International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 5 Volume No.6 Aug-Sept – 2022
DOI:10.46647/ijetms.2022.v06i05.046 ISSN: 2581-4621

Comparison of YOLO Versions for Object Detection from Aerial Images

Cina Mathew[1] , Binny S[2], Roji Thomas[3], Cini Joseph[4], Shincy K Kurian[5]

[1] *Assistant Professor, Department of Computer Application, Kristu Jyoti College of Management and Technology, Changanacherry, Kerala*

[2] *Associate Professor, Department of Computer Application, Kristu Jyoti College of Management and Technology, Changanacherry, Kerala*

[3] *Associate Professor & Head, Department of Computer Application, Kristu Jyoti College of Management and Technology, Changanacherry, Kerala*

[4] *Assistant Professor, Department of Computer Application, Kristu Jyoti College of Management and Technology, Changanacherry, Kerala*

[5] *Assistant Professor, Department of Computer Application, Kristu Jyoti College of Management and Technology, Changanacherry, Kerala*

ABSTRACT

Deep learning techniques are used across a wide range of fields for several applications. Deep learning-based object recognition from aerial or terrestrial photos has grown in popularity as a research topic in recent years. On this work, object detection was used by training the YOLOv2 and YOLOv3 algorithms in the Google Colaboratory cloud service using the DOTA dataset, which consists of aerial pictures, and the Python programming language. For assessment, 4 aerial pictures of 5 class items were used. Large vehicle, aeroplane, ship, basketball court, and swimming pool are some examples of these classifications. The outcomes of accuracy analyses of these two algorithms were compared in accordance with recall, precision, and F1-score for 5 classes. The top score with YOLOv2 was 99 percent F1 in the aeroplane class, whereas the best score with YOLOv3 was 83 percent in the pool class. While YOLOv2 can identify items in an average photo in 43 seconds, YOLOv3 has outperformed its predecessor in terms of speed, identifying objects on average in just 2.5 seconds.

Keywords: Deep Learning, Object Detection, YOLOv2, YOLOv3, Aerial Image

Introduction

Digital image processing is a technique for altering the image to produce an improved version, converting the image to a digital format, and extracting data from it. Deep learning for digital image processing has become used for many purposes in computer vision, such as face recognition (Atik and Duran, 2020), object detection and classification (Atik and Ipbuker, 2020; Atik and Ipbuker 2021), etc. Deep learning-based object recognition is frequently employed, particularly with photos captured via remote sensing and photogrammetric techniques. (Yang et al., 2019). The effectiveness of deep learning methods for object detection can be enhanced by the use of larger datasets and the creation of more potent models. Success of region-based techniques and region-based convolutional neural networks(R-CNN) has led to the most important advances in object detection. (Chen et al., 2016).Convolutional neural network-based object identificationconsists of basically two different classes, two-stage and single-stage. Two-stage CNNs: R-CNN (He et al., 2017), Fast R-CNN (Girshick, 2015), Faster R-CNN (Ren et al.,2015) and R-FCN (Dai et al., 2016). Compared to single-stage procedures, two-stage techniques operate more slowly yet achieve results. One of the single-stage strategies was the You Only Look Once (YOLO) strategy employed in the study. When using bounding boxes that are geographically dispersed, YOLO describes object detection as a regression problem. (Gavrilova , 2019)

This research seeks to use YOLOv2 to detect things,(Redmon and Farhadi, 2017) and YOLOv3 (Redmon and Farhadi, 2018) deep learning algorithms with aerial images in the DOTA data set.As assessment criteria, precision, recall, and F-score were employed.

Related Works

Recent years have seen a large number of works in the area of deep learning for object detection published. Lu et al(2017) looked on object detection in autonomous cars' negative trends. The YOLO sensor's detection rate was tested while simulating the operation of a vehicle with printed licence plates. Fast YOLO, a novel technique that speeds up real-time object recognition from video in embedded **devices, was the** subject of a study by Shafiee et al. (2017).

First, YOLOv2 utilized evolutionary deep intelligence to develop the network architecture, and with only 2% IOU (Intersection Over Union - calculated as the area where the intersection of two rectangles divided by the area of the combination of these two rectangles), an optimized with 2.8 times fewer parameters. A multi-target tracking method based on YOLO has been suggested in the Tan et al. (2018) study to further improve the efficacy and accuracy of multi-target tracking. Depth extraction was performed after acquiring target, size, position, and other details. The feature extraction technique requires less calculation and time because the image noise has been eliminated. Li et al. (2017) examined at the causes of why conventional two-stage algorithms take longer to identify objects than single-stage object detectors like YOLO and SSD. While the R-FCN produced a big score map, the Faster R-CNN had two completely linked layers for ROI identification. As a result, it has resolved the issue with Fast R-CNN and R-FCN before and after ROI skewing that required extensive calculation. As a result, it has been discovered that these networks' slow pace is caused by their architectural design. Liu, et al. (2020) have suggested a UAV-YOLO solution in their work based on deep learning algorithms to address the challenges encountered in identifying small objects from UAV-based images. In particular, the study intends to enhance the human detection performance and enhance the neural network structure of the YOLO method by creating an image dataset gathered from the UAV platform. The YOLOv3 algorithm was utilized in the investigation, and a Darknet software framework enhancement was built for the study.

A comparison of the YOLOv2 and YOLOv3 algorithms over aerial photos is reported in this paper. It is feasible to assess the study's methodologies from several angles, particularly because the data set contains both small and large items.

Material and Method

Data Used

DOTA is a large dataset for object recognition in aerial photos (Xia et al., 2018; Ding et al., 2018; Ding et al., 2021). DOTA-v1.0 has the following object categories: helicopter, football field, basketball court, tennis court, basketball court, baseball field, tennis court, basketball court, highway field, harbour, and swimming pool. Some samples from the dataset are presented in Figure 1. The China Center for Resources Satellite Data and Application has contributed Google Earth, GF-2, and JL-1 satellite pictures to the DOTA collection. In addition, the spatial resolution information of each image is presented in its metadata.

Convolutional Neural Networks (CNN)

A deep learning technique that can distinguish different objects from an input picture is the convolutional neural network. Convolutional neural networks are modelled after how the visual cortex in the human brain is organized and operates. CNNs' most significant feature is that it minimizes the number of parameters in ANNs. (Albawi et al., 2017). Convolution layers use filters to extract information from the image at various levels. By combining the first filter with the featured image, a feature type is defined. Then, a second filter is used to detect a different feature type in a second image. Convolution layer, nonlinear layer, pooling layer, smoothing layer and fully connected layer form the convolutional neural network architecture (Atik and Ipbuker, 2021).

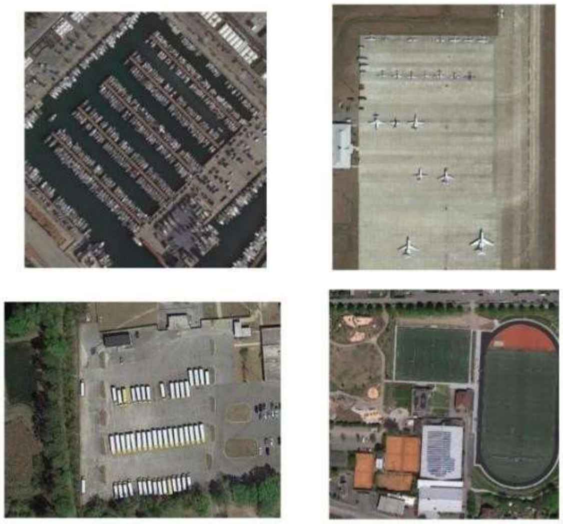


Fig. 1: Samples from DOTA Dataset

YOLO

The open-source object identification method You Only Look Once (YOLO) (Redmon et al., 2016) is built on convolutional neural networks. One of the most popular deep learning algorithms is YOLO, and its single-stage detection architecture sets it apart for its quickness. (Figure 2). Detection systems prior to YOLO reuse classifiers or localizers for object detection.

Multiple scales and locations are used to apply the model on the image. The image's high-scoring areas are classified as objects. In YOLO, object detection is handled as a regression issue while a single neural network is applied to the whole image. The network divides the picture into areas and bounding boxes, then estimates probabilities for each region. Based on the predicted probability, these bounding boxes have been given weights.

In YOLO, object detection is handled as a regression issue while a single neural network is applied to the whole image.

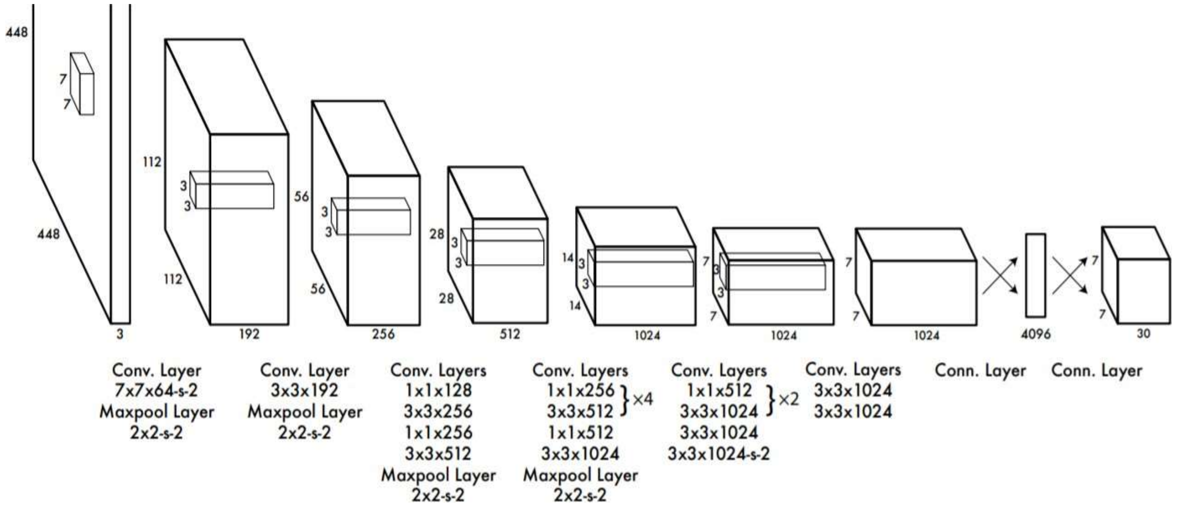


Fig. 2: Architecture of YOLO algorithm (Redmon et al., 2017).

In order to perform detection, YOLO first splits the input image into SxS grids. Depending on the versions, these grids may have different sizes. Each grid must determine if an item is present in the field, whether it is its middle, as well as its length, height, and class. These actions result in the creation of bounding boxes. After that, each grid obtains its own estimate vector. Within the prediction vector are the confidence score, Bx (x coordinate of the object's midpoint), By (y coordinate of the midpoint of the object), B_w (the width of the object), B_h (the height of the object) and the dependent class probability.

YOLOv2

Significant localization mistakes are made by YOLO. As a result, it may be inferred that YOLO had a significant recall error. For this, the primary goals of YOLOv2 (Redmon and Farhadi, 2017) are to improve recall and localisation while preserving classification accuracy.

With Yolov2, a new network design was introduced by removing the full connection layer and batch normalization (Sang et al., 2018). In order to enhance performance, batch normalisation was added to all convolutional layers in YOLO. This resulted in an improvement of more than 2% above mAP. A high-resolution classifier was further trained. Resolution of the classifier was increased from 224 to 448. As a result, when switching to detection, the network also needs to adjust to the new input resolution and switch to learning object detection. Another breakthrough in YOLOv2 is the enhancement in accuracy and performance in multi-object identification with the introduction of Anchor Box (Redmon and Farhadi, 2017).

YOLOv3

Because of this, whenever the network switches to detection, it also needs to adjust to the increased input resolution and move to learning object detection. Another new feature of YOLOv2 is the usage of Anchor Boxes, which improves multi-object detection performance and accuracy. Using logistic regression, YOLOv3 estimates a confidence score for each bounding box. The input image is divided into small grid cells SxS using the YOLOv3 algorithm.The grid cell must detect an object if it enters a core cell. Each cell calculates the objectivity scores of these bounding boxes and estimates the location information for the B bounding boxes. This approach states that if the bounding box covers an item with known ground accuracy more than the other bounding boxes did, the confidence score should be 1. (Zhao and Li, 2020).

Only one bounding box is given by the system to each item with known ground accuracy. An item with known location accuracy will not experience any coordinate or class estimations loss if a prior bounding box is not allocated to it. In certain instances, the box doesn't have the highest IOU but nevertheless completely covers a precision item. In such situations, the prediction is discarded. In order to forecast classes during training, binary cross-entropy loss is used. (Zhao and Li, 2020).

Using independent logistics classifiers, an object can be perceived as a woman and a person at the same time.The shortcut connections used in the algorithm have provided advantages over other algorithms in findingsmall objects. Using this linkage method provides more detailed information from the previous feature map. However, YOLOv3 performs less well on medium and large-sized items than YOLOv2.

Evaluation Metrics

Three metrics were determined for the analysis of the results; precision (Eq. 1), recall (Eq. 2), and F-score (Eq. 3). These metrics are calculated according to the confusion matrix. The confusion matrix shows the distribution of object detection. The confusion matrix consists of 4 parameters: true positive (TP), true negative(TN), false positive (FP) and false negative (FN). TP: Prediction is positive and ground truth is positive. TN: Prediction is negative and ground truth is positive. FP: Prediction is positive and ground truth is negative. FN: Prediction is negative and ground truth is positive (Gonultas et al., 2020). Precision refers to the number of correct detections of the method, while recall is the metric of correctly detected objects that actually exist. F1-score is a function of precision and recall. Evaluation metrics are also calculated using the confusion matrix (Atik et al., 2021).

Precision = TP / (TP + FP) (Eq.1)

Recall = TP / (TP + FN) *Eq.*(2)

*F1-score = 2 (Precision * Recall) / (Precision + Recall)* (Eq.3)

The models were trained using the DOTA dataset and tested using test images using the YOLOv2 and YOLOv3 methods. In order to compare and assess the algorithms, or in other words, to undertake accuracy analysis, 43 images for both models were collected as output files following the successful completion of the control phase.Outputs were obtained in both algorithms for 43 images and these results were evaluated one by one (Figure 3 and Figure 4).

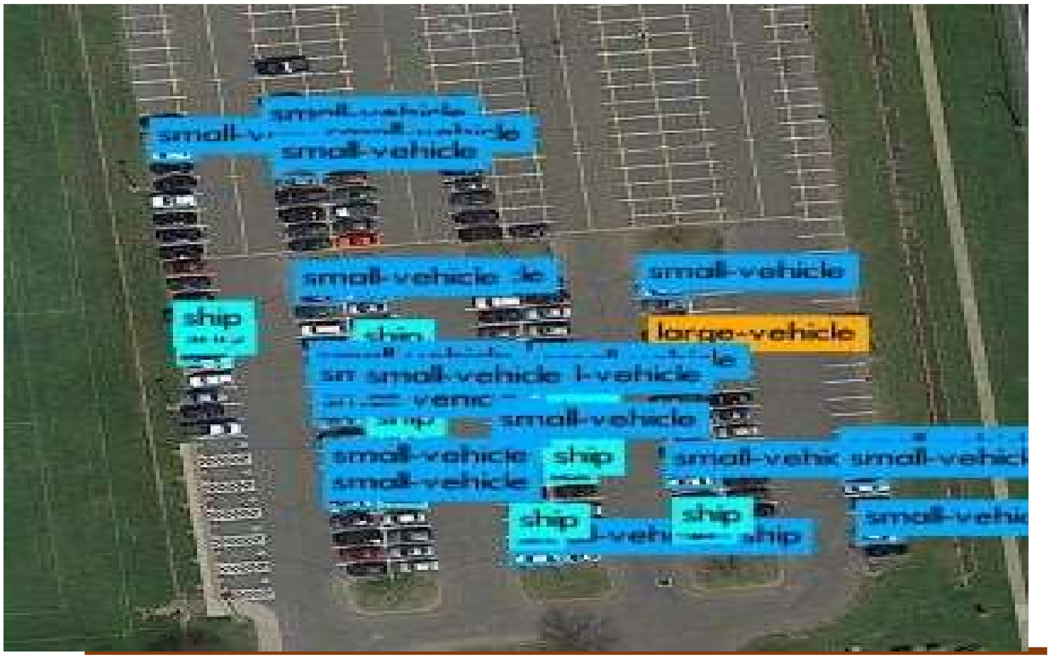
Implementation of YOLO

In order to give the best results in object detection in aerial photographs, the DOTA dataset consisting of aerial photographs was used. The implementation of YOLOv2 and YOLOv3 algorithms has been carried out on the Google Colaboratory platform with free high GPU (Graphics Processing Unit) support.

Google Colaboratory is a cloud service application that can use Tesla K80 GPU for free and develop deep learning applications. The service basically runs on the Python scripting language. In this study, Google Colaboratory was chosen to get support from the Tesla K80 GPU, showing a high performance especially intraining models.

Label data is defined as "x1 y1 x2 y2 x3 y3 x4 y4category difficult" (image coordinates and category number of each corner of the object, respectively) in the DOTA dataset, but the desired format for Darknet algorithms is "category-id xy width height" (respectively. category number and width and length measurements). Data conversion has been made in these files in order to make them suitable for algorithms.

Fig.



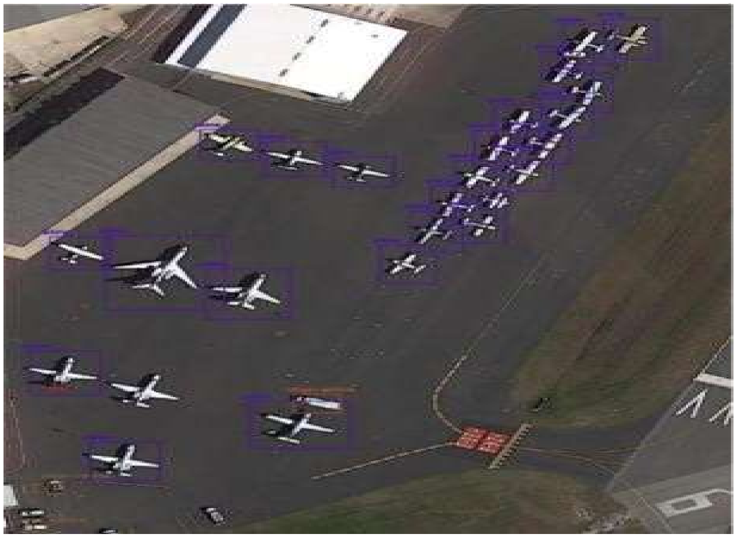


Fig. 4: Detection of objects in the DOTA dataset usingYOLOv3

Results and Discussion

In accordance with the requirements derived from the error matrix, three alternative assessment metrics were employed. Recall is the first, precision is the second, and the F- score is the third. Tables 1 and 2 display the recall, precision, and F-measure accuracies of 5 classes of YOLOv2 and YOLOv3 algorithms.

Accuracy comparisons between the YOLOv2 and YOLOv3 algorithms were performed on the identification of 5 object types, including large vehicles, planes, ships, basketball courts, and swimming pools. In light of the data from the error matrix, recall, precision (Precision), and F-measure (F-measure) were employed as accuracy criteria. Since it considers both recalls and sensitivity requirements, the F-score has been used as the accuracy metric for evaluating the algorithms' results. 43 images from a range of classifications were chosen for review.

Table 1. Results of YOLOv2 algorithm. The values are given as %.

Metric	Precision	Recall	F-score
Large Vehicle	99	24	39
Plane	99	99	99
Ship	99	62	76
Basketball Court	100	64	78
Swimming Pool	100	43	60

Table 2. Results of YOLOv3 algorithm. The values are given as %.

Metric	Precision	Recall	F-score
Large Vehicle	100	54	70
Plane	100	43	66
Ship	100	50	67
Basketball Court	0	0	0
Swimming Pool	100	71	83

It has been observed that the YOLOv3 algorithm gives better results in detecting large vehicles than theYOLOv2 algorithm with an F-score of 70% and an F- score of 39%. In determining the swimming pool, YOLOv3 gave a more successful result than YOLOv2 with an F-score of 83% and an F-score of 60%.

In plane class, YOLOv2 achieved the highest accuracy of all classes and two algorithms, and YOLOv3 achieved an F-score of 66%. Since theYOLOv3 algorithm could not find any basketball courts, the F-score value was 0, but YOLOv2 achieved 78% F- score success in this area.

Since the same data are utilized for both training and evaluation, the variations in outcomes are caused by variations in the methods. In the comparison phase, YOLOv2 had a five-class advantage while YOLOv3 had a four-class edge. Additionally, class accuracy scores are greater than class recall values. This indicates that false detection rates are often quite low for approaches. They are unable to find every real thing, though. As a result, YOLOv2 has more class accuracy than YOLOv3. However, YOLOv2 takes an average of 43 seconds to identify an item, compared to YOLOv3's average detection time of 2.5 seconds. It has been clearly seen that YOLOv3 outperforms YOLOv2 in terms of speed performance for object detection.

Conclusions

Using the YOLOv2 and YOLOv3 approaches, object recognition on aerial photos was carried out in this work. We used the DOTA dataset, which consists of several classifications and aerial photos. Future research will employ additional aerial photos in the training dataset, as well as images from more classes and the same class at different sizes, to enhance the performance of both algorithms. This will result in more accurate results and a greater success rate for item detection.

References

Albawi, S., Mohammed, T. A., Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. *In 2017 International Conference on Engineering and Technology (ICET)* :1-6

Atik, M. E., Duran, Z. (2020). Deep Learning-Based 3D Face Recognition Using Derived Features from Point Cloud. *In The Proceedings of the Third International Conference on Smart City Applications*, 797-808).Springer, Cham.

Atik, M. E., Duran, Z., Seker, D. Z. (2021). Machine Learning-Based Supervised Classification of Point Clouds Using Multiscale Geometric Features. *ISPRS International Journal of Geo-Information*, 10(3), 187.

Atik, S. O., Ipbuker, C. (2020). Instance SegmentationOf Crowd Detection In The Camera Images. *In Proceeding of Asian Conference on Remote Sensing 2020 (ACRS 2020).*

Atik, S. O., Ipbuker, C. (2021). Integrating Convolutional Neural Network and Multiresolution Segmentation for Land Cover and Land Use Mapping Using Satellite Imagery. *Applied Sciences*, *11*(12), 5551.

Atik, S. O., Ipbuker, C. (2021). Ship Detection fromSatellite Images with Instance Segmentation (Uydu Görüntülerinden Örnek Segmentasyonu ile Gemi Tespiti). *18. Harita Bilimsel ve Teknik Kurultayı*, 29- 29 Mayıs 2021, Ankara.

Cepni, S., Atik, M. E., Duran, Z. (2020). Vehicle detection using different deep learning algorithms from image sequence. *Baltic Journal of Modern Computing*, 8(2), 347-358.

Chen, E., Gong, Y., Tie, Y. (2016). Advances in Multimedia Information Processing. Category Aggregation Among Region Proposals for Object Detection. China: *17th Pasific Rim Conference on Multimedia* Xi'an, 210-211.

Dai, J., Li, Y., He, K., Sun, J. (2016). R-fcn: Object detection via region-based fully convolutional networks. *In Advances in neural information processing systems* (pp. 379-387).

Ding, J., Xue, N., Long, Y., Xia, G. S., Lu, Q. (2018).
Learning RoI transformer for detecting oriented objects in aerial images. arXiv preprint

arXiv:1812.00155.

Ding, J., Xue, N., Xia, G. S., Bai, X., Yang, W., Yang,
M. Y., ... & Zhang, L. (2021). Object detection in aerial images: A large-scale benchmark and challenges. arXiv preprint arXiv:2102.12219.

Gavrilova, M., Chang, J., Thalmann N. M., Hitzer, E., Ishikawa, H. (2019). Advances in Computer Graphics. Object Perception in the RGB Image. Canada: *36th Computer Graphics International Conference*, 478-430.

Girshick, R. (2015). Fast r-cnn. *In Proceedings of the IEEE international conference on computer vision*(pp. 1440-1448).

Gonultas, F., Atik, M. E., Duran, Z. (2020). Extraction of roof planes from different point clouds using RANSAC algorithm. *International Journal ofEnvironment and Geoinformatics*, 7(2), 165-171.

He, K., Gkioxari, G., Dollár, P., Girshick, R. (2017). Mask r-cnn. *In Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).

He, S., Lau, R. W. H., Liu, W., Huang, Z., Yang, Q. (2015). SuperCNN: A Superpixelwise Convolutional Neural Network for Salient Object Detection. *International Journal of Computer Vision*. doi 10.1007/s11263-015-0822-0.

Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Canada: University of Toronto.

Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., Sun, J. (2017). Light-Head R-CNN: In Defense of Two- Stage Object Detector. China: Tsinghua University. preprint arXiv: 1711.07264v2

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. *In European conference on computer vision* (pp. 740- 755). Springer, Cham.

Liu, M., Wang, X., Zhou, A., Fu, X., Ma, Y., Piao, C. (2020). UAV-YOLO: small object detection onunmanned aerial vehicle perspective. *Sensors*, 20(8), 2238.

Lu, J., Sibai, H., Fabry, E., Forsyth, D. (2017). NO need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles. USA: University of Illinois. arXiv preprint arXiv: 1707.03501v1.

Redmon, J., Farhadi, A. (2017). YOLO9000: better, faster, stronger. *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263-7271).