Deep learning-based software for detecting population density of Antarctic birds

Sinan Uğuz

Abstract

Monitoring populations of bird species living in Antarctica with current technologies is critical to the future of habitats on the continent. Studies of bird species living in Antarctica are limited due to climate, challenging geographic conditions, and transportation and logistical constraints. The goal of this study is to develop Deep Learning-based software to determine the population densities of Antarctic penguins and endangered albatrosses. Images of penguins and albatrosses obtained from internet sources were labeled using the segmentation technique. For this purpose, 4144 labeled data were trained with five different convolutional neural network architectures TOOD, YOLOv3, YOLOF, Mask R-CNN, and Sparse R-CNN. The performance of the obtained models was measured using the average precision (AP) metric. The experimental results show that the TOOD-ResNet50 model with 0.73 AP^{50} detects the Antarctic birds adequately compared to the other models. At the end of the study, a software was developed to detect penguins and albatrosses in real time.

Keywords: Deep Learning, Antarctic birds, Remote sensing, Population estimation, Convolutional neural network.

MSC 2020: 68T07, 68T45, 62P12.

1 Introduction

As a result of global warming in the world, the melting of sea ice has negatively affected the feeding ecosystems of populations of bird species such as penguins, albatrosses, skuas, cormorants, petrels, and Arctic

doi:10.56415/csjm.v31.11

terns living in Antarctica. Polar regions have difficult conditions for scientific studies due to climatic difficulties, geographic barriers, transportation and logistical constraints, and ecological limitations. Despite these difficult conditions, knowledge of bird species populations is very important for the region's habitat. Because many seabird species are closely spaced in colonies or breed in large numbers in inaccessible areas, population estimates of these birds are difficult [1]. Penguins are among the most important bird species living in Antarctica. It is estimated that the total number of penguin pairs breeding in Antarctica, where there are 18 different species of penguins, is about 20 million. Although penguins cover a large geographic area, they are concentrated in coastal areas with less harsh climates [2]. Another important bird species that lives in Antarctica is the albatross. All but seven of the world's 22 albatross species are threatened with extinction. Every year, tens of thousands of albatrosses die because they get caught in large nets while fishing behind fishing boats [3].

Determining the population densities of both bird species will provide information on the breeding behavior of these birds and make an important contribution to the protection of the Antarctic habitat. To this end, researchers have used various remote sensing technologies such as satellite imagery and computer vision techniques. In [4], satellite imagery with a resolution of 10 m was used to survey the penguin population and determine their reproductive behavior. The same authors used commercial satellite imagery at 30 cm resolution for albatross colonies in their 2017 study [5]. In [6] were identified breeding colonies of the bird species Thalassoica Antarctica using images in six spectral bands from Landsat-8. In another study using satellite imagery, in [7] was used a convolutional neural network (CNN) called U-Net for albatross colony detection. According to the researchers, the main limitations of the research are noise in satellite images, cloud cover, and complex background images.

The main disadvantage of studies based on satellite imagery is that the spatial resolution of satellite imagery is not high enough to clearly detect birds. While it is possible with military satellites to detect objects in an area of 10 cm^2, these satellites cannot be used by researchers because they are not open to the public [8]. Another problem that arises when using satellite imagery is that the number of birds detected in the colony cannot be accurately estimated. In particular, birds approaching each other for warmth in very cold time periods cause the population to be miscalculated. Satellite imagery can be of great use in determining the location of new colonies. However, various solutions need to be developed to determine the population density at that location.

The most successful methods for bird colony population detection are Deep Learning, which has recently achieved great success in all fields, and image processing techniques [9]. In image processing, researchers must manually extract features from images and apply various machine-learning techniques to solve each problem. This is a very long process. Also, in order to define an object, all the features must be defined by the researchers; for example, to create a penguin's feature map using image processing, researchers must define features such as its beak, arm, and leg. However, in the Deep Learning-based approach, feature extraction is done automatically using Deep Learning architectures [10]. In [9], images of birds near lakes and on agricultural land were collected by unmanned aerial vehicles (UAVs). This dataset includes popular CNN models, Faster Region-based CNN (Faster R-CNN), Region-based Fully Convolutional Network (R-FCN), Single Shot MultiBox Detector (SSD), RetinaNet, and You Only Look Once (YOLO). The best accuracy was achieved with Faster R-CNN and the fastest result production with YOLO.

In the other study [11], a penguin dataset was created. The unique feature of this dataset is the use of the dot annotations technique. That is, each penguin in the dataset is represented by dots. The researchers proposed a CNN model based on the VGG16 classification architecture. As a result of the study, the population density of penguins was determined by creating kernel density maps. In [12], it was aimed to detect Black-browed albatross and Southern Rockhopper penguin colonies using drone imagery. Approximately 37.000 data labels were made. Training was conducted using the RetinaNet architecture. The mAP for the albatross model was 97.66% and the mAP for the penguin model was 87.16%. In another study, the Penguin Counting model developed by[1] is still in the project phase. The main objective of the project is to count the penguins in the images captured by the camera traps deployed in Antarctica. For this purpose, the Microsoft Azure platform and the PyTorch framework were used.

Though several approaches were presented for population prediction of Antarctica birds, there exist some significant challenges in it. Some Antarctic seabirds gather at the sea surface, while albatrosses and penguins are not seen in groups in the sea. For this reason, it is necessary to estimate the population of albatrosses and penguins from land rather than from the surface of the sea. For land imaging, the resolution of commercial satellite images is insufficient. Another problem is the difficult geographic conditions when taking images with a drone. On the other hand, population estimates can be made with deep learning techniques using camera images placed in specific regions.

The aim of this study is to develop a Deep Learning-based remote sensing software for penguin and albatross colony prediction. The specific contributions of this study are as follows: (1) A new dataset consisting of 4144 penguin and albatross images is collected to train deep learning models. (2) In contrast to similar studies, a segmentation-based annotation technique was used here. (3) By using various state-of-the-art object detection models, the best AP value of 73% in prediction success was achieved.

2 Material and Methods

The general processes of the project are shown in the diagram in Fig.1. According to this diagram, the first phase of the project is the data preprocessing process. The images of penguins and albatrosses were obtained from the Internet using open-source images and videos. The penguins and albatrosses in the images are individually labeled based on segmentation. In the next step, model trainings were conducted using innovative CNN models. As a result of all trainings, a performance evaluation of the model was performed. The model with the best performance was used in the developed graphical user interface

[1] https://penguin-counting-app.azurewebsites.net

(GUI) software.

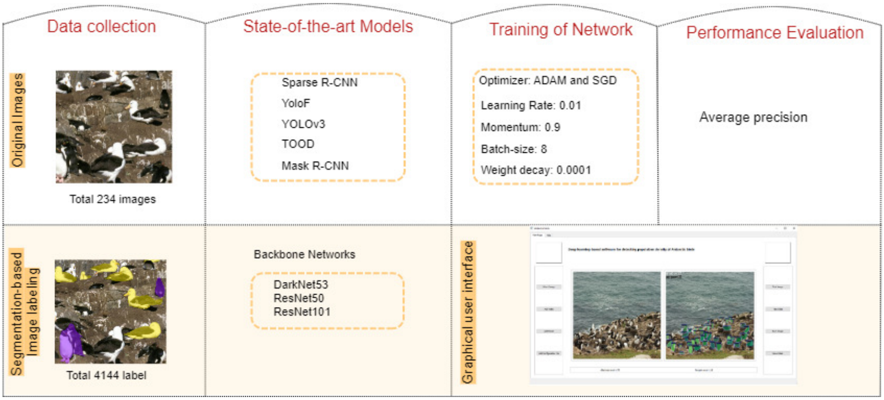

Figure 1. General operating diagram for experimental studies

2.1 Data collection and preparation

To create the dataset, images of penguins and albatrosses were obtained from the Internet via open-source images and videos. The VLC player program was used for the images obtained from the videos. With the help of this program, one image per 20 frames was obtained from each of the video images of albatross and penguin colonies. The common feature of the obtained images is that they are images taken from a high angle and not in the form of a drone or satellite image. This is because the software we developed is designed to recognize camera images taken from a specific height and angle. In addition, birds in colonies were preferred for all images. Penguin and albatross images in the dataset were labeled based on segmentation via the Supervisely platform[2].

An example of labeling is shown in Fig.2. It can be seen that the labeled penguins in the image are colored purple. On the Supervisely platform, segmentation by rectangles and polygons can be done using the tools on the left. Each record stores the coordinate information of the bird in that image. In this study, a total of 4.144 labels were

[2]https://supervise.ly

created. The fact that the labeling process is based on segmentation means that more time is required. It took the project team about 50 hours to label 4.144 images. The distribution of the number of images and labels in the dataset is shown in Table 1.

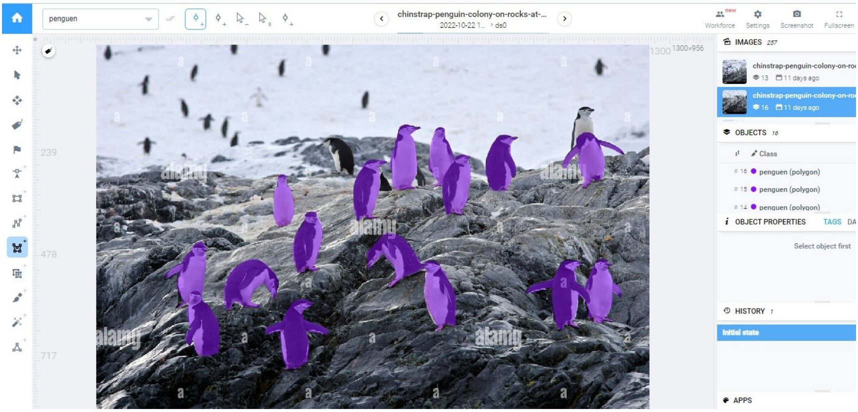

Figure 2. Segmentation-based labeling

Table 1. Numerical distribution of classes in the dataset

Classes	Image Count	Label Count	Size
Penguin	128	2072	
Albatross	106	2072	
Total	234	4144	55MB

2.2 Implementation details of CNN models

CNN architectures first appeared in 1998 with LeNET-5 [13], then came AlexNet in 2012 [14], and later several other CNN architectures were developed that provided successful solutions to artificial intelligence problems. With these architectures, numerous convolutional operations, pooling operations, and different types of activation functions were tried.

In this study, experiments were conducted using state-of-the-art CNN architectures to detect population densities of Antarctic birds.

The experiments in this study were conducted using the MMDetection[3], a state-of-the-art PyTorch-based modular object detection library released by the OpenMMLab project[4]. The toolbox directly supports state-of-the-arts deep learning frameworks. Sparse R-CNN, YOLOF, TOOD, YOLOv3, and Mask R-CNN frameworks were preferred in this study. For these frameworks, ResNet-50, ResNet-101, and DarkNet-53 architectures were chosen as backbones.

In this study, 90% of the enlarged data was randomly selected as the training set and the remaining 10% as the test set. Of these, 10% of the training set was selected as the validation set. All models were configured to use the base models, and the training hyperparameters were predefined and remained static for all experiments. For training the state-of-the-art models, the number of iterations was started with 100 and gradually increased. In the selection of parameters, the preferred values in some literature studies [15]–[18] were used. Accordingly, Adam and SGD were used as learning algorithms. The learning rate was initially set to 0.01 and the mini-batch size was set to 8. The weight decay and momentum coefficients were set to 0.0001 and 0.9, respectively.

The Supervisely platform provides the ability to run MMDetection architectures. The training and testing processes on the Supervisely platform are performed on a workstation with the configurations of an Intel Xeon CPU, 16GB Nvidia Quadro RTX5000 GPU with 16GB RAM.

2.3 Performance evaluation of CNN models

Object detection problems are challenging tasks that involve both classifying the objects on the image and determining the coordinates of the images. In these problems, the labeled regions are expressed as ground truth (G), and the regions detected by the trained model are expressed as default boxes (D). As can be seen in Fig.3, the ratio of the intersection of the G and D regions to the union is defined as the intersection over the union (IoU).

[3] https://github.com/open-mmlab/mmdetection

[4] https://github.com/open-mmlab

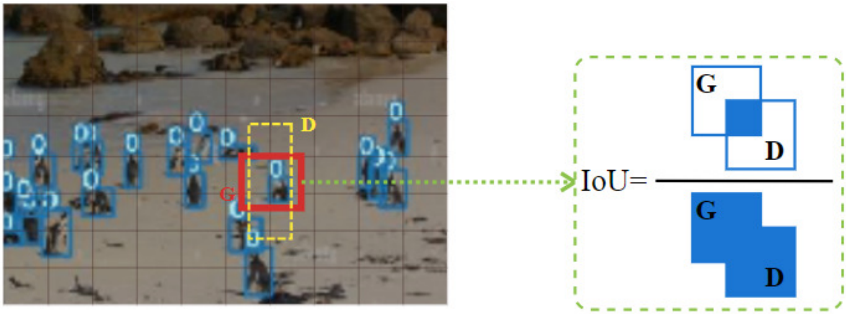

Figure 3. Ground-truth and default-box match

IoU takes values between 0 and 1, and $IoU = 1$ means that the boxes overlap [19]. The greater the overlap, the better the detection success. Average precision (AP) was used as a performance evaluation metric for the Sparse R-CNN, YOLOF, TOOD, YOLOv3, and Mask R-CNN architectures used in this study. AP given in Eq.(1) is a popular metric used for the performance of the Microsoft Common Objects in Context (COCO) dataset [20].

$$AP = \frac{1}{10}(AP^{50} + AP^{55} + AP^{60} + ... + AP^{95}). \tag{1}$$

For the performance evaluation of the system in this study, the AP was preferred, which is commonly used in the literature [21], [22] for object detection problems. Accordingly, the AP^{50} metric expresses the calculated AP values for $IoU > 0.5$. As stated in Section 3 of this study, the best results were obtained with the AP^{50}.

3 Results and Discussion

The performance results obtained as a result of training for the state-of-the-art CNN architectures used in this study are shown in Table 2. The experiments for five different models lasted approximately 48.5 hours. In this study, the ResNet50, ResNet101, and DarkNet-53 architectures were preferred as backbone networks. The main task of the backbone networks is to classify the objects detected in the image as albatross or penguin. As can be seen in Table 2, the best results were obtained with the AP^{50} metric. Accordingly, the TOOD_ResNet50 model was the best performing model with 73.0%. The lowest performance was achieved with the YOLOv3 architecture with 49.2%.

Table 2. Performance comparison of state-of-the-art models

Model	Backbone	$AP^{50:0.5:9}$	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
Sparse R-CNN	ResNet50	33.3	54.1	37.0	16.0	33.2	50.7
YOLOF	ResNet50	31.0	59.8	27.4	09.0	34.3	50.9
TOOD	ResNet101	47.4	68.7	55.2	23.7	53.0	63.9
TOOD	ResNet50	51.3	73.0	61.6	35.1	54.0	56.5
YOLOv3	DarkNet-53	22.8	49.2	17.1	2.8	24.5	32.6
Mask R-CNN	ResNet50	43.0	60.1	54.0	29.3	47.1	54.5

The training loss curves are shown in Figure 4, where the coordinates represent epoch and loss values.

As the number of training iterations increased continuously, the loss values of all models decreased abruptly at first and then slowly. The epoch number is limited to 500 in this study. Increasing the number of epochs will help to further reduce the oscillations in the loss curve. However, increasing the iteration number would significantly increase the duration of the experiments [23]. When considering the loss curves, the Sparse R-CNN and YOLOv3 models are the models with the least oscillations. The loss curve of TOOD-ResNet50, the best prediction model, gradually converged toward 0.162 after 250 iterations, while that of YOLOv3, the lowest prediction model, converged toward 306.75. The loss values for the YOLOF, Sparse R-CNN, and Mask R-CNN models decreased to 0.215, 5.25, and 0.217, respectively. It is shown that the TOOD-ResNet50 model has a lower loss value among all models and learns the attributes of penguin and albatross images effectively.

It can be seen that the loss is fixed at a certain value in all models except the YOLOF model. In the YOLOv3 model, the loss decreased to a certain level in the first iterations, but no significant decrease was observed in the next iterations.

In this study, a graphical user interface (GUI) was developed for the

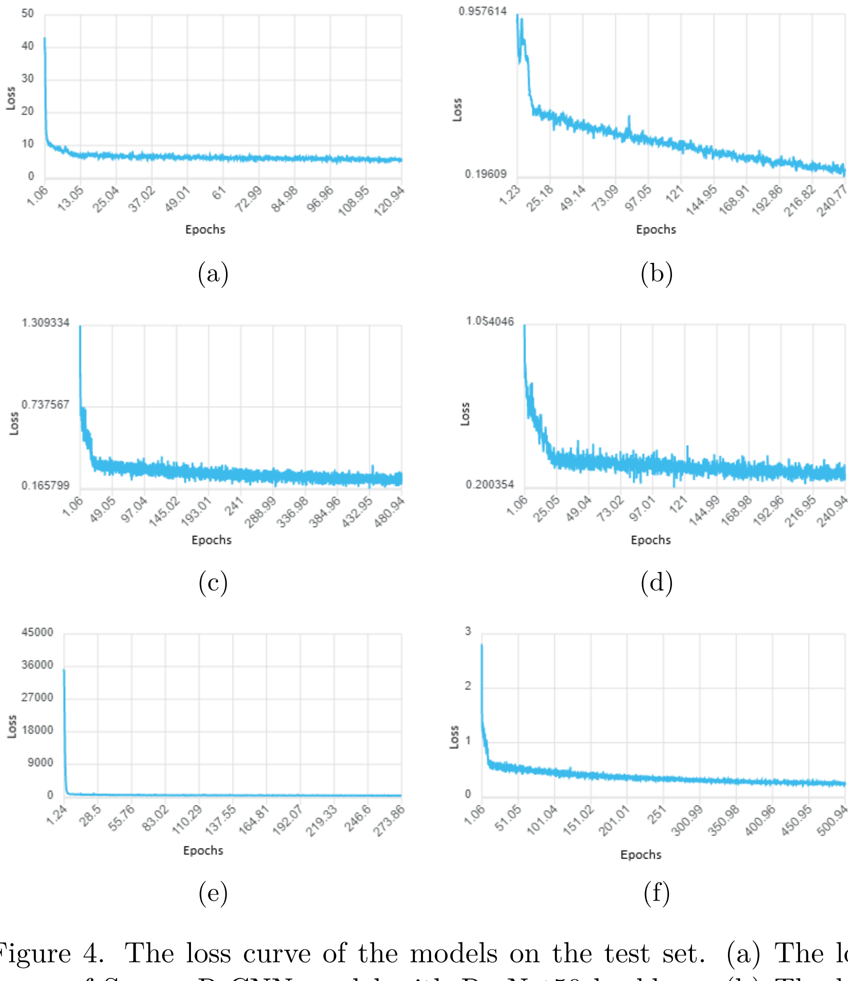

Figure 4. The loss curve of the models on the test set. (a) The loss curve of Sparse R-CNN model with ResNet50 backbone (b) The loss curve of YOLOF model with ResNet50 backbone (c) The loss curve of TOOD model with ResNet101 backbone (d) The loss curve of TOOD model with ResNet50 backbone (e) The loss curve of YOLOv3 model with DarkNet-53 backbone (f) The loss curve of Mask R-CNN model with ResNet50 backbone

state-of-the-art application using PyQt5, one of Python's libraries. In the GUI screen shown in Figure 5, the image is first selected from the file system using the "Select Image" button. The "Add Model" button allows the selection of the file with extension .pth, which contains the weights of the best state-of-the-art model. The "Add Configuration File" button is used to load the configuration file created by the Supervisely platform. The program also detects penguins and albatrosses on video. For this purpose, the "Add Video" button must be selected. After all the selection processes are completed, the results can be viewed in real time on the right side of the screen. The obtained results can be saved in the database.

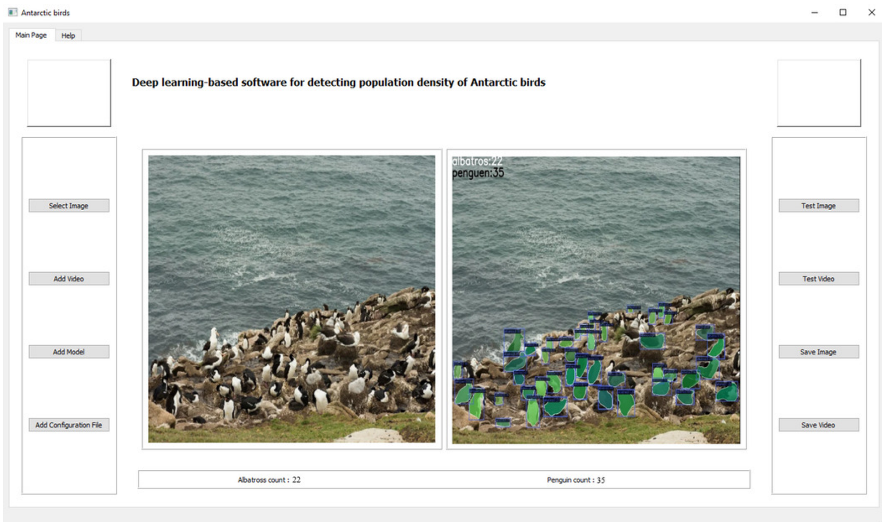

Figure 5. GUI developed for detection of Antarctic birds

The sample results obtained for the best model are shown in Figure 6.

Table 3 shows the comparison between this study and some other studies. Other than this study, there is only one study [12] that detects both penguins and albatrosses. This is one of the two studies using data from the Internet as the data source. It can be seen that the studies are divided into two areas: Image processing and CNN as the method

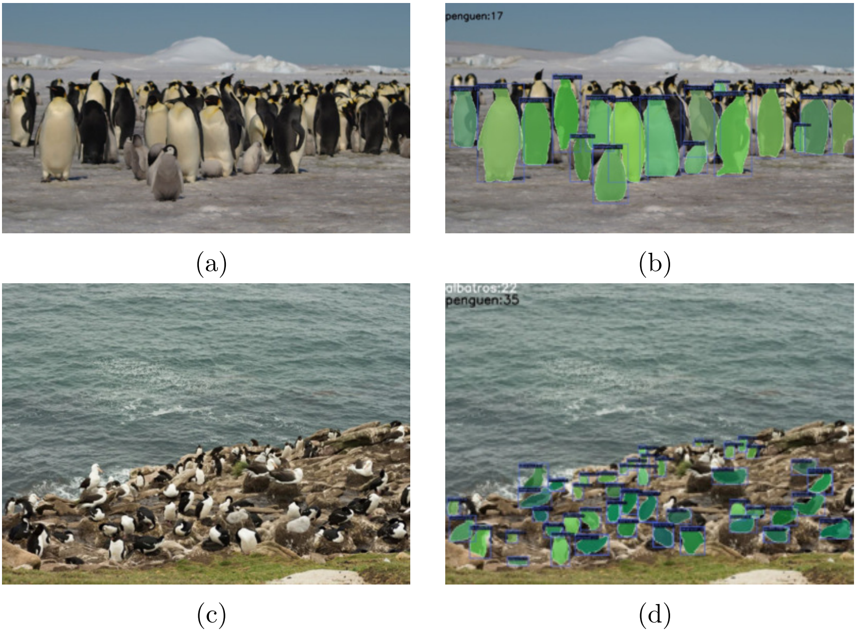

Figure 6. Results of the best model. (a) Original image of penguin colony (b) The result produced by the model (c) Original image with penguins and albatrosses (d) The result produced by the model

used. The five different CNN models used in this study have not been used in any previous study.

Apart from this study, there is only one study [11] that uses segmentation labeling. In CNN-based studies, [12] achieved a prediction success of 97.6% for black-browed albatrosses and 90% for southern rockhopper penguins using the RetinaNet architecture. [9], on the other hand, achieved a prediction success between 85% and 95.4% in his experiments. In our study, a prediction rate of up to 73% was achieved.

Detection of colony populations can be said to be a difficult problem for several reasons. One of these reasons is that birds in Antarctica stay close together to protect themselves from the cold. This makes the angle of the camera very important. For example, in the image in Figure 6a, the camera took an image from the horizontal position. For

Table 3. Results from previous studies on detection of birds in Antarctica

Ref.	Birds	Data Source	Method	Labeling	GUI
[4]	Penguin	Satellite	Image Pro.	-	No
[5]	Albatross	Satellite	Image Pro.	-	No
[6]	Penguin	Satellite	Image Pro.	-	No
[7]	Albatross	Satellite	U-NET	Rectangle	No
[9]	Land birds	Drone	R-CNN,SSD	Rectangle	No
[11]	Penguin	Internet	VGG-16	Point	Yes
[12]	Penguin Albatross	Drone	RetinaNet	Rectangle	No
This Paper	Penguin Albatross	Internet	TOOD YOLOv3 YOLOF Faster R-CNN Sparse R-CNN	Segmentation	Yes

this reason, some penguins left behind are not detected by the software. It is recommended that researchers wishing to conduct similar studies work with images taken from the top angle.

Complex background images are the type of problem that researchers find difficult in deep learning applications. Since the data was not collected under controlled conditions, background trees, rocks, etc. affect the prediction success. However, the results obtained in this study are encouraging for studies with other Antarctic birds such as skuas, cormorants, petrels, and arctic terns.

In this study, 4,144 labeling operations took approximately 50 hours. The fact that the labeling process is based on segmentation means that more time is spent. As the number of tags increases, the models are more successful, so datasets with more tags can be created. However, limited image resources on the Internet are a major problem. In particular, data collection with cameras in Antarctica will provide much more successful results.

4 Conclusion

The protection of Antarctic habitat is considered important everywhere in the world. However, reasons such as climate, geographic conditions, and an insufficient research budget require the use of new technological solutions in Antarctic research. In this study, Deep Learning-based software was developed for real-time detection of penguin and albatross colonies in Antarctica. The dataset obtained from internet sources was trained with five different convolutional neural network architectures: YOLOv3, YOLOF, Faster R-CNN, and Sparse R-CNN. In addition to the close proximity of albatrosses and penguins in the colony images, the complex background structure complicates the problem. However, the experimental results show that the TOOD-ResNet50 model with 0.73 AP^{50} adequately detects the birds compared to the other models.

References

[1] G. P. Rush, L. E. Clarke, M. Stone, and M. J. Wood, "Can drones count gulls? minimal disturbance and semiautomated image processing with an unmanned aerial vehicle for colony-nesting seabirds," *Ecology and evolution*, vol. 8, no. 24, pp. 12 322–12 334, 2018.

[2] B. A. Surveys, "Penguins," https://www.bas.ac.uk/about/antarctica/wildlife/penguins/, 2022, accessed: 2022-10-10.

[3] B. A. Surveys, "Albatross," https://www.bas.ac.uk/about/antarctica/wildlife/albatross/, 2022, accessed: 2022-10-10.

[4] P. T. Fretwell, M. A. LaRue, P. Morin, G. L. Kooyman, B. Wienecke, N. Ratcliffe, A. J. Fox, A. H. Fleming, C. Porter, and P. N. Trathan, "An emperor penguin population estimate: the first global, synoptic survey of a species from space," *PloS one*, vol. 7, no. 4, p. e33751, 2012.

[5] P. T. Fretwell, P. Scofield, and R. A. Phillips, "Using super-high resolution satellite imagery to census threatened albatrosses," *Ibis*, vol. 159, no. 3, pp. 481–490, 2017.

[6] M. R. Schwaller, H. J. Lynch, A. Tarroux, and B. Prehn, "A continent-wide search for antarctic petrel breeding sites with satellite remote sensing," *Remote Sensing of Environment*, vol. 210, pp. 444–451, 2018.

[7] E. Bowler, P. T. Fretwell, G. French, and M. Mackiewicz, "Using deep learning to count albatrosses from space: Assessing results in light of ground truth uncertainty," *Remote Sensing*, vol. 12, no. 12, p. 2026, 2020.

[8] L. Goddijn-Murphy, N. J. O'Hanlon, N. A. James, E. A. Masden, and A. L. Bond, "Earth observation data for seabirds and their habitats: an introduction," *Remote Sensing Applications: Society and Environment*, vol. 24, p. 100619, 2021.

[9] S.-J. Hong, Y. Han, S.-Y. Kim, A.-Y. Lee, and G. Kim, "Application of deep-learning methods to bird detection using unmanned aerial vehicle imagery," *Sensors*, vol. 19, no. 7, p. 1651, 2019.

[10] S. Uğuz and N. Uysal, "Classification of olive leaf diseases using deep convolutional neural networks," *Neural Computing and Applications*, vol. 33, no. 9, pp. 4133–4149, 2021.

[11] C. Arteta, V. Lempitsky, and A. Zisserman, "Counting in the wild," in *European conference on computer vision.* Springer, 2016, pp. 483–498.

[12] M. C. Hayes, P. C. Gray, G. Harris, W. C. Sedgwick, V. D. Crawford, N. Chazal, S. Crofts, and D. W. Johnston, "Drones and deep learning produce accurate and efficient monitoring of large-scale seabird colonies," *The Condor*, vol. 123, no. 3, p. duab022, 2021.

[13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[15] J. Peng, D. Wang, X. Liao, Q. Shao, Z. Sun, H. Yue, and H. Ye, "Wild animal survey using uas imagery and deep learning: modified faster r-cnn for kiang detection in tibetan plateau," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 169, pp. 364–376, 2020.

[16] M. Favorskaya and A. Pakhirka, "Animal species recognition in the wildlife based on muzzle and shape features using joint cnn," *Procedia Computer Science*, vol. 159, pp. 933–942, 2019.

[17] M. Fennell, C. Beirne, and A. C. Burton, "Use of object detection in camera trap image identification: Assessing a method to rapidly and accurately classify human and animal detections for research and application in recreation ecology," *Global Ecology and Conservation*, vol. 35, p. e02104, 2022.

[18] J. Brown, Y. Qiao, C. Clark, S. Lomax, K. Rafique, and S. Sukkarieh, "Automated aerial animal detection when spatial resolution conditions are varied," *Computers and Electronics in Agriculture*, vol. 193, p. 106689, 2022.

[19] W. Li, P. Chen, B. Wang, and C. Xie, "Automatic localization and count of agricultural crop pests based on an improved deep learning pipeline," *Scientific reports*, vol. 9, no. 1, pp. 1–11, 2019.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision.* Springer, 2014, pp. 740–755.

[21] K. Tong, Y. Wu, and F. Zhou, "Recent advances in small object detection based on deep learning: A review," *Image and Vision Computing*, vol. 97, p. 103910, 2020.

[22] Q.-J. Wang, S.-Y. Zhang, S.-F. Dong, G.-C. Zhang, J. Yang, R. Li, and H.-Q. Wang, "Pest24: A large-scale very small object data set of agricultural pests for multi-target detection," *Computers and Electronics in Agriculture*, vol. 175, p. 105585, 2020.

[23] S. Uğuz, G. Şikaroğlu, and A. Yağız, "Disease detection and physical disorders classification for citrus fruit images using convolutional neural network," *Journal of Food Measurement and Characterization*, pp. 1–10, 2022.

Sinan UĞUZ

Received February 3, 2023
Revised April 14, 2023
Accepted April 25, 2023

Sinan UĞUZ
ORCID: https://orcid.org/0000-0003-4397-6196
Department of Computer Engineering
Faculty of Technology
Isparta University of Applied Sciences
E14 Block (3. Floor) West Campus 32260 Isparta/TURKEY
E–mail: sinanuguz@isparta.edu.tr