Journal of Korea Robotics Society (2022) 17(4):425-430
https://doi.org/10.7746/jkros.2022.17.4.425 ISSN: 1975-6291 / eISSN: 2287-3961

로봇시스템에서 작은 마커 인식을 하기 위한 사물 감지 어텐션 모델

Small Marker Detection with Attention Model in Robotic Applications

김 민 재[1] · 문 형 필†
Minjae Kim[1], Hyungpil Moon†

Abstract: As robots are considered one of the mainstream digital transformations, robots with machine vision becomes a main area of study providing the ability to check what robots watch and make decisions based on it. However, it is difficult to find a small object in the image mainly due to the flaw of the most of visual recognition networks. Because visual recognition networks are mostly convolution neural network which usually consider local features. So, we make a model considering not only local feature, but also global feature. In this paper, we propose a detection method of a small marker on the object using deep learning and an algorithm that considers global features by combining Transformer's self-attention technique with a convolutional neural network. We suggest a self-attention model with new definition of Query, Key and Value for model to learn global feature and simplified equation by getting rid of position vector and classification token which cause the model to be heavy and slow. Finally, we show that our model achieves higher mAP than state of the art model YOLOr.

Keywords: Unloading Robot, Object Detection, Deep learning

1. 서 론

딥러닝은 컴퓨터 비전, 머신 러닝 그리고 패턴 인식 등 다양한 분야에서 여전히 뜨거운 주제로 많은 연구가 진행되고 있다[1]. 로봇 비전 분야에서도 딥러닝은 큰 관심을 끌었고 많은 로봇들이 머신 비전에 딥러닝 알고리즘을 결합하여 로봇이 이전에 만난 적이 없는 상황에 대처가 가능하도록 설계하고 있다. 이러한 이점을 활용하여 물류로봇에서도 딥러닝을 활용한 머신 비전이 다양하게 사용되고 있다[2].

물류로봇에서 딥러닝은 주로 화물의 형태, 종류 그리고 위치 등을 파악하는 일에 사용된다. 인식한 화물의 형태를 이용해 파지점을 찾거나[3], 인식한 화물의 종류에 따라 엔드 이펙터(end-effector)를 변경하는 등[4] 여러가지 용도로 사용되고 있다. 이외에도 물류로봇에서 비전의 용도는 다양하며 그 중에 우리는 화물이 깨지거나 부서지기 쉬운 물체인지를 딥러닝을 이용해 알아내려고 한다. 하지만 대부분의 화물은 여러 종류의 다양한 박스들로 포장되어 있기에 딥러닝으로 학습하여 박스 안 사물의 종류를 알아내는 것은 쉽지 않다. 그렇기에 화물의 박스 포장 위에 작은 취급주의, 상하주의 마커를 이용하여 해당 사물이 취급주의를 요하는 사물인지 판단하는 것이 현실적이다.

딥러닝을 사용하여 화물위의 작은 마커를 인식하는 것은 쉬운 작업이 아니다. 이미지에서 작은 물체가 큰 물체보다 인식하는 것이 어려운 첫 번째 이유는 갖고 있는 정보의 양이 적다는 것이다. 이미지에서 사물이 갖고 있는 정보량은 이미지 상에서 사물이 차지하고 있는 픽셀의 수와 비례하기 때문이다. 그 다음은, 우리가 알고 있는 사물 감지 딥러닝 모델들은 Convolution Neural Network(CNN) [5]에 기반한 것이 대부분인데 깊은 계층으로 들어갈수록 특징 이미지가 작아지게 되고 이는 작은 마커를 더욱 작게 만들기 때문에 인식이 힘들어지게 된다.

Received : Sep. 7. 2022; Revised : Oct. 7. 2022; Accepted : Oct. 31. 2022
※ This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-2020-0-01460) supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation
1. Graduate Student, Mechanical Engineering, Sungkyunkwan University, Suwon, Korea (qqq1204@g.skku.edu)
† Professor, Corresponding author: Mechanical Engineering, Sungkyunkwan University, Suwon, Korea (hyungpil@g.skku.edu)

[Table 1] Comparison of object detectors on the MS COCO dataset (AP = Average Precision)

Model	AP_{50}	AP_S	AP_M	AP_L
YOLOv4[6]	64.9%	24.3%	26.1%	55.2%
YOLOv3[7]	55.3%	15.2%	33.3%	42.8%
SSD[8]	43.1%	6.6%	25.9%	41.4%
RFBNet[9]	54.2%	16.2%	37.1%	47.4%

과거부터 작은 물체 탐지는 1 계층 사물 감지 모델(1 Stage Object Detection)들에게 가장 큰 문제였다. 아래 표 [Table 1]는 1 계층 사물 감지 모델에서 사물의 크기에 따라 정확도 (Average Precision: AP)를 나타낸 것이다. 아래 표 [Table 1]와 같이 현재 대부분의 모델들은 작은 물체에서의 정확도(AP_S)가 30%에도 미치지 못한다. AP란 Precision-Recall 그래프를 그릴 경우 그래프 선 아래의 면적으로 계산된다. Precision은 쉽게 말해 모델이 정답이라고 인식한 것 중 실제 정답인 비율, Recall이란 실제 정답들 중에서 모델이 정답이라고 인식한 비율이다.

위 같이 작은 물체에서의 낮은 정확도 때문에 여전히 작은 물체 인식을 위한 여러가지 전략들이 개발되고 있다. 가장 대표적인 것은 입력(Input) 이미지의 크기를 크게 하는 것이다[10]. 여기서 입력 이미지 크기란 모델이 미리 정해 놓은 이미지의 크기이다. 다른 크기의 이미지가 들어올 경우 강제로 늘리거나 줄여서 고정된 입력 이미지 크기로 들어간다. 그렇기에 입력 이미지 크기가 작다면 비교적 큰 이미지는 입력 이미지 크기로 맞추기 위해 많은 정보를 소실하며 강제로 작아지게 된다. 입력 이미지 크기를 크게 한다면 이미지 정보의 손실을 최대한 줄여서 사용할 수 있게 된다. 두 번째는 학습 데이터에 존재하는 작은 물체들을 여러 번 반복적으로 복사하고 무작위한 위치에 붙여 넣는 방법이다[11]. 세 번째는 두 번째와 비슷하게 학습 데이터를 조작하는 Augmentation 방식이다. 그 중에서도 큰 이미지를 여러 개의 작은 이미지로 나누는 Tiling이나 필요한 부분만 잘라내는 크롭(Crop)이 대표적이다[12]. 이는 첫 번째 방식과 유사한 방식이다. 둘 다 중요한 정보를 갖고 있는 입력 이미지의 특정 부분이 손실없이 모델에 들어간다. 이러한 기법을 사용하면 이미지의 손실을 줄여서 이전보다 큰 사이즈로 학습이 가능해진다. 마지막으로는 YOLOv3 에서도 사용된 Feature Pyramid Network (FPN)[13]이 있다. FPN을 사용하게 되면 CNN이 가장 깊은 계층의 결과 뿐만 아니라 이전 계층의 결과값을 같이 사용하도록 설계되어 여러 사이즈의 결과가 마지막 인식에 영향을 끼치게 된다.

하지만 로봇 시스템에서 Real-Time (30FPS)을 만족시키려면 위의 방법으로는 한계가 있다. 입력 이미지 크기가 커지면 그만큼 픽셀 수가 늘어나고 필요한 연산이 점점 늘어나기 때문이다. 그렇기에 본 연구에서는 CNN이 아닌 자연어 처리에서 사용되는 Self-Attention을 이용한다. Self-Attention 이란 자연어 처리 모델인 Trans-former[14]에서 소개된 하나의 자연어 처리 기법이다. 그러나 요즘에는 이미지 처리에서 CNN과 양대 산맥으로 사용되고 있다. Self-Attention은 연속된 입력값에서 전연결계층을 사용해 쿼리(Query: Q), 키(Key: K) 그리고 밸류(Value: V)를 만들어 아래 식 (1)과 같이 연산하여 다음 계층을 만들어가는 방식이다.

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

이 방식의 특징은 입력값들을 섞어 버리기 때문에 지역적 특징(Local Feature)으로 특징 이미지를 만드는 CNN과는 달리 전체적 특징(Global Feature)을 추출한다. Self-Attention이 이미지 처리로 넘어오면서 쿼리, 키 그리고 밸류를 만드는 방법들이 Cvt[15] (Convolution to Transformer), ViT[16] (Vision Transformer) 같은 모델 마다 다르다.

본 연구에서는 단순하게 화물이 있는 이미지에서 작은 마커를 개별적으로 인식하는 것이 아니라 Self-Attention의 전체적 특징을 사용하여 박스위에 무언가를 마커라고 인식하게 만들고자한다. 이를 위해 우리는 새로운 Attention Block을 정의하였다. 이를 통해 작은 물체의 인식 성능을 전보다 정교하면서 동시에 Real-Time으로 처리가 가능하게 만들었다.

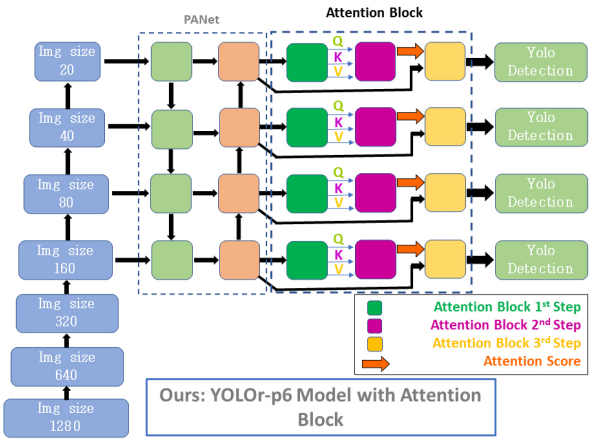
본론에서는 우리가 만든 모델의 구조와 연산과정 및 새로운 값들을 정의한다. 그리고 학습의 결과를 정리한다. 실험 결과에서는 학습 결과에 대한 해석 및 실제 테스트 이미지 결과를 설명한다. 마지막으로, 결론에서는 우리 모델의 다른 분야에서의 활용 용도를 제안하며 남겨진 문제에 대해 기술한다.

2. 본 론

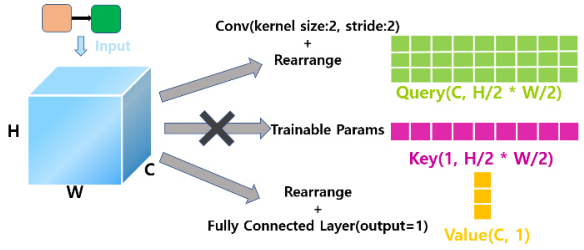
2.1 Model Structure

설계의 기본 모델로는 현재 실시간 사물 감지에서 가장 높은 성능을 보이는 YOLOr[17]을 사용하였다. [Fig. 1] 는 YOLOr 구조에 Attention Block을 포함하여 작은 물체를 인식할 수 있도록 개선한 모델이다. 그림과 같이 PANet[18]과 Detection task 사이에 우리의 Attention Block이 삽입된다.

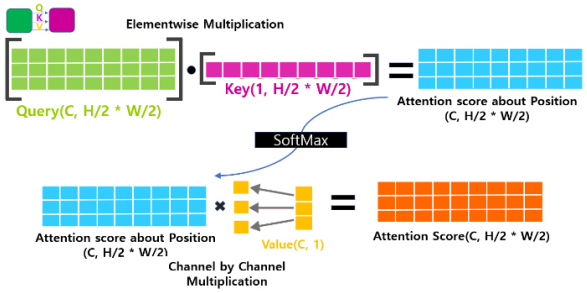
본 연구의 Attention Block에서 쿼리는 입력 특징 이미지에 Depthwise-Convolution을 사용하여 구현되며, 사이즈는 (Channel, Height/2, Width/2)이다. Depthwise-Convolution이란모든 채널마다 각기 다른 Kernel을 사용하는 방법이다. 키는 학습가능한 값이고 입력 특징 이미지와 상관없는 값으로 사이즈는 (1, Height/2, Width/2)이다. 밸류는 입력 특징 이미지에서 전연결계층(Fully connected layer)을 사용하여 사이즈는 (Channel, 1)로 구현 되어



[Fig. 1] Model Architecture - YOLOr with Attention Block



[Fig. 2] Attention Block 1st Step - Definition of Query, Key and Value

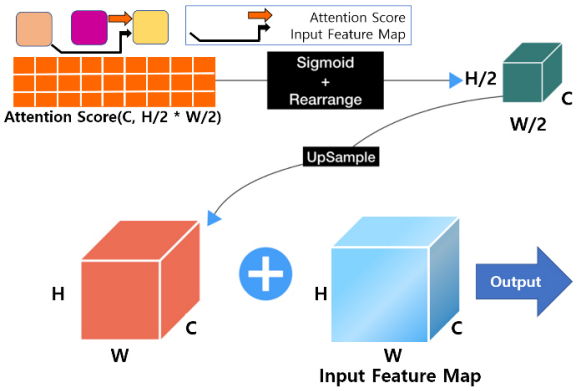


[Fig. 3] Attention Block 2nd Step - Multiplication of Query, Key and Value

있다. [Fig. 2] 은 위에 대한 전반적인 그림이다. 그 후 [Fig. 3] 와 같이 연산을 한다. 아래 [Fig. 3]의 연산은 식 (2) 같이 간단하게 나타낼 수 있다. 식 (2)의 ⊙ 기호는 Elementwise 곱을 나타낸다.

$$Attention(Q,K,V) = softmax(Q \odot K)\,V \qquad (2)$$

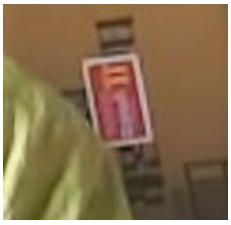
쿼리와 키의 연산에서 Position을, 밸류와의 연산에서 Channel에 대한 가중치(Weight)를 학습하게 된다. 마지막은 이전의 입력 이미지 형태로 돌려주는 단계이며 원래의 특징 이미지와 같은 크기로 돌려줌으로써 여러가지 모델에 사용할 수 있는 범용성을 갖게 되며 이는 [Fig. 4]와 같다. 이후 출력(Output)값은 Yolo Detection 단 계로 들어간다. Yolo Detection 단계는 기존 Yolo[19,20]에서 사용하는 방법이 동일하게 사용된다. 이 단



[Fig. 4] Attention Block 3rd Step - Return to Input Size

(a)

(b)



(c)

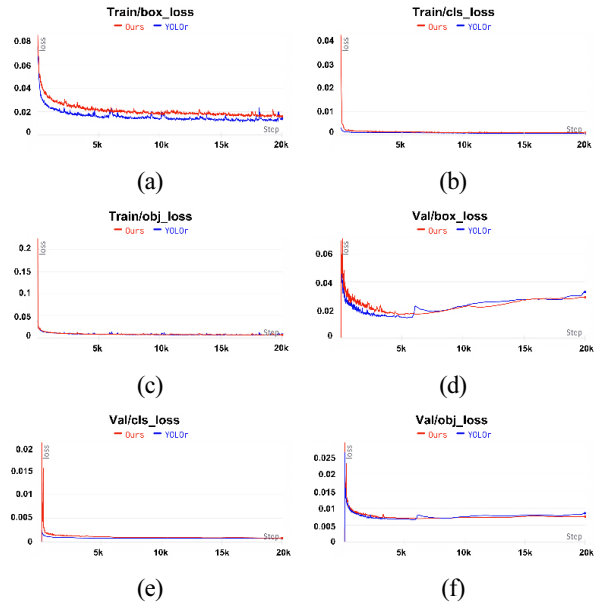
[Fig. 5] Custom Cargo Dataset. (a) Sample Full Image, (b) Cropped Image 1, (c) Cropped Image 2

계는 이미지를 여러 개의 Grid Cell 로 나눈 후 각각의 Cell에서 여러 개의 Bounding Box를 출력하는 단계이다.

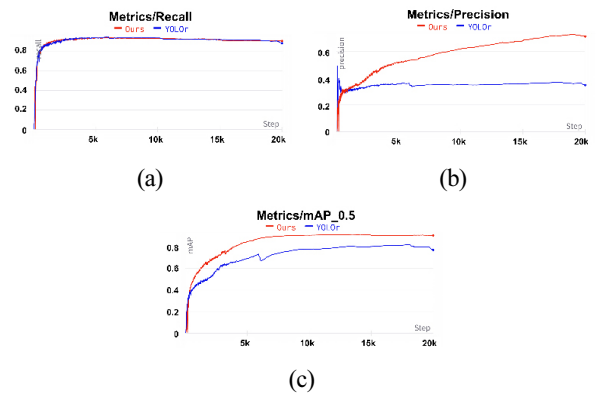
2.2 Training Result

학습은 [Fig. 5]와 같이 화물위에 취급주의 마커가 붙어있는 자체 제작 데이터를 사용하였으며 가장 작은 마커는 536 x 2048 사이즈의 이미지에 30 x 30으로 전체 이미지 대비 1/3495 배의 사물이다. 학습의 결과는 [Fig. 6], [Fig. 7] 과 같다. [Fig. 6] 에서 (a), (b) 그리고 (c)는 학습 데이터 셋에서의 Box(사물의 위치와 형태) Loss, Classification (Class에 대한 확률) Loss, 그리고 Object-ness (박스가 있을 확률) Loss이다. (d), (e) 그리고 (f)는 검증 데이터셋에서의 Loss들이다. Classi-fication과 Object-ness Loss에서는 큰 차이가 없고 Box Loss에서 기존 SOTA 모

델인 YOLOr이 0.011로 우리의 모델이 보여주는 0.014보다 약 0.003 더 낮아 약간의 우위를 보인다. 또한, 일정 epoch 이상에서 [Fig. 6]의 (a)의 학습 Box Loss는 떨어지고 (d)의 검증용 Box Loss는 조금씩 증가하는 양상을 보인다. 이는 다른 Loss에서는 나타나지 않았지만 Box Loss 에서는 약간의 과적합이 일어나고 있다고 볼 수 있다. 하지만 그 정도가 심하지 않아 mAP에는 영향을 주지 못하였다. 다음 [Fig. 7]에서 두 모델은 큰 차이를 보였다. 새로 만든 모델이 Precision에서 기존 모델을 크게 앞서 나가는 것을 볼 수 있다. 우리의 모델은 72.0% 가 나온 반면 기존 모델은 최대 36.4%로 35.6%라는 커다란 차이를 보였다. 즉, 모델이 정답이라고 인식한 물체들 중 실제 물체가



[Fig. 6] Graph of Training Result. (a) Training Box Loss, (b) Training Classification Loss, (c) Training Objectness Loss, (d) Validation Box Loss, (e) Validation Classification Loss.,(f) Validation Objectness Loss



[Fig. 7] Graph of Training Result. (a) Recall, (b) Precision, (c) mAP (mean Average Precision) > 0.5

있던 경우가 기존 YOLOr 모델보다 늘었다는 것이다. Loss가 거의 차이가 없는데도 mAP에서 차이가 난 이유는 Loss는 임계값(thres-hold)을 제외한 예측값들로 계산을 하지만, 실제 정확도 측정에서는 신뢰도를 사용한 임계값이 적용되기 때문이다. 그렇기에 딥러닝 모델 성능에서 가장 중요한 지표인 mAP 에서 본 연구에서 새로 만든 모델은 91.1%, 이전의 SOTA모 델인 YOLOr은 81.6%이며 9.5% 라는 커다란 차이를 보였다.

3. Discussion

본 연구에서는 기존의 사물 감지 YOLOr 모델보다 작은 물체에 강하게 작용하는 모델을 만들고자 하였다. [Table 2]는 테스트 셋에서 실험한 결과이며 우리의 모델이 정확도에서 9.5% 라는 차이로 YOLOr보다 좋은 성능을 보여주었다. [Table 3]을 보면 False Positive(FP)를 줄이는 효과가 있는 것을 확인할 수 있다. [Fig. 8]는 Grad-CAM[21]을 이용하여 우리가 추가한 어텐션 블록(Attention Block)이 어떠한 픽셀에 얼마나 영향을 끼치는지를 시각화한 결과와 예측 결과이다. (a)는 대상 마커가 카메라에서 다른 거리에 부착되어 이미지에서 차지하는 비율이 다른 4개의 경우를 설정한 실험 환경을 나타내고 (b)는 기존 방법인 YOLOr의 히트맵(Heatmap) 결과를 나타낸다. 기존 방법은 예상한 바와 같이 인식 대상인 마커의 화소 수 가 작은, 거리 300.0cm(오른쪽 위)의 경우 어텐션 생성에 실패하였고, 따라서 (c)에서 인식에 실패하는 결과를 보여준다. 이에 반해 우리의 모델은 (d)에서 확인할 수 있는 바와 같이 박스 위의 모든 마커에 붉은 색과 노란색의 히트맵이 잘 집중되어 있는 것을 확인할 수 있다. 상자와 상자 사이 바닥, 혹은 상자의 테두리도 히트맵에서 높은 점수로 나타나지만 최종 인식결과인 (e)에서 확인할 수 있는 바와 같이 인식 대상인 마커만을 성공적으로 예측함을 알 수 있다. 또한 우리가 제안하는 방법은 마커의 인식 신뢰도도 기존 방법의 0.7수준에서 0.8~0.9 수준으로 크게 향상되었음을 알 수 있다.

[Fig. 9] 은 마커가 붙어 있는 상자, 마대, 아이스 박스, 파우치와 같이 다양한 화물이 적재되어 있는 상황에서 테스트한 예측 결과이다. (a)는 기존 방법인 YOLOr의 인식 결과이고 (b)

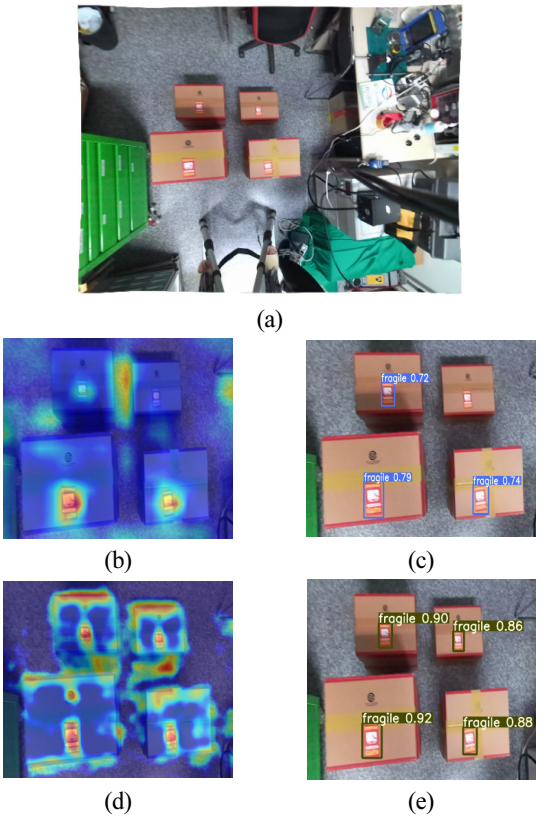
[Table 2] Comparison of our object detector and YOLOr

Model	Recall	Precision	mAP
Ours	95.2%	72.0%	91.1%
YOLOr	94.2%	36.4%	81.6%

[Table 3] Confusion Matrix of our object detector and YOLOr

Model	TP	FP	FN
Ours	2028	788	102
YOLOr	2006	3505	124

※Number of Total Object = 2130



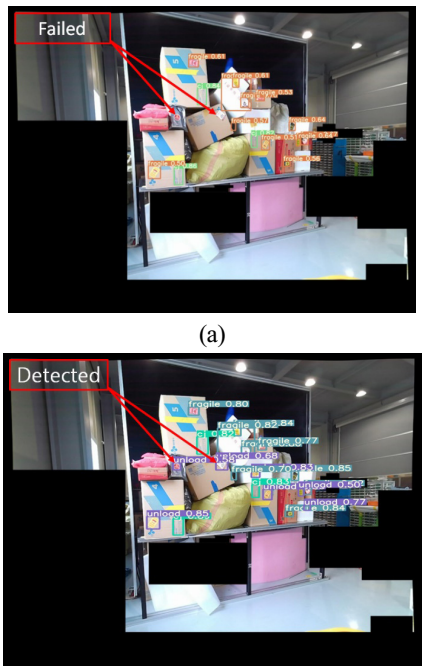
(a) (b) (c) (d) (e)

[Fig. 8] Grad-CAM and Prediction result according to distance from the camera (Left-Bottom: 235.5 cm, Right-Bottom: 257 cm, Left-Top: 278.5 cm, Right-Top: 300 cm). (a) Original Image, (b) Grad-CAM of YOLOr, (c) Prediction of YOLOr, (d) Grad-CAM of Ours, (e) Prediction of Ours

는 우리가 만든 모델의 인식 결과를 표시한다. 화살표가 가리키는 작은 마커의 경우 YOLOr 이 인식하지 못하였지만 우리의 모델에서는 인식을 성공한 사례이다.

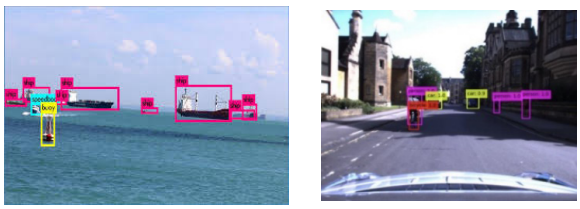
4. 결 론

본 연구에서는 새로운 Attention Block을 제안하였고 쿼리를 Depthwise-Convolution으로 추출, 키는 완전히 새로운 파라미터, 그리고 밸류는 다른 SOTA 모델과 비슷하게 전연결계층을 사용하여 새롭게 정의하였다. 이를 실시간 처리에 맞게 기존의 Self-Attention에서 여러 번 반복하던 연산을 한 번만 처리하도록 변형하였으며 연산과정에서 Position Embedding이나 Classification Token등 이전에 Self-Attention에서 사용하는 변수들을 생략하였다. 학습은 화물위에 작은 마커를 붙인 데이터셋을 사용하였고 유사한 화물 테스트셋으로 작은 물체에 대해 Precision과 mAP에서 더 높은 성능을 보여주는 것을 확인하였다. coco-dataset에서 차량 주행 시 볼 수 있는 작은 자동차들의 사이즈는 이미지 대비 1/2438배 정도이며 이는 우리가 만든 데이터셋에 비해 더 작은 크기의 사물이다. 그렇기에 우리 모델은 단순 물류로봇을 위한 취급주의 마커 인식 뿐만 아니라 자동차 자율주행에서 멀리 있는 피사체를 인식하는 데에도 응용될 수 있을 것이다. 또한, 작은 배나 선박이 멀리서 작게 보이는 사물을 [Fig. 10]처럼 더욱 빠르게 인지해 올바른 경로를 하는 데에도 응용될 수 있다. 마찬가지로 조그마한 크랙이나 흠집 등을 잡는 불량품 검출에서도 한 번에 큰 이미지를 사용할 수 있게 만들어 작업속도를 향상시킬 수 있을 것이다.



(a) (b)

[Fig. 9] Test Result. (a) Result of YOLOr, (b) Result of Ours



(a) (b)

[Fig. 10] Small object detection. (a) Object Detection with Sailing, (b) Object Detection with Driving

이후에는 사물감지 뿐만 아니라 Instance Segmentation 작업에서도 사용이 가능한지를 실험해볼 계획이다. 그 후 Grad-CAM 이외에 설명가능 한 AI (eXplainable Artificial Intelligence, XAI) 모델 중 하나인 SHapley Additive exPlana-tions (SHAP)[22]를 사용하여 우리의 모델을 해석하려고 한다. 또한, 현재는 작은 크기의 사물들이 포함 되어있는 데이터셋으로만 실험을 해보았기 때문에 여러가지 크기의 사물들이 섞여 있는 coco-dataset 같은 open dataset에서 실험을 해본 뒤, 다른 모델과의 정량적 비교도 진행하려고 한다.

References

[1] J. Ruiz-del-Solar, P. Loncomilla, and S. Naiomi, "A survey on deep learning methods for robot vision," *Computer Vision and Pattern Recognition*, 2018, DOI : 10.48550/arXiv.1803.10862.

[2] Y. Wu and D. Ge. "Key technologies of warehousing robot for intelligent logistics," *The First International Symposium on Management and Social Sciences (ISMSS 2019)*, Atlantis Press, 2019, DOI: 10.2991/ismss-19.2019.16.

[3] H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang. "Pointnetgpd: Detecting grasp configurations from point sets," *2019 International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada, pp. 3629-3635, 2019, DOI: 10.1109/icra.2019.8794435.

[4] A. Zeng, S. Song, K. T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, N. Fazeli, F. Alet, N. C. Dafle, R. Holladay, I. Morona, P. O. Nair, D. Green, I. Taylor, W. Liu, T. Funkhouser, and A. Rodriguez, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," *2018 IEEE International Conference on Robotics and Automation (ICRA),* Brisbane, QLD, Australia, 2018, DOI: 10.1109/icra.2018.8461044.

[5] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation,* vol. 1, no. 4, pp. 541-551, 1989, DOI: 10.1162/neco.1989.1.4.541.

[6] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *Computer Vision and Pattern Recognition,* 2020, DOI: 10.48550/arXiv.2004.10934.

[7] R. Joseph and A. Farhadi, "Yolov3: An incremental improvement, *Computer Vision and Pattern Recognition,* 2018, DOI: 10.48550/arXiv.1804.02767.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector." *European Conference on Computer Vision*, pp. 21-37. Springer, Cham, 2016, DOI: 10.1007/978-3-319-10578-9_23.

[9] L. Deng, M. Yang, T. Li, Y. He, and C. Wang, "RFBNet: deep multimodal networks with residual fusion blocks for RGB-D semantic segmentation," *Computer Vision and Pattern Recognition*, 2019, DOI: 10.48550/arXiv.1907.00135.

[10] B. Koonce, "Efficientnet," *Convolutional Neural Networks with Swift for Tensorflow*, Apress, Berkeley, CA., USA, 2021, pp 109-123, DOI: 10.1007/978-1-4842-6168-2_10.

[11] A. Ajaz, A. Salar, T. Jamal, and A. U. Khan, "Small Object Detection using Deep Learning," *Computer Vision and Pattern Recognition,* 2022, DOI: 10.48550/arXiv.2201.03243.

[12] F. O. Unel, B. O. Ozkalayci, and C. Cigla, "The power of tiling for small object detection," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),* Long Beach, CA, USA, 2019, DOI: 10.1109/cvprw.2019.00084.

[13] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, DOI: 10.1109/cvpr.2017.106.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Computation and Language*, 2017, DOI: 10.48550/arXiv.1706.03762.

[15] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "Cvt: Introducing convolutions to vision transformers," *Computer Vision and Pattern Recognition*, 2021, DOI: 10.48550/arXiv.2103.15808.

[16] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale, *Computer Vision and Pattern Recognition*, 2020, DOI: 10.48550/arXiv.2010.11929.

[17] C. Y. Wang, I. H. Yeh, and H. Y. M. Liao, "You only learn one representation: Unified network for multiple tasks," Computer Vision and Pattern Recognition, 2021, DOI: 10.48550/arXiv.2105.04206.

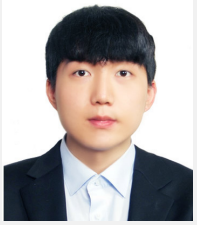
[18] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "Panet: Few-shot image semantic segmentation with prototype alignment," *2019 IEEE/CVF International Conference on Computer Vision (ICCV),* Seoul, Korea (South), 2019, DOI: 10.1109/iccv.2019.00929.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, Las Vegas, NV, USA, DOI: 10.1109/cvpr.2016.91.

[20] J. Redmon, and A. Farhadi, "YOLO9000: better, faster, stronger," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, DOI: 10.1109/cvpr.2017.690.

[21] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, DOI: 10.1109/iccv.2017.74.

[22] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Artificial Intelligence*, 2017, DOI: 10.48550/arXiv.1705.07874.



김 민 재

2020 성균관대학교 기계공학과, 시스템경영공학과(공학사)

2021~현재 성균관대학교 기계공학과 (공학석사)

관심분야: Logistics Robot, Mobile Robot, Deep Learning, Image Processing

문 형 필

1996 포항공과대학교 기계공학과(공학사)

1998 포항공과대학교 기계공학과(공학석사)

2005 Mechanical Engineering, University of Michigan, Ann Arbor.(공학박사)

2008~현재 성균관대학교 정교수

관심분야: Robotic Manipulation, Polymer-based sensor and actuators, Visual recognition