International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhanced Animal Detection in Complex Outdoor Environments Using Modified Yolo V7

Johnwesily Chappidi, Divya Meena Sundaram

Submitted: 11/01/2024 **Revised**: 17/02/2024 **Accepted**: 25/02/2024

Abstract: Detecting animals accurately and quickly in complicated outdoor settings is important for getting work done efficiently. But it's not easy because the places where animals live have complicated environmental conditions. This research proposes a novel animal detection method that uses the YOLO V7 network to overcome these challenges. Thorough evaluations and comparisons are performed on various detection networks like YOLO V3-spp, YOLO V5s, Faster R-CNN, and YOLO V7, which are meticulously conducted. The rigorous assessments identify YOLO V7 as the preeminent performer. The findings are noteworthy, as the model exhibits exemplary detection capabilities and robust adaptability in complex field environments. It attains a noteworthy mean Average Precision (mAP) of 96.03%, accompanied by impressive precision, recall, F1 score, and an average detection time of 94.76%, 95.54%, 95.15%, and 0.025 seconds per image, respectively. This study underscores the profound efficacy of uniting YOLO V7 for animal detection within challenging field conditions.

Keywords: *Anima detection, YOLO v7, Challenging image conditions, R-CNN*

1. Introduction

In ecology and wildlife conservation research areas, detecting wild animals is crucial. Identifying animal species and tracking their movements in their natural habitats becomes essential for the purpose of monitoring biodiversity and gaining insights into the dynamics of ecosystems. However, manual detection in complex natural habitats poses significant challenges and requires extensive resources [1]. To address this issue and harness the potential of wild animal monitoring, there is a need to develop rapid and accurate detection methods for animals in their natural environments. Automatic and mechanised approaches could significantly improve efficiency, reduce equipment and manpower costs, and promote advancements in wildlife research and conservation efforts [2]. One key technology in achieving automated wild animal detection is the swift and precise localisation of animals in challenging field environments. Hence, developing efficient methods for detecting and identifying wild animals in diverse landscapes becomes paramount [3]. Such advancements could lead to significant benefits, including better wildlife management, improved understanding of animal behaviour, and enhanced conservation strategies [4].

Current methods in detecting wild animals mostly depend on sophisticated imaging technology [5–7]. Conventional techniques for animal detection primarily rely on characteristics like colour and shape to distinguish animals from different objects [8]. However, these intricate algorithms with set boundaries have inherent drawbacks, resulting in mistakes when detecting animals in complex field environments and lacking the necessary reliability [9]. To address these issues, deep learning algorithms have emerged as a revolutionary advancement in wild animal detection. They efficiently extract important features of the target animals in challenging environments, surpassing the limitations of traditional methods [10]. One standout deep learning algorithm, the Convolutional Neural Network (CNN), has garnered widespread adoption due to its ability to perform convolution computations and its deep architecture. As a representative technique in the domain of deep learning, CNN has demonstrated its effectiveness in animal classification techniques [11], localisation techniques [12], detection techniques [13], and segmentation techniques [14]. Several convolutional neural network (CNN) detection techniques, including YOLO v3 [15], YOLO V5 [16,17], and Faster R-CNN [18], have been employed to identify wild animals in their natural habitats. Consequently, in this research, we will utilize image detection technology and harness the capabilities of CNN to aid in detecting wildlife in their native habitats.

YOLO, a widely employed single-stage target detection algorithm, is recognized for its high accuracy and speed [19, 20]. This approach has demonstrated its effectiveness in identifying small and partially hidden targets within intricate field settings, surpassing numerous other deep learning algorithms in terms of speed [21,22]. The most recent addition to the YOLO series is YOLO V7, which introduces trainable enhancements to improve real-time detection accuracy without adding extra computational

School of Computer Science and Engineering, VIT-AP University, Amaravathi
divyameena.s@vitap.ac.in

load. It also integrates strategies like extension and compound scaling to reduce parameters and computations effectively, resulting in a substantial increase in detection speed [23]. YOLO V7 has not yet been utilised for wild animal detection despite its potential. Hence, in this study, we explore the application of YOLO V7 as an innovative detector to detect and identify wild animals in their natural habitats efficiently.

Wild animal detection in natural environments presents a complex and challenging task, where the images of animals might be affected by numerous factors such as different lighting conditions, partial or heavy occlusion, and diverse background interferences, leading to potential false detections or missed targets [25]. To ensure the effectiveness of our detection model, it is essential to incorporate a diverse range of scenes in the training data, enabling the extraction of robust features and overcoming complexities present in the field [26]. However, the limited number of animal image data captured in the environment due to wildlife habitat limitations and the time spent collecting it poses a challenge for deep learning systems.

2. Yolo V7 Network Architecture

A notable upgrade to the YOLO architecture is YOLO V7, which stands out as a powerful object detection network tailored for wild animal detection. It boasts exceptional attributes such as rapid detection speed, high precision, and ease of training and deployment. The network's accuracy, as well as speed ranging from 5 to 160 FPS, surpasses the performance that currently exists in object detectors. Specifically, in the same volume, YOLO V7 exhibits a remarkable 120% faster speed than YOLO V5 (FPS). Moreover, the experimental results of this detector on the MS COCO dataset surpass those of the YOLO V5 detector, giving it an excellent option for applications involving the detection of wild animals. [30]. Figure 1 illustrates the well-designed network structure of YOLO V7, showcasing its advanced architecture and the intricate features that contribute to its exceptional performance in detecting and localizing wild animals with great efficiency and accuracy.

Regarding its detection of wildlife in nature, the YOLO V7 network is organized into three essential elements. It starts with the input network; the next step is the backbone and head networks. The YOLO V7 network initiates the detection process by preprocessing the animal image and resizing it to a predetermined size of 640 × 640 × 3. This resized image is then inputted into the backbone network.

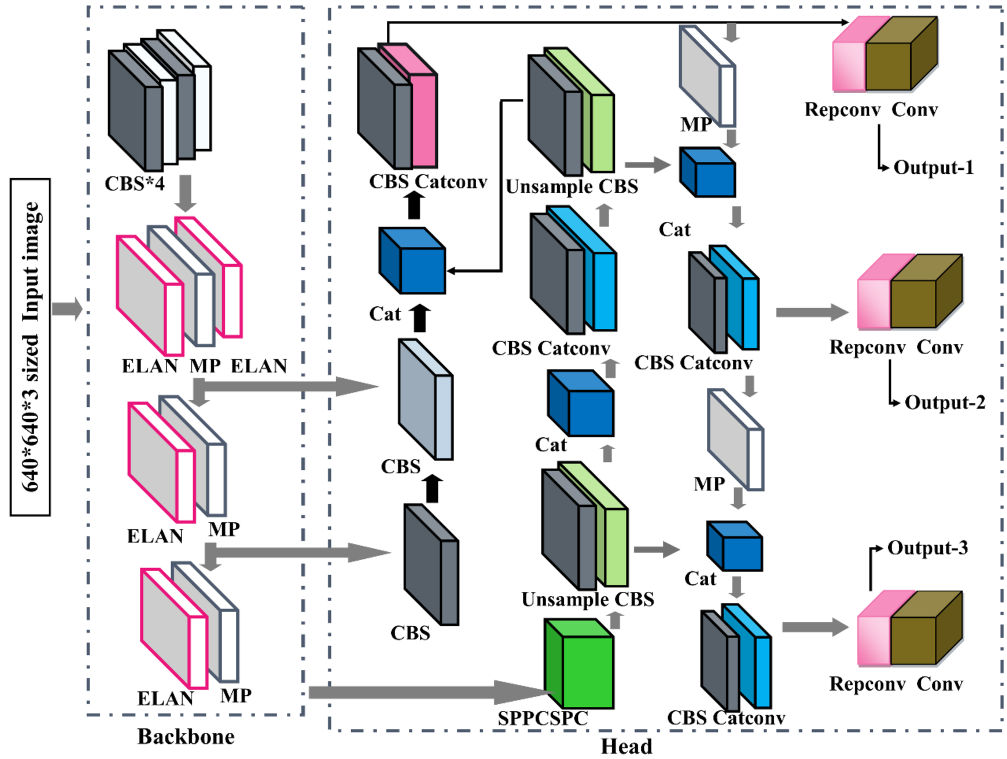

Fig 1. Architecture of YOLO V7 network

The YOLO V7 network incorporates the CBS composite module, ELAN stands for efficient layer aggregation networks module [31], and the MP module sequentially reduces the feature map's dimensions by half. At the same time, the number of output channels is doubled compared to the number of input channels.

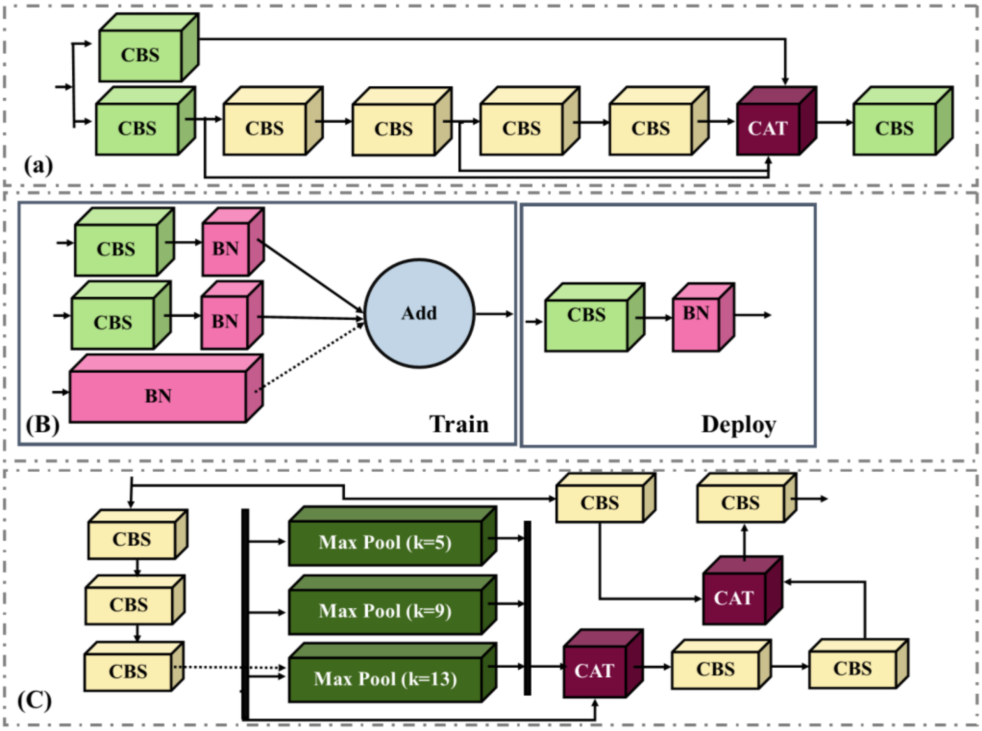

Fig 2. The framework of each module. (a) ELAN; (b) Repconv; and (c) SPPCSPC

The CBS composite module applies convolution, batch normalization (BN), and an activation function to the input feature map, as illustrated in Figure 2. Like YOLO V5, Silu is utilised as the activation function in YOLO V7 for its effectiveness in enhancing the model's detection capabilities. By integrating these essential components, the YOLO V7 network becomes adept at detecting and localizing wild animals in the images, achieving remarkable accuracy and speed in the detection process. This well-structured network empowers wildlife researchers and conservationists with an efficient and effective tool for monitoring and studying animals in their natural habitats.

A novel module known as ELAN has been developed to detect wild species. To improve the model's accuracy, this module uses a combination of shuffle, expand, and merge cardinality strategies to keep the original gradient route and continuously increase the network's learning potential. Several convolutions are used to construct the ELAN structure, with group convolution employed to increase the channel and cardinality of the computational units while preserving the original architecture's channel count. Consequently, the quantity of channels emitted by the ELAN module is inversely proportional to the input. Furthermore, the MP (Max-pooling) module contributes substantially to the operation of the network. The upper branch of the MP module applies max pooling to reduce the length and width of the feature map by half, and convolution is utilised to decrease the number of channels by half. Conversely, the lower branch employs the initial convolution to decrease the number of channels, followed by a second convolution of half the length and breadth of the feature map, utilising a kernel size of three and a stride of two. In conclusion, the outputs generated by the upper and lower branches are aggregated to produce an output feature map proportionally sized in length and breadth, with an equivalent quantity of input and output channels. The accuracy and learning capabilities of the untamed animal detection model are enhanced through the integration of ELAN and MP modules. It detects and localises animals in their natural habitats with ease. The adaptability of the network to diverse wildlife characteristics and complex environmental conditions improves the efficacy and precision of wildlife monitoring and conservation efforts.

Expanding upon the three-layer output of the backbone network, the head network of the YOLO V7 model generates three additional layers of feature maps featuring varying proportions. These feature maps are then processed by the Repconv module in order to modify the ultimate number of output channels. Following this, the final results are generated by employing three layers of convolution operations with a kernel size of 1 (1×1) for the purposes of objectness, class, and box prediction. This facilitates image detection. The head network is composed of several modules, which are as follows: the SPPCSPC module, a sequence of CBS modules, the MP module, the Cat conv module, and three Rep conv modules that follow. Comparable to the SPPF (Spatial Pyramid Pooling) implemented in YOLO V5, the SPPCSPC module aids in expanding the network's receptive field. The module processes the input feature map with three convolution operations, improving network capability for object detection and feature extraction from the images. The feature map has dimensions of 512 × 20 × 20.

Two phases comprised the development of the animal object detection model: training and testing. The training set was utilized during the training phase to train the YOLO V7 neural network. After acquiring the model weights, assessment indicators were computed on the validation set in order to evaluate the performance of the model. As preliminary animal detection model, the model exhibiting the highest-performing weights was chosen. During the testing phase, the prediction results of the detection model's application to the test set were assessed in order to determine whether the model could be generalized for future use in animal detection systems, such as selecting machines. The model establishment process's workflow is illustrated in Figure 3. The neural network's ultimate result comprised the bounding outlines of the identified animal objects, accompanied by the probability (confidence) that signified the category of the object.

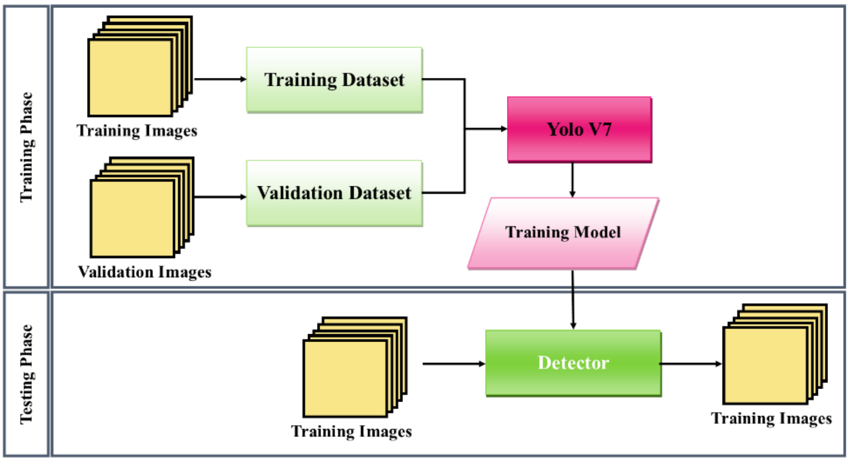

Fig 3. Workflow of the proposed study

The CIOU loss function was used to objectively quantify the error between the predicted bounding boxes and the ground truth calibration boxes in evaluating the animal detection model [34,35]. The information required to determine the CIOU is contained in the parameters: calibration box A, prediction box B, the distance (L1) between the centres of the two boxes, and the diagonal length (L2) of the smallest rectangle that can contain both boxes. Here is how CIoU was calculated:

$$loss_{CIoU} = 1 - IOU + \frac{l_1^2}{l_2^2} + \propto v \quad \textbf{(1)}$$

From the above equation, $\propto$ is the balancing factor that determines the contribution of both v and IoU to the loss, while v stands for the aspect ratio similarity between the calibration box and the prediction box.

3. Experimental Framework

3.1 Platform for training and setting parameters.

The training and testing were performed on a computer with the Windows 11 operating system specifically for animal detection. The hardware configuration comprises a 12th Generation Intel(R) Core (TM) i7-12700H CPU processor operating at a frequency of 2.30 GHz, accompanied by 16.0 GB of RAM. Additionally, it features an NVIDIA GeForce GTX 3060Ti graphics card with 8 GB of video memory, specifically selected to fulfil the GPU computing prerequisites. Python 3.8 was the programming language, with PyTorch as the underlying deep-learning framework. The software tools employed included CUDA 11.3, CUDNN 8.2, OpenCV 3.4.5, and Visual Studio 2017.

Transfer learning methods were employed to train the animal detection model in this study, which is based on YOLO V7. There were 300 epochs in the training process, and each batch size was 8. It was set to 640 x 640 for the entry size. Regularisation was used every time the model's weight was changed through the BN layer. The rate at which the weight fell was set to 0.0005, and the momentum factor was 0.937. The first value of the vector was set to 0.01. Augmentation values of 0.015 were set for hue (H), 0.7 for saturation (S), and 0.4 for lightness (V). The Tensor-board visualisation tool recorded data and tracked the loss during training. At the end of each epoch, the model weights were saved.

3.2 Dataset description

Two distinct types of datasets are used in this study. The "iNat Challenge 2019" sample was picked to see how well the YOLO algorithm works. This dataset has 1,010 species and a training and validation set of 268,243 images from iNaturalist that were gathered and checked by various users. The "Animal 10N" Dataset has 55,000 pictures of five pairs of animals: a cat and a lynx, a tiger and a cheetah, a wolf and a coyote, a gorilla and an ape, and a hamster and a guinea pig. There are 50,000 pictures in the training set and only 5,000 images in the test set.

3.3 Performance metrics

Several evaluation metrics were used in this study to correctly and objectively rate how well the animal detection model worked. Precision, F1 scores, recall, and mAP, were used for the comprehensive evaluation. Precision is an evaluation tool that shows how many correctly identified targets are compared to the total number of targets found. More precision means better success in detection.

$$P = \frac{TP}{TP+FP} \times 100\% \quad \textbf{(2)}$$

A recall is a performance metric that checks how well a model can find all the important objects of the class in a dataset. It is calculated by dividing the number of correct guesses by the total number of correct predictions and false negatives:

$$R = \frac{TP}{TP+FP} \times 100\% \quad \textbf{(3)}$$

The area under the precision-recall curve for a certain class is called the Average Precision (AP). This number shows the average accuracy found by calculating different memory levels for that class.

$$AP = \int_0^1 P(r)dr \quad \textbf{(4)}$$

Mean Average Precision (mAP) is a metric that measures the average precision across different levels of recall. mAP utilised to evaluate the accuracy of YOLOv7 detection models. it achieved a mAP of 95.74%, indicating a high average precision in detecting objects across various scenarios in the dataset.

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_I \quad \textbf{(5)}$$

Although precision may not yield a comprehensive evaluation, mAP, Recall, and F1 scores were implemented to provide a more thorough assessment.

The F1 Score is calculated as the harmonic mean of Precision and Recall, offering a balanced measure between the two. A higher F1 Score signifies an improved equilibrium between Precision and Recall. The YOLOv7 model got an F1 Score of 93.67%, signifying commendable overall performance in terms of both Precision and Recall.

$$F1 = 2 \times \frac{P \times R}{P+R} \quad \textbf{(6)}$$

Here, TP (True Positive) is the number of animals that have been correctly recognised., FP (is the number of other objects wrongly identified as animals., and FN (False Negative) represents the number of undetected/missed animal objects. These metrics assess the model's ability to recognise animals thoroughly.

4. Results and Discussion

The YOLO V7 model for animal detection was trained using a wild animal dataset curated for this purpose. The training process involved training the YOLO V7 network on this dataset.

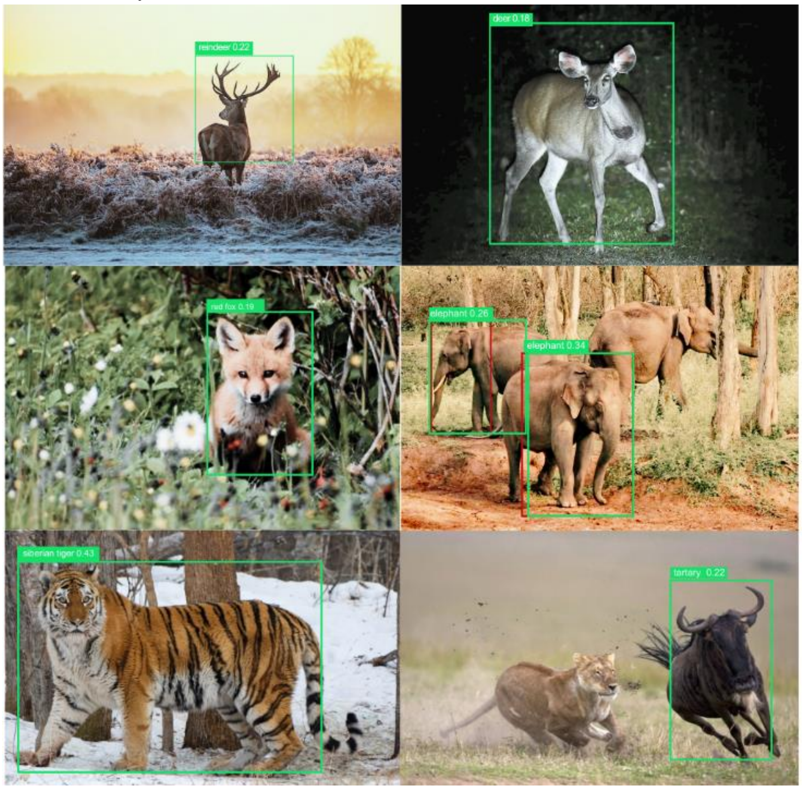

Fig 4. Detected animal images.

Several well-known models, including Faster RCNN, YOLO V5s, and YOLO V3-spp, were tested against the YOLO V7 model to see how well it detected animals. These models were trained on the same COCO pre-training weights and utilised the original animal detection dataset. Table 1 compares the models using several evaluation criteria, such as mAP, Precision, Recall, F1 score, and detection speed. Compared to YOLO V5s, YOLO V3-spp, and Faster RCNN, the YOLO V7 model improved 0.4%, 4.5 %, and 34.86.6 %, respectively, in Precision. Similarly, the mAP of the YOLO V7 model showed enhancements of 0.98%, 11.44%, and 4.24% over YOLO V5s, YOLO V3-spp, and Faster RCNN models, respectively. Regarding Recall, the YOLO V7 model demonstrated 3.95% and 5.63% improvements over YOLO V5s and YOLO V3-spp, respectively. The Recall of Faster RCNN was slightly higher than that of YOLO V7, but the other evaluation metrics were significantly lower. The YOLO V7 model achieved the highest F1 score of 93.67%, indicating a strong balance between Precision and Recall. Furthermore, the YOLO models exhibited fast detection times, with the YOLO V7 model performing exceptionally well at only 0.025 seconds per image. In contrast, the Faster RCNN model, being a two-stage detection model, had an average detection time of 5.167 seconds per image, considerably slower than the YOLO models. In summary, the YOLO V7 model for animal detection exhibited superior accuracy and efficiency compared to YOLO V5s, YOLO V3-spp, and Faster RCNN models. It achieved higher Precision, improved mAP, increased Recall, and a superior F1 score, while maintaining fast detection speeds, making it a reliable and efficient choice for animal detection tasks.

Table 1. Performance Comparison Among Different Models

Models for Detecting Targets	**mAP in %**	**Precision in %**	**Recall (%)**	**Fl_Score (%)**	**Detection Speed on Average (s/ image)**
Faster RCNN	91.50	59.35	93.59	73.00	5.167
YOLO V5s	84.30	89.70	87.50	86.90	0.072
YOLO v3-spp	94.76	93.81	89.18	91.44	0.054
YOLO v7 (Ours)	**95.74**	**94.21**	**93.13**	**93.67**	**0.025**

Using the test set to identify animals allowed us to compare various models' capacity for generalization. Table 2 displays the results of the comparison of performance. When compared to the YOLO V5s, YOLO V3-spp, and Faster RCNN models, the YOLO V7 model performed better and was able to detect more items. In comparison to the other models, the YOLO V7 model also showed a lower rate of missing and erroneous detections. Figure 9 displays the detection results of each model on the test set, which consists of various difficult situations. The blue circles in the pictures stand for false positives, and the green ones stand for false negatives that were missed. By looking at the examples in Figure 9, it is clear that YOLO V7 can identify animals in difficult lighting conditions or with little obstruction. In scenarios with backlighting or high occlusion, the YOLO V7 model demonstrated fewer inaccurate and missing detections than YOLO V5s, YOLO V3-spp, and Faster RCNN models.

Table 2. Detection results of testing the "iNat Challenge 2019" Dataset

No. of Objects	**No. of Actual Objects**	**Models for Detecting Targets (Accuracy %)**			
		Faster RCNN	**YOLOV3-spp**	**YOLOV5**	**YOLOV7**
No.of detected Animals	268,243	68%	77%	91%	**97.6%**
No. of correct detected animals	1,010	72%	78.4%	93%	**98%**
No. of wrong animals	0	16%	8%	6%	**0.7%**
No. of missed Animals	0	9.8%	6.3%	4%	**0**

Compared to the YOLO V5s, YOLO V3-spp, and Faster RCNN models, the YOLO V7 model demonstrated superior performance and resilience when detecting animals in difficult situations. In situations with occlusion or backlighting, it could detect more objects with fewer false positives and missed detections.

Table 3. Detection results of testing the " Animal 10N " dataset

No.Of Objects	**No.of Actual Objects**	**Models for Detecting Targets (Accuracy %)**			
		Faster RCNN	**YOLOV3-SPP**	**YOLOV5**	**YOLOV7**
No. of Animals detected	55,000	62%	73%	86%	**98%**
No. of correct detected animals	5,000	58%	67%	82%	**96%**
No. of wrong animals	65	26%	12%	8%	**1.3%**
No. of missed Animals	0	12%	7.6%	3%	**0**

5. Conclusions

The YOLO V7 target detection network and several data augmentation techniques created a real-time and precise animal detection approach. The goal was to find animals in intricate field sceneries. Using a dataset of animal picture data, a detection model originally developed on the YOLO V7 network was reserved for animal detection. This model performed better at detecting targets than Faster R-CNN, YOLO V3-spp, and YOLO V5s. Averaging only 0.025 seconds for detection, the YOLO V7 model outperformed the competition with a remarkable mAP of 95.74%, F1 score of 93.67%, Precision of 94.21%, and recall of 93.13%. The dataset was enhanced in various ways to enhance the model's recognition capability. These included mirroring, rotating, adding Gaussian noise, adjusting picture brightness, and mosaic augmentation. A more effective detection method, DA-YOLO V7, was trained using the expanded dataset.

With an F1 score of 95.15%, mAP of 96.03%, Precision of 94.76%, and recall of 95.54%, the ideal animal detection model was DA-YOLO V7. Combining the YOLO V7 target detection network with various data augmentation approaches might efficiently and accurately detect animals in complicated scenes. Tasks like mechanical harvesting activities could benefit greatly from this method's implementation. Plans are afoot to incorporate an end-effector system into the suggested model for future work. This will allow animal detection, positioning, picking angle modifications, and end-effector position adjustments. Also, this paper is a great resource for those interested in the theory behind animal identification.

References

[1] Lahoz-Monfort, J. J., & Magrath, M. J. (2021). A comprehensive overview of technologies for species and habitat monitoring and conservation. *BioScience*, *71*(10), 1038-1062.

[2] Petso, T., Jamisola Jr, R. S., & Mpoeleng, D. (2022). Review on methods used for wildlife species and individual identification. *European Journal of Wildlife Research*, *68*(1), 3.

[3] Peng, J., Wang, D., Liao, X., Shao, Q., Sun, Z., Yue, H., & Ye, H. (2020). 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, 169, 364-376.

[4] Schindler, F., & Steinhage, V. (2021). Identification of animals and recognition of their actions in wildlife videos using deep learning techniques. Ecological Informatics, 61, 101215.

[5] Saxena, A., Gupta, D. K., & Singh, S. (2021). An animal detection and collision avoidance system using deep learning. In *Advances in Communication and Computational Technology: Select Proceedings of ICACCT 2019* (pp. 1069-1084). Springer Singapore.

[6] Carl, C., Schönfeld, F., Profft, I., Klamm, A., & Landgraf, D. (2020). Automated detection of European wild mammal species in camera trap images with an existing and pre-trained computer vision model. European journal of wildlife research, 66, 1-7.

[7] Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Newton, E. J., Boughton, R. K., Ivan, J. S., & Miller, R. S. (2020). Improving the accessibility and transferability of machine learning algorithms for

identification of animals in camera trap images: MLWIC2. *Ecology and evolution*, *10*(19), 10374-10383.

[8] Sabeenian, R., Deivanai, N., & Mythili, B. (2020). Wild animals intrusion detection using deep learning techniques. Int. J. Pharm. Res, 12(4), 1053-1058.

[9] Sayagavi, A. V., Sudarshan, T. S. B., & Ravoor, P. C. (2021). Deep learning methods for animal recognition and tracking to detect intrusions. In *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2020, Volume 2* (pp. 617-626). Springer Singapore.

[10] da Silva Santos, A., de Medeiros, V. W. C., & Gonçalves, G. E. (2023). Monitoring and classification of cattle behavior: A survey. *Smart Agricultural Technology*, *3*, 100091.

[11] Xu, W., Zhang, X., Yao, L., Xue, W., & Wei, B. (2020). A multi-view CNN-based acoustic classification system for automatic animal species identification. *Ad Hoc Networks*, *102*, 102115.

[12] Su, X., Ullah, I., Liu, X., & Choi, D. (2020). A review of underwater localization techniques, algorithms, and challenges. *Journal of Sensors*, *2020*.

[13] Petso, T., Jamisola Jr, R. S., & Mpoeleng, D. (2022). Review on methods used for wildlife species and individual identification. *European Journal of Wildlife Research*, *68*(1), 3.

[14] Li, L., Dong, B., Rigall, E., Zhou, T., Dong, J., & Chen, G. (2021). Marine animal segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(4), 2303-2314.

[15] Manasa, K., Paschyanti, D. V., Vanama, G., Vikas, S. S., Kommineni, M., & Roshini, A. (2021, July). Wildlife surveillance using deep learning with YOLOv3 model. In *2021 6th International Conference on Communication and Electronics Systems (ICCES)* (pp. 1798-1804). IEEE.

[16] Ma, D., & Yang, J. (2022, October). Yolo-animal: An efficient wildlife detection network based on improved yolov5. In *2022 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML)* (pp. 464-468). IEEE.

[17] Shanthakumari, R., Nalini, C., Vinothkumar, S., Govindaraj, B., Dharani, S., & Chindhana, S. (2022, March). Image detection and recognition of different species of animals using deep learning. In *2022 International Mobile and Embedded Technology Conference (MECON)* (pp. 236-241). IEEE.

[18] Altobel, M. Z., & Sah, M. (2021). Tiger Detection Using Faster R-CNN for Wildlife Conservation. In *14th International Conference on Theory and Application of Fuzzy Systems and Soft Computing–ICAFS-2020 14* (pp. 572-579). Springer International Publishing.

[19] Yang, W., Liu, T., Jiang, P., Qi, A., Deng, L., Liu, Z., & He, Y. (2023). A Forest Wildlife Detection Algorithm Based on Improved YOLOv5s. *Animals*, *13*(19), 3134.

[20] Wang, F., He, P., Zhang, T., & Liang, D. (2022, October). Wildlife detection algorithm based on Inv-YOLOv5m. In *2022 Global Reliability and Prognostics and Health Management (PHM-Yantai)* (pp. 1-6). IEEE.

[21] XiaoAn, B., LinQing, Z., XiaoMei, T., Biao, W., QingQi, Z., YuTing, J., & Na, Z. (2023). Wildlife target detection based on improved YOLOX-s network.

[22] Dai, W., Wang, H., Song, Y., & Xin, Y. (2021, May). Wildlife small object detection based on enhanced network in ecological surveillance. In *2021 33rd Chinese Control and Decision Conference (CCDC)* (pp. 1164-1169). IEEE.

[23] Roy, A. M., Bhaduri, J., Kumar, T., & Raj, K. (2023). WilDect-YOLO: An efficient and robust computer vision-based accurate object localization model for automated endangered wildlife detection. *Ecological Informatics*, *75*, 101919.

[24] Panicker, S. A., Kumar, R. V., Ramachandran, A., & Padmavathi, S. (2021). Analysis of Image Processing Techniques to Segment the Target Animal in Non-uniformly Illuminated and Occluded Images. In *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020* (pp. 15-26). Springer Singapore.

[25] Francisco, F. A., Nührenberg, P., & Jordan, A. (2020). High-resolution, non-invasive animal tracking and reconstruction of local environment in aquatic ecosystems. *Movement ecology*, *8*(1), 1-12.

[26] Nguyen, H., Maclagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., ... & Phung, D. (2017, October). Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In *2017 IEEE international conference on data science and advanced Analytics (DSAA)* (pp. 40-49). IEEE.

[27] Kellenberger, B., Marcos, D., Lobry, S., & Tuia, D. (2019). Half a percent of labels is enough: Efficient animal detection in UAV imagery using deep CNNs and active learning. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(12), 9524-9533.

[28] Dulal, R., Zheng, L., Kabir, M. A., McGrath, S., Medway, J., Swain, D., & Swain, W. (2022, November). Automatic Cattle Identification using YOLOv5 and Mosaic Augmentation: A Comparative Analysis. In *2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA)* (pp. 1-8). IEEE.