



## Detection of Object (Weapons) with Deep Learning Algorithms from Images Obtained by Unmanned Aerial Vehicles

Mustafa BURGAZ<sup>1\*</sup>, Cafer BUDAK<sup>2</sup>

<sup>1</sup>PhD student of Department of Electronic-Communication Engineering Yıldız Technical University, [mustafa.burgaz@std.yildiz.edu.tr](mailto:mustafa.burgaz@std.yildiz.edu.tr)  
Orcid No:0000-0001-7525-2649

<sup>2</sup>Department of Electric-Electronic Engineering, Dicle University, [cafer.budak@dicle.edu.tr](mailto:cafer.budak@dicle.edu.tr) Orcid No:0000-0002-8470-4579

### ARTICLE INFO

#### Article history:

Received 13 May 2022  
Received in revised form 22 June 2022  
Accepted 24 June 2022  
Available online 28 June 2022

#### Keywords:

Deep learning, unmanned aerial vehicles, regional based convolutional neural networks, ResNet

Doi: 10.24012/dumf.1116534

\* Corresponding author

### ABSTRACT

Today, the rapid development of Artificial Intelligence technologies is effective in the success of deep learning algorithms in different application areas. These applications detect many objects that even the human eye cannot detect in object detection in videos and images with deep learning algorithms. In this study, it is aimed to detect weapons from images obtained from Unmanned Aerial Vehicle (UAV) by using deep learning algorithms. Images were obtained from the UAV at 200 different angles and heights. Images from different angles and heights obtained from the unmanned aerial vehicle are trained by Regional Based Convolutional Neural Networks (R-CNN) and Residual Neural Network (ResNet). Two-thirds of the images we obtained were split into training images and one-third into test images. The feature maps extracted from the images used for training were compared with the test images. By bringing these compared images closer to the desired images, 99% of the desired image detection is achieved. Performance evaluation of the algorithms was made using Loss plot, mAP curves, Precision, Recall and F1-Score. The performance evaluation of the detected images is discussed, and the success of deep learning algorithms used in object detection is presented. The ResNet model showed higher performance with 64% accuracy, 94% recall and 76% F1 score.

## Introduction

There has been rapid development in the field of artificial intelligence in recent years. The source of this development is seen as a result of the success of machine learning science in different application areas. Deep learning algorithms, which have started to be used widely in machine learning science, have opened the door to new technologies [1]. Deep learning algorithms play an important role in the design of the technologies produced for different purposes. Thanks to the algorithms used in deep learning, applications have been developed in many different areas. Increasing production in agriculture and rural development [2], reducing noise in the health and biomedical images [3], making images meaningful in the field of culture and art [4], analyzing crowd density in the field of security and arranging social areas, etc. areas are shown as examples of these applications. Deep learning also gives successful results in object detection with multiple algorithms. Object detection is a technologically challenging and practically useful problem. Object detection is concerned with the detection of various objects. Recognition of objects in images is one of the challenging problems in computer vision, especially when the number of objects is large. While humans can

recognize thousands of object types, most existing object recognition systems will only recognize a few [5]. Object detection is defined as bringing the region, which has a feature in its entirety, to the fore with different image processing methods. Therefore applications that are made by making use of features such as classification and comparison, which form the basis of object detection, are becoming widespread.

Among the deep learning algorithms that make these applications very fast and successful, Regional-Based Convolutional Neural Networks (R-CNN) are preferred in object detection.

R-CNN provides a transition in object detection in the image classification as simple as possible. Implementing and training R-CNNs has become valuable as it is straightforward and provides a unified solution for object detection and segmentation [6].

ResNet algorithm has been developed to improve its performance in R-CNN image classification. ResNets ILSVRC 2015 achieves state-of-the-art performance in its classification task and allows training deep networks of up to 1000 layers. Similar to road networks, it uses identity shortcut links that enable the flow of information without the attenuation caused by multiple stacked nonlinear transforms. Thus, remaining networks are transmitted as

input [7]. Faster R-CNN and ResNet algorithms detect the given images by considering possible locations.

In this study, it is aimed to make the images obtained from UAVs work in real-time with object detection algorithms.

## Literature Review

There are many studies in the literature on object detection with UAV images [11]. Sought an automatic solution to the problem of detecting and counting cars on UAV images. They did not start by dividing the input image into small homogeneous regions that could be used as candidate locations for vehicle detection and then used deep learning architecture to adopt a highly descriptive feature from these windows by removing a window around each region. They used the trained neural network system in conjunction with the SVM classification to classify the regions as "automobile" and "car free". All regions with a bounding box greater than 200 in width or length were eliminated in advance, as the regions extracted in practice have variable dimensions, and considering that an average car known in the test images is about  $200 \times 90$  pixels, it is pointless to examine regions larger than  $200 \times 200$  pixels. For the regions within the size limitations, a  $160 \times 160$ -pixel window is kept around each region and access to the CNN algorithm is provided for feature extraction. The sensitivity analysis experiment concluded that the  $160 \times 160$ -pixel window size at an 80% level provided the best overall accuracy of 93.6% for  $200 \times 200$  pixels.

A new UAV study focusing on complex scenarios, created 14 feature maps with approximately 80,000 representative frame bounding boxes selected from 10 hours of raw videos. They divided object detection into single object tracking and multiple object tracking. Later, they conducted a detailed quantitative study for each task using the latest technology algorithms. Experimental results show that the newest methods available perform relatively poorly in the dataset due to new challenges arising in real scenes based on UAVs, such as high density, small objects and camera movement [8].

Radovic et al. (2017) detailed the parameters used in CNN training in a series of aerial images for efficient and automatic object detection. They attributed the accuracy and reliability of CNNs to the training of the network and the selection of operational parameters and CNN [9].

They detailed the training procedure and parameter selection. They used a new dataset of 267 images containing 540 aircraft to test the CNN recognition accuracy. Object detection results showed that by choosing an appropriate parameter set, a CNN could detect and classify objects with high accuracy (97.5%) and computational efficiency. They showed that CNN was able to recognize "airplane" objects in the data set with 97.5% accuracy (526 out of 540 "aircraft" objects), only 16 samples were miscategorized (14 aircraft were not identified). An incorrectly categorized sample was an example where the image contained an airplane but was not recognized by the network. The tagged results calculated the positive predictive value for CNN as 99.6%, false discovery rate 0.4%, true positive rate 97.4% and false negative rate 2.6%.

Ye et al. (2018) presented a new approach to detection and tracking from a single camera mounted on a UAV with moving object detection and tracking algorithms in a video. Initially, they predicted background movements through a perspective transformation model. They then identified moving objects in the background extracted image through a deep learning classifier trained on manually tagged data sets. They found spatial-temporal properties for each candidate moving object through optical flow matching and then pruned according to their motion patterns compared to the background. They used Kalman Filter to increase the temporal consistency of detecting moving objects. The experimental results in the real video data set showed that the deep learning method could increase detection accuracy with the help of appearance information. The results showed that the algorithms could detect and monitor small UAVs with limited computing resources. They used multiple target UAVs with various views and shapes for the data set. They corrected 40 videos as a training set for deep learning to create a basic accuracy data set for training and performance evaluation. The deep learning method was fully utilized from the manually tagged training dataset with a classification accuracy of over 95% [10].

## Method

The CNN algorithms (Faster R-CNN and ResNet) used in the study take into account the weights that will adversely affect training in order to train the images obtained with UAVs during the object detection phase. Pre-trained IMAGENET weights are used for ResNet and R-CNN. The block diagram of the system is shown in Fig. 1.

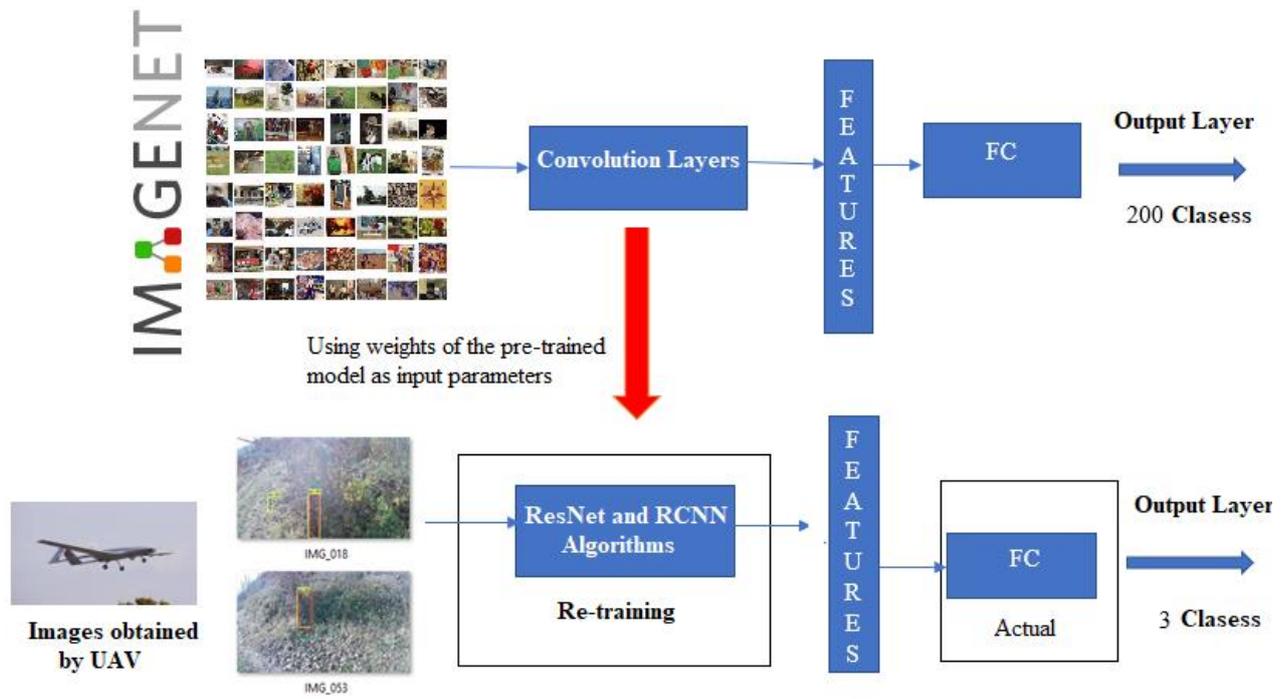


Figure 1. The block diagram of the system

In the block diagram in Fig. 2, R-CNN and ResNet models whose weights have been saved from the images taken with the UAV perform object detection.



Figure 2. Object detection process

R-CNN, a successful object detection algorithm, has derivatives such as Fast R-CNN, Faster R-CNN and Mask R-CNN. These deep learning algorithms are used in many deep learning libraries. TensorFlow library is one of them. TensorFlow is a software library developed by the Google Brain-Machine within Google's Machine Learning Intelligence research organization to conduct machine learning and deep neural network research [12].

In order to improve something in machine learning, it often needs to be measured. TensorBoard is used as an important tool to provide the measurements and visualizations required during machine learning's object detection. TensorBoard provides the function of tracking trial metrics such as loss and accuracy, visualizing the model plot, projecting nodes into a lower-dimensional area and much more [13].

**Performance Evaluation**

AP (Average sensitivity) is a popular metric used to measure object detection accuracy. AP calculates the AP value for the recall value from 0 to 1 [14]. While the average sensitivity continues to be used while measuring the accuracy of the object detection models, we can also measure the accuracy of the object detection models used by creating a bounding box.

It regroups many challenging tasks such as object detection, classification and regression tasks. During the object detection process, the models create many bounding boxes with different confidence values. The Intersection over Union (IoU) field is the overlapping area between the ground truth box and the predicted box. A higher IoU is a sign of a better-predicted bounding box position. Usually, all bounding box candidates are kept with an IoU area greater than or equal to some threshold value. True positive (TP), False positive (FP), and False negative (FN) are used in sensitivity and recall calculations to determine the performance of a model.

Table 1. Tables of Precision, Recall and F1-Score metrics [15]

| Metric    | Equation  |
|-----------|---|
| Precision | $\frac{TP}{TP + FP}$                                |
| Recall    | $\frac{TP}{TP + FN}$                                |
| F1-Score  | $2 * \frac{Precision * Recall}{Precision + Recall}$ |

Detection boxes\_Precision mAP graph is shown in Fig. 3. The scores in the mAP graph are related to capturing objects between 32<sup>2</sup> and 96<sup>2</sup> pixels. In the figure, The approximate value of the Faster R-CNN architecture is

0.38. This value indicates that it can detect small size objects.

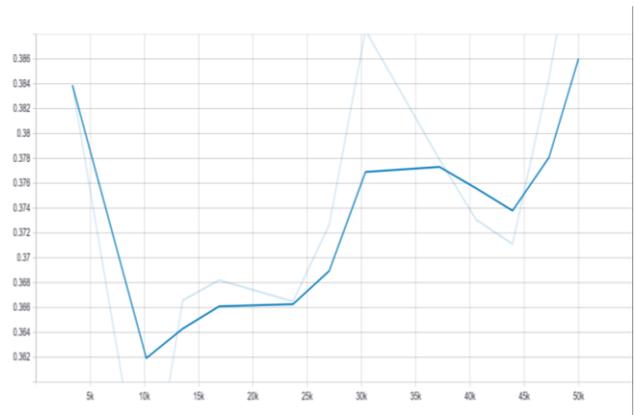


Figure 3. Detection boxes precision map chart

Detection boxes Precision mAP graph is shown in Fig. 4. The scores in the mAP graph are related to capturing objects located between pixels lower than 32<sup>2</sup>. In the figure, the approximate value of the ResNet architecture is 0.29. This value indicates that it can detect small size objects

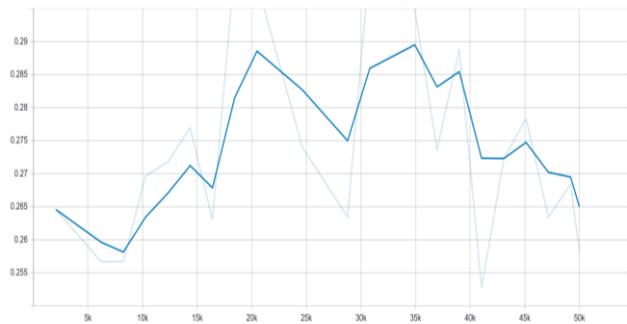


Figure 4. Detection boxes precision map chart

R-CNN architectures used for object detection are algorithms that give very successful results in the field of object detection. These algorithms identify the desired object by classifying the object detection. Since the network structures that use the location information of the object are multi-layered, many outputs are obtained. Each of these outputs has its unique Loss (error) graphs. The Loss (Error) graphs of the Faster R-CNN and ResNet architectures from the R-CNN architecture we used in our study are compared in Fig. 5 and Fig. 6.

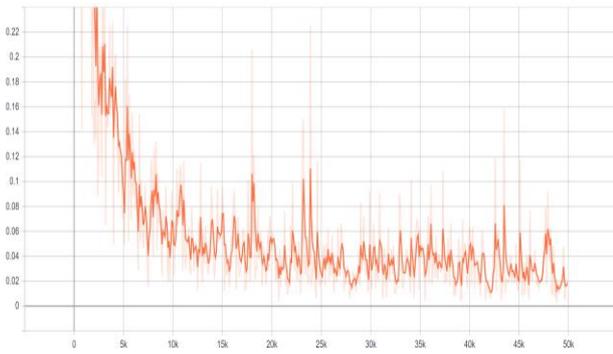


Figure 5. Faster R-CNN architecture loss graph

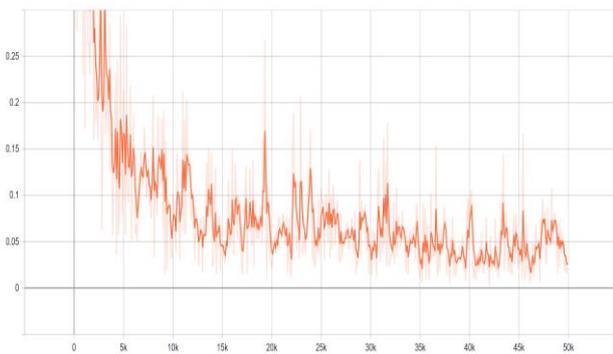


Figure 6. ResNet architecture loss graph

Faster R-CNN and ResNet architecture, Loss Error graphs approaching zero (0) show that both architects are successful. The loss value of the Faster R-CNN architecture is (0.02) and the Loss value of the ResNet architecture is (0.05), the error values of the two architectures are approximately zero, indicating that the R-CNN architectures are successful in object detection and identification applications for object detection.

The algorithms used during object detection make an estimate. The prediction success of the algorithms used in the study was evaluated with the test data set at hand. The targeted values in the data set are called ground truth (absolute reference). In Fig. 7 and Fig. 8, our ground truth values of the Faster R-CNN and ResNet architectures can be seen. The area framed in red is the ground truth, while the area framed with other colors shows the predicted (predict) objects.

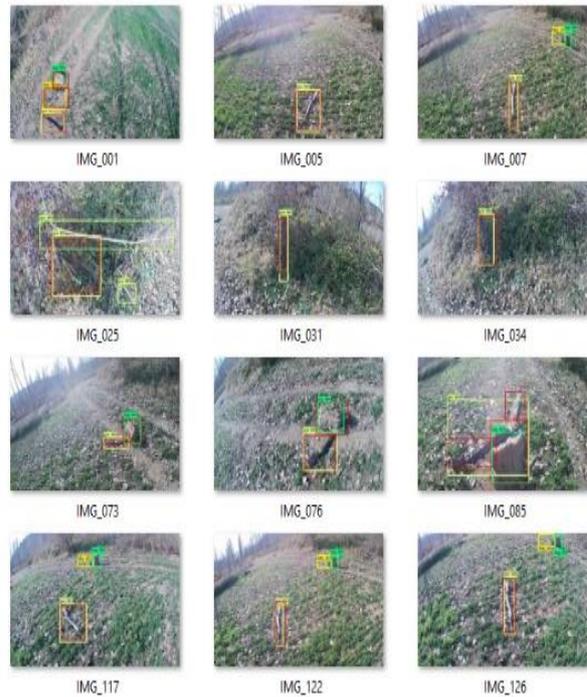


Figure 7. Faster R-CNN ground truth values

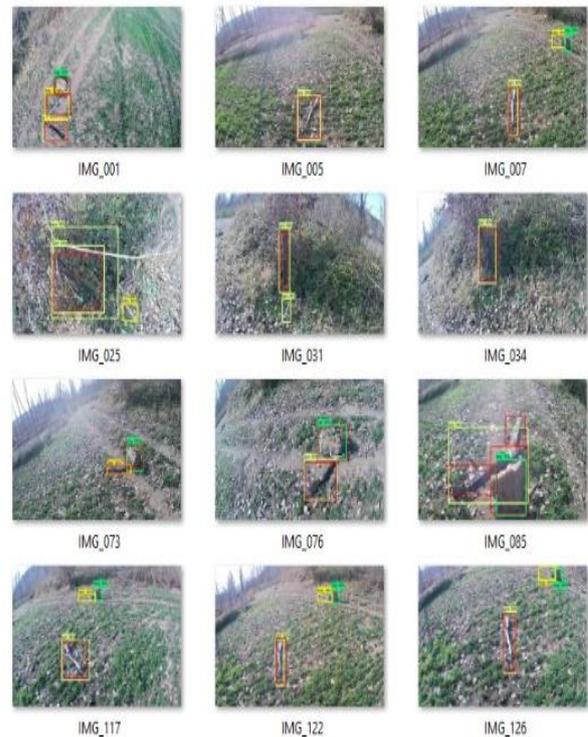


Figure 8. ResNet ground truth values

When the TP, FP and FN values obtained in the test data of the Faster R-CNN architecture used in the study are examined, it is seen that the correct classification values and the wrong classification values are quite high. The TP, FP, and FN values obtained for Faster R-CNN are shown in Table 2, while the results obtained for ResNet are shown in Table 3.

Table 2. TP, FP and FN values of Faster R-CNN algorithm

|             | TP    | FP    | FN    | P1    | P     | R1    | R     | P.R   | P+R   | F1    |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| IMG_001.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_007.jpg | 1     | 2     | 0     | 3     | 0,33  | 1     | 1     | 0,33  | 1,33  | 0,49  |
| IMG_016.jpg | 1     | 0     | 0     | 1     | 1     | 1     | 1     | 1     | 2     | 1     |
| IMG_022.jpg | 1     | 0     | 0     | 1     | 1     | 1     | 1     | 1     | 2     | 1     |
| IMG_025.jpg | 1     | 2     | 0     | 3     | 0,33  | 1     | 1     | 0,33  | 1,33  | 0,49  |
| .....       | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| .....       | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| .....       | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| IMG_135.jpg | 1     | 2     | 0     | 3     | 0,33  | 1     | 1     | 0,33  | 1,33  | 0,49  |
| IMG_155.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_167.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_182.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_189.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |

Table 3. TP, FP and FN values of ResNet algorithm

|             | TP    | FP    | FN    | P1    | P     | R1    | R     | P.R   | P+R   | F1    |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| IMG_001.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_007.jpg | 1     | 2     | 0     | 3     | 0,33  | 1     | 1     | 0,33  | 1,33  | 0,49  |
| IMG_016.jpg | 1     | 0     | 0     | 1     | 1     | 1     | 1     | 1     | 2     | 1     |
| IMG_022.jpg | 1     | 0     | 0     | 1     | 1     | 1     | 1     | 1     | 2     | 1     |
| IMG_025.jpg | 1     | 2     | 0     | 3     | 0,33  | 1     | 1     | 0,33  | 1,33  | 0,49  |
| .....       | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| .....       | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| .....       | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| IMG_155.jpg | 2     | 2     | 1     | 4     | 0,5   | 3     | 0,66  | 0,33  | 1,16  | 0,56  |
| IMG_167.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_182.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_189.jpg | 3     | 0     | 0     | 3     | 1     | 3     | 1     | 1     | 2     | 1     |
| IMG_202.jpg | 3     | 2     | 0     | 5     | 0,6   | 3     | 1     | 0,6   | 1,6   | 0,75  |

When the TP, FP and FN values obtained in the test data of the ResNet and RCNN architecture used in the study are examined, it is seen that the correct classification values and the wrong classification values are quite high.

Table 4. Recall, Precision, and F1-Score Values for R-CNN and ResNet when IoU=0.5

| IoU=0.5   |        |          |           |        |          |
|-----------|--------|----------|-----------|--------|----------|
| R-CNN     |        |          | ResNet    |        |          |
| Precision | Recall | F1-score | Precision | Recall | F1-score |
| 0.62      | 0.88   | 0.72     | 0.64      | 0.94   | 0.76     |

The precision and Recall curve shows Precision and Recall equilibrium at different threshold values. High calling at the points indicated by the curve indicates the points where high precision intersects. This curve is understood from the curve directly related to the low value of FP and the low value of FN if the value of R is high for P to be high. The high P and R values in the curve are interpreted to be correct results in the classification made during object detection, and most of the results are detected correctly. In Fig. 9 and Fig. 10, P and R curves of the Faster R-CNN and ResNet architecture are shown.

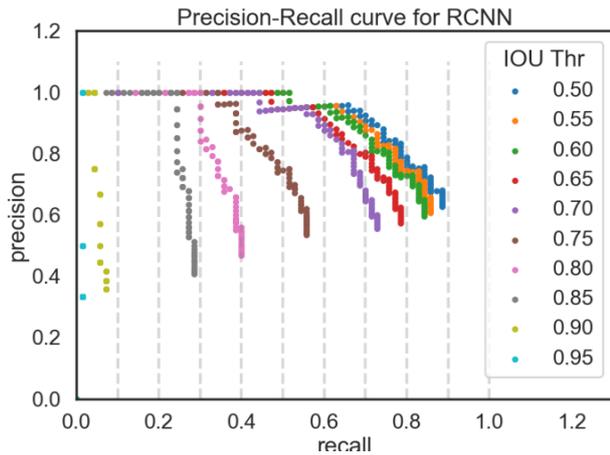


Figure 9. Precision recall curve of the Faster R-CNN architecture.

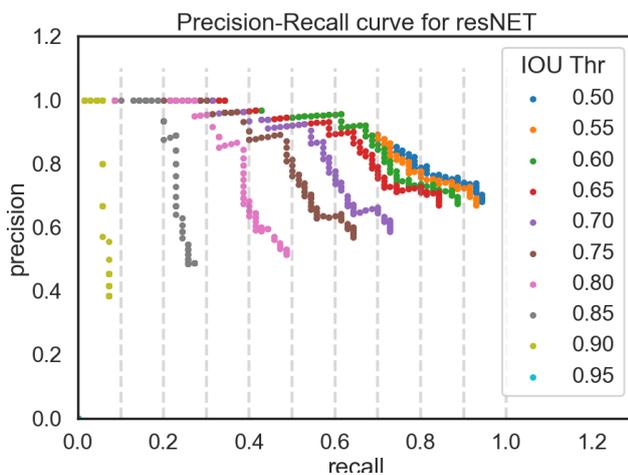


Figure 10. Precision recall curve of the ResNet architecture

The mAP value of the Faster R-CNN architecture in our training (50.84) and the mAP value of the ResNet architecture (52.17) are seen to be close to the sensitivity values of both architectures.

## Results and Discussion

Object detection application was made by using deep learning algorithms in the data set of 200 images created in the study. The UAV used in the application made object detection using Faster R-CNN and ResNet algorithms from the data sets obtained from a height of 50 meters. In this application, it is seen in the results that Faster R-CNN and ResNet architectures are successful in object detection. The architectures used for Loss Error graphs are close to 0, indicating that the erroneous outputs are minimal during training. Obtaining the Faster R-CNN Loss value of (0.02) and the ResNet Loss value of (0.05) revealed the reason why both architectures were preferred in the detection of objects (weapons) in the images during training. Because the training was carried out in 50,000 steps in two architectures, the threshold value (0.5 - 0.95) ensured that the Precision and Recall values reached the

appropriate values for object detection. It is seen in the object detection on the image in the training results that the two architectures used can reach the accuracy values with 99% estimation.

Evaluation metrics Precision, Recall and F1-Score are used for the R-CNN and ResNet algorithms. In the metrics used, the recall rate of the ResNet algorithm was 0.94, and the recall rate of the RCNN algorithm was 0.88. When the precision values of these algorithms were compared, ResNet algorithms had a 0.64 ratio to R-CNN algorithms and a 0.62 ratio to ResNet algorithms. It seems to work more precisely than R-CNN algorithms.

In addition, the IoU calculation is calculated by dividing the area where the two rectangles intersect (intersection) by the junction area of these two rectangles. In our study, the 0.5 IoU value shows that ResNet and R-CNN algorithms are successful for object detection.

With the development of UAV technologies, the air flight altitude will vary, and the ResNet and R-CNN algorithms we use are considered to be successful at different altitudes.

Applications that are successful in object detection will be preferred in many areas in the future. This preferred R-CNN and ResNet will be transported to UAV usage areas and will be a very important step in catching up with today's technologies.

## References

- [1] C. Gaszczak and T. Breckon, J. Han, "Real-time people and vehicle detection from uav imagery", *Intelligent Robots And Computer Vision*, pp. 1-13, 2011, Available:[https://www.researchgate.net/publication/253357566\\_Realtime\\_People\\_and\\_Vehicle\\_Detection\\_from\\_UAV\\_Imagery](https://www.researchgate.net/publication/253357566_Realtime_People_and_Vehicle_Detection_from_UAV_Imagery)
- [2] Y. Uzun, M. Bilban and H. Arıkan, "Tarım ve kırsal kalkınmada yapay zekâ kullanımı", VI. KOP Bölgesel Kalkınma Sempozyumu-KOPBKS, Konya, 2018.
- [3] C. Budak, M. Türk and A. Toprak, "Reduction in impulse noise in digital images through a new adaptive artificial neural network model", *Neural Comput & Applic*, vol. 26, no.4, pp. 835-843, 2015.
- [4] D. K. Prasad, "Survey of the problem of object detection in real images", *International Journal of Image Processing*, vol. 6, no. 6, pp. 441-466, 2012.
- [5] M. Kızrak and B. Bolat, "Derin öğrenme ile kalabalık analizi üzerine detaylı bir araştırma", *Bilişim Teknolojileri Dergisi*, vol. 11, no.: 3, pp. 263-285, 2018.
- [6] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.38, no.1, pp.142-158, 2016.
- [7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385,2015.

- [8] D. Du, Y. Qi and H. Yu, "The unmanned aerial vehicle benchmark: object detection and tracking", *Computer Vision Foundation*, 2018, Available: <https://arxiv.org/abs/1804.00518>
- [9] M. Radovic, O. Adarkwa and Q. Wang, "Object recognition in aerial images using convolutional neural networks", *Journal of Imaging*, vol. 3, no. 21, pp. 3-29, 2017.
- [10] D. Ye, J. Li and Q. Chen, "Deep learning for moving object detection and tracking from a single camera in unmanned aerial vehicles (uavs)", *Society for Imaging Science and Technology*, 2018, Available: <http://docserver.ingentaconnect.com/deliver/connect/ist/24701173/v2018n10/s17.pdf?expires>
- [11] N. Ammour, H. Alhichri and Y. Bazi, "Deep learning approach for car detection in UAV imagery", *Remote Sensing*, 2017, pp. 312
- [12] G. Zaccane, *Getting started with TensorFlow*. Kindle edition, pp.2, 2016.
- [13] Tensorflow, 2020, Available: <https://www.tensorflow.org/>
- [14] The PASCAL Visual Object Classes (VOC) Challenge, by Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn and Andrew Zisserman
- [15] C. Budak, V. Mençik, Detection of ring cell cancer in histopathological images with region of interest determined by SLIC superpixels method. *Neural Comput & Applic* (2022). <https://doi.org/10.1007/s00521-022-07183-8>