Journal of KIIT. Vol. 17, No. 8, pp. 93-105, Aug. 31, 2019. pISSN 1598-8619, eISSN 2093-7571 93
http://dx.doi.org/10.14801/jkiit.2019.17.8.93

RGB 영상 및 LiDAR 포인트 클라우드 합성을 통한 YOLO 기반 실시간 객체 탐지

김진수*, 조정호**

YOLO-based Real-Time Object Detection Scheme Combining RGB Image with LiDAR Point Cloud

Jinsoo Kim*, Jeongho Cho**

이 논문은 2018 정부(교육부)의 재원으로 한국연구재단 (No.2018R1D1A3B07041729) 및 순천향대학교 학술연구비의 지원을 받아 수행된 연구임

요 약

차량 스스로의 판단만으로 도로의 주행을 목표로 하는 자율주행의 구현을 위해 다양한 객체 탐지 알고리즘을 통한 실시간 주행환경 감지 연구가 활발히 진행되고 있다. 이를 위해 일반적으로 RGB 카메라를 통한 객체 탐지가 이루어지고 있지만, 주행환경 감지 성능 향상을 위해 또 다른 감지 센서와의 융합을 통한 상호보완이 이루어지고 있는 추세이다. 따라서, 본 논문에서는 객체 탐지 성능 고도화 및 실시간 감지를 위해 RGB 영상 데이터와 LiDAR 포인트 클라우드의 합성을 통한 YOLO 기반의 객체 탐지 시스템을 제안한다. 제안된 시스템의 성능평가를 위해 KITTI Benchmark Suite을 활용하였으며, 그 결과 RGB 카메라를 단독으로 활용하였을 때보다 훨씬 우수한 객체 탐지율을 보여주었으며 이로써 낮은 미검출율을 가능하게 할 수 있음을 확인하였다.

Abstract

In order to realize autonomous driving, detection studies on real-time driving environment through various object detection algorithms are actively being carried out. For this purpose, object detection is generally performed by RGB cameras. However, in order to improve the sensing performance of the driving environment, it is being complemented by fusion with other sensors. Therefore, in this paper, we propose a YOLO-based object detection system by combining RGB image data and LiDAR point cloud for object detection enhancement and real-time detection. The KITTI Benchmark Suite was used to evaluate the performance of the proposed system. As a result, the object detection rate was much better than that of the RGB camera alone, which enabled a low detection rate.

Keywords
RGB image, YOLO, LiDAR point cloud, real-time, object detection

* 순천향대학교 전기통신시스템공학과
- ORCID: http://orcid.org/0000-0002-7669-1384
** 순천향대학교 전기공학과 조교수(교신저자)
- ORCID: http://orcid.org/0000-0001-5162-1745

· Received: Jun. 24, 2019, Revised: Jul. 09, 2019, Accepted: Jul. 12, 2019
· Corresponding Author: Jeongho Cho
Dept. of Electrical Engineering, Soonchunhyang University, Asan, Korea,
Tel.: +82-41-530-4960, Email: jcho@sch.ac.kr

Ⅰ. 서 론

운전자의 개입이 필요 없이 도로를 주행하는 완전 자동화를 목표로 실시간 주행환경 감지에 관한 자율주행의 연구가 활발하게 진행되고 있다. 주행환경 감지는 안전과의 매우 밀접한 관계로 인해 자율주행의 필수적인 분야로 자리 잡았으며 심층학습 알고리즘의 접목을 통해 큰 성능 향상이 이루어졌다[1]. 심층학습 알고리즘은 신경망 구조를 바탕으로 많은 양의 입력 데이터에 대한 학습을 진행하며, 특히 컨볼루션 신경망(CNN, Convolutional Neural Network) 구조가 제안된 이후 자율주행의 주행환경 감지에도 적극적으로 활용되고 있다[2][3].

RGB 카메라는 기본적으로 사람의 시각과 유사하게 사물의 형태와 색상을 인식하여 기본적인 객체 탐지 성능이 높다. 하지만 사물로부터 반사된 가시광선을 영상 데이터로 나타내기 때문에 조명, 날씨, 사물의 잘림 등의 외부환경적 요인에 취약하다는 단점을 가진다. 또한, RGB 카메라를 통해 탐지한 객체의 정확한 3차원 거리 정보를 획득하기에는 많은 어려움이 있다. 따라서 최근에는 객체 탐지의 성능을 높이기 위해 라이다(LiDAR, Light Detection And Ranging)를 RGB 카메라와 함께 사용하여 한계점을 보완하는 많은 연구가 진행되고 있다[4].

라이다는 레이저를 방출하여 측정 범위 내의 사물들로부터 반사된 신호를 포인트 클라우드 데이터(PCD, Point Cloud Data)로 나타낸다. 센서 자체에서 파생한 레이저로부터 반사된 신호를 측정하기 때문에 가시광선을 측정하는 RGB 카메라와는 다르게 외부환경적 요인에 강인하다는 장점을 가진다. 또한, 표면 성질에 따른 반사율 정보와 반사된 시간에 따른 거리 정보를 포함하여 객체와의 정확한 거리 측정이 가능하다. 그러나 반사된 레이저 신호만을 측정하기 때문에 반사 영역에만 포함되는 환경정보를 나타내며 이로 인해 PCD로 표현되는 데이터의 해상도는 영상 데이터의 10% 이내로 매우 작아 실제 환경의 정보를 모두 표현하는 데에 한계를 갖는다[5].

이처럼 RGB 카메라와 라이다는 상호보완적인 장단점을 가지고 있어 이들 센서의 정보를 융합하여 객체 탐지 성능을 고도화하는 센서 융합 기술의 제안이 활발하게 이뤄지고 있다[6]. [7]에서는 영상 데이터와 PCD를 바탕으로 서포트 벡터 머신(SVM, Support Vector Machine)을 통해 특징들을 추출하여 하나의 단일벡터로 결합한 후, 결합한 단일 벡터를 변형 가능한 모델의 입력 데이터로 활용하여 객체 탐지 결과를 융합함으로써 보행자의 탐지 성능을 개선하였다. [8]에서는 영상 데이터와 PCD를 기반으로 세분화 기법을 통해 객체를 추론하고 컨볼루션 기반으로 특징 맵을 활용하여 의사 결정 수준에서 분류된 출력을 확률 기반으로 융합함으로써 보행자, 차량, 자전거의 다중 객체 분류 성능을 개선하였다. 이외에도 [9]에서는 영상 데이터와 PCD에서 추출한 객체가 존재할법한 후보 지역의 이미지에서 추출한 특징을 융합한 후 객체 탐지 모델을 학습하여 자동차를 탐지하는 방식도 제안되었다. 이처럼 각각의 신호를 다른 모델로 처리한 후 객체 탐지 결과를 융합하는 방식을 통해 객체 탐지 성능의 향상이 이루어지고 있지만, 실제 차량의 자율주행 중 큰 사고로 이어질 수 있는 객체의 미검출(Missed-detection)에 대한 탐지 성능 개선에 관한 연구는 상대적으로 미비하다. 또한 [7]과 같이 SVM 등의 머신러닝 기반 객체 탐지 알고리즘을 활용하는 방식은 실시간으로 객체를 탐지하는 데에 어려움을 겪는다.

이에 본 논문에서는 실시간 객체 탐지에 적합한 YOLO(You Only Look Once: Real-Time Object Detection)를 활용하여 영상 데이터와 PCD를 바탕으로 독립적으로 객체 탐지를 실행한 후 각각의 결과를 융합하여 미검출에 대한 탐지 성능이 향상된 YOLO 기반의 적응형 객체 탐지 시스템을 제안한다. 반사율 및 거리 정보를 포함하는 PCD와 영상 데이터를 바탕으로 CNN 기반의 3가지 YOLO에 대한 객체 탐지 학습을 각각 실행하고 각 모델에서의 객체에 대한 경계상자와 신뢰도 점수를 예측한다. 이후 객체 탐지 결과를 융합하기 위해 경계상자의 좌표를 해당 객체에 대한 신뢰도 점수를 기반으로 가중평균을 통해 최종 경계상자를 결정한다. 이로써 더 높은 신뢰도 점수를 가진 모델의 경계상자에 가깝게 최종 경계상자의 좌표가 결정된다.

제안된 객체 탐지 시스템의 성능평가를 위해 자율주행 벤치마킹 플랫폼 ‘KITTI Benchmark Suite’

[10]를 활용하여 자동차를 대상으로 객체 탐지를 진행하였다. 제안된 가중평균을 통한 센서 융합 결과 RGB 카메라를 단독적으로 활용할 때보다 훨씬 우수한 객체 탐지율을 보였으며, 어느 한 YOLO 모델이 객체의 탐지를 놓치는 경우에도 전체 모델로 부터의 탐지 결과를 가중함으로써 미검출율의 저하를 가능하게 할 수 있었다.

II. RGB 영상 기반 객체 탐지

2.1 CNN 기반의 객체 탐지

기존의 영상 신호처리 분야에서의 객체 탐지는 영상 데이터에서 객체의 특징을 사전에 추출하고 해당 특징을 기반으로 객체를 탐지하는 방식으로 진행되었다. 특징점을 찾기 위해 영상 내부의 지역적인 특징점들을 추출하는 SIFT(Scale Invariant Feature Transform), 분할된 영상의 에지의 방향을 히스토그램으로 나타내는 HOG(Histogram of Oriented Gradients) 등이 활용되었으며 추출된 특징을 기반으로 기계 학습의 전통적인 분류 알고리즘인 SVM 등이 객체 탐지에 적용되었다[11][12]. 하지만 영상처리 기반의 방식은 객체 탐지 성능에 직접적인 영향을 미치는 특징을 직접 찾는 과정이 필요하다는 단점을 내재하고 있다.

CNN의 등장으로 신경망이 자체적으로 특징을 추출하고 학습하는 종단간 학습이 가능해짐으로써 객체 탐지의 큰 성능 개선이 이루어졌다. CNN 기반의 객체 탐지 알고리즘은 크게 지역 기반과 단일 회귀 방식 두 종류로 나뉜다. 지역 기반의 방식은 대표적으로 객체가 존재할법한 후보 관심 영역(ROI, Region of Interest)을 생성하고 해당 영역에서 특징을 추출하여 분류 알고리즘과 경계상자에 대한 회귀학습을 통해 ROI 내부의 객체를 탐지하는 R-CNN이 있으며, 기존의 객체 탐지 알고리즘에 비해 높은 성능 향상을 보였다. 하지만 특징 추출, 분류의 단계가 나뉘어 있으며 각각의 ROI를 CNN에 입력하여 추출한 특징들을 개별적으로 학습해야 하므로 학습에 많은 시간이 소요된다는 단점을 가진다.

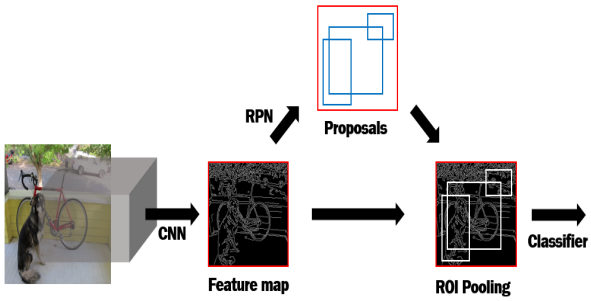


그림 1. Faster R-CNN의 블록 다이어그램
Fig. 1. Block diagram of faster R-CNN

이와 같은 단점을 보완하기 위해 학습과 탐지 속도가 향상된 Fast R-CNN과 Faster R-CNN이 제안되었다. Fast R-CNN은 ROI에서 분류기와 경계상자의 손실을 동시에 학습하는 멀티태스크 학습을 통해 CNN의 연산 과정과 학습 단계를 단순화하여 학습 소요 시간을 감소시켰으며, Faster R-CNN은 CNN의 마지막 계층에 ROI를 생성하는 영역 제안 네트워크(RPN, Region Proposal Network)를 적용하여 학습 속도를 더욱 빠르고 탐지 성능 또한 높였다. 그러나, 여전히 지역 기반 방식의 객체 탐지 알고리즘은 ROI를 생성하고 영역 내부의 객체를 분류하는 두 가지 작업을 순차적으로 진행하기 때문에 탐지 성능은 우수하지만, 탐지 속도가 느리다는 단점을 가진다[13]-[15].

한편, 단일 회귀 방식은 ROI를 찾지 않고 영상 데이터 전체에 대하여 객체의 경계상자 예측과 분류를 동시에 진행하기 때문에 실시간에 근접한 빠른 속도로 탐지한다는 장점을 가진다. 이러한 방식으로는 YOLO, SSD(Single Shot Detector) 등이 존재한다[16][17].

2.2 YOLO

YOLO는 입력된 이미지 내부의 객체에 대한 경계상자의 예측과 분류를 동시에 실행하는 통합탐지(Unified detection)를 특징으로 한다. YOLO에 입력되는 영상 데이터는 해상도에 따라 $S \times S$개의 격자 구역으로 나뉘고 CNN 구조의 신경망을 통해 특징이 추출되며, 완전 연결 노드(Fully connected layer)를 통해 최종적으로 그림 2와 같이 예측 텐서(Prediction tensor)가 출력된다.

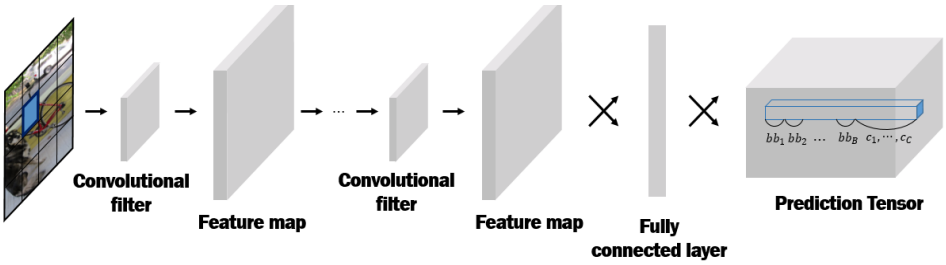


그림 2. YOLO의 네트워크 구조
Fig. 2. Structure of YOLO network

예측 텐서는 $(S\times S)$의 크기와 $(B\times 5+C)$의 길이를 가진다. 여기서, $S\times S$은 격자 구역의 개수, B는 중심점이 격자 구역 내부에 포함된 후보 경계상자의 개수, C는 분류할 수 있는 객체의 개수를 의미한다. 각각의 격자 구역은 $(B\times 5+C)$의 길이를 가지는 벡터로 나타나며 $S\times S$개의 격자 구역의 집합이 $S\times S\times(B\times 5+C)$의 예측 텐서를 구성한다.

격자 구역은 B개의 경계상자를 예측하는데 경계상자는 (x,y,w,h,S_{conf})의 5가지 정보를 포함한다. (x,y)는 경계상자의 중심좌표, (w,h)는 폭과 높이, S_{conf}는 식 (1)과 같이 경계상자에 객체가 포함될 확률인 $Pr(Object)$와 경계상자가 얼마나 정확하게 경계상자를 예측했는지를 나타내는 실제값(Ground-truth)과 교차영역의 상대적인 넓이(IOU, Intersection of Union)인 IOU_{pred}^{truth}와의 곱을 의미한다. 실제값과 예측한 경계상자의 중심좌표가 같은 격자 구역에 포함된 경우에 경계상자에 객체가 포함된 것으로 간주하며 $Pr(Object)$는 1로 계산되고 각각 다른 격자 구역에 포함되는 경우에는 0으로 계산된다.

$$S_{conf}=Pr(Object)\times IOU_{pred}^{truth} \tag{1}$$

$$IOU_{pred}^{truth}=\frac{area(b.b_{pred}\cap b.b_{truth})}{area(b.b_{pred}\cup b.b_{truth})} \tag{2}$$

IOU는 두 영역의 교차영역의 넓이를 합의 영역의 넓이로 나눈 값으로 식 (2)와 같으며 실제값의 경계상자($b.b_{truth}$)에 대해 예측한 경계상자($b.b_{pred}$)의 정확도를 평가하기 위해 사용되는 지표이다. 또한 격자 구역은 경계상자 내부에 포함된 객체의 종류가 분류할 수 있는 C개의 객체 중 어떤 객체일지를 나타내는 조건부 확률을 계산하여 식 (3)과 같이 P_{class}로 나타낸다.

$$P_{class}=Pr(Class|Object) \tag{3}$$

이와 같이 $(B\times 5+C)$의 길이를 가지는 텐서가 $(S\times S)$의 모든 격자 구역에 대한 예측을 진행한 이후에는 식 (4)를 통해 경계상자에 객체가 포함되는 확률인 S_{conf}와 포함된 객체가 어떤 객체일지를 나타내는 P_{class}를 객체를 분류하기 위한 신뢰도 점수(CS_{conf})로 확장한다.

$$\begin{aligned} CS_{conf}&=S_{conf}\times P_{class} \\ &=Pr(Object)\times IOU_{pred}^{truth} \\ &\quad\times Pr(Class|Object) \\ &=Pr(Class)\times IOU_{pred}^{truth} \end{aligned} \tag{4}$$

식 (1) 의 S_{conf}와 식 (3)의 P_{class}를 곱함으로써 예측한 경계상자 내부에 객체가 포함될 확률과 분류한 객체가 실제값과 일치하는 확률을 모두 나타내는 CS_{conf}가 계산된다. 최종적으로 분류한 객체에 대하여 입력 텐서의 예측된 B개의 경계상자 중에서 가장 높은 CS_{conf}를 가진 경계상자가 해당 객체의 경계상자로 선택된다[16].

III. YOLO 기반 센서 융합 객체 탐지 시스템

본 논문에서 제안하는 가중평균 기반의 YOLO 기반 객체 탐지 시스템은 데이터 전처리부와 센서 융합부로 구성된다.

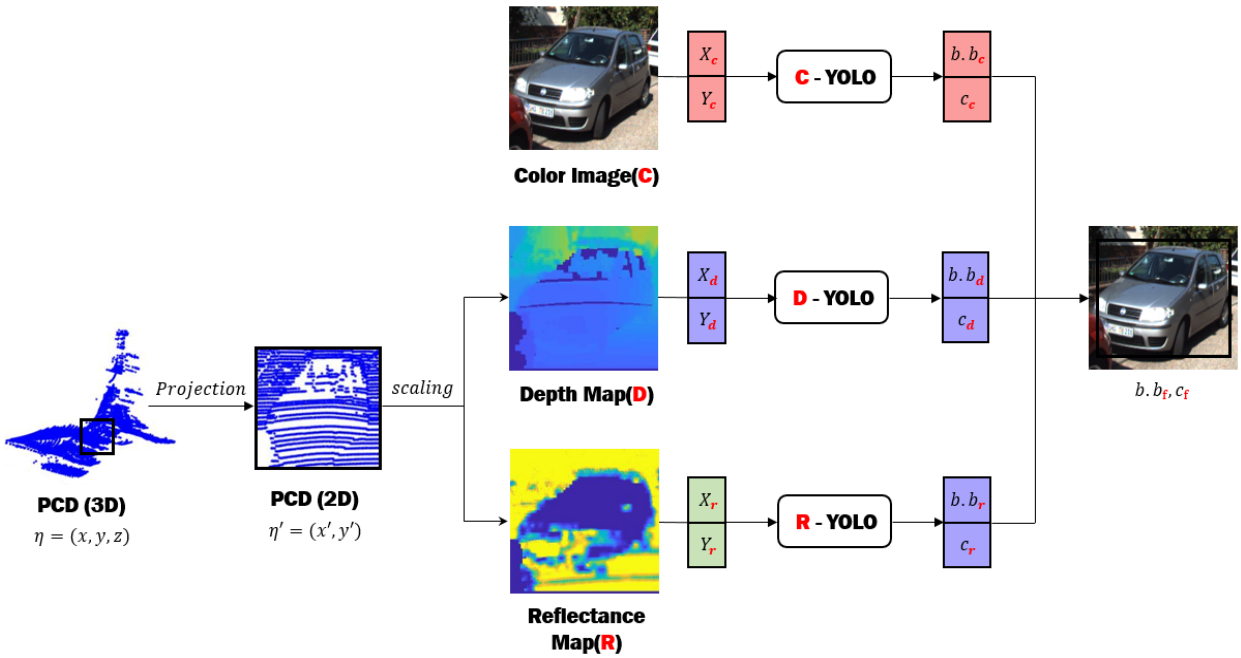


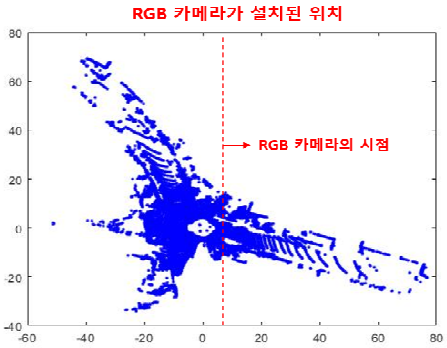
그림 3 YOLO 기반 센서 융합 객체 탐지 시스템의 블록 다이어그램
Fig. 3. Block diagram of sensor fused object detection system based on YOLO

전처리 과정에서는 3차원 공간 정보를 나타내는 PCD를 RGB 카메라의 시점과 동일하게 맞춰주는 좌표보정을 통해 2차원 공간에 투영한다. 투영과정을 거친 이후에는 PCD가 포함하는 거리, 반사율 정보에 따라 깊이 맵과 반사율 맵을 생성하여 객체 탐지에 활용한다. 센서 융합 과정에서는 RGB 카메라의 영상 데이터, 전처리된 PCD의 깊이 맵과 반사율 맵을 각각 YOLO 기반의 모델을 통해 객체를 탐지하고 가중평균을 적용하여 경계상자의 좌표와 크기를 조정한다. 라이다에서 파생된 레이저 신호는 다른 감지 센서보다 높은 펄스를 가져 장거리의 측정이 가능하며 센서 자체에서 파생한 신호로부터 반사된 정보를 측정하기 때문에 외부환경적 요인에 강인하다는 장점을 가진다.

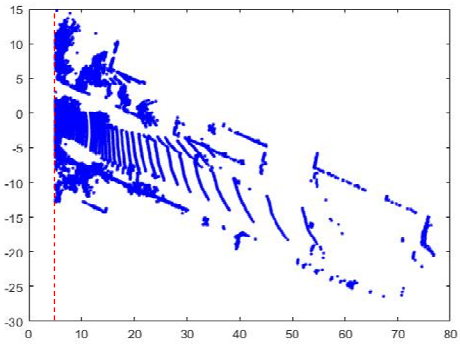
3.1 데이터 전처리

라이다는 반사된 레이저 신호를 $\eta = (x, y, z, r)$의 3차원 좌표값(x, y, z)과 반사율 정보(r)를 제공하는데, 반사율 정보는 지면 및 물체의 반사면의 거친 정도, 색상 및 재질 등에 따라 반사된 신호의 강도를 의미한다. 이를 활용한 객체 탐지는 3차원 좌표값을 그대로 사용하거나 이를 탑뷰 또는 전면뷰의 2차원 공간으로 투영시켜 객체를 탐지하는 경우로 나뉜다. 탑뷰를 활용한 객체 탐지는 차량의 진행 방향 및 운동 속도를 추출하기 용이하지만 객체 탐지의 연산 과정이 복잡한 반면, RGB 카메라와 운전자가 바라보는 시점과 동일한 전면뷰를 활용한 객체 탐지는 탑뷰를 활용한 객체 탐지 대비 연산이 간단하다[18].

본 논문에서는 PCD를 RGB 카메라의 시야각(FOV, Field Of View)과 동일한 전면뷰로 투영하는 변환과정을 통해 PCD의 차원과 좌표계를 영상 데이터와 같은 2차원 픽셀 좌표계로 변환하여 활용한다. 픽셀 좌표계는 영상 데이터에 포함된 픽셀의 2차원 기준 좌표계를 의미하며 영상 데이터의 좌측 상단 모서리를 기준으로 우측 방향은 x의 증가 방향, 하단 방향은 y의 증가 방향을 의미한다. PCD는 라이다를 기준으로 그림 4(a)와 같이 360°의 전 방향에서 취득된 데이터를 나타내기 때문에 RGB 카메라의 FOV에서 표현되는 PCD만 그림 4(b)와 같이 분리한다. 그림 4(a)에서 라이다의 위치는 원점, RGB 카메라의 위치는 라이다로부터 x축이 5만큼 이동한 지점이며 FOV의 중심축은 x축과 평행한 방향이므로 $x > 5$의 조건을 만족하는 PCD만 분리하여 활용한다.

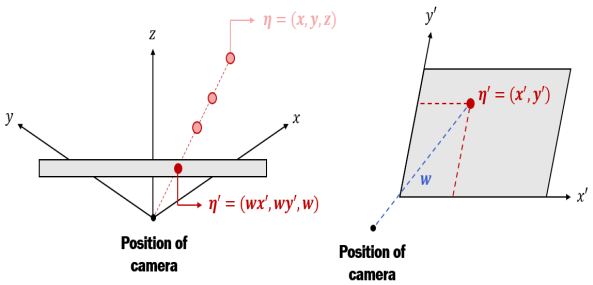


(a) 라이다의 PCD(탑뷰)
(a) PCD from a LiDAR(top view)



(b) 추출된 라이다의 PCD(탑뷰)
(b) Extracted PCD from a LiDAR(top view)

(c) RGB 카메라의 영상 데이터
(c) Image from a RGB camera
그림 4. 임의의 학습 데이터
Fig. 4. Arbitrary training data

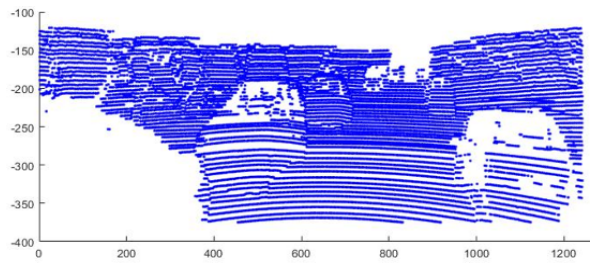


(a) (b)
그림 5. 3차원 PCD의 투영 과정, (a) 투영된 2차원 PCD(동차좌표), (b) 투영된 2차원 PCD(픽셀 좌표)
Fig. 5. Projection process of 3D PCD, (a) Projected 2D PCD(homogeneous coordinate), (b) Projected 2D PCD(pixel coordinate)

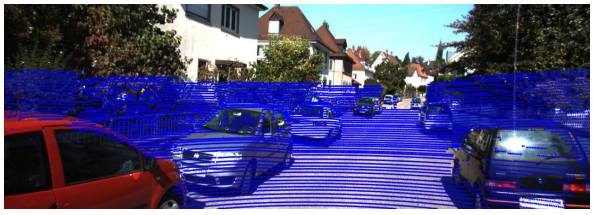
분리된 PCD의 3차원 좌표계는 영상 데이터의 픽셀 좌표계와 다르기 때문에 투영 변환을 통해 3차원 공간에서의 PCD를 2차원 픽셀 좌표계로 투영한다. 그림 5(a)와 같이 분리된 PCD에서 3차원 좌푯값을 추출한 후 투영 행렬을 곱하여 2차원 평면의 한점으로 투영되는 η'를 구한다. η'는 동차좌표로 나타나기 때문에 $\eta' = (wx', wy', w)$로 표현할 수 있는데 동차좌표란 (x,y)를 0이 아닌 w에 대하여 차원을 확장하여 (wx, wy, w)로 표현되는 것으로 투영 변환을 통해 3차원 공간의 좌표가 2차원으로 투영되었을 때 2차원 좌표는 3차원으로 확장된 동차좌표 형태로 나타난다.

따라서 η'는 2차원 좌표의 차원이 카메라의 위치와 η의 거리를 나타내는 w에 대한 동차좌표이므로 그림 5(b)와 같이 2차원 좌표에 w를 곱해주면 $\eta' = (x', y')$와 같이 픽셀 좌표로 변환되며 이를 $H = (X, Y)$라 정의한다.

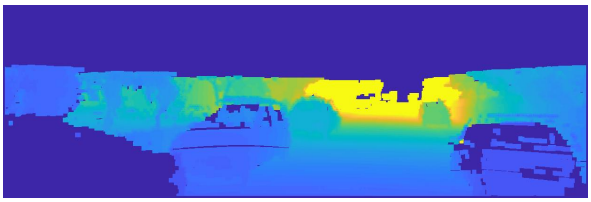
그림 6(a)에는 2차원 픽셀 좌표계에서의 PCD를 도식화하였으며 그림 6(b)를 통해 RGB 카메라의 FOV와 동일한 전면뷰로 PCD가 투영된 것을 확인할 수 있다. 하지만 PCD는 영상 데이터와 비교하여 해상도가 낮아 데이터의 정보가 희소하게 나타나기 때문에 양자 필터(Bilateral filter)[19]를 이용하여 고해상도로 샘플링한 후 객체 탐지에 활용한다.



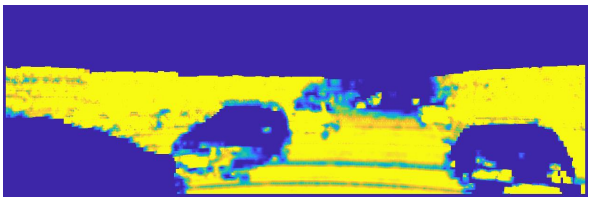
(a) 픽셀 좌표계로 변환된 PCD
(a) PCD converted to a pixel coordinate



(b) 영상 데이터에 투영된 PCD
(b) PCD projected to a image
그림 6. 이미지 평면에 투영된 PCD
Fig. 6. Projection of PCD on a image plane



(a) 깊이 맵
(a) Depth map



(b) 반사율 맵
(b) Reflectance map
그림 7. 투영된 고해상도 PCD의 스케일링
Fig. 7. Scaling of projected high-resolution PCD

양자 필터는 후광 현상(Halo Artifact)을 억제하여 영상내부의 가장자리를 보존하면서 이미지를 흐리게 하거나 노이즈를 완화하는 비선형 필터이다. 이를 활용해 PCD가 존재하는 픽셀과 인접한 픽셀들이 가지는 거리, 반사율 정보로 나타나는 가중된 픽셀값 을 인접한 픽셀들의 픽셀값에 적용하여 그림 7과 같이 스케일링 된 고해상도의 깊이 맵과 반사율 맵을 생성하고 각각의 맵이 가지는 픽셀의 좌표를 $H_d=(X_d, Y_d)$, $H_r=(X_r, Y_r)$, 영상 데이터가 가지는 픽셀의 좌표를 $H_c=(X_c, Y_c)$라 정의한다.

3.2 영상 및 포인트 클라우드 기반 YOLO 학습

전처리 과정 이후에는 영상 데이터, 깊이 맵, 반사율 맵을 바탕으로 각각의 객체 탐지 모델 C-YOLO, D-YOLO, R-YOLO를 통해 학습을 진행한다. CNN의 구조는 24개의 컨볼루션 계층과 2개의 완전 연결 계층으로 구성하였으며 격자 구역의 크기와 개수를 결정하는 $S=7$, 각각의 격자 구역이 예측하는 경계상자의 개수인 $B=2$, 탐지할 객체는 자동차를 선정하여 $C=1$로 설정하였다. 영상 데이터, 깊이 맵, 반사율 맵으로 분류된 데이터를 개별적으로 학습을 진행하였기 때문에 각각의 데이터에 대해 최적화된 파라미터를 이용하여 독립적으로 객체 탐지가 진행된다. 학습된 객체 탐지 모델은 데이터에 포함된 객체의 위치와 크기를 나타내는 경계상자의 정보 $b.b_k=(x_k, y_k, w_k, h_k)$와 경계상자 내부의 분류된 객체가 정답일 확률을 나타내는 CS_{conf}인 c_k를 출력한다($k=c,d,r$). 모델을 통해 객체를 탐지한 이후에는 탐지 결과를 가중 평균을 통해 융합한다.

가중 평균은 데이터의 중요도를 나타내는 변수를 가중치로 반영한 평균값으로, 학습된 YOLO 기반 모델의 객체 탐지 결과는 (x, y, w, h)의 경계상자의 기하학적 정보와 탐지한 객체가 실제값과 일치하는 지를 나타내는 CS_{conf}로 나타난다. CS_{conf}는 객체가 분류된 확률의 신뢰성을 반영하여, 높은 CS_{conf}를 가지는 객체 탐지 결과의 경계상자는 실제값의 경계상자와 겹치는 면적이 넓어져 IOU가 높게 나타난다. 실제로 많은 객체 탐지 알고리즘에 실제값이 한 개의 객체를 포함할 때 객체 탐지 결과가 2개 이상인 경우, 가장 높은 CS_{conf}의 경계상자 이외의 다른 경계상자를 억제하는 비최대값 억제 알고리즘 적용되고 있다. 따라서 3가지 모델에서 탐지된 객체의 CS_{conf}를 가중하여 경계상자의 기하학적 정보의 평균값을 식 (5)와 같이 구한다.

$$b.b_f=\left(\frac{\sum_k x_k c_k}{\sum_k c_k}, \frac{\sum_k y_k c_k}{\sum_k c_k}, \frac{\sum_k w_k c_k}{\sum_k c_k}, \frac{\sum_k h_k c_k}{\sum_k c_k}\right) \quad (5)$$

여기서 $b.b_c=(x_c, y_c, w_c, h_c)$, $b.b_d=(x_d, y_d, w_d, h_d)$, $b.b_r=(x_r, y_r, w_r, h_r)$는 3가지 모델의 객체 탐지 결과로 나타나는 경계상자이다.

그림 8은 제안된 객체탐지 시스템 (WM-YOLO, Weighted Mean-YOLO)의 탐지 결과로 나타날 수 있는 5가지 상황에 대한 예시를 나타내었다. 이미지 평면에서 점선의 경계상자는 객체의 실제값, 실선의 경계상자는 시스템의 객체 탐지 결과이다. 시나리오 ①은 3가지 모델이 모두 객체를 탐지한 경우이므로 객체의 실제값에 대하여 3개의 경계상자 ($b.b_c$, $b.b_d$, $b.b_r$)와 $CS_{conf}(c_c, c_d, c_r)$가 출력된다. 각각의 경계상자가 가지는 (x_k, y_k, w_k, h_k) 값을 식 (5)와 같이 CS_{conf}로 가중하여 평균값을 구한다. 따라서 3개의 경계상자의 (x_k, y_k, w_k, h_k)와 CS_{conf}에 따라 가중된 평균값을 가져 높은 IOU를 가지는 하나의 경계상자로 나타나게 된다.

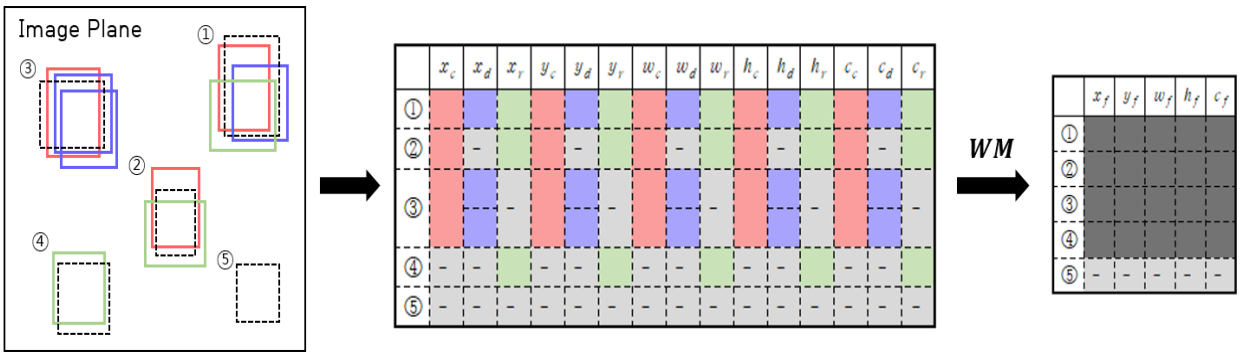
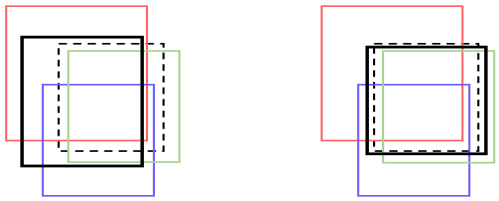


그림 8. 가중 평균 기반의 센서 융합 예시
Fig. 8. Example of sensor fusion based on weighted mean



(a) 좌표기반의 평균 (b) 가중 평균
그림 9. 경계상자 비교
Fig. 9. Comparison of bounding box
(a) Mean of coordinate base, (b) Weighted mean

가중 평균된 경계상자가 높은 IOU를 가지는 이유는 기존의 객체 탐지 모델의 결과에서 CS_{conf}가 높을수록 실제값의 경계상자와 겹치는 면적이 더 넓기 때문이다. 그림 9(a)와 같이 가중치를 사용하지 않고 3개의 경계상자의 평균을 구하는 경우 실제값과는 관계없이 각각의 경계상자의 기하학적 정보만을 기반으로 융합된다. 하지만 CS_{conf}를 가중치로 사용하여 3개의 경계상자의 평균을 구하는 경우 실제값과의 IOU가 반영되기 때문에 그림 9(b)와 같이 높은 IOU를 가지는 검은 실선의 경계상자를 얻을 수 있다.

시나리오 ②, ③, ④는 C-YOLO, D-YOLO, R-YOLO, 세 모델중 최소 1개 이상의 모델이 객체를 탐지한 경우이다. 예를 들어 시나리오 ②에서는 깊이 맵에서만 객체를 탐지하지 못하였으나 영상 데이터, 반사율 맵에서 탐지된 결과를 바탕으로 가중 평균을 통해 깊이 맵이 놓칠 수 있는 객체를 탐지할 수 있도록 보완함으로써 탐지 성능을 향상시킬 수 있게 된다.

Ⅳ. 실험 결과

본 논문에서 시험평가에 사용된 KITTI 데이터셋은 RGB 카메라와 Velodyne Lidar 등의 센서가 장착된 차량으로 도시 지역에서 추출되었으며 7481개의 시퀀스의 학습 데이터로 구성되어 있다. 학습 데이터는 9가지 객체의 종류와 51,867개의 라벨을 포함하고 있으며 이 중 55%(4,145개)는 학습, 45%(3336개)는 성능평가에 활용하였고 객체는 자동차로 선정하였다. 학습을 위한 신경망의 알고리즘은 YOLO를 선택하고 학습을 진행한 워크스테이션의 OS는 Ubuntu 16.04.5(4.15.0-38 kernel), GPU는 2개의 GTX 1080 Ti(11GB), 라이브러리는 Cuda V8.0.44, Cudnn 8.0, Opencv 3.4.0을 사용하였다. YOLO에서 입력받는 학습 이미지 데이터의 기본 크기는 416×416의 해상도로 설정되어 있는데 KITTI에서 제공하는 이미지 데이터의 기본 크기는 1392×512의 해상도를 가져 학습 결과에 좋지 않은 영향을 미칠 수 있다. 따라서 YOLO의 입력 데이터의 기본 크기를 1392×512로 변경하고 학습 횟수는 45,000회로 설정하였으며 각각의 YOLO 모델을 학습하는데 소요된 시간은 33시간이다. 성능평가를 위해 IOU 기반의 객체 탐지 성능 평가지표로 활용되는 평균 정밀도(AP, Average Precision)와 병렬 구조로 시스템을 구축한 경우의 처리시간을 확인하였다.

AP는 객체 탐지의 성능을 평가하는 지표로 객체를 탐지하지 못하는 미검출과 객체를 다른 객체로 탐지하는 오검출(False-alarm)을 동시에 고려하는 평가지표이다. AP를 계산하는 경우 미검출과 오검출

은 정밀도(Precision)와 재현율(Recall)로 정의되며 식 (6)과 같이 나타난다.

$$Precision = \frac{TP}{TP+FP} = \frac{TP}{all\ detections}$$
$$Recall = \frac{TP}{TP+FN} = \frac{TP}{all\ ground truths} \quad (6)$$

여기서, 탐지할 객체를 올바르게 탐지하는 경우는 TP(True Positive), 탐지하지 못한 경우는 FN(False Negative)로, 탐지할 객체 이외의 다른 객체를 탐지하지 않은 경우는 TN(True Negative), 탐지한 경우는 FP(False Positive)로 정의된다. 정밀도는 모든 검출 결과 중에서 객체를 올바르게 탐지한 경우의 비율을, 재현율은 모든 실제값 중에서 객체를 빠트리지 않고 탐지한 경우의 비율을 의미한다.

정밀도와 재현율은 IOU 값의 영향을 받으며, IOU를 조절하며 얻은 정밀도와 재현율을 곡선으로 나타낸 것을 AP 곡선이라 부르며 IOU에 따른 정밀도에 대한 재현율의 증가량의 곱(해당 곡선의 면적)을 나타낸 수치를 AP라 정의한다. 제안된 객체 탐지 시스템의 성능을 평가하기 위해 모든 객체에 대한 성능평가, KITTI 데이터 셋 기준의 성능평가, 외부 환경 변화에 대한 성능평가로 구분하여 진행하였다. 또한, KITTI 데이터 셋의 기준에 따라 지역 기반 방식의 탐지 알고리즘 중 빠른 탐지 속도를 가지는 Faster R-CNN[15]을 활용하여 RGB 카메라 기반 객체 탐지 결과 및 [9]과 같이 RGB 카메라와 라이다의 정보를 융합한 객체 탐지의 비교평가를 진행하였다.

제안된 시스템의 단계별 처리 시간을 그림 10에 나타내었다. 병렬 구조로 시스템을 구축하는 경우 입력된 데이터로부터 각각의 모델의 객체 탐지 결과를 융합하는데 평균 $77ms$의 시간이 소요되었다.

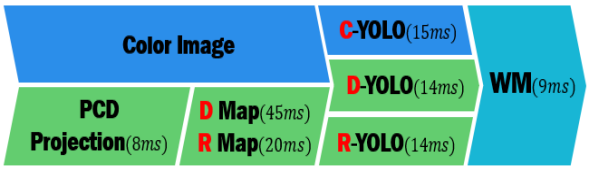


그림 10. 시스템의 단계별 평균 처리 시간
Fig. 10. Average processing time of the system

특히 각각의 객체 탐지 모델이 프레임 당 최대 $14ms$의 빠른 속도로 객체를 탐지하며, 프레임 당 $2s$의 속도로 객체를 탐지하는 Faster R-CNN 보다 자동차 탐지속도가 하는 훨씬 빠른 것을 확인 할 수 있었다.

4.1 단일·융합 객체 탐지 시스템 성능평가

본 논문에서 제안한 시스템은 C-YOLO, D-YOLO, R-YOLO의 객체 탐지 결과를 가중 평균을 통해 융합하여 객체 탐지의 성능을 고도화하는 것을 목적으로 한다. 따라서 단일 객체 탐지 시스템의 성능을 평가한 이후에 제안된 융합을 통한 객체 탐지 시스템의 시험평가를 통해 비교분석을 진행하였다. 단일 객체 탐지 시스템의 AP를 측정한 결과 영상 데이터로 학습한 C-YOLO가 84.31%로 가장 높게 나타났고 D-YOLO와 R-YOLO는 C-YOLO보다 약 15% 낮은 검출성능을 보였다.

데이터 전처리 과정에서 저해상도의 PCD를 양자 필터를 이용하여 고해상도로 샘플링 하였지만, 깊이 맵, 반사율 맵의 해상도는 영상 데이터의 35~45% 이하 수준으로 나타나기 때문에 영상 데이터와 비교하여 객체 정보의 희소성으로 인해 영상 데이터의 객체 탐지 성능이 가장 높게 측정되었다. 하지만 C-YOLO는 외부환경적 요인에 취약하기 때문에 그림자에 의해 배경이 어두워지는 경우, 장애물에 의하여 객체의 일부분이 가려진 경우에는 객체 탐지 성능이 저하되어, D- YOLO와 R-YOLO의 객체탐지 성능이 더 우수하였다. 결과적으로, 단일 객체 탐지 시스템의 탐지 결과를 가중 평균을 통해 융합한 결과 AP가 90.8% (IOU=0.7)로 향상되었으며, IOU에 따른 AP[%]를 표 1에 정리하였다.

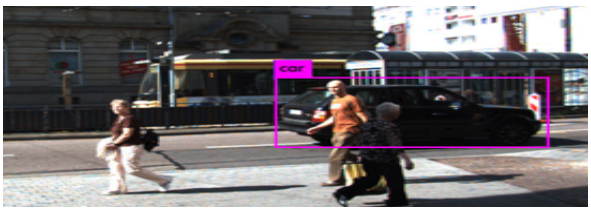
표 1. 단일·융합 객체 탐지 시스템 성능평가
Table 1. Performance evaluation of a single and combined object detection system

2D AP(%)			
	IOU=0.3	IOU=0.5	IOU=0.7
C-YOLO	89.50	87.97	84.31
D-YOLO	82.23	76.81	69.82
R-YOLO	80.47	75.02	67.79
WM-YOLO	95.05	93.80	90.80

(a) C-YOLO

(b) WM-YOLO

(c) C-YOLO



(d) WM-YOLO

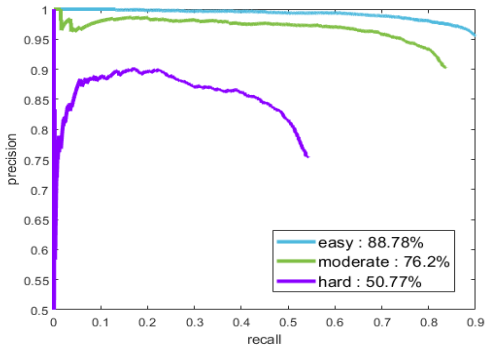
그림 11. 객체 탐지 결과

Fig. 11. Object detection results

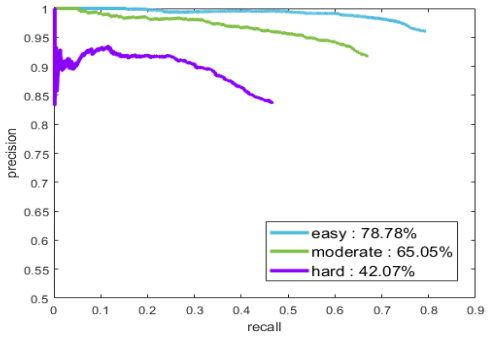
또한, 단일 객체 탐지 시스템의 결과가 서로 상이한 경우 이들의 융합을 통해 서로 보강됨으로써 성능이 향상되는 것을 확인할 수 있었으며 융합된 탐지 결과 예시는 그림 11에서 보여준다. C-YOLO에서 탐지하지 못한 경계상자를 D-YOLO와 R-YOLO는 각각 410개, 370개씩 탐지하였으며, D-YOLO에서 탐지하지 못한 경우 C-YOLO와 R-YOLO는 1,150개, 764개, R-YOLO에서 탐지하지 못한 경우 C-YOLO와 D-YOLO는 1,267개, 921개의 경계상자를 탐지하였다.

4.2 객체의 난이도에 따른 성능평가

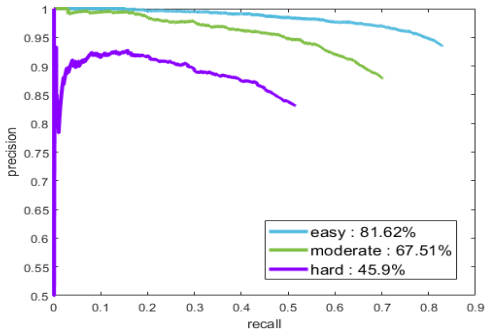
KITTI 데이터셋의 성능평가 방식은 탐지할 객체의 크기와 잘림 정도에 따라 'easy', 'moderate', 'hard'의 3가지 난이도로 나뉜다. 'easy'는 잘림 정도가 'fully visible', 픽셀의 높이가 최소 40픽셀, 'moderate'는 잘림 정도가 'partial occlusions', 픽셀의 높이가 최소 25픽셀, 'hard'는 잘림 정도가 'higer occlusions'이며 픽셀의 높이는 'moderate'와 같다.



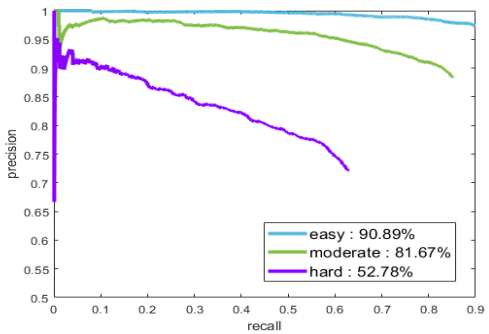
(a) C-YOLO



(b) D-YOLO



(c) R-YOLO



(d) WM-YOLO

그림 12. 난이도에 따른 정밀도와 재현율

Fig. 12. Precision and Recall according to degree of difficulty

그림 12에 C-YOLO, D-YOLO, R-YOLO와 제안된 WM-YOLO 를 통한 IOU가 0.7일 때의 AP를 3가지 난이도에 따 라 나타내었으며 타 시스템과의 성능 비교 결과를 표 2에 나타내었다.

표 2. 난이도에 따른 성능 비교평가
Table 2. Performance comparisons with respect to degree of difficulty

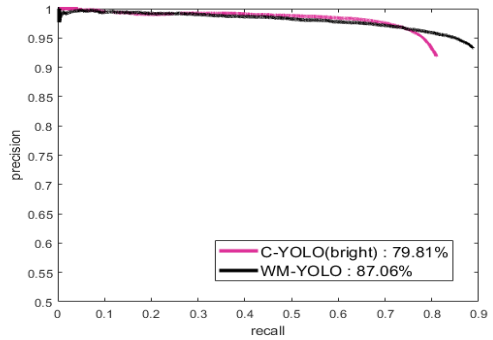
Detector	Detection time [ms]	2D AP(%)		
		easy	moderate	hard
Faster R-CNN[15]	2000	86.71	81.84	71.12
MV[9]	240	89.80	79.76	78.61
WM-YOLO	77	90.89	81.67	52.78

난이도에 따른 융합 결과 각각의 난이도에서 모두 WM-YOLO를 통한 객체 탐지의 성능이 향상된 것을 확인할 수 있었다. ‘easy’는 2.4%, ‘moderate’은 11.17%, ‘hard’는 25.48%로 가장 큰 성능 향상이 나타났다. 특히 ‘hard’에서는 독립적인 객체 탐지 모델의 성능은 비슷하게 나타났지만 가중 평균을 통해 WM-YOLO의 검출성능이 크게 향상되었다. 이러한 원인은 각각의 센서의 특징에 따라 탐지하는 객체가 다르기 때문이다. 영상 데이터는 0~255까지의 픽셀값을 가지기 때문에 표현되는 픽셀의 범위가 넓지만, 외부환경적 요인에 취약하다.

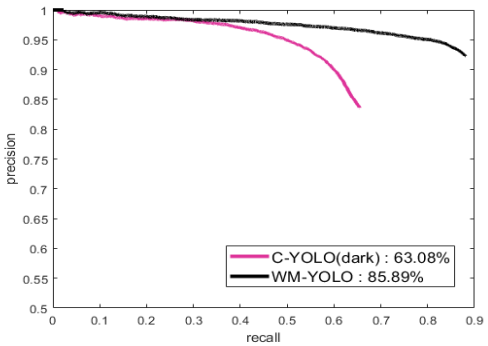
또한, 성능평가 결과 객체의 크기가 작거나 장애물에 의하여 객체가 잘린 경우 깊이 맵, 반사율 맵보다 객체 탐지 성능이 저하되는 것을 확인하였다. 이러한 이유는 깊이 맵과 반사율 맵의 해상도가 낮게 나타나 공간적인 특성을 가져 객체의 형태가 더 잘 표현되기 때문이다. 하지만 깊이 맵 및 반사율 맵은 픽셀이 거리, 반사율 정보에 따라 스케일링 되었기 때문에 영상 데이터보다 나타낼 수 있는 픽셀 값의 범위가 좁게 나타난다. 이처럼 각각의 센서의 탐지 성능이 독립적으로 나타나는 경우 ‘hard’와 같이 탐지하기 어려운 객체에 대하여 센서 융합을 통한 성능 향상이 두드러졌다. 난이도에 따른 융합된 탐지 결과와 타 시스템과의 성능 비교결과 제안한 시스템은 가장 빠른 탐지 속도를 보여주었으며 ‘easy’에서는 가장 높은 AP가 나타났지만, ‘hard’에서는 가장 낮은 AP가 나타났다. ‘hard’에서 낮은 AP를 얻게된 이유는 제안된 시스템의 YOLO가 이미지를 임의의 격자 구역으로 나누어 객체를 탐지하므로 여러 개의 객체가 겹치거나 크기가 작은 경우 탐지 성능이 낮기 때문으로 판단된다.

4.3 외부환경에 대한 강인성 평가

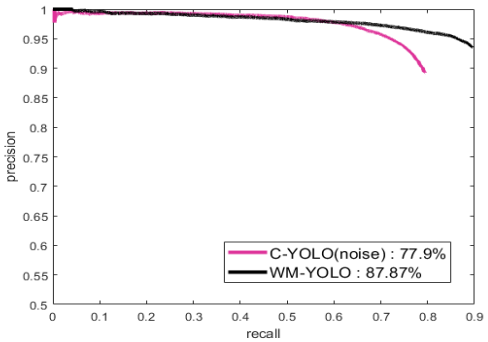
다음으로는 외부환경적 특성에 취약한 RGB 카메라의 한계점에 대한 성능 평가를 진행하기 위해 영상 데이터의 명암을 밝게 또는 어둡게 변화시키고, 가우시안 백색 잡음을 추가해 가며 다양한 환경 변화에 따른 AP를 확인하였다.



(a) 밝은 환경
(a) Bright environment



(b) 어두운 환경
(b) Dark environment



(c) 잡음이 존재하는 환경
(c) Noise-induced environment

그림 13. 외부환경 변화 고려시 정밀도와 재현율
Fig. 13. Precision and Recall considering external environmental changes

명암이 밝은 영상은 순간적으로 낙뢰가 발생하거나 다른 차량의 상향등의 영향을 받는 경우, 명암이 어두운 영상은 태양에너지가 존재하지 않는 터널 내부나 야간, 그리고 가우시안 백색 잡음의 경우는 눈, 비가 내리거나 안개가 낀 날씨의 외부환경을 묘사하기 위해 영상 데이터를 전처리하였다. 명암을 조절하기 위해 0~255로 나타나는 영상 데이터의 픽셀값 I에 명암을 조절하는 파라미터 δ를 적용하여 평균 픽셀값의 범위를 $mean(I)-\delta \sim mean(I)+\delta$ 수준으로 나타내었다. 그리고 실제 환경에서 나타날 수 있는 일반적인 잡음을 데이터에 추가하기 위해 평균이 0, 분산이 0.005인 가우시안 백색 잡음을 추가하여 시험 데이터를 생성하였다. 각각의 상황에 대하여 AP를 확인한 결과, 외부 환경적 요인에 의해 RGB 카메라를 통한 객체 탐지 결과에 악영향을 미쳤을지라도 라이다를 통한 객체 탐지 결과를 가중 평균함으로써 그림 13과 같이 C-YOLO보다 향상된 객체 탐지 결과를 얻을 수 있음을 확인하였다.

Ⅴ. 결론 및 향후 과제

본 논문에서는 자율주행에서의 객체 탐지 성능 고도화를 위하여 RGB 카메라, 라이다의 객체 탐지 결과의 융합을 통해 검출성능을 고도화하는 가중 평균 기반의 적응형 객체 탐지 시스템을 제안하였다. RGB 카메라의 영상 데이터와 라이다의 고해상도로 크기조정 된 PCD를 통해 거리, 반사율 정보에 따라 깊이 맵과 반사율 맵을 생성한 후 C-YOLO, D-YOLO, R-YOLO 모델을 통해 각각 객체 탐지를 진행하였다. 이후 가중 평균을 기반으로 하는 융합을 통하여 최종적인 검출성능 고도화 결과를 도출하였다. 특히 데이터의 해상도가 높으나 외부환경적 요인에 취약한 영상 데이터와 외부환경적 요인에 강인하지만, 해상도가 낮은 PCD의 객체 탐지 결과를 가중 평균을 통해 보강하였을 때 객체 탐지 성능이 향상되며 실시간에 적합한 처리 속도로 최종 객체 탐지 결과를 도출하는 것을 확인하였다. 또한, 실제 주행환경에서 외부환경적 요인의 영향을 고려한 경우에도 제안된 WM-YOLO를 통해 객체 탐지 성능이 향상되는 것을 확인하였다. 향후 보행자, 자전거 등의 객체를 추가로 학습한 후 문제점 분석결과를 바탕으로 3차원 예측 텐서의 스택기반 학습 또는 신뢰도 점수, 경계상자의 기하학적 정보 등의 효율적인 추출을 통해 시스템 성능보완을 지속해 나아갈 계획이다.

References

[1] H. Shin, H. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. Summers, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning", IEEE Trans. Medical Imaging, Vol. 35, No. 5, pp. 1285-1298, 2016.

[2] S. Nazeer, G. Yu, T. Tan Dat, D. Sin, and J. Kim, "Real-Time Implementation of Human Detection in Thermal Imagery Based on CNN", The Journal of KIIT, Vol. 17, No. 1, pp. 107-121, Jan. 2019.

[3] Y. Byeon and K. Kwak, "Comparative Analysis of Performance Using Faster RCNN and ACF in People Detection", The Journal of KIIT, Vol. 15, No. 6, pp. 11-21, Jun. 2017.

[4] J. Kocić, N. Jovičić, and V. Drndarević, "Sensors and Sensor Fusion in Autonomous Vehicles", 26th Telecommunications forum TELFOR 2018, pp. 420-425, Nov. 2018.

[5] C. Premebida, L. Garrote, A. Asvadi, A. Pedro Ribeiro, and U. Nunes, "High-Resolution LIDAR-based Depth Mapping using Bilateral Filter", IEEE Int. Conf. Intelligent Transportation Systems, Rio de Janeiro, Brazil, pp. 2469-2474, Nov. 2016.

[6] R. Omar, C. Garcia, and O. Aycard, "Multiple Sensor Fusion and Classification for Moving Object Detection and Tracking", IEEE Trans. Intelligent Transportation Systems, Vol. 17, No. 2, pp. 525-534, Feb. 2016.

[7] C. Premebida, J. Carreira, J. Batista, and U. Nunes, "Pedestrian Detection Combining RGB and Dense LIDAR Data", IEEE/RSJ International Conference on Intelligent Robots and Systems,

Chicago, IL, USA, pp. 4112-4117, Sep. 2014.

[8] S. Oh and H. Kang, "Object Detection and Classification by Decision-Level Fusion for Intelligent Vehicle Systems", Sensors, Vol. 17, No. 1, pp. 207, Jan. 2017.

[9] C. Xiaozhi, M. Huimin, W. Ji, Li. Bo, and X. Tian, "Multi-View 3D Object Detection Network for Autonomous Driving", IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, pp. 1907-1915, Jul. 2017.

[10] A. Geiger, P. Lenz, and R. Urtasun, "Are we Ready for Autonomous Driving? The KITTI Vision Benchmark Suite", IEEE Conf. Computer Vision and Pattern Recognition, Providence, RI, USA, pp. 3354-3361, Jun. 2012.

[11] D. Lowe and G. David, "Distinctive Image Features from Scale-invariant Keypoints", International Journal of Computer Vision, Vol. 60, No. 2, pp. 91-110, Nov. 2004.

[12] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection", Int. Conf. Computer Vision & Pattern Recognition, San Diego, CA, USA, pp. 886-893, Jul. 2005.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", IEEE Conf. Computer Vision and Pattern Recognition, Columbus, OH, USA, pp. 580-587, Oct. 2014.

[14] G. Ross, "Fast R-CNN", IEEE Int. Conf. Computer Vision, pp. 1440-1448, Apr. 2015.

[15] R. Shaoqing, H. Kaiming, G. Ross, and S. Jian, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 39, No. 6, pp. 1137-1149, Jun. 2017.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", IEEE Conf. Computer Vision and Pattern Recognition, Las Vegas, NV, USA, pp. 779-788, May 2016.

[17] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. Berg, "SSD: Single Shot MultiBox Detector", European Conference on Computer Vision, pp. 21-37, Dec. 2016.

[18] F. Garcia, D. Martin, A. Escalera, and J. Armingol, "Sensor Fusion Methodology for Vehicle Detection", IEEE Intelligent Transportation Systems Magazine, Vol. 9, No. 1, pp. 123-133, Spring 2017.

[19] S. Paris, P. Kornprobstm, J. Tumblin and F. Durand, "Bilateral Filtering: Theory and Applications: Series: Foundations and Trends in Computer Graphics and Vision", Computer Graphics and Vision, Vol. 4, No. 1, pp. 1-74, Jan. 2009.

저자소개

김 진 수 (Jinsoo Kim)

2019년 2월 : 순천향대학교 전기공학과(공학사)
2019년 3월 ~ 현재 : 순천향대학교 전기통신시스템공학과 석사과정
관심분야 : 적응신호처리, 기계학습

조 정 호 (Jeongho Cho)

2004년 12월 : Univ. of Florida 컴퓨터및전기공학과(공학박사)
2005년 ~ 2006년 : Univ. of Florida 의용공학과 박사후연구원
2006년 ~ 2007년 : 삼성전자 책임연구원
2007년 ~ 2014년 : 한국항공우주 구원 선임연구원
2017년 3월 ~ 현재 : 순천향대학교 전기공학과 조교수
관심분야 : 시스템 FDE, GNSS 및 보강시스템, 기계학습