딥러닝 기반 객체 인식 기술 동향

Object detection is an important computer vision task that detects and processes instances of specific classes of visual objects in digital images. With the rapid development of deep learning technology in recent years, object detection has been widely used in industries such as autonomous driving, robot vision, and video surveillance. This paper describes the object detection method and model and divides it into a one-shot detector and a two-shot detector.

문종원(qaz3482@naver.com) 세종대 석사과정

Ⅰ. 서론

객체 인식은 디지털 영상에서 특정 클래스의 visual object의 인스턴스를 검출해 처리하는 중요한 컴퓨터 비전 작업이다. 객체 인식의 목적은 컴퓨터 비전 응용 프로그램에 필요한 가장 기본적인 정보 중 하나를 제공하는 계산 모델과 기술을 개발하는 것이다. 컴퓨터 비전의 여러 문제 중 하나인 객체 인식은 instance segmentation, image captioning, object tracking과 같이 다른 많은 컴퓨터 비전 작업의 기초를 형성한다. 최근 몇 년 동안 딥러닝 기술의 급속한 발전으로 자율주행, 로봇비전, 비디오 감시 등과 같은 산업에 광범위하게 활용되고 있다.

딥러닝 기술은 데이터에서 자동으로 딥러닝 기술은 데이터에서 자동으로 특징 표현을 학습하는 강력한 방법으로 등장했다. 2012년 ILSVRC(Large Scale Visual Recognition Challenge)에서 엄청난 이미지 분류 정확도를 달성한 AlexNet[1]이 나오게 되었다. 이것은 사람을 뛰어넘는 인식률을 달성하였고 이후로 대부분의 연구가 딥러닝을 이용한 방법에 초점을 맞추었고, 이러한 기술은 객체 인식에 큰 개선을 제공했다.

객체 인식은 여러 물체에 대해 어떤 객체인지 분류하는 classification 문제와 객체가 어디 있는지 box를 통해 위치 정보를 나타내는 localization 문제를 해결해야 하는데 전자의 대한 문제는 어느정도 해결이 되었지만, 후자에 대한 문제는 또다른 문제였다. 따라서 최근에 객체의 위치를 검출하는 방법에 대한 연구가 등장하였다.

객체 인식은 크게 one-stage 검출기와 two-stage 검출기로 나눌 수 있다. One-stage 검출기는 앞서 말한 classification과 localization문제를 동시에 수행하는 방식이고, two-stage 검출기는 두 문제를 순차적으로 수행하는 방식이다. 따라서 one-stage의 검출기가 비교적 빠르지만, 정확도가 낮고 two-stage 검출기는 느리지만, 비교적 높은 정확도를 가지고 있다.

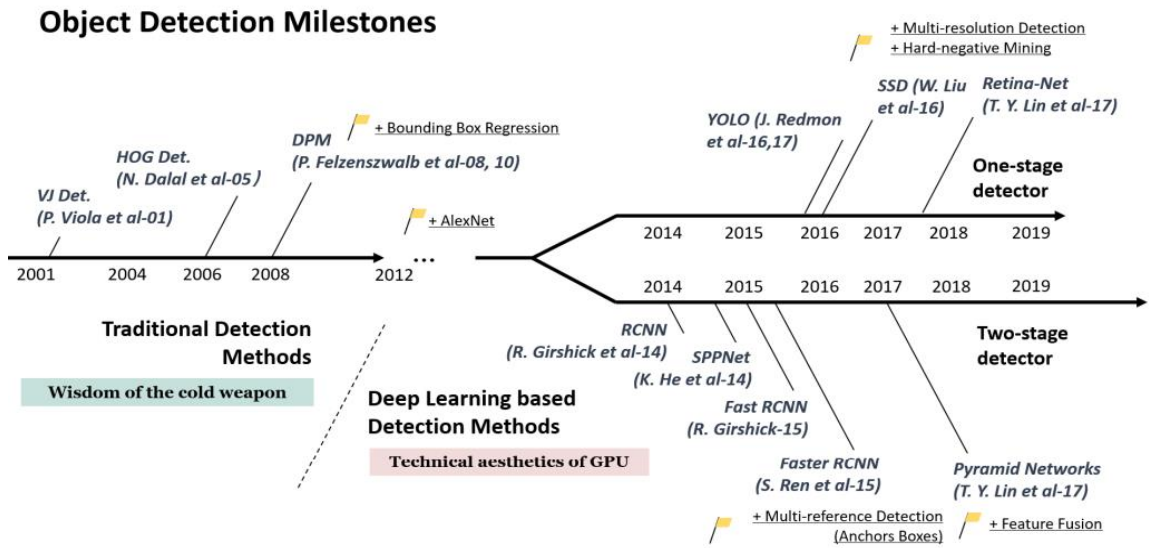
Two-stage 방식인 대표적인 검출기 RCNN(Region-based Convolutional Neural Networks)[2]은 처음으로 CNN을 적용시킨 검출기다. RCNN은 많은 연산량 때문에 검출 속도가 매우 느렸다. 이를 보완하기 위해 Fast RCNN[3]이 나오게 되었지만, Region proposal을 CNN network가 아닌 selective search를 적용하여서 여전히 느렸다. Faster RCNN[4]은 위의 단계를 CNN으로 해결함으로써 객체 인식 속도가 크게 개선되었다. 하지만 실시간으로 동작하도록 요구되는 자율주행과 로봇 등에 적용하기에는 처리 속도가 만족되지 못했다.

이 후 one-stage 방식인 YOLO(You Only Look Once)[5]가 등장하였다. YOLO는 한 번에 객체 위치와 클래스 확률을 계산할 수 있어서 기존 방법과 비교해 상당히 빠른 속도를 보여주었다. 하지만 작은 물체를 인식하는 것에 있어 좋지 못하였고, SSD(Single Shot MultiBox Detector)[6]가 등장하였다. SSD는 다양한 스케일의 특징을 사용하면서 이를 해결하였다. 그 후 YOLOv2[7]가 등장해 입력 이미지 해상도를 변경해가면서 학습하였고, 최근에는 YOLOv3[8]과 같은 빠른 검출 속도를 보이는 방법들이 제안되고 있다.

본고에서는, CNN기반 객체 인식 방법 뿐 아니라 객체 인식 모델의 발전 동향 또한 다룰 것이다.detector

Ⅱ. 객체 인식 기술 동향

1. 객체 인식 방법



(그림 1) 객체 인식의 로드 맵

[출처]ZOU, Zhengxia, et al. Object detection in 20 years: A survey. *arXiv preprint arXiv:1905.05055*, 2019.[18]

1절은 (그림 1)의 로드맵에 나와있는 CNN을 기반으로 한, 객체 인식 발전 동향을 다뤄 보려 한다. 그 중 대표적인 two-stage 검출기와 one-stage검출기로 나누어서 각 방법의 장·단점을 소개한다.

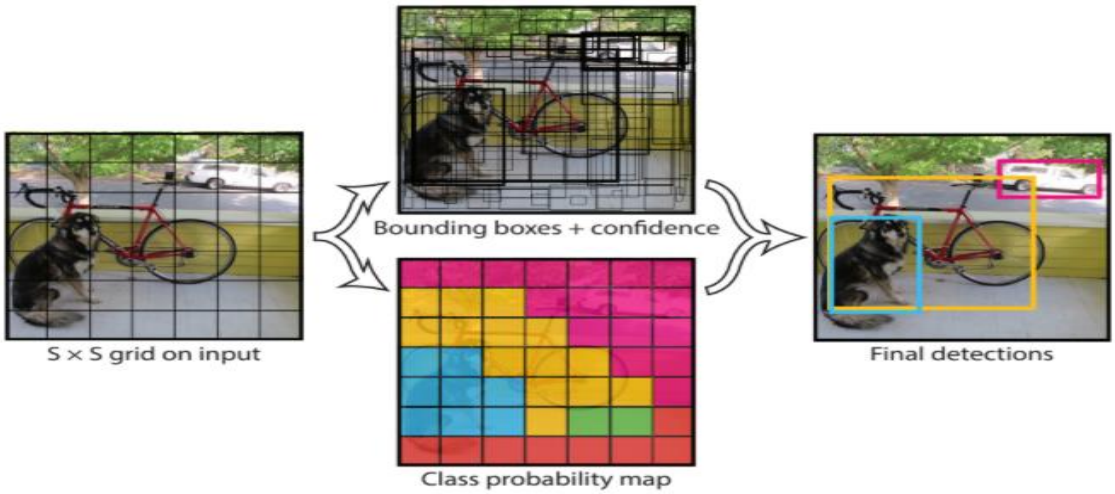
1.1 CNN 기반 Two-stage 검출기

가. RCNN

RCNN[2]은 selective search[9]방법을 통해 객체 후보 상자를 추출한다. 그 다음 후보 상자들을 고정 크기 이미지로 재조정되고, ImageNet에서 사전 학습된 CNN모델을 통해 특징을 추출하게 된다. 마지막으로 선형 SVM 분류기를 통해 각 영역내에서 객체의 유무를 예측하고, 해당 물체가 어떤 클래스 인지를 인식하게 된다. RCNN은 이전에 존재했던 검출기 대비 큰 발전을 이루었지만 단점도 존재한다. 하나의 이미지에서 selective search를 통해 2000개의 박스를 CNN에 통과시키기 때문에 많은 연산량으로 검출기의 속도가 매우 느리다는 단점이 있다.

나. Fast RCNN

Faster RCNN[3]은 RCNN의 일부 단점을 해결하기위해 제안되었다. softmax분류기, SVM, Bounding box를 별도로 학습시키는 것이 아니라 softmax 분류기, 클래스별 bounding box 좌표위치를 동시에 학습하도록 간소화된 학습 과정을 제안하여 end-to-end 방식으로 학습을 가능하게 한다. Fast RCNN은 region proposal간에 convolution 계산을 공유하고, 마지막 convolution layer와 첫번째 Fully connected layer사이에 ROI pooling layer를 추가해 각 region proposal에 대한 고정 크기의 feature를 추출하게 하였다. Fast RCNN은 RCNN보다 훈련에서는 3배, 테스트에서는 10배 더 빠르고, mAP까지 향상시켰다.



(그림 2) YOLO 모델

[출처] REDMON, Joseph, et al. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 779-788.

다. Faster RCNN

Fast RCNN은 selective search방법으로 Region proposal을 생성하는데, cpu에서 돌기 때문에 속도가 느리다. 따라서 Faster RCNN[4]은 위와 같은 사항을 개선하기 위해 Region Proposal Network(RPN)을 통해 RoI를 계산한다. RPN은 convolution을 통과해 나온 featuremap에서 RoI를 생성하고, bounding box의 위치를 예측하도록 하는 네트워크이다. Faster RCNN은 RPN에서 사용될 CNN과, bbox classifer를 위한 CNN의 네트워크를 공유하게 된다. 따라서 Fast RCNN은 최초의 실시간 딥러닝 검출기라고 불릴 정도로 Fast RCNN에 비해 속도와 성능면에서 많은 개선이 있었다.

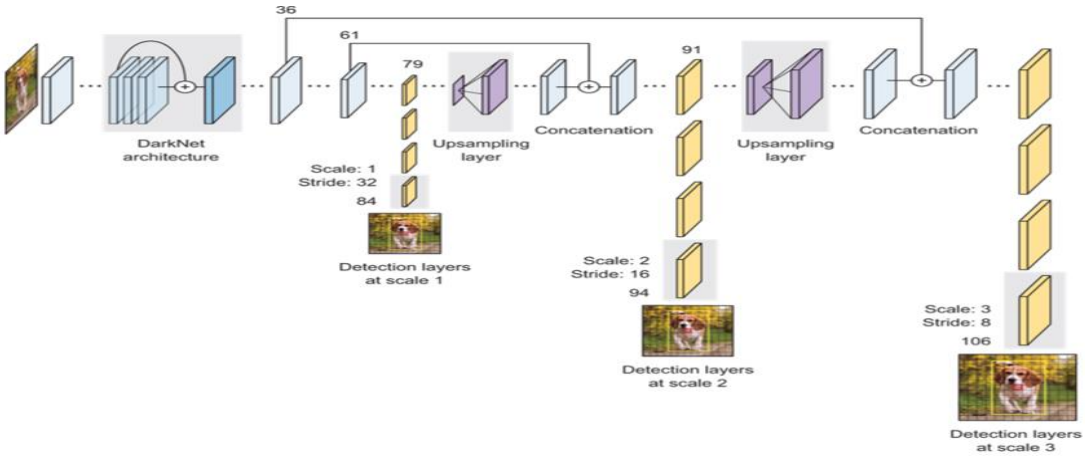
라. Feature Pyramid Networks

Feature Pyramid Networks(FPN)[10] 전 까지는 대부분의 딥러닝 기반 검출기는 네트워크의 최상단 layer를 통해 검출을 실행해왔다. 객체 검출 분야에서 작은 scale의 object를 검출하는 것은 어려운 문제이다. FPN은 low resolution과 high resolution의 featuremap을 묶는 방식을 사용해 각 레이어 마다 다양한 scale에 대응되도록 객체를 검출할 수 있게 한 방법이다. Convolution의 층이 깊어 질수록 손실되는 지역정보를 FPN을 통해 보완해줌으로써 전체적으로 높은 수준의 의미 있는 정보들을 갖춘 피라미드를 생성하게 된다. FPN을 적용한 Faster RCNN은 기존 모델보다 높은 성능을 보여주었다.

1.2 CNN 기반 One-stage 검출기

가. YOLO

YOLO[5]는 객체 검출을 하나의 회귀 문제로 보고, 한 번에 객체 위치와 클래스 확률을 계산해줌으로써 훨씬 빠른 검출 속도를 내게 되었다. (그림 1)에서 YOLO는 입력 이미지에 대해 SxS grid로 나눈다. 각각의 grid cell은 객체 위치에 대한 box, box에 대한 confidence score



(그림 3) YOLOv3 구조

[출처] REDMON, Joseph; FARHADI, Ali. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

를 예측하게 된다. YOLO는 구성이 단순하며, 1초에 45장의 영상을 처리할 수 있을 정도로 빠르지만, Faster RCNN보다 정확도는 다소 떨어진다. 그리고 YOLO는 작은 객체들을 검출하는데 어려움이 있다.

나. SSD

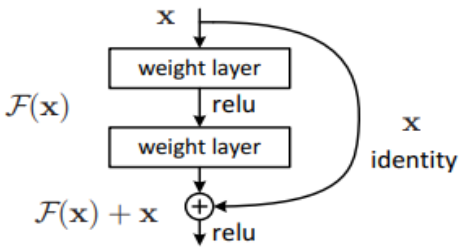
SSD[6]는 Faster RCNN의 RPN과 다양한 스케일의 특징을 효과적으로 결합해 높은 정확도를 보이면서 빠른 검출 속도를 나타내는 검출기이다. SSD는 YOLO와 마찬가지로 고정된 수의 bounding box및 score를 예측한 후, NMS 단계를 거쳐 최종적으로 검출이 된다. SSD는 VOC 2007 데이터에서 mAP 약3% 증가를 보였고, YOLO보다 높은 fps를 보여주었다.

다. YOLOv2

YOLOv2[7]은 custom GoogLeNet[11]을 Darknet19로 변경하였다고 한다. Dropout 대신에 모든 convolution layer마다 batch normalization[12]을 적용시킴으로써 mAP의 상승이 있었다. 그리고 기존에 YOLO가 가지고 있던 작은 물체 검출에 대응하기 위해서 FC layer를 제거하였다고 한다. FC layer를 제거하였기 때문에 입력 이미지의 해상도를 바꿔가면서 학습하였고, 이에 따라 기존의 YOLO와 SSD보다 높은 정확도를 가지게 되었다.

라. YOLOv3

YOLOv3[8]는 backbone을 residual block을 이용해 darknet53이라는 네트워크를 사용한다. 이는 기존 resnet-101보다 1.5배 더 빠르다고 한다. YOLOv3는 (그림)과 같이 FPN을 이용해 3가지 크기의 검출 결과가 나오도록 하였다. 이는 물체의 스케일을 고려해 작은 물체는 큰 featuremap에서, 큰 물체는 작은 featuremap에서 검출할 수 있었다. 인퍼런스시에는 SSD와 성능이 비슷하지만, 3배이상 빠른 속도를



(그림 4) Residual 학습

보였다.

2. 객체 인식 모델

이번 장은 객체 인식을 위해 사용되는 CNN기반 백본 네트워크에 대해 설명하려 한다.

가. AlexNet

8개의 layer로 구성된 AlexNet[1]은 컴퓨터 비전에서 딥러닝 기반 최초의 CNN모델이다. AlexNet은 2012 ImageNet LSVRC대회에서 우승을 차지하였다.

나. VGG

VGG(Visual Geometry Group)[13]은 모델의 깊이를 16-19 layer로 늘리고, 이 전에 AlexN-et에서 사용되었던 5x5, 7x7 convolution filter 대신 더 작은 3x3 filter를 사용하였다. VGG는 그 당시에 ImageNet 데이터를 가지고 최고 성능을 달성하였다.

다. GoogLeNet

GoogLeNet[11]은 2014년에 Google이 제안한 CNN모델이다. 이 모델은 CNN의 width와 depth를 모두 증가시켰고 총 22개의 layer를 사용한다. Inception 모듈이라는 것을 통해 연산량을 줄이고, Vanishing Gradient문제를 방지하였다.

라. ResNet

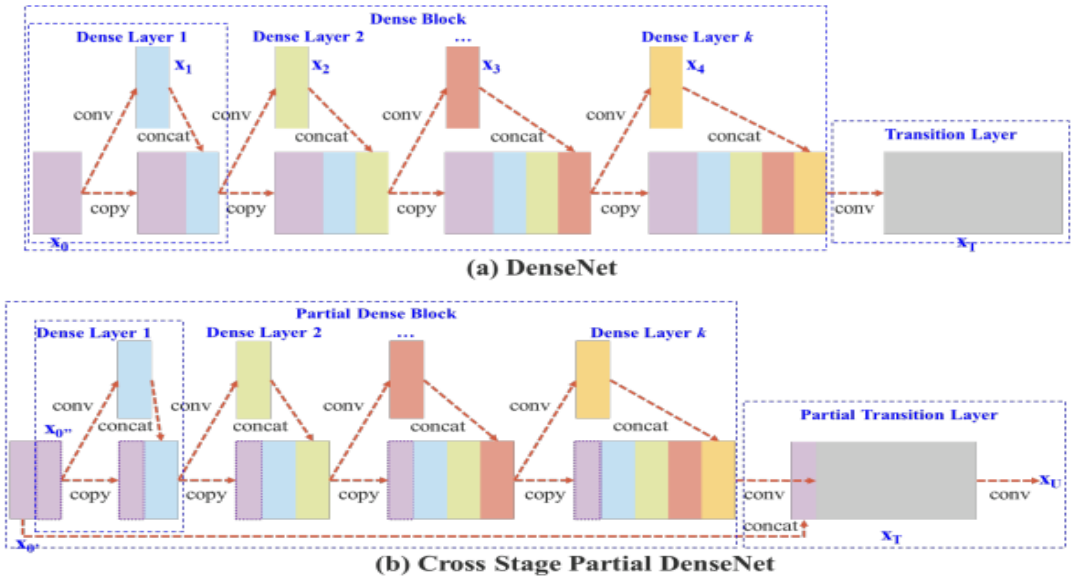
ResNet(The Deep Residual Networks)[14]은 이전에 사용된 모델보다 훨씬 많은(152개) layer를 쌓은 구조이다. ResNet은 기울기 소실과 폭발 문제를 해결하기 위해 (그림 4)와 같은 구조를 제안하였다. 입력 x를 몇번의 layer를 거친 후 출력 값에 더해주는 방식으로 residual을 최소가 되도록 학습해, 훈련을 더욱 용이하게 한다. ResNet은 2015년 ImageNet detection등과 같은 여러 대회에서 우승을 차지했다.

마. DenseNet

DenseNet[15]은 ResNet의 shortcut 구조를 좀 더 효율적으로 사용하기 위해 제안되었다. 이전 레이어를 다음 모든 레이어에 직접적으로 연결을 하고 여러 개의 레이어로 구성된 블록을 도입하였다. 기존 ResNet과 비교해 연산량을 줄일 수 있었고 더 빠른 학습이 가능했다.

바. SENet

SENet[16]은 featuremap의 채널별 중요도를 학습한다. 가중치가 큰 채널은 중요한 특징을 담고 있다는 의미로 해석하여 채널 간의 가중치를 계산해 성능을 끌어올린 모델이다. SENet은 ILSVRC 2017 분류 대회에서 1위를 차지했다.



(그림 5) DenseNet과 CSPNet 구조 비교

사. CSPNet

CSPNet[17]의 목적은 gradient combination이 만들어지는 동안 연산량을 줄이는 것이다. (그림 5)와 같이 채널의 일부를 쪼개어 나중에 합치는 방식을 사용하므로 연산량이 줄어들게 된다. 이 방식으로 인해 CSPNet은 CNN의 학습 능력이 더 강화되었고, 연산량 감소, 메모리 감소와 같은 많은 장점을 가지게 되어 YOLOv4[-19]의 backbone으로 사용되었다.

Ⅲ. 객체 인식 기술 성능 비교

지금까지 CNN기반 one-stage, two-stage방식의 검출기와 CNN network 모델에 대해 살펴보았다. 객체 인식 방법은 정확도와 속도 측면에서 장단점을 가지고 있다. (그림6)은 대표적인 객체 인식 방법인 Faster RCNN과 YOLO, YOLOv2를 속도와 성능을 비교하여 나타낸다.

	Detection Frameworks	Train	mAP †	FPS	PS (mAP×FPS)	PS Orde
Less Than Real-Time Detectors	1 Fastest DPM [26]	2007	30.4	15	456	11
	2 R-CNN Minus R [27]	2007	53.5	6	321	13
	3 Faster R-CNN ZF[20]	2007+2012	**62.1**	**18**	1118	9
	4 YOLO VGG-16[24]	2007+2012	**66.4**	**21**	1394	8
	5 Fast R-CNN[22]	2007+2012	70.0	0.5	35	14
	6 Faster R-CNN VGG-16[20]	2007+2012	73.2	7	512	10
	7 Faster R-CNN ResNet[20]	2007+2012	76.4	5	382	12
Real-Time Detectors	1 Fast YOLO [24]	2007+2012	52.7	155	8169	1
	2 YOLO(YOLOv1)[24]	2007+2012	63.4	45	2853	7
	3 YOLOv2 288×288[24]	2007+2012	69.0	91	6279	2
	4 YOLOv2 352×352[24]	2007+2012	73.7	81	5970	3
	5 YOLOv2 416×416[24]	2007+2012	**76.8**	**67**	5146	4
	6 YOLOv2 480×480[24]	2007+2012	**77.8**	**59**	4590	5
	7 YOLOv2 544×544[24]	2007+2012	**78.6**	**40**	3144	6

(그림 6) 검출기의 성능 비교

PASCAL 데이터셋에 대해서, Fast YOLO는 155fps로 가장 빠른 속도를 보여주었다. 대부분의 검출기들은 정확도 측면과 속도에서 trade-off관계를 보인 것을 확인할 수 있다.

Ⅳ. 결론

객체 인식 기술이 많은 발전을 이루었지만 여전히 특정 환경이나 작은 물체를 인식하는데 어려움이 있다. 이 전에 큰 장면에서 작은 객체를 인식하기 위해 다양한 노력을 해왔지만, 여전히 어려운 문제다. 이를 해결하기 위해선 더 많은 연구가 필요하다.

참고문헌

[1] KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 2012, 25: 1097-1105.

[2] GIRSHICK, Ross, et al. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 2015, 38.1: 142-158.

[3] GIRSHICK, Ross. Fast r-cnn. In: *Proceedingsof the IEEE international conference on computer vision*. 2015. p. 1440-1448.

[4] REN, Shaoqing, et al. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 2015, 28: 91-99.

[5] REDMON, Joseph, et al. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 779-788.

[6] LIU, Wei, et al. Ssd: Single shot multibox detector. In: *European conference on computer vision*. Springer, Cham, 2016. p. 21-37.

[7] REDMON, Joseph; FARHADI, Ali. YOLO9000: better, faster, stronger. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. p. 7263-7271.

[8] REDMON, Joseph; FARHADI, Ali. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[9] UIJLINGS, Jasper RR, et al. Selective search for object recognition. *International journal of computer vision*, 2013, 104.2: 154-171.

[10] LIN, Tsung-Yi, et al. Feature pyramid networks for object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. p. 2117-2125.

[11] SZEGEDY, Christian, et al. Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015. p. 1-9.

[12] IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: *International conference on machine learning*. PMLR, 2015. p. 448-456.

[13] SIMONYAN, Karen; ZISSERMAN, Andrew. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] HE, Kaiming, et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 770-778.

[15] HUANG, Gao, et al. Densely connected convolutional networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. p. 4700-4708.

[16] HU, Jie; SHEN, Li; SUN, Gang. Squeeze-and-excitation networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018. p. 7132-7141.

[17] WANG, Chien-Yao, et al. CSPNet: A new backbone that can enhance learning capability of CNN. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2020. p. 390-391.

[18] ZOU, Zhengxia, et al. Object detection in 20 years: A survey. *arXiv preprint arXiv:1905.05055*, 2019.

[19] BOCHKOVSKIY, Alexey; WANG, Chien-Yao; LIAO, Hong-Yuan Mark. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.