ISSN: 2148-9173 **Vol: 9 Issue:2 June 2022**

International Journal of Environment and Geoinformatics (IJEGEO) is an international, multidisciplinary, peer reviewed, open access journal.

Comparison of YOLO Versions for Object Detection from Aerial Images

Muhammed Enes ATIK, Zaide DURAN, Roni OZGUNLUK

Abstracting and Indexing: TR DIZIN, DOAJ, Index Copernicus, OAJI, Scientific Indexing Services, International Scientific Indexing, Journal Factor, Google Scholar, Ulrich's Periodicals Directory, WorldCat, DRJI, ResearchBib, SOBIAD

International Journal of Environment and Geoinformatics 9(2):087-093 (2022)

Research Article

Comparison of YOLO Versions for Object Detection from Aerial Images

Muhammed Enes Atik*, Zaide Duran, Roni Ozgunluk

Istanbul Technical University, Faculty of Civil, Geomatics Engineering Department, Istanbul, Turkey.

* Corresponding author: M.E. Atik
E-mail: atikm@itu.edu.tr

Received: 16.09.2021
Accepted : 01.12.2021

How to cite: Atik et al., (2022). Comparison of YOLO Versions for Object Detection from Aerial Images, *International Journal of Environment and Geoinformatics (IJEGEO)*, 9(2):087-093 doi. 10.30897/ijegeo.1010741

Abstract

Many different disciplines use deep learning algorithms for various purposes. In recent years, object detection by deep learning from aerial or terrestrial images has become a popular research area. In this study, object detection application was performed by training the YOLOv2 and YOLOv3 algorithms in the Google Colaboratory cloud service with the help of Python software language with the DOTA dataset consisting of aerial photographs. 43 aerial photographs containing 9 class objects were used for evaluation. These classes are large vehicle, small vehicle, plane, harbor, storage tank, ship, basketball court, tennis court and swimming pool. Accuracy analyzes of these two algorithms were made according to recall, precision and F1-score for nine classes, and the results were compared accordingly. YOLOv2 gave better results in 5 out of 9 classes, while YOLOv3 gave better results in recognizing small objects. While the best result with YOLOv2 was obtained in airplane class with 99% F1-score, the best result with YOLOv3 was obtained in swimming pool class with 83%. YOLOv2 can detect objects in an average photograph in 43 seconds, YOLOv3 has achieved superior performance in terms of time by detecting objects in an average of 2.5 seconds.

Keywords: Computer Vision, Deep Learning, Object Detection, YOLO, Aerial Image

Introduction

Digital image processing has been positively affected by developments in science and technology and has gained great popularity in photogrammetry, remote sensing and computer vision (Cepni et al., 2020). Digital image processing is a method of performing some operations on the image to obtain an enhanced image, converting the image into a digital format and extracting information 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... Object detection based on deep learning is widely used, especially with images obtained by remote sensing and photogrammetric methods (Yang et al., 2019). Larger datasets can be used and more powerful models can be developed to improve the performance of deep learning approaches to object detection. The most significant breakthroughs in object detection stem from the success of region-based methods and region-based convolutional neural networks (R-CNN) (Chen et al., 2016). Convolutional neural network-based object identification consists 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). Two-stage techniques work slower and produce results than single-stage techniques. The You Only Look Once (YOLO) method used in the study is one of the single-stage techniques. YOLO defines object detection as a regression problem and detects objects with spatially separated bounding boxes. (Gavrilova et al., 2019). YOLO produces results faster than other two-stage object identification methods thanks to its way of handling the problem.

This study aims to detect objects with YOLOv2 (Redmon and Farhadi, 2017) and YOLOv3 (Redmon and Farhadi, 2018) algorithms based on deep learning with aerial imagery in the DOTA data set, to investigate, compare, and reveal the deficiencies of the methods. Precision, recall and F-score were used as evaluation metrics.

Related Works

Many studies in the field of object detection with deep learning have been published in recent years. Cepni et al. (2020) compared different deep learning algorithms using terrestrial and UAV-based aerial images in their study. For this purpose, vehicles were chosen as the object to be detected and they analyzed their data with YOLOv3-YOLOv3-spp and YOLO-tiny models, which they trained on the COCO dataset (Lin et al., 2015) Google Colab. The study by Lu et al. (2017) investigated negative patterns in object detection in autonomous vehicles. Negative examples were created against both the YOLO detector and the traffic sign classification and then these negative images were suppressed. The process of a car driven with printed plates was simulated, and the YOLO sensor's detection rate was checked. In the study by Shafiee et al. (2017), they investigated YOLOv2, a new algorithm called Fast YOLO that accelerates real-time object detection from video in embedded devices.

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. Architecture (referred to here as O-YOLOv2) has been produced. Deep with O-YOLOv2 based on transient motion characteristics while maintaining performance to reduce power consumption in embedded devices further. In the study by Tan et al. (2018), a multi-target tracking algorithm based on YOLO has been proposed to further increase the accuracy and efficiency of multi-target tracking. After obtaining target, size, location, and other information, depth extraction was made. The noise in the image is removed and the computational and time cost of the feature extraction process is reduced. In the study by Krizhevsky et al. (2012), a deep convolutional neural network was trained to divide 1.2 million high-resolution images from the ImageNet LSVRC-2010 competition into 1000 different classes. In the test data, the two most common error rates of 37.5% and 17.0% were achieved, which are much better than the previous technology. He et al. (2015) studied explicit object detection with convolutional neural networks. Current computational models for salient object detection are based on handcrafted features that can only capture low levels of contrast information. Hierarchical contrast properties were learned by formulating conspicuous object detection as a binary tagging problem using deep learning techniques. Li et al. (2017) investigated the reasons why typical two-stage methods are slower than single-stage object detectors such as YOLO and SSD. The Faster R-CNN included two fully interconnected layers for ROI recognition, while the R-FCN generated a large score map. Thus, it has solved the intensive computation problem of Fast R-CNN and R-FCN before and after ROI skewing. Therefore, the speed of these networks has been found to be slow due to their architectural design. Ren et al. (2015) investigated the rich feature hierarchy for accurate object perception and semantic segmentation. As measured in the canonical PASCAL VOC dataset, object detection performance has increased significantly over the past few years. It has been found that the best-performing methods are often complex community systems that combine multiple low-level display features with high-level content. In their study based on deep learning algorithms, Liu, et al. (2020) have proposed a UAV-YOLO solution to solve the difficulties experienced in detecting small objects from UAV-based visuals. The study aims to create an image dataset obtained from the UAV platform, especially to improve the human detection performance and improve the neural network structure of the YOLO algorithm. In the study, the YOLOv3 algorithm was chosen and an improvement was made for the study by using the Darknet software framework.

In this study, a comparison of YOLOv2 and YOLOv3 methods over aerial images is presented. Especially since the data set includes both small and large objects, it is possible to evaluate the methods used in the study from different perspectives.

Material and Method

Data Used

DOTA (Xia et al., 2018; Ding et al., 2018; Ding et al., 2021) is a large-scale dataset for object detection with aerial images. Object categories in DOTA-v1.0 include airplane, ship, storage tank, baseball field, tennis court, basketball court, highway field, harbor, bridge, large vehicle, small vehicle, helicopter, football field and swimming pool. Some samples from the dataset are presented in Figure 1. The DOTA dataset includes Google Earth, GF-2 and JL-1 satellite images provided by the China Center for Resources Satellite Data and Application. In addition, the spatial resolution information of each image is presented in its metadata.

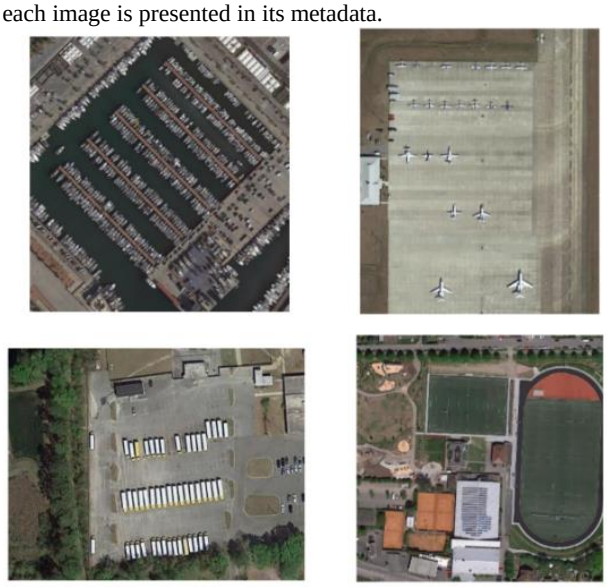

Fig. 1: Samples from DOTA Dataset

Convolutional Neural Networks (CNN)

A convolutional neural network (ConvNet/Convolutional neural networks / CNN) is a deep learning algorithm that can separate various views/objects from an input image. Convolutional neural networks are inspired by the organization and functionality of the visual cortex in the human brain. The most prominent aspect of CNNs is that it reduces the number of parameters in ANNs (Albawi et al., 2017). Convolution layers apply filters to the image to extract features at different levels from the image. Thus, a featured image is obtained, which, together with the first filter, defines a feature type. Then a second filter is applied to a second image for detecting another feature type. A convolutional neural network uses predictions from layers to represent the probability of a particular feature belonging to a particular class. Thus it produces a final output that presents a vector of probability points. Convolution layer, nonlinear layer, pooling layer, smoothing layer and fully connected layer form the convolutional neural network architecture (Atik and Ipbuker, 2021).

YOLO

You Only Look Once (YOLO) (Redmon et al., 2016) is an open-source object detection algorithm based on convolutional neural networks. YOLO is among the most well-known deep learning algorithms, and it stands out with its speed thanks to its single-stage detection architecture (Figure 2). Detection systems prior to YOLO reuse classifiers or localizers for object detection.

They apply the model to the image at multiple locations and scales. High-scoring regions of the image are defined as objects. As a different approach in YOLO, a single neural network is applied to the whole image, object detection is treated as a regression problem. In this network, the image is divided into regions and bounding boxes and probabilities are estimated for each region. These bounding boxes are weighted based on the estimated probabilities.

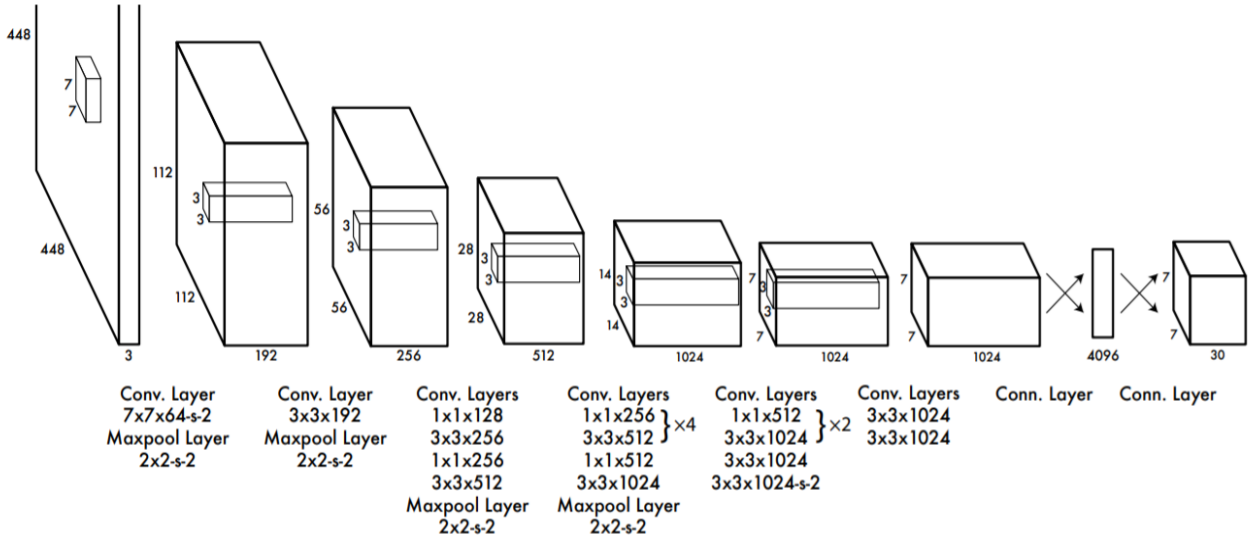

Fig. 2: Architecture of YOLO algorithm (Redmon et al., 2017).

YOLO first divides the input picture into grids of SxS for detection. The sizes of these grids may differ according to the versions, for example, grids such as 3x3 5x5 19x19 can be used. Each grid within itself is responsible for finding out whether there is an object in the field, if it is in its midpoint, if it is within its midpoint, its length, height, and class. As a result of these operations, bounding boxes are created. Then, an estimation vector is created for each grid. 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

YOLO makes significant localization errors. Therefore, it can be said that YOLO made a relatively high recall error. Therefore, in YOLOv2 (Redmon and Farhadi, 2017), the focus is mainly on enhancing recall and localization while maintaining classification accuracy.

With Yolov2, a new network design was introduced by removing the full connection layer and batch normalization (Sang et al., 2018). By adding batch normalization to all convolutional layers in YOLO to achieve better performance, more than 2% improvement from mAP was achieved. In addition, a high-resolution classifier was used for training. Classifier resolution has been increased from 224 to 448. This means that when switching to detection, the network must simultaneously switch to learning object detection and set to the new input resolution. Another innovation in YOLOv2 is the increase in accuracy and performance in multi-object recognition with the use of Anchor Box (Redmon and Farhadi, 2017).

YOLOv3

The YOLOv3 (Redmon and Farhadi, 2018) was developed as a result of the development and optimization of YOLOv2. Object detection has been identified as a regression problem in the YOLOv3 method. YOLOv3 predicts a confidence score for each bounding box using logistic regression. The YOLOv3 method splits the input image into small grid cells S×S. If an object falls inside a central cell, the grid cell must detect the object. Each cell estimates the position information of the B bounding boxes and calculates the objectivity scores of these bounding boxes. According to this algorithm, the confidence score should be 1 if the bounding box covers an object of known ground accuracy more than the other bounding boxes previously (Zhao and Li, 2020).

The system assigns only one bounding box for each object of known ground accuracy. If a previous bounding box is not assigned to an object of known location accuracy, it will not create any coordinate or class estimates loss. In some cases, the box does not have the highest IOU but covers a precision object more than some threshold. In these cases, the prediction is ignored. It uses binary cross-entropy loss for class predictions during training (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 finding small objects. Using this linkage method provides more detailed information from the previous feature map. But compared to YOLOv2, YOLOv3 has less achievement on medium and large-sized objects.

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).

$$Precision = TP / (TP + FP) \quad \text{(Eq.1)}$$

$$Recall = TP / (TP + FN) \quad \text{Eq.(2)}$$

$$F1\text{-}score = 2\,(Precision \times Recall) / (Precision + Recall) \quad \text{(Eq.3)}$$

Implementation (YOLO)

In this study, YOLOv2 and YOLOv3 algorithms were trained with the DOTA dataset. 43 aerial images were used for testing. DOTA-v1.0 version, which is the first version, was used in the application. Object categories in the dataset include: airplane, ship, storage tank, baseball field, tennis court, basketball court, highway field, harbor, bridge, large vehicle, small vehicle, helicopter, intersection, football field and swimming pool. In the dataset, the location of each object is explained by bounding boxes that can be represented as "x1, y1, x2, y2, x3, y3, x4, y4". The dataset contains 2806 different aerial views.

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 in training models.

Label data is defined as "x1 y1 x2 y2 x3 y3 x4 y4 category 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. In both YOLOv2 and YOLOv3 algorithms, the models were trained in the DOTA dataset, and the working of the models was checked with test photos. After the successful completion of the control phase, output files were obtained from 43 photographs for both models in order to compare and evaluate the algorithms, in other words, to perform accuracy analysis. Outputs were obtained in both algorithms for 43 images and these results were evaluated one by one (Figure 3 and Figure 4).

Fig. 3: Detection of objects in the DOTA dataset using YOLOv2.

Fig. 4: Detection of objects in the DOTA dataset using YOLOv3.

Results and Discussion

Three different evaluation metrics were used according to the conditions coming from the error matrix. The first is recall, the second is precision, and the other is the F-score. Recall, Precision and F-measure accuracies of 9 classes of YOLOv2 and YOLOv3 algorithms are shown in Table 1 and Table 2.

The detection of 9 object classes consisting of large vehicles, small vehicles, storage tanks, ships, tennis courts, ports, planes, basketball courts and swimming pools was subjected to accuracy comparisons in YOLOv2 and YOLOv3 algorithms. Recall, precision (Precision) and F-measure (F-measure) were used as accuracy criteria in the light of the information obtained from the error matrix. In comparing the success of the algorithms, the F-score has been the criterion for accuracy since it takes both recalls and sensitivity criteria into account. 43 photographs with various classes were selected for evaluation.

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

Metric	Precision	Recall	F-score
Large Vehicle	99	24	39
Small Vehicle	99	3	6
Plane	99	99	99
Harbor	100	79	88
Storage Tank	86	11	20
Ship	99	62	76
Basketball Court	100	64	78
Tennis Court	100	67	80
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
Small Vehicle	99	32	48
Plane	100	43	66
Harbor	100	28	44
Storage Tank	100	25	40
Ship	100	50	67
Basketball Court	0	0	0
Tennis Court	100	47	64
Swimming Pool	100	71	83

It has been observed that the YOLOv3 algorithm gives better results in detecting large vehicles than the YOLOv2 algorithm with an F-score of 70% and an F-score of 39%. In small vehicles, it has been determined that the YOLOv3 algorithm is 8 times more successful than YOLOv2, which yields a 48% F-score with a 6% F-score. In storage tanks, YOLOv3 was twice more successful than YOLOv2, and F-score values of 40% and 20% were obtained, respectively. 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%.

The YOLOv2 algorithm, with an F-score of 76%, gave a more successful result than YOLOv3 with an F-score of 76% in ship detection. YOLOv2, which gave an F-score of 80% in finding tennis courts, gave a better result than YOLOv3, which gave an F-score of 64%. In finding ports, YOLOv2 is twice as successful as YOLOv3, and the F-score values of 88% and 44% were obtained, respectively. In plane class, YOLOv2 achieved the highest accuracy of all classes and two algorithms, and YOLOv3 achieved an F-score of 66%. Since the YOLOv3 algorithm could not find any basketball courts, the F-score value was 0, but YOLOv2 achieved 78% F-score success in this area.

The differences in the results arise from the differences in the algorithms as the same data are used in training and evaluation. In the comparison phase, there was a 5 class's advantage in YOLOv2 and 4 classes in YOLOv3. Also, the precision values of classes are higher than recall values. This means that methods usually have a very low rate of false detection. However, they cannot detect all objects that actually exist. Therefore, the class accuracy of YOLOv2 is more than YOLOv3. However, while YOLOv3 can detect objects in an average of 2.5 seconds, YOLOv2 has an average of 43 seconds to detect the object. The superiority of YOLOv3 over YOLOv2 in terms of speed performance in object detection has been clearly observed.

Conclusions

In this study, object detection was performed on aerial images using YOLOv2 and YOLOv3 methods. The DOTA dataset, consisting of aerial photographs and containing many classes, was used. In future studies, to improve the results in both algorithms, increasing the number of aerial images used in the training dataset, increasing the images belonging to more classes and the same class at different scales will provide more positive results and a higher accuracy rate in order to increase the success in object 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 Segmentation Of 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 from Satellite 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 of Environment 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 on unmanned 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).

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

Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

Ren, S., He, K., Girshick, R., Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 91-99.

Sang, J., Wu, Z., Guo, P., Hu, H., Xiang, H., Zhang, Q., Cai, B. (2018). An improved YOLOv2 for vehicle detection. *Sensors*, 18(12), 4272.

Shafiee, M. J., Chywl, B., Li, F., Wong, A. (2017). Fast YOLO: A Fast You Only Look Once System for Real-Time Embedded Object Detection in Video. Canada: University of Waterloo. preprint arXiv: 1709.05943v1.

Tan, L., Dong, X., Ma, Y., Yu, C. (2018). A Multiple Object Tracking Algorithm Based on YOLO Detection. *In 2018 11th International Congress on Image and Signal Processing*, BioMedical Engineering and Informatics (CISP-BMEI 2018). China: Beijing Technology and Business University.

Xia, G. S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., ... & Zhang, L. (2018). DOTA: A large-scale dataset for object detection in aerial images. *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3974-3983).

Yang, M. Y., Liao, W., Li, X., Cao, Y., Rosenhahn, B. (2019). Vehicle Detection in Aerial Images. *Photogrammetric engineering and remote sensing: PE&RS*, 85(4), 297-304.

Zhao, L., Li, S. (2020). Object detection algorithm based on improved YOLOv3. *Electronics*, 9(3), 537.