Pest Birds Detection Approach in Rice Crops Using Pre-trained YOLOv4 Model

Ismael Diakhaby, Mouhamadou Lamine Ba, and Amadou Dahirou Gueye(✉)

Université Alioune Diop de Bambey, Bambey, Senegal
{Ismael.diakhaby,mouhamadoulamine.ba,dahirou.gueye}@uadb.edu.sn

Abstract. In Senegal, farmers in general and rice growers particularly are still facing many issues such as climatic hazards and water-scare environments in their daily life. A very acute and challenging problem for rice crops remains, however, their destruction by pest birds. These latter attack the rice crops when they are mature, leaving the farmers in disarray and without solution. Indeed, such an attack results in a drastic reduction in yields during harvest. Over time, many repellent techniques like scarecrow have been used, but show their limitations. In this paper, we tackle this problem and propose a pest birds detection approach in rice crops using pre-trained YOLOv4 detector and transfer learning. To show the efficiency of our model we conduct experiments on a real bird dataset, exhibiting a mean average precision of 96%.

Keywords: Object detection · Bird · Deep learning · YOLOv4 · Rice crops · Performance evaluation

1 Introduction

Senegalese Agriculture plays an important role in the country's economic development. It contributes about 14% to the Gross Domestic Product (17% for the primary Agricultural sector), employs 60% of the active population and concerns a surface area of more than 2, 5 million out of a potential of 3, 8 million of available arable lands. The irrigable area is estimated to 275, 000 hectares, of which 105, 000 hectares are developed and more than 75, 000 hectares are actually cultivated.

Cereals represent the most predominant crops in Senegal with 2,541,470 tons in 2018: in details we have corn, millet, paddy rices and sorghum with yields of 417,259 tons, 891,069 tons, 1,007,277 tons, and 225,865 tons respectively. Rice yield has increased by a factor of 2,3 between 2010 and 2019. One has to observe that rice is the most used cereals in Senegal. The growth of the production in rice and groundnut noted between 2012 and 2019 is due to the prioritization of these crops given their strategic role in the country's economy and food security Unfortunately, despite all the efforts made by the Senegal government, natural disasters are still hindering the development of the agricultural sector. Some of

A. D. Mambo et al. (Eds.): InterSol 2022, LNICST 449, pp. 223–234, 2022.
https://doi.org/10.1007/978-3-031-23116-2_19

these disasters have a relationship with climate change. For instance, climate change has resulted in a specific scourge that is not negligible, that is *invasion of rice crops by pest birds*. These hamfull birds arrive at a time when the rice is reaching maturity and eat the seeds on a one hand and drop lot of them on another hand. This heavily reduces expected yields during the harvest of the crops. Indeed the amount of lost yields is estimated to one thousands hectares out of fifty thousands hectares planted in 2020.

To fight against the pest birds, local residents have no effective monitoring system so far. Indeed, they are using traditional repellent methods which have quickly shown their limits. For instance, *scarecrows*, *clap empty bottles or waves of cassette wires* are highly harnessed to scare away birds. For a more productive and efficient Agriculture it is necessary to provide innovations based on new technological advances in particular artificial intelligence, sensors and IoT as sketched in [1,10,15]. Indeed, over the past years, general purpose or domain specific object detection models built on deep learning have received much attention both in the research community and people from industry [5,17–19]. For instance, the one-stage YOLOv4 [2] detector has proven its efficiency for real-time object detection applications.

In this paper, we propose a pest birds detection approach using a pre-trained YOLOv4 detection model for rice crops in the North Valley in Senegal We build our approach by using YOLOv4 as the basis because this latter is suitable for real-time detection and provides a lightweight version which can be deployed on computers with low resources. We rely on transfer learning in order to fineturne a pre-trained YOLOv4 on two large datasets (ImageNet and MS-COCO) to improve the accuracy of our final model within our setting. To show the efficiency of our model we conduct experiments on a real bird dataset, exhibiting an mean average precision (mAP) (of 96%. The main contributions of this work are as follows.

– Building a real dataset of images about pest and non-pest birds in rice crops in Senegal.
– Designing and evaluating a promising pest birds detection model based on a pre-trained YOLOv4 model.

The remaining of the paper is organized as follows. In Sect. 2, we first review the state-of-the-art. Then, we detail the building block of our proposed detector in Sect. 3 and evaluate its performance over a real dataset in Sect. 4. Finally, we conclude in Sect. 5 by giving some research perspectives.

2 Related Work

We briefly review in this section several research directions pertaining to the problem of object detection using deep learning in general, and in particular for the task of monitoring pest birds in the field of Agriculture.

2.1 Deep Leaning Based Object Detection Models

Over the past years, deep learning based object detection models have received much attention both in the research community and people from industry. A thorough presentation of the state-of-the-art of such types of object detectors is available in [5,17–19]. For instance in [19], Zhengxia et al. have proposed a up-to-date survey of the object detection field in the last 20 years. They made a review of more than 400 papers about object detection in the light of its technical evolution, spanning over a quarter-century's time (from the 1990 s to 2019). They covered a number of topics in the paper, including the milestone detectors in history, detection datasets, metrics, fundamental building blocks of the detection system, speed up techniques, etc. The paper has also reviewed some important detection applications, such as pedestrian detection, face detection, text detection, etc. It made an in-deep analysis of their challenges as well as technical improvements in recent years.

As highlighted in [5], state-of-the-art general purpose and domain-specific object detection models can be categorized into two families: the *one-stage detectors* and *the two-stage detectors*. As examples of popular one-stage detectors we can cite YOLO variants [2,12,13], SDD [9], and RetinaNet [6]. Faster R-CNN [14] is an example of existing two-stage detection models. While methods in both families tackle the same problem, they differ regarding the performance criteria they put in forward. Indeed, two-stage detectors are better in localization and object recognition accuracy, whereas the one-stage detectors are faster. In recent years, proposed object detectors such as YOLOv4 [2] often add some additional layers, seen as the neck of the detector, between backbone and head in order to collect feature maps from different stages. The neck has been introduced in order to improve the accuracy of the prediction, in particular for one-stage detectors. Within the same family, detectors can be also distinguished in terms of performances and application scenarios. For instance, YOLOv4 model proposes a lightweight version that can be used and deployed in computers with low resources such as sensors and micro controllers.

2.2 Pest Birds Monitoring Approaches

Monitoring pest birds is still an acute activity in the field of Agriculture because till now currently used traditional methods remain inefficient in particular for rice crops in Sub-Saharan countries like Senegal. Recently, automatic monitoring and repellent systems for pest birds are starting to be investigated and proposed in the research field [1,10,15]. The most promising works are based on Artificial Intelligence and sensors.

In [15] the authors review the literature of systems and methods employed for autonomous bird pest control in Agriculture. The study points out the fact that using natural bird predator is one of the effective methods used for bird deterring. However, since the predators cannot be controlled, designing artificial predators or systems that act like predators is the focus of most systems reviewed in that paper. A conceptual system was proposed with emphases on the use of offensive strategy when designing an effective bird deterrent system More specifically, [10] presents a CNN based system to detect flocks of birds and to trigger an actuator that will scare the objects only when a flock passes through the monitored space. Before teaching the network, video cameras and a differential algorithm are used to detect all items moving in the vineyard. In terms of function, the algorithm is implemented in a module consisting of a microcomputer and a connected video camera. When a flock is detected, the micro controller will generate a signal to be wirelessly transmitted to the module, whose task is to trigger the scaring actuator. Despite the interesting studies to automate detection and repellent of pest birds in Agriculture, the main observation we can made on the existing proposals is a very few attention made on the use of existing object detection models in the proposed systems or approaches, in particular to fight against the pest birds in rice crops.

To the best of our knowledge this study is the first attempt to use inherent object detection models to automate the monitoring of pest birds in rice crops in Sub-Saharan countries like Senegal.

3 Pest Birds Detection Model

We build and propose our approach to automatically localize and classify pest birds on images (e.g. from a real time image acquisition system) by using YOLOv4 model and transfer learning. The YOLOv4 represents our basis detector model and the transfer learning procedure has been used to mitigate the impact of the limited size of our birds dataset in the learning process of the optimal parameter values of the detection model. The transfer learning helps us to use and fit YOLOv4 in our setting by considering the same hyper-parameter values as those obtained with the basic YOLOV4 model trained and validated on two large and reference datasets in the domain of object detection based on deep learning methods. We start the description of the setting up of our detection model for pest birds in rice crops in Senegal by presenting the architecture of YOLOv4 model.

3.1 YOLOv4 Model Architecture

YOLOv4 [2] belongs to the class of one stage object detectors, also known as dense detectors, which prioritize the inference speeds while ensuring a high detection accuracy. Recall that in one-stage detector models ROI (Region of Interest) is not selected, the classes and the bounding boxes for the complete image are simultaneously predicted. Thus, this makes them faster than two-stage detectors. YOLOv4, corresponding to the fourth version of the YOLO detection model, introduces great improvements of the speed and the accuracy over its predecessors (e.g. [12,13]). Similarly to the other one-stage detectors, YOLOv4 model architecture [2], as shown in Fig. 1, consists of three main components which play each a crucial role in the object detection process: *the backbone*, *the neck* and *the head* representing the dense prediction layer. In YOLOv4, the following choice has been made for each component of its architecture after intensive performance evaluation on large image datasets (MS COCO and ImagineNet).

- CSPDarknet53 [16], a Convolutional Neural Network based on the DenseNet design, as the backbone. Thanks to its two blocks (Convolutional Base Layer and Cross Stage Partial Block), it concatenates the previous inputs with the current input before proceeding into the dense layers - this is referred to as the dense connectivity pattern.
- SPP [4] additional module and PANet path-aggregation [8] as the neck where feature maps are collected from the different stages of the backbone. Then, it mixes and combines the collected feature maps to prepare them for the next step. To this end, an additional block called SPP (Spatial Pyramid Pooling) is added in between the CSPDarkNet53 backbone and the feature aggregator network (PANet) which latter improves the process of instance segmentation. This helps to increase the receptive field and separates out the most significant context features and has almost no effect on network operation speed. It is connected to the final layers of the densely connected convolutional layers of CSPDarkNet.
- YOLOv3 [13] as the head. The main function here is locating bounding boxes and performing classification. We defer the reader to [2] for the details about this function.

In practical, YOLOv4 models an object-detection task as a regression task followed by a classification task. Regression predicts classes and bounding boxes for the whole input image in single run and helps to identify the object position while classification determines the object's class. More specifically, the model receives an input consisting of the training images which will be fed to the network - they are processed in batches in parallel by the GPU. Then, the Backbone and the Neck components perform the feature extraction and aggregation. The detection Neck and detection Head together can be called as the Object Detector. At last, the head (or dense prediction layer), responsible for the detection (both localization and classification) ends the process. For further improvements of the accuracy without extra running time, YOLOv4 introduces and uses Bag of Freebies (BoF) and Bag of Specials (BoS) strategies which usually represent data augmentation techniques.

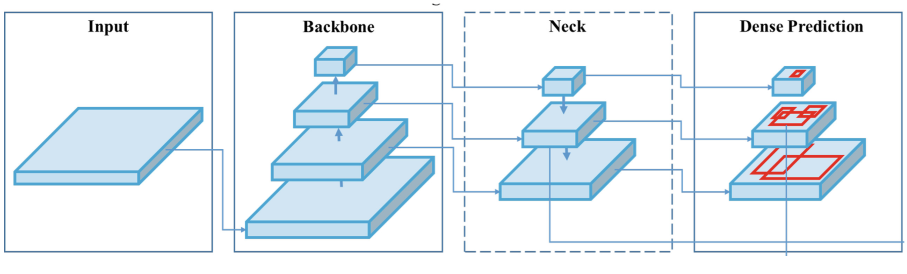

Fig. 1. YOLOv4 main components

3.2 Building Pest Bird Real Data

Our collection of real world data consists of a set of pest bird images and non pest bird images that are frequently present in rice crops in the North Valley in Senegal. As a collection methodology, we first established the list of the names of all such birds. Then, we relied on Web search engines and related tools (Google Search or Bing) and Web scrapping tools to collect images corresponding to the list of bird names of interest. The used tools include *Bing Bulk Image Downloader* script and *Google Chrome extension* for bulk images downloading named *image downloader*. After the collection of the images from the Web, we did a pre-processing stage in order to first delete irrelevant images regarding our application domain, altered images and duplicated images. Then, we proceeded to putting all the images in the same size, i.e. resizing step with 416 × 416 size. Finally, we proceeded to the labelling of our images needed when fitting our supervised detection model. For the labelling step, we have used *labelImg* which is an open source GUI image annotation tool written in Python[1]. We used bounding-box techniques to draw the ROI of each image and associated the corresponding class name to each box. Using this process, we have generated the labels of our entire set of images according to YOLO format.

We ended up the data collection phase with an annotated dataset of 2443 images with 1338 images belonging to the class of pest birds and 1105 images corresponding to non pest birds. An excerpt of our dataset is given in Fig. 2.

(a) Quelea (b) Amarante

Fig. 2. Excerpt of images in our bird dataset

3.3 Training Phase

We describe below the training phase of our pest bird prediction model using Yolo v4. We start by detailing the pre-trained YOLOv4 object detection model proposed by Bochkovskiy et al. in [2].

[1] https://github.com/tzutalin/labelImg.

Pre-training of YOLOv4. YOLOv4 in its basis has been trained and evaluated on two different large datasets: ImageNet (ILSVRC 2012) [3] with $1,000$ distinct classes and MS COCO (test-dev 2017) [7] which latter consists of 80 different object classes. For the purpose of this study, we rely on the same values of YOLOv4 hyper-parameters fitted and tested using those two images datasets. In ImageNet, for the classification of images, the default hyper-parameters of Yolov4 for the various conducted experiments have been set as follows: (i) the training iteration set to $8,000,000$; (ii) the batch size and the mini-batch size are 128 and 32, respectively; (iii) the polynomial decay learning rate scheduling strategy is adopted with initial learning rate of 0.1; (iv) the warm-up steps is 1000; and (v) the momentum and weight decay are respectively set as 0.9 and 0.005. All the BoS (Bag of Specials) experiments use default hyper-parameters while BoF (Bag of Freebies) experiments consider additional 50% training iterations. The training has been done with a 1080Ti or 2080Ti GPU. In MS-COCO object detecton experiments, the authors of YOLOv4 set the default hyper-parameters as follows: (i) the training iterations is $500,500$; (ii) the step decay learning rate scheduling strategy is adopted with initial learning rate 0.01 and multiply with a factor 0.1 at the 400,000 steps and the 450,000 steps, respectively; and (iii) the momentum and weight decay are respectively set as 0.9 and 0.0005 All architectures use a single GPU to execute multi-scale training in the batch size of 64 while mini-batch size is 8 or 4 depend on the architectures and GPU memory limitation. Except for using genetic algorithm for hyper-parameter search experiments, all other experiments use default setting. Genetic algorithm used YOLOv3-SPP to train with IoU loss and search 300 epochs for min-val $5k$ sets. We adopt a search learning rate of 0.00261, momentum of 0.949, IoU threshold forassigning ground truth 0.213, and loss normalizer 0.07 for genetic algorithm experiments. We have verified on a large number of BoF, including grid sensitivity elimination, mosaic data augmentation, IoU threshold, genetic algorithm, class label smoothing, cross mini-batch normalization, self-adversarial training, cosine annealing scheduler, dynamic mini-batch size, Drop Block, Optimized Anchors, different kind of IoU losses. We also conduct experiments on various BoS, including Mish, SPP, SAM, RFB, BiFPN, and Gaussian YOLO. For all experiments in the pre-training phase, we only use one GPU for training, e.g. techniques such as syncBN that optimizes multiple GPUs are not used.

Fine-Tuning Phase. To adapt the pre-trained YOLOv4 model in our application case, we implemented additional layers trained and evaluated on our pest birds dataset. To obtain the final detection model, we used YOLOv4 pre-trained weights by harnessing transfer learning. The fine-tuning phase has been done with the following parameter settings: (i) linebatch $= 64$; (ii) subdivisions $= 16$; (iii) width $= 416$ height $= 416$ as network input size; (iv) maxbatches $= 6000$. (v) filters $= 21$ on each convolutional layer before each YOLOv4 layer; and (vi) number of classes $= 2$.

4 Experiments and Validation

We detail in the section the experiments we conducted on a real dataset to evaluate the performance of our proposed model. For the details of the description of the used real bird dataset we refer the reader to Sect. 3.2. We start by present our experiment setting.

4.1 Setting Up Experiments

We present here the implementation of our approach and the experimental environment.

Implementation Details. For the purposes of fitting and evaluating our proposed approach, we did the implementation of our detection model using DarketNet library available in C and CUDA. For the labelling of the images in our dataset, we did it using labellImg a graphical user interface written in Python. For the fine-turning of YOLOv4 model it comes out to configure the file *YOLOv4-custom.cfg* by specifying the values of the hyper-parameters that will be considered for the training of the model in our pest bird training set.

Testing Environment. The model has been trained and tested using the Google Colaboratory environment to benefit from the free avalaible GPU and computation power. Colab provides several types of NVIDIA based GPU: GPU NVIDIA K80, P100, P4, T4, V100 and A100 that offer a variety of computation options which suit our workload.

Splitting of the Dataset. With respect to the recommendation of the authors of YOLOv4, we split our dataset considering the following proportions : 90% for the training set and 10% for the testing.

4.2 Performance Measures

We present below the metrics used to evaluate performance of our detection model. Some of them are based on the confusion matrix that sums up the number of true positive (TP), false positive (FP), false negative (FN), and true negative (TN) after the testing of our detection and comparison with the ground truth.

Intersection over Union. Intersection over Union (IoU) measures the difference between the ground truth mask (gt), the predicted objects and the predicted mask (pd) as follows.

$$IoU = \frac{area(gt) \cap area(pd)}{area(gt) \cup (area(pd)} \tag{1}$$

The value of the IoU metric varies from 0 to 1 where 0 means no overlapping and 1 implies a perfect prediction rate. .

Precision. Precision (P) evaluates the ability of the detector to identify exactly the correct classes of objects.

$$P = \frac{TP}{TP + FP} \tag{2}$$

Recall. Recall (R) measures the ability of the model to find correct classes of the given objects, i.e. the proportion of true positives detected among all object classes to be predicted.

$$R = \frac{TP}{TP + FN} \tag{3}$$

A good model is one that can detect correctly the classes of most of the objects.

F1-Score. It represents the weighted average of the precision and the sensitivity (recall). Therefore, this score takes into account both false positives and false negatives. Intuitively, it is not as easy to understand as precision, but F1-score is generally more useful than precision, especially if you have an unequal class distribution values:

$$F1 - score = 2 \times \frac{P \times R}{P + R} \tag{4}$$

Mean Average Precision. The mean average precision (mAP) is a metric used to measure the accuracy of object detectors over all classes in a specific database. The mAP is simply the average precision over all classes [11] and is given by the following formula:

$$mAP = \frac{1}{n} \sum_{i=0}^{n} AP_i \tag{5}$$

where AP_i being the average precision in the ith class and n is the total number of classes.

Loss Function. We consider the sum-squared error between the predictions and the ground truth to calculate loss. The loss function encompasses the classification loss and the localization loss (errors between the predicted boundary box and the ground truth).

ROC Curve. A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.

4.3 Results and Analysis

Figure 3 shows the results of testing our final model on the two images given in Fig. 2; the model successfully detects both birds and classifies them as pest (image in left) and non pest (image in right).

(a) Quelea class detection

(b) Amarante detection

Fig. 3. Examples of detection outputs of our model

The performance of our YOLOv4 pest birds detection model are summarized in Table 1 and Fig. 4. Table 1 contains the precision, recall, F1-score, the mAP, and the IoU measures of the model. The mAP is equal to 96.5% meaning that the average precision of our detection model is very high. This trend is also verify by the recall of our model which is equal to 94%.

Table 1. Performance measures of our detection model

mAP	F1-Score	Precision	Recall	IoU
0.965	0.89	0.84	0.94	63.31

Figure 4 shows the curve representing the evolution of the mAP (red curve) and the loss function (blue curve) as a function of the number of iterations. We note that the loss function exponentially decreases from 1 up to about 200 iterations and then from there, continues to linearly decrease. The loss at the last iteration is 0.1398 showing a very low error rate of our detection model. On the other hand, after the first 1000 iterations, the average accuracy (mAP) is 74%. This percentage grows exponentially until 92%. Then from this point, the average accuracy follows a monotonic trend up to 96% until convergence. The average of the mAP at the end of the training is 96.5%. In sum, we can deduce that our model present promising performance in its ability to correctly detect pest birds and non pest birds. This may be due to the transfer learning process and the high quality of the set of bird images used to fine-turne the pre-trained model in our application case.

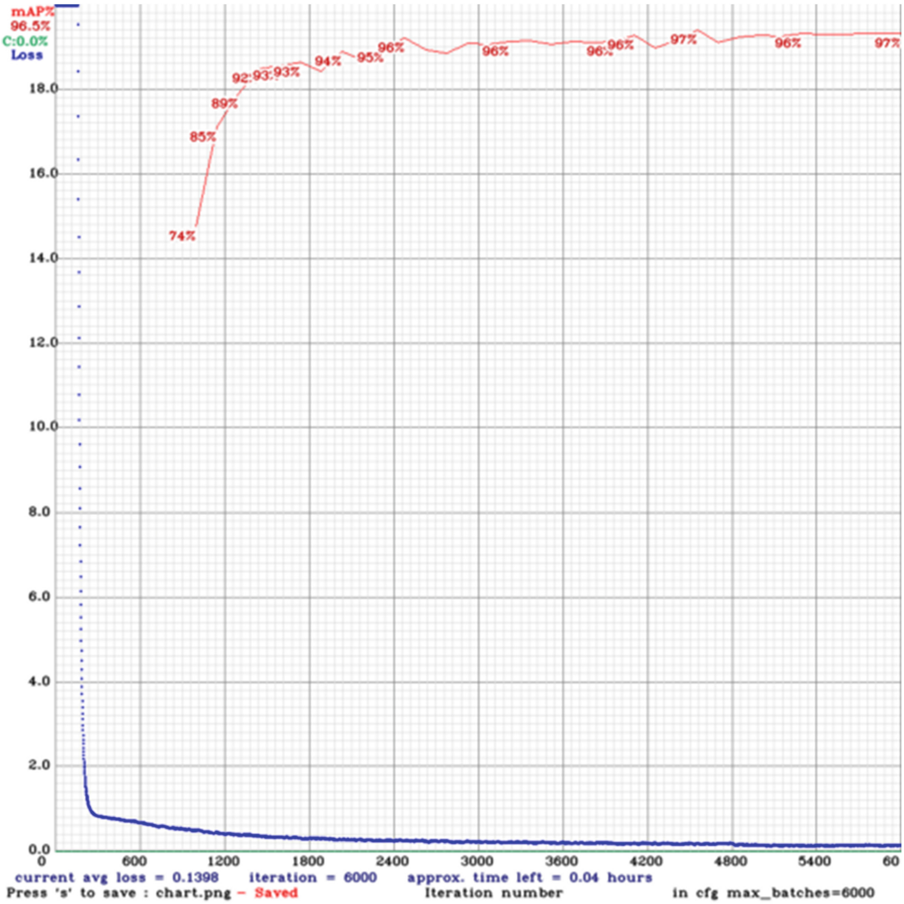

Fig. 4. ROC curve of the training accuracy and lost of our detection model

5 Conclusion and Perspectives

This paper has tackled the problem of pest birds monitoring in rice crops in Senegal and proposed a detection approach built on top of a pre-trained YOLOv4 detector and transfer learning. To show the efficiency of our model we conducted experiments on a real bird dataset, exhibiting a mean average precision of 96%. We use transfer learning to solve the problem of limitations in terms of domain specific data size and resources. As perspectives, we plan to evaluate other one-stage detectors such as SDD and RetinaNet in our setting and compare their performance to those of YOLOv4 in order to see if they can outperform this latter. We also plan to deploy and evaluate our approach in the rice yards in the North Valley in Senegal to see how the model reacts in real environment. At last, we envision to evaluate the running time of our model in order to assess its applicability in real life applications.

References

1. Bhusal, S., Bhattarai, U., Karkee, M.: Improving pest bird detection in a vineyard environment using super-resolution and deep learning. IFAC-PapersOnLine **52**(30), 18–23 (2019)
2. Bochkovskiy, A., Wang, C.-Y., Mark Liao, H.-Y.: YOLOv4: optimal speed and accuracy of object detection. CoRR, abs/2004.10934, 2020
3. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei, L.F.: ImageNet: a large-scale hierarchical image database. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009)
4. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
5. Jiao, L., Zhang, F., Liu, F., Yang, S., Li, L., Feng, Z., Qu, R.: A survey of deep learning-based object detection. IEEE Access **7**, 128837–128868 (2019)
6. Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of IEEE International Conference on Computer Vision, pp. 2999–3007 (2017)
7. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
8. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: Proceedings of IEEE and CVF Conference on Computer Vision and Pattern Recognition, pp. 8759–8768 (2018)
9. Liu, W., et al.: In: Proceedings of ECCV **9905**, 21–37 (2016)
10. Marcon. P., et al.: A system using artificial intelligence to detect and scare bird flocks in the protection of ripening fruit. Sensors **21**(12), 4244 (2021)
11. Padilla, R., Netto, S.L., SilvaEduardo da, A.B.: A survey on performance metrics for object-detection algorithms. In: Proceedings of International Conference on Systems, Signals and Image Processing, pp. 237–242 (2020)
12. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
13. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
14. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks (2016)
15. Suleiman, I., Babawuya, A., Oyewole, A., Salihu, B., Adeoti O., Alhaji, S.Y.: A review of bird pest repellent systems in farms (2021)
16. Wang, C.-Y., Mark Liao, H.-Y., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., Yeh, I.-H.: CSPNet: a new backbone that can enhance learning capability of CNN. In: Proceedinds of Conference on Computer Vision and Pattern Recognition, Workshops, pp. 1571–1580. Computer Vision Foundation and IEEE (2020)
17. Abbas Zaidi, S.S., Ansari, M.S., Aslam, A., Kanwal, N., Asghar, M., Lee, B.: A survey of modern deep learning based object detection models. Digit. Sig. Process. **126**, 103514 (2022)
18. Zhao, Z.-Q., Zheng, P., Shou-Tao, X., Xindong, W.: Object detection with deep learning: a review. IEEE Trans. Neural Netw. Learn. Syst. **30**(11), 3212–3232 (2019)
19. Zou, Z., Shi, Z., Guo, Y., Ye, J.: Object detection in 20 years: a survey. CoRR, http://arxiv.org/abs/1905.05055arXiv:1905.05055 (2019)