악한다. 이 정보는 이웃하는 엣지장치로부터 수신한 정보와 함께 V2N (Vehicle-to-Network)을 통해 주변 차량에 다시 전송되며, 후진이 불가능한 자율주행 차량은 이를 바탕으로 양방향 단일차선 진입 여부를 결정한다. 엣지장치가 전송하는 상황인지 정보에는 이동 객체의 timestamp, speed, location 이 포함되며, 선택적으로 status (Approach, Enter, Leave) 정보도 포함한다.

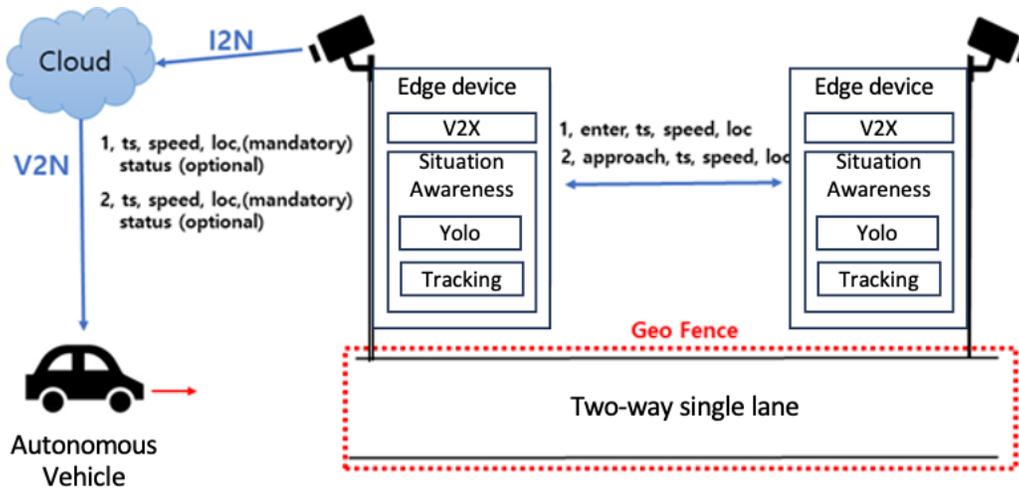


Figure 3. Architecture of the developed technology for recognizing a two-way single lane situation

그림 4 는 각 객체의 상태 (지오펜스에 대한 Enter, Approach, Leave 상태) 를 판별하는 상황인식 알고리즘이다. 상황인식 알고리즘은 객체 인식 모듈로부터 지속적으로 객체를 가져오고 (line 4) 이 객체에 대한 궤적을 객체 추적 모듈로부터 가져온다(line 6). 다음으로 궤적을 이용해 해당 객체의 이동방향 (heading)을 계산한다(line 8). 또한, 지오펜스의 양 측 중 가까운 측의 두개의 모서리와 객체의 현재 위치의 각도를 구한다(line 9 ~ 11). 마지막으로 객체의 현재위치가 지오펜스에 포함되는지 여부와 이동방향과 두 모서리와의 각도를 비교하여 상태를 판별한다. 제안하는 상황인지 알고리즘은 이동방향을 정확하게 계산하는 것이 매우 중요하다. 이동방향에 의해서 상황을 인식하기 때문이다. 실험에서 확인한 바로는 이동방향의 계산에는 상황에 따라 오류가 발생할 수 있다. 2차원 객체인식 및 객체인식의 오류로 인한 위치 정확도 저하가 발생할 때 특히 오류가 발생하게 된다. 또는 지오펜스로 진입하는 방향으로 이동하다가 멈추거나 지나가 버리는 경우에도 오류가 발생할 수 있다.

1. input : GF = $((x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3))$ // GF : locations of the four corners of a geo-fence
2. while (True)
3. // get next detected object from object detection module
4. obj = GetDetectedObj()
5. // get obj's trajectory consists of past n positions from object tracking module
6. objtraj $[(x_{t-n}, y_{t-n}), \dots, (x_{t-1}, y_{t-1}), (x_t, y_t)]$ = GetTrajectory(obj) // t is current time
7. // compute vehicle's heading
8. objheading = $\frac{\text{objtraj}.x_t - \text{objtraj}.x_{t-1}}{\text{objtraj}.y_t - \text{objtraj}.y_{t-1}}$
9. NC[2] = get two corners of the near side of GF
10. NC_0_angle = compute angle between NC[0] and (objtraj. x_t , objtraj. y_t)
11. NC_1_angle = compute angle between NC[1] and (objtraj. x_t , objtraj. y_t)
12. if (objtraj. x_t , objtraj. y_t) in GF
13. status = 'enter'
14. elseif (NC_0_angle <= objheading <= NC_1_angle) or (NC_1_angle <= objheading <= NC_0_angle)
15. status = 'approach'
16. else
17. status = 'leave'

Figure 4. Situation-awareness algorithm

III. 구현 및 실험

이 논문에서는 제안하는 양방향 단일차선 상황인지 시스템의 프로토타입을 구현하고 실험을 수행하여 타당성을 검증하였다. 실험을 위해 한국교통대학교 정문 주변의 양방향 1차선 도로와 유사한 환경을 실험 구역으로 선정하고 데이터를 수집하였다. 2022년 9월 2일부터 2022년 9월 7일까지 총 1만 여장 (1.1GB)의 이미지를 수집하였다. 수집한 데이터에 대해 통행에 방해가 될 수 있는 객체들을 모두 "vehicle"로 어노테이션 하였다. 1만 여장 중 실제 어노테이션이 수행된 이미지는 총 1,800 장이다.



Figure 5. Data collection area (yellow rectangle is a geofence)

어노테이션된 1,800 장의 학습데이터를 이용하여 Darknet Yolov4[10]의 사전 학습된 모델을 학습하였다. 학습에 사용한 Yolo 모델의 설정 값은 그림 6 과 같다. 학습된 Yolov4 모델과 Deep Sort 방법을 이용하여 카메라로 촬영하는 영상에 대해 실시간으로 객체를 추적하고 추적한 객체들의 궤적을 분석하여 상황인지를 수행하는 상황인지 모듈을 구현한다. 그림 7 은 구현한 상황인지 모듈의 실제 동작 과정을 보여준다. 그림 7 의 (a)는 ID 가 11 인 차량이 노란색 지오펜스에 접근(Approach 상태)하고 있는 것을 인지하였으며 이 때 위치는 이미지 픽셀 기준 (1242, 210) 임을 보여준다. 그림 7 의 (b)는 ID 가 19 인 차량이 지오펜스에 접근(Approach 상태)하고 있는 것을 인지하였으며 이 때 위치는 이미지 픽셀 기준 (1225, 208)에 있음을 보여준다. 또한, ID 가 11 인 차량은 지오펜스 내에 진입(Enter 상태)한 상태임을 인지 하였으며 위치는 이미지 픽셀 기준 (1197, 292)에 있음을 보여주고 있다.

- batch=64
- subdivisions=16
- width=416
- height=416
- max_batches = 2000
- steps=1600,1800
- classes = 1
- pre-train = yolov4.conv.137

Figure 6. Configuration of Yolo V4 model training

그림 7 의 (c)는 ID 가 19 인 차량이 지오펜스에 접근하다가 진입(Enter 상태)한 상황을 인지하였으며 이 때 위치는 이미지 픽셀 기준으로 (1075, 379)에 있음을 보여준다. 또한 ID 가 11 인 차량 역시 접근 상태에서 진입(Enter 상태) 한 상황을 인지한 것을 보여준다. 그림 7 의 (d)는 ID 가 19 인 차량이 여전히 진입(Enter 상태) 한 상황을 인지하고 있음을 보여주며 ID 가 11 인 차량에 대해서는 진입했던 지오펜스를 벗어나는(Leave 상태) 상황을 인지한 것을 보여준다. 마지막으로 그림 7 의 (e)는 ID 가 19 인 차량이 진입 상태에서 지오펜스를 벗어나는 (Leave 상태) 상황을 인지하는 것을 보여주고 있다.

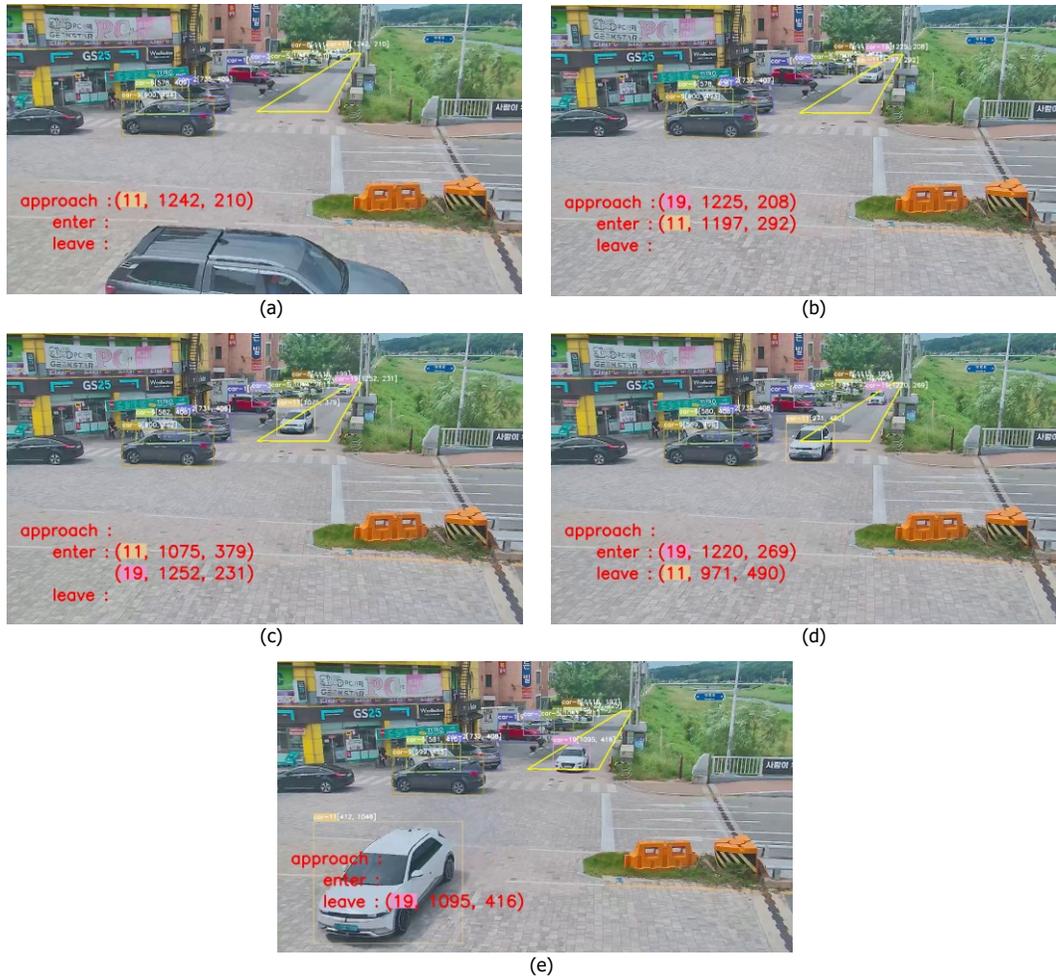


Figure 7. Example of the operation of the implemented situation-awareness module

지금까지 이 논문에서 제안하는 양방향 단일차선 센서 데이터 공유 시스템의 핵심인 상황인지 모듈의 구현 내용에 대해서 설명하였다. 이 논문에서는 구현한 상황인지 모듈의 정확도를 측정하기 위한 실험을 수행하였다. 실험을 위해 9 건의 영상 클립을 준비하였다. 9 건의 영상 각각을 입력으로 하여 상황인지 모듈을 실행하였고 각 이미지 프레임에서 각 객체의 실제 상태와 상황인지 모듈이 인지한 상태를 비교하여 혼동행렬(Confusion Matrix)를 작성하고 이를 기반으로 정확도(Accuracy), 재현율(Recall), 정밀도(Precision), F1-점수(F1-Score)를 측정하였다.

표 1 은 실험에 사용된 하드웨어와 소프트웨어 환경을 보여준다. 실험은 도로에 설치된 인프라 카메라로 촬영한 비디오 클립을 GPU 가 장착된 서버에 저장하고 실험이 시작되면 저장된 비디오 클립 9 개를 순차적으로 읽어서 상황인지 모듈에 입력하면서 상황인지를 수행한다. 상황인지 모듈은 9 개의 비디오 클립에 대해서 상황인지를 수행하여 각 객체별로 지오펜스에 대해 접근, 진입, 진출 상태를 출력한다.

Table 1. Experimental environments (HW, SW, Data Set)

DIVISION	CONTENT
HW	<ul style="list-style-type: none"> CPU : Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz Memory : 32GB * 4 GPU : NVIDIA Corporation GP102 TITAN Xp OS : ubuntu 20.04 LTS server
SW	<ul style="list-style-type: none"> Python 3.8.10, Tensorflow 2.3.0
Data Set	<ul style="list-style-type: none"> Training data: 1,300 images collected from KNUT test bed Test data: 9 video clips (500 images) collected from KNUT test bed

상황인지 모듈의 출력을 이용하여 표 2 와 같이 비디오 클립별로 각 상태 (Approach, Leave, Enter)에 대한 혼동행렬을 생성하였다. 그리고, 혼동행렬을 이용하여 각 상태 클래스에 대한 성능을 계산하였다. 또한 각 비디오 클립에 대해 모든 상태 클래스에 대한 평균 성능을 계산하였다. 마지막으로, 모든 비디오 클립의 평균 성능을 측정하였다. 전체 평균은 정확도 0.9247, 재현율 0.8871, 정밀도 0.8871, F1-점수 0.8871 이다. 상태 클래스 별로 측정된 성능을 살펴보면 접근 0.8005, 진출 0.9058, 진입 0.9312 이다. 비디오 클립별로 살펴봐도 전반적으로 접근 클래스에 대한 재현율과 정밀도는 0.6 에서 1.0 사이로 편차가 매우 크며 다른 클래스에 비해서 전반적으로 낮은 수치를 보인다. 접근 클래스의 성능이 낮은 원인은 차량이 지오펜스 방향으로 이동한 후 이미지의 편의점 부근에 주차를 하거나 다시 지오펜스에서 멀어지는 상황이 발생할 때 이를 제대로 인식하지 못한 것이 원인으로 분석된다. 또한, 객체 인식을 2 차원에서 수행하므로 일부 객체에 대한 이동방향(Heading) 계산이 정확하게 되지 않는 경우가 있는 것으로 보인다.

진출 클래스의 경우에 5 번 비디오에서 정확도 0.6689, 재현율 0.4600, 정밀도 0.5169, F1-점수 0.4868 로 매우 낮은 것을 확인할 수 있었다. 영상을 확인해 본 결과 지오펜스에 진입한 차량이 진출하기전에 경계선에서 한동안 멈춰 있는 것이 원인으로 분석된다. 경계선에서 일단 정지한 후 매우 천천히 진출하는 과정에서 진출, 진입 클래스를 번갈아 가면서 인식된 것이 원인이 된 것으로 보인다. 이때 진입 클래스도 FN(False Negative) 이 27로 다소 높게 측정된 것을 볼 수 있다. 즉, 진입임에도 이를 진출로 인식해서 발생한 오류이다.

이 실험에서 진출 클래스의 경우 진출을 진입 클래스로 인식하더라도 자율주행차량에는 큰 문제가 되지 않는다. 자율주행 차량은 진출임에도 진입으로 인식한 경우에는 양방향 단일차선에 진입하지 않고 정지하면 되기 때문이다. 경계선을 명확히 지나가면 다시 진출로 인식이 되므로 자율주행 차량은 다시 양방향 단일차선에 진입할 수 있다. 접근 클래스의 경우에는 접근임에도 이를 진입 클래스나 접근 클래스로 인식하지 못하게 되면 문제가 발생할 수 있다. 자율주행 차량은 접근 중인 차량이 있음에도 상황을 제대로 인식하지 못하고 양방향 단일차선 진입을 할 수 있기 때문이다. 이동방향 계산 오류로 인해 진입 상황을 제대로 인식하지 못하는 것은 자율주행 차량에 문제가 될 수 있으므로 이에 대한 개선이 필요해 보인다.

Table 2. Confusion matrix and performance for each video clip

Clip No.	Status	TP	TN	FP	FN	Accuracy	Recall	Precision	F1-Score
1	Approach	71	130	0	3	0.9853	0.9595	1.0000	0.9793
	Leave	48	153	3	1	0.9805	0.9796	0.9412	0.9600
	Enter	81	120	1	0	0.9950	1.0000	0.9878	0.9939
	Average					0.9869	0.9804	0.9804	0.9804
2	Approach	116	416	38	0	0.9333	1.0000	0.7532	0.8593
	Leave	229	341	0	0	1.0000	1.0000	1.0000	1.0000
	Enter	187	345	0	38	0.9333	0.8311	1.0000	0.9078
	Average					0.9556	0.9333	0.9333	0.9333
3	Approach	69	487	24	0	0.9586	1.0000	0.7419	0.8519
	Leave	221	359	0	1	0.9983	0.9955	1.0000	0.9977
	Enter	266	290	1	24	0.9570	0.9172	0.9963	0.9551
	Average					0.9713	0.9570	0.9570	0.9570
4	Approach	108	214	29	0	0.9174	1.0000	0.7883	0.8816
	Leave	111	240	0	0	1.0000	1.0000	1.0000	1.0000
	Enter	103	219	0	29	0.9174	0.7803	1.0000	0.8766
	Average					0.9449	0.9174	0.9174	0.9174
5	Approach	81	142	54	16	0.7611	0.8351	0.6000	0.6983
	Leave	46	150	43	54	0.6689	0.4600	0.5169	0.4868
	Enter	69	197	0	27	0.9078	0.7188	1.0000	0.8364
	Average					0.7793	0.6689	0.6689	0.6689
6	Approach	188	694	24	100	0.8767	0.6528	0.8868	0.7520
	Leave	371	534	96	5	0.8996	0.9867	0.7944	0.8802
	Enter	315	652	12	27	0.9612	0.9211	0.9633	0.9417
	Average					0.9125	0.8688	0.8688	0.8688
7	Approach	112	414	33	12	0.9212	0.9032	0.7724	0.8327
	Leave	273	285	12	1	0.9772	0.9964	0.9579	0.9767
	Enter	141	398	0	32	0.9440	0.8150	1.0000	0.8981
	Average					0.9475	0.9212	0.9212	0.9212
8	Approach	93	228	6	61	0.8273	0.6039	0.9394	0.7352
	Leave	118	200	70	0	0.8196	1.0000	0.6277	0.7712
	Enter	101	272	0	15	0.9613	0.8707	1.0000	0.9309
	Average					0.8694	0.8041	0.8041	0.8041
9	Approach	67	325	17	29	0.8950	0.6979	0.7976	0.7444
	Leave	187	213	30	8	0.9132	0.9590	0.8618	0.9078
	Enter	137	291	0	10	0.9772	0.9320	1.0000	0.9648
	Average					0.9285	0.8927	0.8927	0.8927
Aggregate	Approach	1039	3447	252	266	0.8965	0.7962	0.8048	0.8005
	Leave	1823	2804	299	80	0.9243	0.9580	0.8591	0.9058
	Enter	1578	3192	14	219	0.9534	0.8781	0.9912	0.9312
	Average					0.9247	0.8871	0.8871	0.8871

IV. 결론

이 논문에서는 자율주행 차량의 주행 안전성 확보를 위해 양방향 단일차선의 상황인지를 위한 센서 데이터 공유 기술을 제안하고 핵심 기술인 상황인지 모듈을 구현하고 실험을 통해서 가능성을 확인하였다. 개발한 양방향 단일차선 상황인지 모듈은 객체 인식 및 객체 추적기술을 이용하여 양방향 단일차선에 접근하거나 진입한 차량들에 대한 궤적을 실시간으로 추적하고 각 궤적에서 도출한 차량의 이동방향과 위치를 이용해서 상황인지를 수행한다. 개발한 상황인지 모듈의 정확도를 측정하기 위해 실제 양방향 단일차선에서 수집한 데이터를 이용하여 실험을 수행하였다. 실험은 양방향 단일차선에 차량이 접근(Approach), 진입(Enter), 진출(Leave) 하는지 인지하고 인지한 결과와 직접 육안으로 관찰한 상황과 비교를 통해서 혼동 행렬을 작성하고 이를 바탕으로 정확도, 재현율, 정밀도, F1-점수를 측정한다. 측정 결과 F1-점수는 0.8871 이었다. 분석결과 양방향 단일차선의 경계에서 멈추거나 느리게 진행하는 차량에 대해 상황인지를 제대로 못하는 경우가 발생하였고 이에 따라 정확도 감소가 발생하는 것을 볼 수 있었다. 향후 연구에서는 딥러닝을 이용해서 차량의 궤적을 분석하여 상황을 인지하는 방법을 개발한다.

V. 감사의 글

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant 23AMDP-C160549-02).

VI. 참고문헌

- [1] A. Vinel, N. Lyamin, and P. Isachenkov, "Modeling of V2V communications for C-ITS safety applications: A CPS perspective," *IEEE Commun. Lett.*, Vol. 22, No. 8, pp. 1600-1603, Aug. 2018.
- [2] S. Maaloul, H. Aniss, M. Kassab, and M. Berbineau, "Classification of C-ITS services in vehicular environments," *IEEE Access*, Vol. 9, pp. 117868-117879, 2021.
- [3] A. Kim, B. Woo, S. Tak, and S. Lim, "Comparison Analysis of the Road Environment between Urban and Suburban Area for Connected and Automated Driving(CAD) Mobility Services," *J. Korea Inst. Intell. Transp. Syst.*, Vol. 21, No. 5, pp. 287-300, Oct. 2022.
- [4] A. Kim and S. Lim, "A study to improve the effectiveness of mobile services in traffic underprivileged area based on autonomous vehicles using the KANO model," in *Proc. of the 88th Conf. Korean Soc. Transp.*, April 2023, pp. 467-468.
- [5] G. Eom and S. Jang, "Policy Requirements for Implementing Mobility Services with Autonomous Vehicles for the Disabled," in *Proc. of the 88th Conf. of Korean Soc. of Transp.*, April. 2023, pp. 168-169.
- [6] Y. S. Song and J. D. Choi, "Analysis of Level 4 Autonomous Driving System Requirements and Architecture for Mobility Services in Poor Transport Provision Area," in *Proc. KSAE 2021 Annu. Autumn Conf. Exhib.*, Nov. 2021, pp. 391-391.
- [7] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A Review of Yolo algorithm developments," *Procedia Comput. Sci.*, Vol. 199, pp. 1066-1073, 2022
- [8] A T. L. Dang, G. T. Nguyen, and T. Cao, "Object tracking using improved deep SORT YOLOv3 architecture," *ICIC Express Lett.*, Vol. 14, No. 10, pp. 961-969, Oct. 2020.
- [9] "YOLO-ROS Docker Image," Docker Hub, [Online]. Available: <https://hub.docker.com/r/cjh2626002/yolo-ros/tags>
- [10] M. Stevens and E. Atkins, "Geofence definition and deconfliction for UAS traffic management," *IEEE Trans. Intell. Transp. Syst.*, Vol. 22, No. 9, pp. 5880-5889, 2020.

저자소개



김성중 (Seongjong Kim)

2022 년 2 월 한국교통대학교 컴퓨터공학과(공학사)
2022 년 3 월 ~ 현재 : 한국교통대학교 컴퓨터공학과 석사과정

관심분야: 인공지능, 이동객체, 컴퓨터 비전 등



송석일(Seokil Song)

2000 년 2 월 충북대학교 정보통신공학과(공학석사)
2003 년 2 월 충북대학교 정보통신공학과(공학박사)
2003 년 7 월 ~ 현재 : 한국교통대학교 컴퓨터공학과 교수

관심분야 : 데이터베이스, 스토리지 시스템, 분산 병렬 시스템, 빅데이터, 딥러닝 등