

Fast moving object detection with event camera and Yolo

2021학년도 1학기 미래자동차로봇 캡스톤디자인

소프트웨어융합학과 3학년

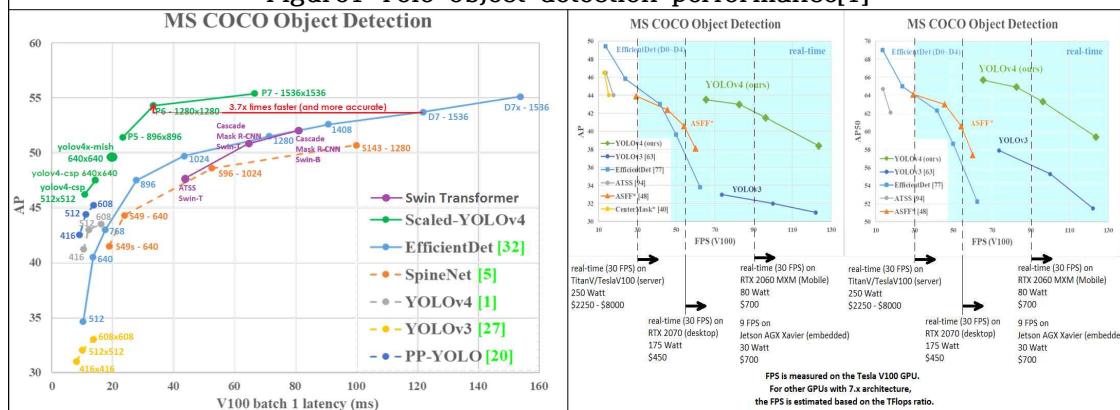
2017103762 최명규

1. 서론

최근 4차 산업 혁명이 조명됨에 따라 로보틱스 산업이 빠르게 발전하고 있다. 빅데이터, 인공지능 로봇, 사물인터넷, 자율주행 자동차 등이 4차 산업의 핵심 분야로 뽑히는 대표적인 분야이다. 로보틱스 산업의 발전이 가속화되면서 컴퓨터비전 또한 4차 산업 전 분야에서 다각도로 활용되고 있다. 그 중 object detection은 컴퓨터 비전의 핵심 분야로서 이미지와 비디오 등의 디지털 시각 정보에서 특정 계열의 객체 인스턴스(e.g. 사람, 건물, 자동차)를 감지하는 일을 다룬다. 특히 object detection은 자율주행 자동차 산업에서 보행자, 차량, 표지판 등을 감지 및 탐지하는 중요한 역할을 맡고 있다.

자율주행 자동차에서 object detection의 핵심 쟁점은 처리 지연시간을 얼마나 낮출 수 있는가에 달려 있다. 높은 지연시간은 deadline의 초과로 이어지며, 이는 곧 고속으로 움직이는 차량의 특성상 돌이킬 수 없는 인명 피해와 재산 피해로 이어지기 때문이다. 예를 들어 object detection이 기존 상용 카메라와 비슷한 프레임율인 30fps의 성능을 가지고 있다고 가정한다면 그 처리 지연시간은 프레임당 0.0333s으로 환산할 수 있다. 이를 통해 가령 100km/h(27.8m/s)로 움직이는 차량에서 object detection 처리 지연시간 동안 움직인 거리 즉, 처리 지연거리는 1m/processing로 추정할 수 있다. 따라서 object detection이 낮은 처리 지연시간과 높은 프레임율을 바탕으로 real time 지원을 지원할 수 있는 가는 자율주행 자동차가 반드시 충족해야 할 필요조건으로 여길 수 있다.

Figure1 Yolo object detection performance[1]



이러한 요구조건을 충족 시키기 위해 object detection의 처리 지연시간을 낮추려는 연구가 활발히 진행되고 있다. 이번 캡스톤디자인에서 탐구 분야로 뽑은 Yolo(You Only Look

Once: Unified, Real-Time Object Detection)[2]는 낮은 처리 지연시간으로 학계에서 주목 받고 있는 대표적인 object detection이다. Yolo는 1-stage 전략을 취하고 있는 object detection으로 다른 알고리즘과 비교해 정확도는 다소 낮지만 빠른 프레임율과 낮은 지연시간을 특징으로 가지고 있다. 이러한 특징으로 Yolo는 현재 real time object detection을 선두하는 연구로 주목 받고 있다. 관련 논문에 따르면 Yolo는 이상적인 환경에서 최대 300fps 이상을 성능을 지원한다.

object detection이 고프레임율을 지원하기 위해선 input data의 대상이 되는 디지털 이미지와 비디오 또한 고프레임율을 지원해야 할 필요가 있다. 그러나 기존 상용 카메라는 고정된 프레임율(30~60fps)을 한계로 지니고 있기 때문에 고속으로 움직이는 물체에서의 object detection에 크게 두 가지 문제점을 제공하고 있다. 첫 번째 문제는 관측하는 대상이 카메라가 지원하는 프레임율보다 빠르게 움직임에 따라 발생하는 motion blur 현상이며, 두 번째 문제는 input data의 프레임율(30~60fps)이 object detection의 성능(300fps)에 미치지 못함에 따라 발생하는 전반적인 성능 저하이다.

Figure2 motion blur 현상[3]



최근 들어 기존 전통적인 카메라가 내재한 motion blur 현상과 저프레임율 문제를 해결하기 위해 다양한 카메라 센서가 개발 및 연구되고 있다. 특히 이번 캡스톤 디자인에서 탐구 분야로 뽑은 이벤트 카메라는 이 두 가지 영역의 문제를 해결할 획기적이고 새로운 센서로서 주목 받고 있다. 기존 전통적인 카메라는 프레임을 측정의 단위로 채택하고 있는 것에 반해 이벤트 카메라는 빛의 밝기 변화로 발생하는 비동기적인 이벤트를 측정 단위로 하고 있다. 즉, 기존 전통적인 카메라는 주어진 모든 픽셀을 동기적으로 관측하는 것에 반해 이벤트 카메라는 빛의 밝기가 변화가 있는 픽셀만을 비동기적으로 관측한다. 이를 통해서 이벤트 카메라는 기존 전통적인 카메라와 비교해 높은 프레임율을 지원함과 동시에 motion blur free라는 특성을 갖추고 있다.

Figure3 Typical characteristics of image sensors[4]

Sensor	Dynamic range (dB)	Equivalent framerate
Human eye	30-40	200-300
High-end DSLR camera	44.6	120
Ultrahigh-speed camera	64	12,500
Event camera	120	1,000,000

그러나 이벤트 카메라는 비동기적 발생 이벤트를 관측의 단위로 삼고 있는 카메라 센서이기 때문에 기존 전통적인 카메라를 기반으로 발전 및 연구된 컴퓨터비전 알고리즘을 직접 적용하기에는 어려움이 다수 존재한다. 이에 이벤트 카메라를 활용한 컴퓨터비전 분야의 연구는 크게 두 가지 방향으로 진행되어왔다. 그 중 하나는 새로운 패러다임의 전환으로써 이벤트 카메라를 인정하여 이에 적합한 새로운 컴퓨터비전 알고리즘을 개발하려는 연구이며, 다른 하나는 이벤트 카메라의 관측 데이터를 기존 전통적인 카메라의 관측 데이터인 이미지로 변환하는 과정을 거친 후 변환된 이미지를 기존 컴퓨터비전 알고리즘에 적용하려는 연구가 바로 그것이다.

이번 미래자동차로봇 캡스톤 디자인에서는 후자에 주목하여 이벤트 카메라를 활용한 2단계 선형 접근법(2-Step Linear Approach) object detection에 관해 탐구해 보고자 한다. 구체적으로 이벤트 카메라로 관측된 이벤트 데이터 스트림을 이미지로 변환한 후, 높은 프레임율을 지원하는 Yolo를 통해 빠르게 움직이는 물체에서의 real time object detection 지원 가능성에 대해서 탐구하고자 한다.

2. 본론

2.1 이벤트 카메라



이벤트 카메라(Event Camera)는 프레임을 측정의 단위로 삼는 기존 전통적인 카메라와 달리 빛의 밝기 변화에 따른 비동기적인(asynchronous) 이벤트를 관측의 단위로 하는 새로운 카메라 센서이다. 이에 따라 이벤트 카메라의 측정값은 이벤트 데이터의 스트림으로 구성되어 있으며 하나의 이벤트는 4차원 벡터인 <time, x address, y address, polarity>로 표현된다 [6].

Figure5. event data stream의 예시

t_0	x_0	y_0	p_0	t_1	x_1	y_1	p_1	t_2	x_2	y_2	p_2	...	t_n	x_n	y_n	p_n
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-----	-------	-------	-------	-------

t = time,

x = x address,

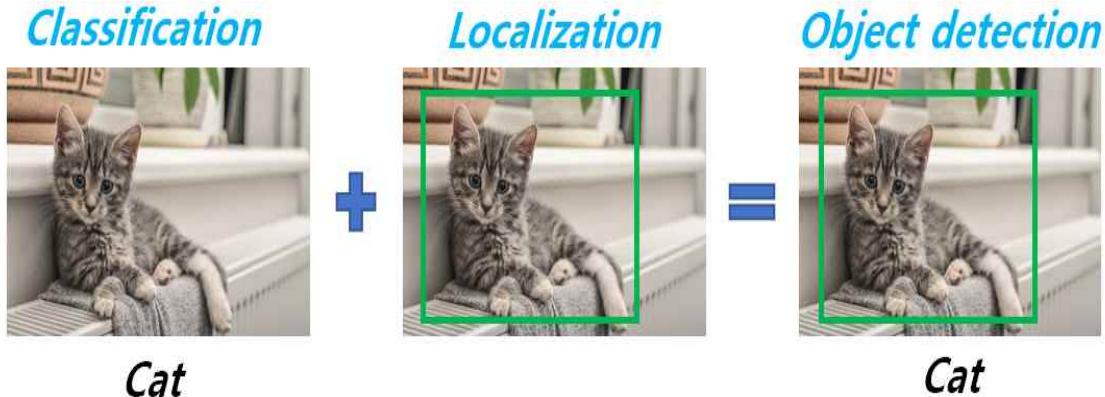
y = y address,

p = polarity of brightness changes

이벤트 카메라는 측정의 단위로 비동기적 발생 이벤트를 채택하고 있기 때문에 기존 전통적인 카메라와 비교해 다양한 장점을 가지고 있다. 높은 high dynamic range(140dB이상)를 지원하며, latency(지연시간)은 마이크로초로 높은 high temporal resolution을 지원한다. 또한 high temporal resolution을 기반으로 기존 전통적인 카메라에서 고속으로 움직이는 물체를 관측할 경우 비번이 발생하는 motion blur 현상에서 비교적 자유롭다는 강점을 가지고 있다[9]. 따라서 이벤트 카메라는 high-speed와 high-dynamic-range가 요구되는 환경에서 기존 카메라를 대체할 새로운 카메라 센서로 대두되고 있다.

2.2 Yolo

Figure6. object detection with classification and localization



object detection은 이미지에 담겨 있는 물체에 대해서 각각 어떤 물체인지 분류하는 classification 문제와 bounding box를 통해 그 물체의 위치 정보를 나타내는 localization 문제 모두를 내포하고 있는 이미지 처리와 관련된 대표적인 컴퓨터비전 분야이다. 최근 deep learning을 활용한 object detection 관련 연구는 크게 1-stage detector와 2-stage detector 두 가지 방향으로 발전되어 왔다. 전자는 classification 문제와 localization 문제를 동시에 해결하는 방법으로 비교적 처리 속도는 빠르지만, 정확도는 낮다는 단점을 가지고 있다. 이와 달리 후자는 두 문제를 순차적으로 해결하기 때문에 비교적 처리 속도는 느리지만, 정확도는 높다는 특징을 가지고 있다.

Yolo(You Only Look Once: Unified, Real-Time Object Detection)는 1-stage detector 전략을 취하고 있는 대표적인 object detection 알고리즘으로 classification 문제와 localization 문제를 하나의 regression 문제 통합했다는 특징을 가지고 있다[2]. 이를 통해 단 하나의 신경망이 단 한 번의 계산만으로 bounding box와 class probability를 동시에 예측한다. 통합된 모델을 기반으로 Yolo 다른 object detection 알고리즘과 비교해 다소 정확도는 낮지만 이를 극복할 빠른 처리 속도(적은 지연시간)를 가지고 있다. 지난 2015년 5월 첫 논문이 발표된 Yolo는 지속적인 개선을 거쳐 2020년 4월 4번째 버전인 Yolo-v4까지 모

두 4차례의 공식적인 논문 발표가 있었다. 논문에 따르면 Yolo의 성능은 최대 40mAP, 300fps 이상을 지원한다[1].

Figure7. Yolo의 신경망 구성[2]

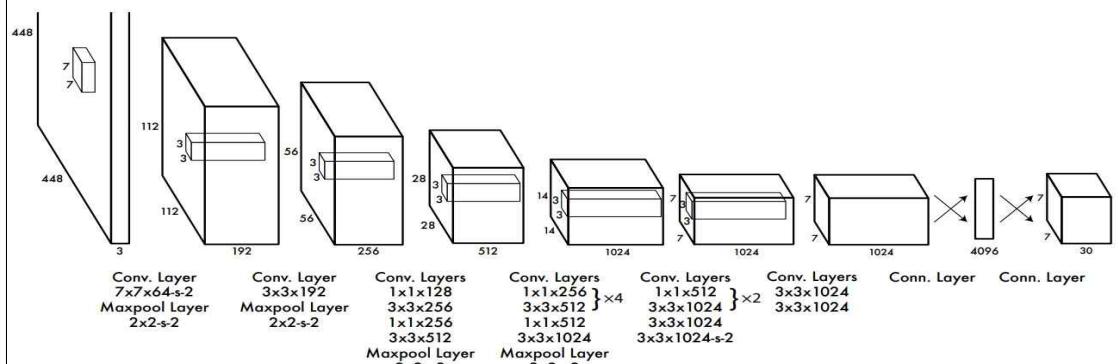
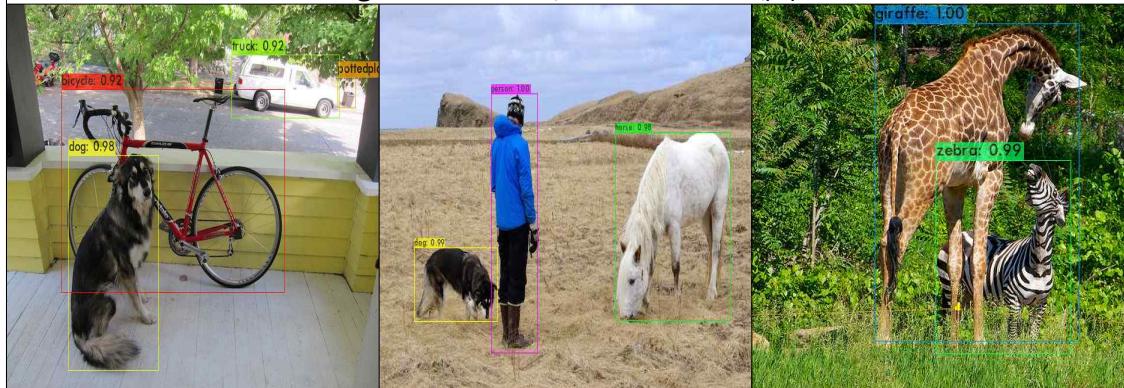


Figure8. Yolo object detection 예시



2.3 이벤트 카메라를 활용한 고속으로 움직이는 물체에서의 object detection

고정된 물체가 아닌 차량과 같이 고속으로 움직이는 물체에서 기존 전통적인 카메라로 object detection을 수행하는 것에는 한계가 존재한다. 이는 고속으로 움직이는 물체에서 object detection의 정확도와 성능을 높이기 위해선 object detection의 자체 성능뿐만 아니라 input data를 제공하는 정확도와 성능 또한 높아야 하기 때문이다. 만약 input data의 프레임율이 object detection의 프레임율에 미치지 못한다면 object detection이 이루어지지 않는 이미지 프레임이 필연적으로 존재할 수밖에 없다. 이는 곧 전반적인 성능 하락을 초래한다. 따라서 input data를 제공하는 성능이 object detection을 수행하는 성능 같거나 큰 이상적인 환경을 가정할 수 있다.

이러한 가정된 환경에서 기존 전통적인 카메라는 고속으로 움직이는 물체에서 object detection을 수행하기에는 정확도와 성능 두 가지 측면 모두에서 부적합하다고 추정할 수 있다. 성능적 측면에서 기존 전통적인 카메라는 고정된 프레임(30~60fps)을 측정의 단위로 삼기 때문에 높은 프레임율 지원에 한계를 가지며, 또한 낮은 프레임율로 인한 motion blur 현상

은 input data의 정확도마저 훼손한다. 예를 들어 기존 상용 카메라가 지원하는 30fps(지연시간 0.0333s) 환경에서 100km/h(27.8m/s)로 움직이는 차량이 object detection을 수행한다고 가정할 때 object detection 처리 지연시간은 0.0333s으로 추정할 수 있다. 따라서 object detection 처리 지연 시간 동안 차량이 움직인 거리 즉, 처리 지연거리는 약 1m로 추정할 수 있다.

Figure9. 고속으로 움직이는 물체에서의 처리 지연거리 추측값					
	fps	처리 지연시간	처리 지연거리		
			100km/h	200km/h	300km/h
기존 카메라	30fps	0.003s	0.9259m	1.8519m	2.7778m
이벤트 카메라	1000fps	0.0003s	0.0277m	0.0555m	0.0833m

이벤트 카메라는 이러한 기존 전통적인 카메라의 문제를 해결할 새로운 패러다임의 카메라 센서로 주목 받고 있다. 이는 기존 전통적인 카메라와 달리 이벤트 카메라는 프레임의 모든 픽셀을 동시에 관측하지 않기 때문이다. 대신 이벤트 카메라는 비동기적으로 발생하는 빛의 밝기 변화 즉, 이벤트를 측정 및 관측의 단위로 삼고 있다. 이러한 특징으로 인해 이벤트 카메라는 기존 전통적인 카메라와 달리 높은 프레임율(1,000fps 이상)과 motion blur free를 지원한다.

그러나 이벤트 카메라는 비동기적 발생 이벤트를 측정하는 새로운 패러다임의 카메라 센서이기 때문에 기존 전통적인 카메라를 기반으로 발전 및 연구되어 온 컴퓨터비전 알고리즘을 직접 적용하기에는 어려움이 있다. 따라서 이번 캡스톤디자인에서는 이벤트 카메라로 측정된 이벤트 데이터 스트림을 디지털 이미지로 재가공한 후 기존 컴퓨터비전 알고리즘에 간접 적용하는 방법에 대해 탐구하고자 한다. 이를 통해 이벤트 카메라가 차량 등과 같이 고속으로 움직이는 물체에서 real time object detection 지원 가능성에 대해 알아보고자 한다. 이를 위해 이벤트 데이터를 대상으로 2단계 선형 접근법(2-Step Linear Approach)을 구성하여 이벤트 카메라 데이터 기반 object detection의 성능을 정량적으로 평가한다.

3. 실험

3.1 실험 구성

이벤트 데이터의 object detection 성능지표 AP(Average Precision)와 latency를 측정하기 위해 다음과 같이 2단계 선형 접근법(2-Step Linear Approach)을 구축하여 모의 실험을 구성하였다.

2-Step Linear Approach :

1. 이벤트 카메라로 관측된 이벤트 데이터 스트림을 디지털 이미지로 재구성한다.

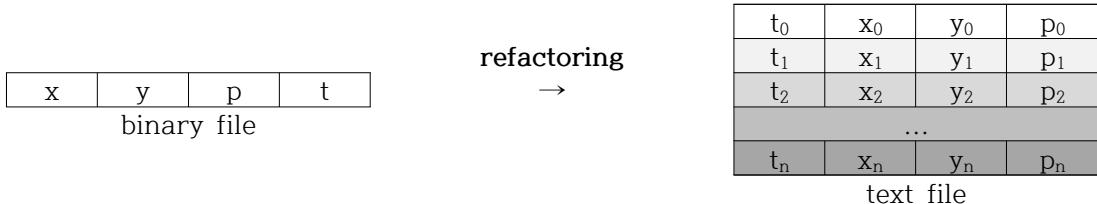
2. 재구성된 이미지를 대상으로 Yolo-v4 object detection을 수행한다.

또한 이벤트 카메라와 기존 전통적인 카메라에 대한 object detection의 성능을 정량적으로 비교하기 위해 Caltech101(이미지)과 N-Caltech101(이벤트)을 이번 모의 실험의 dataset으로 선정하였다. Caltech101은 101가지의 클래스로 구성된 image dataset이며 N-Caltech101은 Caltech101 dataset의 image를 이벤트 카메라 앞에서 앞뒤좌우 모션을 줌으로써 이미지를 이벤트 데이터 스트림으로 변환한 dataset이다[8].

구체적으로 이번 모의실험은 컴퓨팅 자원의 환경의 한계로 인해 구글 코랩의 컴퓨팅 환경을 사용하였으며 detection model은 Yolo-v4에서 제공하는 coco-dataset pretrained model을 활용하였다. 또한 Caltech101과 N-Caltech101의 101개의 클래스 중 dalmatian과 wild_cat 총 2개 클래스를 모의실험 대상 클래스로 한정하였다. dalmatian과 wild_cat을 모의실험 대상 클래스로 선정한 이유는 개와 고양이를 구별하는 문제가 어려울 것이라는 가정과 pretrained model의 클래스와 input 이미지의 클래스를 맞추기 위함이었다.

Step1 : Reconstruction from N-Caltech101 to image

Figure10. refactoring data format



이벤트 데이터 스트림을 디지털 이미지로 재구성하기 앞서 N-Catech101 dataset의 포맷 리팩토링을 진행하였다. <X address, Y address, Polarity, Timestamp> 구성된 N-Caltech101 이벤트 데이터 스트림을 <Time_stamp, X address, Y address, Polarity> 순으로 리팩토링한 후 Time_stamp를 기준으로 이벤트 데이터를 오름차순 정렬하였다. 또한 파일 형식을 binary file에서 text file로 변경하였다.

Median Selection among reconstructed image frames

$$\text{Median Selection}(\text{frame}_1, \text{frame}_2, \dots, \text{frame}_n) = \text{frame}_{\left\lfloor \frac{1+n}{2} \right\rfloor}$$

재배열된 N-Caltech101 dataset은 High Speed and High Dynamic Range Video with an Event Camera[7]를 활용하여 이벤트 데이터 스트림에서 디지털 이미지로 재구성하였다. 재구성된 이미지의 프레임은 이벤트 데이터 스트림의 길이(이벤트의 수)에 따라 유동적임으로 모의 실험에서 이미지(Caltech101)과 비교될 프레임을 선정하는 작업이 필요했다. 이는 Median Selection을 통해 수행했다. 처음 이벤트와 마지막 이벤트에 근접할수록 선행 및 후행 이벤트의 정보 부족으로 인해 이미지의 해상도가 떨어진다는 점을 착안하여 실험에서 비교 대상이 될 이미지는 재구성된 프레임 중 중간값을 활용하였다.

Figure11. median selection for best quality

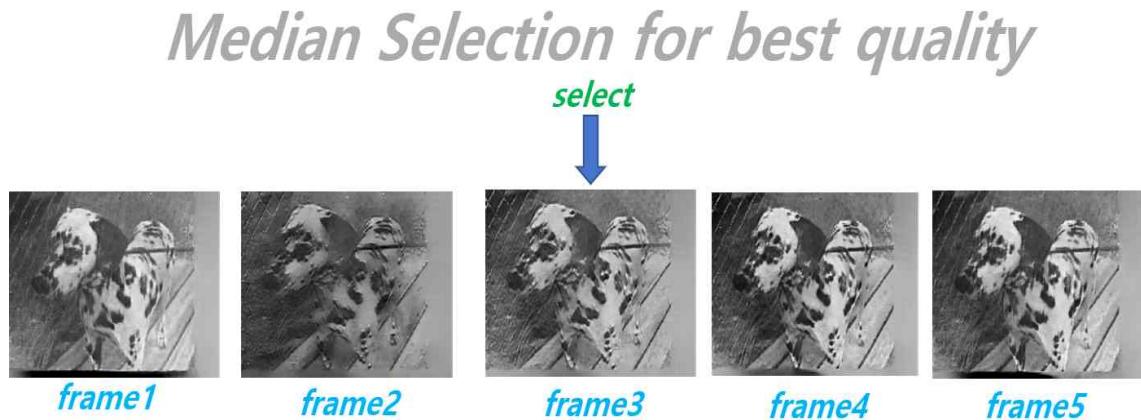


Figure12. Reconstruction from N-Caltech101 to Image

Caltech101	Reconstructed Image from N-Caltech101	Caltech101	Reconstructed Image from N-Caltech101
dalmatian_image_0010		wild_cat_image_0001	
dalmatian_image_0032		wild_cat_image_0024	
dalmatian_image_0055		wild_cat_image_0028	

Step2 : Object detection through Yolo

yolo-v4를 통해 이벤트 데이터 스트림에서 디지털 이미지로 재구성된 이미지를 대상으로 object detection을 수행하였다. 또한 기존 전통적인 카메라와 이벤트 카메라의 object detection 성능을 정량적으로 비교하기 위해 동일 클래스 동일 객체를 담고 있는 디지털 이미지 dataset Caltech101에 대해서도 object detection을 동일한 환경에서 수행하였다.

또한 object detection의 성능을 정량적으로 비교하기 위해 성능평가 지표를 다음과 같이 정의하였다.

성능 평가 지표 :

TP(Ture Positive) : dog을 dog으로 감지 또는 cat을 cat으로 감지

FP(False Positive) : dog을 cat으로 감지 또는 cat을 dog으로 감지

FN(False Negative) : dog 또는 cat 감지 실패

Recall(재현율) : $TP/(TP+FN)$

Precision(정확도) : $TP/(TP+FP)$

PR(Precision-Recall) Curve와 AP(Average Precision)

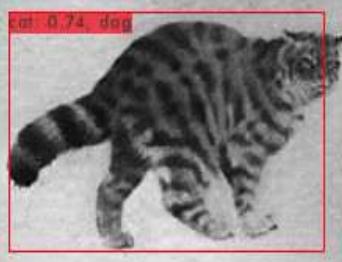
PR Curve는 성능 평가 지표를 object detection confidence 값으로 내림차순 정렬한 후 누적 Precision과 Recall 값으로 계산하였다. 또한 이를 기반으로 11-points interpolation 방법을 적용해 AP를 추정하였다.

11-points interpolation

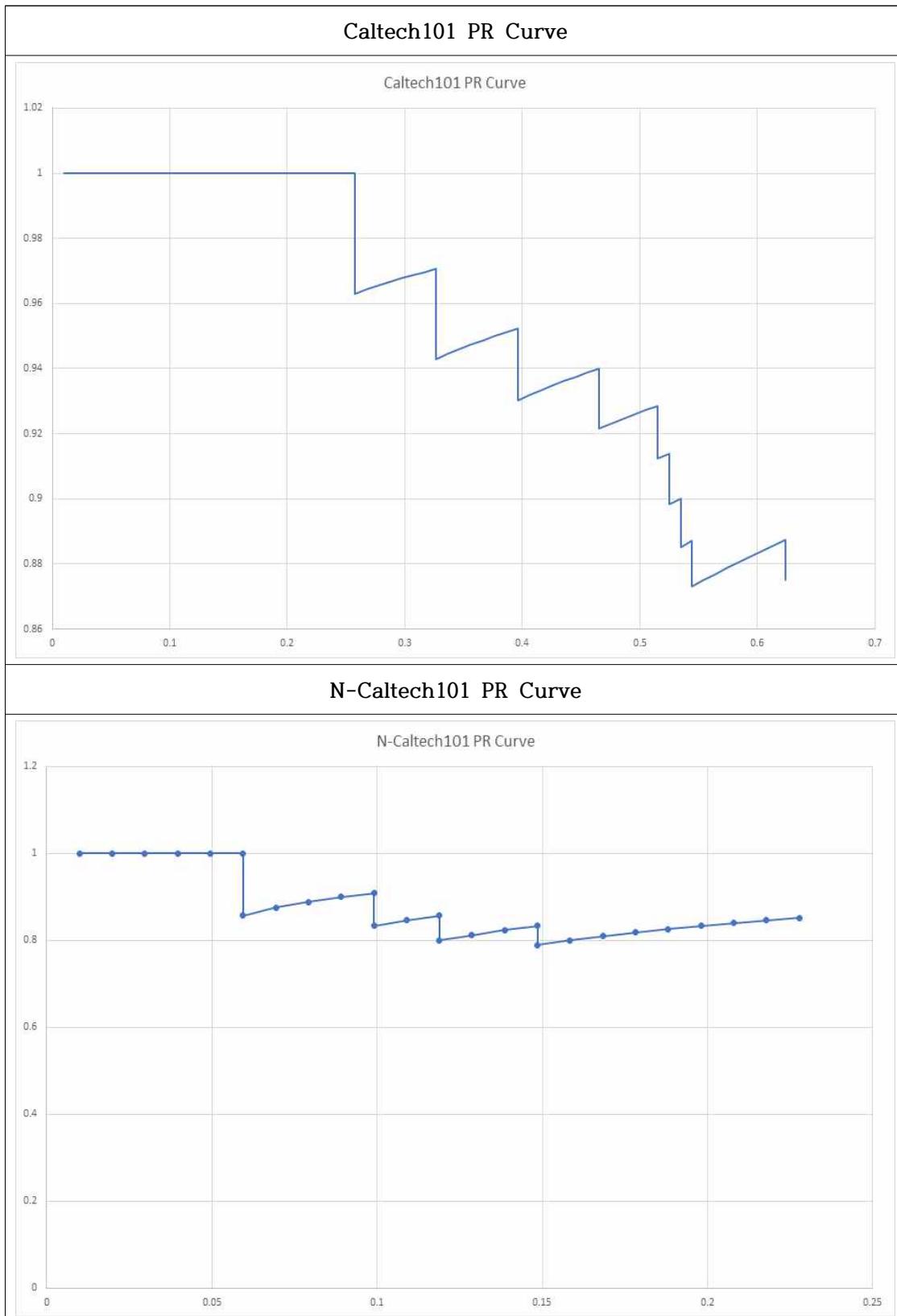
$$AP = \sum_{r \in 0.0, \dots, 1.0} \frac{1}{11} AP_r$$

$$\text{where } p_{interp}(r) = \max_{\tilde{r} \geq r} p(\tilde{r}) = \sum_{r \in 0.0, \dots, 1.0} \frac{1}{11} p_{interp}(r)$$

Figure13. object detection 수행 결과 예시

TP(Ture Positive)	FP(False Positive)	FN(False Negative)
		
dalmatian_image28	wild_cat_image27	dalmatian_image62
		
wild_cat_image25	wild_cat_image29	dalmatian_image64
		
wild_cat_image28	dalmatian_image33	wild_cat_image1
		
dalmatian_image13	dalmatian_image40	wild_cat_image3

3.2 실험 결과



	TP	FP	FN	Precision	Recall	AP
Caltech101	63	9	29	87.5%	68.5%	61.1
N-Caltech101	23	7	74	85.2%	23.7%	13.2

average latency(per frame)			
reconstruction	object detection		total
2.05ms(487.8fps)	Yolo	54.52ms(18.3fps)	56.57(17.6fps)
	Yolo-tiny	10.38ms(96.3fps)	12.43(80.45fps)

모의 실험 결과 디지털 이미지 Caltech101 dataset으로 Yolo object detection을 수행했을 때 Precision, Recall, AP는 각각 87.5%, 68.5%, 61.1을 기록하였다. 특히 이번 모의 실험에서 AP값으로 측정된 61.1은 관련 논문이 MS-COCO dataset을 대상으로 측정한 AP값 62.8%과 유사한 수치를 보였다.

또한 N-Caltech101 dataset으로부터 재구성된 이미지를 대상으로 Yolo object detection을 수행한 결과, Precision, Recall, AP는 각각 85.2%, 23.7%, 13.2로 측정되었다. 이는 Caltech101의 디지털 이미지와 비교해 Precision은 2.3%p, Recall은 44.8%p, AP는 47.9 감소하였다. 특히 주목되는 점은 N-Caltech101으로부터 재구성된 이미지가 Yolo object detection에서 Precision의 경우 매우 작은 감소 폭을 보였지만, Recall과 AP 측정값에서 Caltech101 디지털 이미지 대비 각각 65%, 78% 낮을 정도로 매우 큰 감소 폭을 보였다는 점이다. 이는 기존 디지털 이미지와 비교해 이벤트 데이터 스트림으로 재구성된 이미지를 대상으로 object detection을 수행했을 때 객체를 감지하는 능력은 매우 큰 폭으로 감소하지만, 감지된 객체의 정확도는 높은 것으로 해석할 수 있다.

처리 지연시간은 Yolo-v4 모델을 사용했을 때 reconstruction 단계와 object detection 단계에서 각각 평균 2.05ms와 54.52ms로 측정되었다. 이에 따라 총 처리 지연시간의 평균값은 56.57ms로 측정되었다. 특히 처리 지연시간의 96% 이상이 object detection 단계에서 소모되었고, reconstructin 단계에서 소모된 처리 지연시간은 4% 이하에 불과했다. 따라서 object detection의 지연속도를 낮추기 위해 Yolo-v4-tiny 모델을 사용한 결과 object detection 단계에서 처리 지연시간을 평균 10.38ms까지 낮출 수 있었다. 즉 Yolo-v4-tiny 모델을 사용했을 경우 총 처리 지연시간은 12.43ms로 프레임율로 환산할 경우 80.45fps를 기록하였다.

4. 결론 및 한계와 제언

모의 실험 결과 이벤트 데이터 스트림으로 재구성된 이미지를 대상으로 object detection을 수행했을 때 디지털 이미지 대비 Precision의 감소 폭은 2.3%p 수준으로 디지털 이미지와 거

의 유사한 수치를 보였다. 그러나 Recall 값의 감소 폭은 44.8%p으로 디지털 이미지 대비 65% 이상 매우 큰 감소 폭을 보였다.

이는 object detection 수행에서 이벤트 데이터 스트림으로부터 재구성된 이미지는 객체를 감지하는 재현율을 큰 폭으로 하락시키지만, 감지된 객체의 정확도는 기존 디지털 이미지와 거의 유사한 것으로 해석할 수 있다. 하지만 Recall과 AP가 각각 23.7%, 13.2를 기록한 것을 고려한다면 이벤트 데이터 스트림을 이미지로 재구성해 기존 컴퓨터 비전 알고리즘을 적용하는 2단계 선형 접근법은 현실적인 수준에서 object detection 수행에 부적합하다고 판단할 수 있다.

그러나 이번 모의 실험의 2가지 한계점으로부터 2단계 선형 접근법의 성능을 향상 시킬 수 있는 여지가 있다는 것에 주목할 수 있다. 첫 번째 한계는 모의 실험이 Caltech101 및 N-Caltech101의 모든 클래스를 대상으로 진행되지 않았다는 것이며, 두 번째 한계는 MS-COCO dataset으로 트레이인 된 모델이 이번 모의 실험에서 사용되었다는 점이다. 전자로부터 이번 모의 실험의 결과값이 2단계 선형 접근법의 성능을 대표하는 값으로 추정하는 것은 한계가 있다고 판단할 수 있으며, 후자로부터 objection detection에 사용된 모델이 기존 전통적인 카메라로 관측된 디지털 이미지로만 트레이인 되었다는 점을 통해 Recall과 AP가 큰 감소 폭을 보일 것이라는 예측이 가능하기 때문이다.

따라서 2단계 선형 접근법 object detection의 성능을 개선할 여지가 있다는 것에 무게를 둘 수 있다. 즉, Caltech101 및 N-Caltech101의 모든 클래스에 대해 실험을 진행하여 2단계 선형 접근법의 실험 결과값에 대한 신뢰도를 향상 시킬 수 있다. 또한 디지털 이미지뿐만 아니라 이벤트 데이터 스트림으로부터 재구성된 이미지도 모델 학습에 활용한다면 object detection의 성능을 전반적으로 향상 시킬 수 있을 것으로 기대할 수 있다.

이벤트 데이터 스트림으로부터 재구성된 이미지를 대상으로 object detection을 수행하는 2단계 선형 접근법은 앞선 기술한 바와 같이 Recall과 AP값이 기존 디지털 이미지와 비교해 감소 폭이 크다는 단점이 존재했다. 그러나 이와 관련된 개선이 이루어진다면 높은 프레임율을 지원하는 빠른 처리 속도를 기반으로 고속으로 움직이는 물체에서의 object detection에 활용될 가능성을 엿볼 수 있었다. 이번 모의 실험에서 측정된 프레임당 reconstruction 단계의 지연시간은 2.05ms로 프레임율로 환산할 경우 487.8fps에 해당했다. 이는 object detection 알고리즘의 성능과는 독립적으로 충분한 input data를 제공할 수 있다는 것으로 이해할 수 있다. 이는 곧, 고속으로 움직이는 물체에서 object detection 처리 지연시간 동안 이동한 거리를 나타내는 지연거리를 최소할 수 있는 쟁점이기도 하다.

참고문헌

- [1] Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. 23 Apr 2020
- [2] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. 8 Jun 2015
- [3] Motion blur examples. https://en.wikipedia.org/wiki/Motion_blur
- [4] Typical characteristics of image sensors. https://en.wikipedia.org/wiki/Event_camera
- [5] DAVIS346, The Dynamic Vision Sensor (DVS) from iniLabs
- [6] Daniel Gehrig, Mathias Gehrig, Javier Hidalgo-Carrió, Davide Scaramuzza. Video to Events: Recycling Video Datasets for Event Cameras. 6 Dec 2019
- [7] Henri Rebecq, René Ranftl, Vladlen Koltun, Davide Scaramuzza. High Speed and High Dynamic Range Video with an Event Camera. 15 Jun 2019
- [8] Garrick Orchard, Ajinkya Jayawant, Gregory K. Cohen and Nitish Thakor. Converting Static Image Datasets to Spiking Neuromorphic Datasets Using Saccades. 16 November 2015
- [9] Lan Xu, Weipeng Xu, Vladislav Golyanik, Marc Habermann, Lu Fang, Christian Theobalt. EventCap: Monocular 3D Capture of High-Speed Human Motions using an Event Camera. 30 Aug 2019