Volume: 12 Issue: V Month of publication: May 2024

DOI: https://doi.org/10.22214/ijraset.2024.62130

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue V May 2024- Available at www.ijraset.com

Survey of Techniques and Applications in Object Recognition

Mohammad Adnan[1], Mohammad Imran[2], Er. Sarika Singh[3]
[1, 2, 3]*Department of CSE, SRMCEM (Affiliated with AKTU), Lucknow, Uttar Pradesh, India*

Abstract: Object detection, which is closely related to video analysis and image comprehension, has received attention in years from researchers. Traditional approaches to object detection rely on crafted features and shallow trainable architectures. However, these methods face challenges. Often reach a plateau, in performance improvement. To overcome these limitations complex ensembles are often created by combining low-level image features with high-level context from object detectors and scene classifiers. The emergence of learning has introduced powerful tools that can learn semantics high level and deeper features. These tools address the limitations of architectures by introducing network structures training strategies and optimization functions. This paper provides a review of learning-based frameworks for object detection. The review starts with an overview of the history of learning with a particular emphasis on the Convolutional Neural Network (CNN) as an exemplary tool. It then delves into generic object detection architectures exploring modifications and effective strategies for improving detection performance. Recognizing the characteristics of detection tasks, the paper also briefly surveys several tasks such, as salient object detection, face detection, crowd analysis, and pedestrian detection.

Keywords: deep learning, object detection, YOLO, CNN

I. INTRODUCTION

Object detection is a basic research direction in the fields of computer vision, deep learning, artificial intelligence, etc. It is an important prerequisite for more complex computer vision tasks, such as target tracking, event detection, behavior analysis, and scene semantic understanding. It aims to locate the target of interest from the image, accurately determine the category and give the bounding box of each target. It has been widely used in vehicle automatic driving, video and image retrieval, intelligent video surveillance, medical image analysis, industrial inspection and other fields.[1]

Traditional detection algorithms on manually extracting features mainly include six steps: preprocessing, window sliding, feature extraction, feature selection, feature classification and postprocessing and generally for specific recognition tasks.

Humans can easily detect and identify objects present in an image. The human visual system is fast and accurate and can perform complex tasks like identifying multiple objects with little conscious thought. With the availability of large amounts of data, faster GPUs, and better algorithms, we can now easily train computers to detect and classify multiple objects within an image with high accuracy.[2]

The problem definition of object detection is to determine where objects are located in a given image (object localization) and which category each object belongs to (object classification). So, the pipeline of traditional object detection models can be mainly divided into three stages: informative region selection, feature extraction and classification.

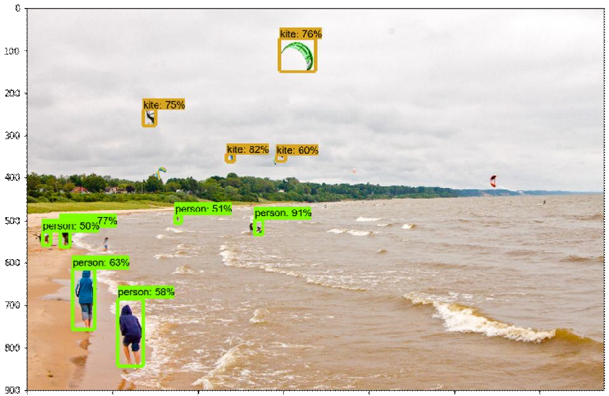


Figure 1

A. Deep Learning

1) Deep learning is a branch of machine learning which is based on artificial neural networks. It is capable of learning complex patterns and relationships within data. In deep learning, we don't need to explicitly program everything. It has become increasingly popular in recent years due to the advances in processing power and the availability of large datasets. Because it is based on artificial neural networks (ANNs) also known as deep neural networks (DNNs). These neural networks are inspired by the structure and function of the human brain's biological neurons, and they are designed to learn from large amounts of data.
2) Deep Learning is a subfield of Machine Learning that involves the use of neural networks to model and solve complex problems. Neural networks are modelled after the structure and function of the human brain and consist of layers of interconnected nodes that process and transform data.
3) The key characteristic of Deep Learning is the use of deep neural networks, which have multiple layers of interconnected nodes. These networks can learn complex representations of data by discovering hierarchical patterns and features in the data. Deep Learning algorithms can automatically learn and improve from data without the need for manual feature engineering.
4) Deep Learning has achieved significant success in various fields, including image recognition, natural language processing, speech recognition, and recommendation systems. Some of the popular Deep Learning architectures include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Deep Belief Networks (DBNs).

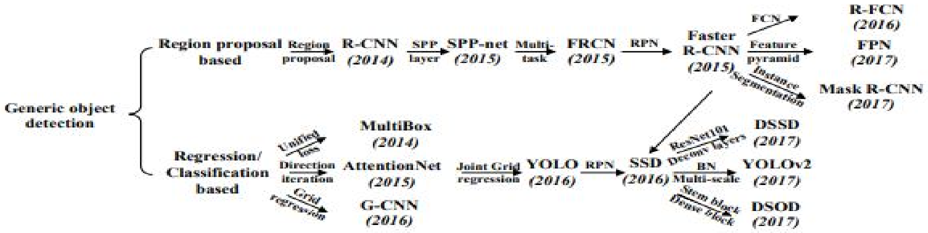


Figure 2 [1]

II. GENERIC OBJECT DETECTION

A. R-CNN

In 2014, the R-CNN algorithm was proposed by Girshick, which is the first real target detection model based on convolutional neural networks. The improved R-CNN model achieves 66% mAP. the model first uses the Selective Search to extract approximately 2000 region proposals of each image to be detected. Then the size of each extracted proposals is uniformly scaled to a fixed-length feature vector and these extracted image features are input into the SVM classifier for classification. Finally, a linear regression model is trained to perform the regression operation of the bounding box. Compared with the traditional detection method, the accuracy of the R-CNN does improve a lot, but the amount of calculation is very large, and the calculation efficiency is too low. Secondly, directly scaling the region proposal to a fixed-length feature vector may cause object distortion.[3]

B. Fast R-CNN

In 2015, Girshick introduced the Fast R-CNN model, achieving an impressive mAP of 70.0% on the joint dataset of VOC2007 and VOC2012. This is represented in figure 2. Unlike its predecessor, R-CNN, Fast R-CNN made three significant changes. It replaced the SVM classifier with a SoftMax function, adopted the pyramid pooling layer from SPP-Net, and utilized the region of interest pooling layer to replace the final pooling layer in the convolutional layer. This allowed the transformation of candidate box features into a fixed-size feature map, providing access to the fully connected layer. Additionally, the last SoftMax classification layer in the CNN network was swapped for two parallel fully connected layers.[4]

C. Faster R-CNN

The Faster R-CNN (where "R" denotes "Region") is considered one of the most efficient methods for object detection within the R-CNN series, utilizing deep learning. By combining the RPN network and the Fast R-CNN network, this technique has proven highly effective. The RPN generates a proposal which is then directly linked to the ROI Pooling layer, providing a comprehensive CNN framework for end-to-end object detection.

In assessing the feasibility of implementing Faster R-CNN with ResNet101 and PVANET, this study examines its performance when using the VGG16 network. Through training with the Caffe deep learning framework, various Faster R-CNN models can be achieved, and by evaluating results using mean average precision (mAP), a superior model can be obtained.[5]

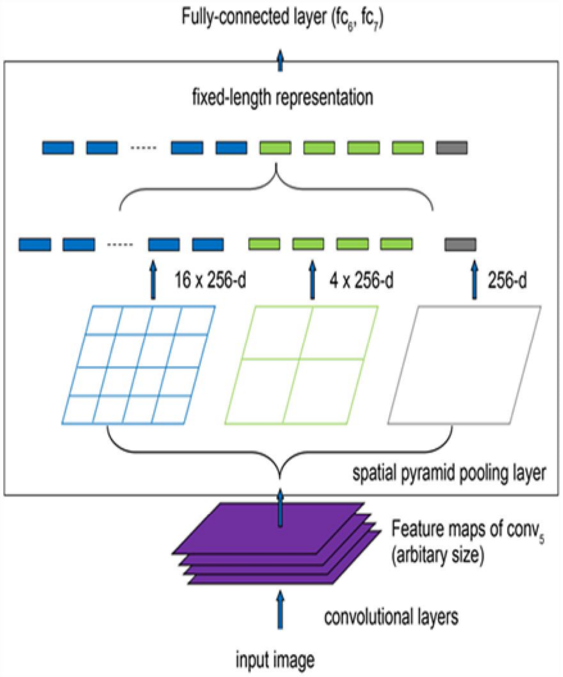

Fully-connected layer (fc_6, fc_7)
fixed-length representation
16 x 256-d
4 x 256-d
256-d
spatial pyramid pooling layer
Feature maps of $conv_5$
(arbitary size)
convolutional layers
input image

Figure 3 [6]

D. SPP-Net

The innovative Spatial Pyramid Pooling (SPP) model tackles crucial issues faced by R-CNN, such as limited detection accuracy and rigid input size demands. In contrast to R-CNN, SPP efficiently extracts region proposal features straight from the feature map in just one convolutional layer pass, reducing redundant computations. With the addition of a spatial pyramid pooling layer following the final convolutional layer, a compact feature vector for the region proposal can be obtained. A key feature of Spp-Net is its ability to perform feature extraction on the entire image in a single shot, eliminating redundant operations and significantly boosting efficiency compared to R-CNN.

E. YOLO-v1

Back in 2016, Joseph Redmon revolutionized the world of object detection with the release of YOLOv1. Breaking away from traditional methods, YOLOv1 simplifies the process by eliminating the region proposal step and utilizing a simple CNN structure. The brilliant concept behind it is to input the entire image into the network and obtain both location and category information for B bounding boxes directly from the output layer. Imagine the image split into a grid of SS cells, where each one predicts B bounding boxes along with their respective confidence scores. This means each cell is responsible for generating B(4+1) values. The result? YOLOv1 boasts a remarkable detection rate of 45 frames per second on a single TitanX, making it a truly real-time model. However, as with any breakthrough, there is a trade-off. While YOLO excels at reducing background errors, it struggles to recognize objects when they appear in groups.[7]

F. YOLO-v2

In 2017, visionary researchers Joseph Redmon and Ali Farhadi revolutionized the field of computer vision by releasing an upgraded version of YOLO, known as YOLOv2. While its predecessor relied on a traditional fully connected layer to directly predict bounding box coordinates, YOLOv2 took inspiration from the popular Faster R-CNN model and incorporated the innovative Anchor mechanism. Utilizing K-means clustering to perfect the Anchor templates resulted in a drastic improvement in the algorithm's recall rate. Furthermore, YOLOv2 expertly combines deep and fine-grained features, elevating its ability to accurately detect smaller objects.[6]

G. YOLO-v3

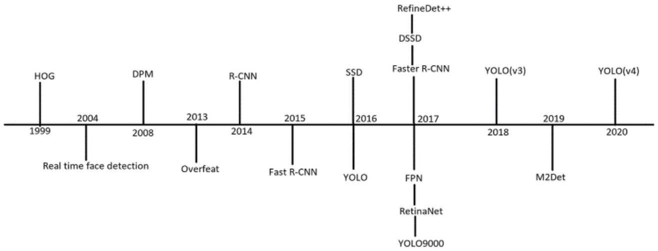
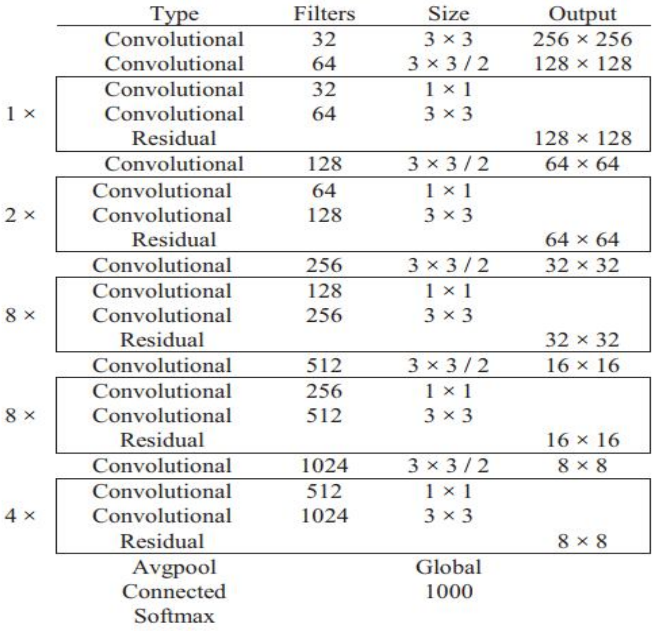


Figure 4 [8]

In 2018, Redmon improved YOLOv2, introducing YOLOv3. It adopts the darknet-53 network for feature extraction, implements a feature pyramid network for multi-scale detection, and utilizes logistic regression for real-time, accurate object detection. YOLOv3's innovative approach applies a single neural network to the entire image, predicting bounding boxes and probabilities for different regions. This global perspective grants it speed advantages, being 1000 times faster than R-CNN and 100 times faster than Fast R-CNN. Unlike R-CNN, it doesn't need thousands of target images for predictions, making it highly efficient with a single network evaluation.[6]

H. Darknet-53

A powerhouse of a convolutional neural network with an impressive 53 layers. A specially trained version of this network has been trained on a whopping one million images from the renowned ImageNet database. This pre-trained network is a master at categorizing images into 1000 object categories, from everyday items like keyboards, mice, and pencils to various animals. With extensive training, the network has honed its feature representations across a wide range of images. Its input size is 256-by-256 pixels, making it a versatile option



	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1 ×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2 ×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8 ×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8 ×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4 ×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 5 [9]

Syntax:
net = darknet53 returns a DarkNet-53 network trained on the ImageNet data set.
net = darknet53('Weights', 'ImageNet') returns a DarkNet-53 network trained on the ImageNet data set. This syntax is equivalent to net = darknet53.
lgraph = darknet53('Weights', 'none') returns the untrained DarkNet-53 network architecture.

I. YOLO-v4

In 2020, Bochkovskiy and others launched YOLOv4 [33]. YOLOv4 conducted a lot of tests on some commonly used Tricks in deep learning and finally selected these useful Tricks: WRC, CSP, CmBN, SAT, Mish activation, Mosaic data augmentation, CmBN, DropBlock regularization, and CIoU loss. YOLOv4 adds these practical skills based on traditional YOLO to achieve the best trade-off between detection speed and accuracy.[6]

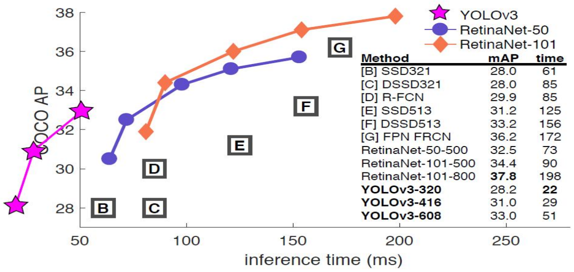
III. YOLO WORKING

A. Input Processing

Image is broken down into a grid, resembling a detective's map. Just like a puzzle, the picture is divided into smaller pieces to aid in detection.

B. Object Localization

For each little square on the grid, YOLOv3 predicts potential clues or "bounding boxes. These boxes come with details like where they start, how wide and tall they are, and a confidence score telling us how likely something is interesting inside.



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

Figure 6 [10]

C. Object Classification

Now, YOLOv3 puts on its detective hat and tries to figure out what's in each box. Instead of trying to recognize everything at once, it looks at each possible object separately. It's like saying, "Okay, what could be hiding in this corner?

D. Anchor Boxes

To be even more accurate, YOLOv3 has a few standard box sizes it considers. These are like templates that can be adjusted to fit the specific shape and size of different objects. Handy for dealing with all sorts of mysteries!

E. Feature Pyramid Network (FPN)

YOLOv3 is not satisfied with just one perspective. It wants to see the big picture and the tiny details. So, it uses a Feature Pyramid Network, like a detective looking at a crime scene from various angles to catch every detail.

F. Non-Maximum Suppression (NMS)

Detectives hate redundancy, and so does YOLOv3. It cleans up its findings by removing overlapping clues. Only the most confident and accurate details make it to the final report.

G. Output

YOLOv3 presents its findings - a list of boxes with probabilities and confidence scores. It can spot and identify multiple objects at once, like a detective solving a complex case.

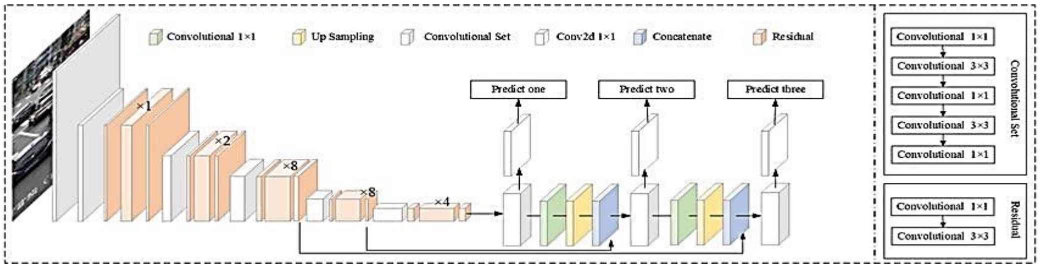


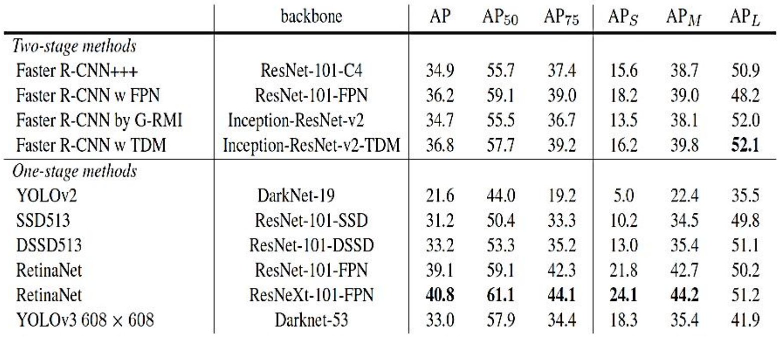
Figure 7 [11]

IV. TABLE

Table 1 Comparison of backbones. Accuracy, billions of operations (Ops), billion floating-point operations per second (BFLOP/s), and frames per second (FPS) for various networks [12]

Backbone	Top-1	Top-5	Ops	BFLOP/s	FPS
DarkNet-19	74.1	91.8	7.29	1246	171
ResNet-101	77.1	93.7	19.7	1039	53
ResNet-152	77.6	93.8	29.4	1090	37
DarkNet-53	77.2	93.8	18.7	1457	78

Table 2 YOLOv3 comparison for different object sizes showing the average precision (AP) for AP-S (small object size), AP-M (medium object size), AP-L (large object size) [13]
Read more at: https://viso.ai/deep-learning/yolov3-overview/



	backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Two-stage methods							
Faster R-CNN+++	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w FPN	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI	Inception-ResNet-v2	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN w TDM	Inception-ResNet-v2-TDM	36.8	57.7	39.2	16.2	39.8	**52.1**
One-stage methods							
YOLOv2	DarkNet-19	21.6	44.0	19.2	5.0	22.4	35.5
SSD513	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513	ResNet-101-DSSD	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
RetinaNet	ResNeXt-101-FPN	**40.8**	**61.1**	**44.1**	**24.1**	**44.2**	51.2
YOLOv3 608 × 608	Darknet-53	33.0	57.9	34.4	18.3	35.4	41.9

V. CONCLUSION

In conclusion, the evolution of YOLO from v1 to v3 features a significant change in object recognition, seamlessly blending traditional techniques and advanced deep learning with YOLO's real-time processing and single-pass image analysis streamline efficiency compared to traditional methods.

Deep learning principles adopted in YOLOv3, including anchor boxes and logistic regression mean scalability for different scenarios and larger objects Recent versions like YOLOv4 continue to innovate, driving accuracy and speed high for complex cases. The change from the traditional approach to YOLO highlights the important role of deep learning in discovery. YOLO’s integrated framework, which processes multiple objects in a single pass, provides a more efficient solution than traditional batch methods.

In the direction of the changing landscape of discovery, Yolo stands as a testament to the transformative power of deep learning. Continuous innovation in YOLO’s architecture is paving the way for a future of real-time, increasing accuracy of object recognition, reshaping the interpretation of visual information across applications

REFERENCES

[1] Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems, 30(11), 3212-3232.

[2] K. K. Sung and T. Poggio, “Example-based learning for view-based human face detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 1, pp. 39–51, 2002.

[3] Girshick, R., Donahue, J., Darrel, T.,Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: Computer Vision and Pattern Recognition. Columbus.2014, pp. 580-587.

[4] Girshick, R. Fast R-CNN.In: Proceedings of the IEEE international conference on computer vision. Santiago.2015, pp. 1440-1448.

[5] Liu, B., Zhao, W., & Sun, Q. (2017, October). Study of object detection based on Faster R-CNN. In 2017 Chinese Automation Congress (CAC) (pp. 6233-6236). IEEE.

[6] Ren, J., & Wang, Y. (2022). Overview of object detection algorithms using convolutional neural networks. Journal of Computer and Communications, 10(1), 115-132.

[7] Deng, J., Xuan, X., Wang, W., Li, Z., Yao, H., & Wang, Z. (2020, November). A review of research on object detection based on deep learning. In Journal of Physics: Conference Series (Vol. 1684, No. 1, p. 012028). IOP Publishing.

[8] Diwan, T., Anirudh, G., & Tembhurne, J. V. (2023). Object detection using YOLO: Challenges, architectural successors, datasets and applications. multimedia Tools and Applications, 82(6), 9243-9275.

[9] Dewi, Christine & Chen, Rung-Ching & Liu, Yan-Ting & Liu, Ye-Shan & Ling-Qi, Jiang. (2020). Taiwan Stop Sign Recognition with Customize Anchor. 51-55. 10.1145/3408066.3408078.

[10] Redmon, J. (n.d.). Yolo: Real-time object detection. https://pjreddie.com/darknet/yolo/

[11] Meel, V. (2023, December 29). Yolov3: Real-time object detection algorithm (guide). viso.ai https://viso.ai/deep-learning/yolov3-overview/

[12] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

[13] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).