# A Review on Real-Time Object Detection Models using Deep Neural Networks

Nkalubo Lenard Byenkya[1] and Nakibuule Rose[2]

[1] Kyambogo University, Department of Data Science and artificial Intelligence, Uganda, lnkalubo@kyu.ac.ug
[2] Makerere University, Department of Computer Science, Uganda, rnakibuule@cis.mak.ac.ug
[*] Kyambogo University, Department of Data Science and artificial Intelligence, Uganda, lnkalubo@kyu.ac.ug

## Abstract.

Object detection is one of the most common issues in computer vision. Various researchers have contributed to a number of application fields, including robots, self-driving cars, and video surveillance. The real-time object detection methods that use deep learning methods are reviewed in this work. Its purpose is to make the readers familiar with the pertinent information, literature, and most recent advancements in cutting-edge methods. Using three sets of keywords: Deep learning, object detection, and convolutional neural networks. This study reviewed electronic records that were gathered from the top scientific databases (IEEE, Google Scholar, Scopus, DOAJ, Science Direct, Elsevier, and other journal publications). The framework for object detection has two unique groups of detectors: conventional detectors and deep learning-based detectors. Deep learning object detectors come in two varieties: one-stage detectors and two-stage detectors. Two-stage detectors produce sparse area recommendations in the first phase before performing regression and classification, in contrast to one-stage detectors that perform classification and regression using dense anchor boxes without first constructing a collection of sparse regions of interest. Crop harvesting, object detector models for the blind, pedestrian identification on the road, traffic sign recognition and classification, text detection, and remote sensing target detection are all applications of object detection. We suggest creating a one-stage object detection model in our upcoming research to aid in directing blind movements.

**Keywords:** Deep learning, object detection, convolutional neural networks

## 1. Introduction

Deep learning (Alzubaidi et al., 2021) (Sarker, 2021) methods are built on artificial neural networks (ANNs) (Lecun et al., 2015). A deep learning-based method gained popularity after winning a computer vision competition handily in 2012. Since 2010, deep learning techniques have improved their performance in complex visual recognition tasks, and by 2015, they had surpassed human accuracy (Lecun et al., 2015). Traditional feature extraction techniques demand human interaction, but deep learning immediately learns from image data (Ahishakiye et al., 2021). Deep convolutional neural networks (CNNs) have achieved state-of-the-art achievements in terms of object detection accuracy and detection speed as a result of the development of deep learning technology in machine vision applications. CNN's capacity to automatically learn from and extract information from an input image is its main benefit (Junos et al., 2021). Computers can now observe, recognize, and evaluate objects in still images and moving images thanks to the study of computer vision. Applications of computer vision include face detection, face identification, pedestrian counting, security systems, vehicle detection, self-driving automobiles, and many others (Zhao et al., 2019) (Pathak et al., 2018). Object localization, object categorization, and object identification are a few computer vision concepts that are related to object detection processing (Kaur & Singh, 2022). In discriminative tasks, deep learning models have advanced significantly. This has been made possible by sophisticated processing, intricate network architectures, and easy access to enormous volumes of data. As a result of the development of convolutional neural networks, deep neural networks have been effectively applied in Computer Vision applications such as image classification, object recognition (Szegedy et al., 2013), and image segmentation (Shorten & Khoshgoftaar, 2019).

One of the most popular problems in computer vision is object detection (Krizhevsky et al., 2012). The foundation of conventional object recognition systems consists of handcrafted characteristics and shallow trainable structures. The creation of complex ensembles, which combine a number of low-level image features with high-level data from object detectors and scene classifiers, can quickly stall their performance. In order to overcome the problems that traditional architectures have, more potent tools that can learn semantic, high-level, and deeper features are now being made available (Zhao et al., 2019). Identifying the locations of items in a given image (object localization) and the categories to which each object belongs (object classification) is referred to as object detection (Zhao et al., 2019). In this study, we will use the term "object recognition" to refer to both object detection (a task that requires an algorithm to pinpoint all of the objects present in an image) and object classification (a work that requires an algorithm to determine which item classes are present in an image) (Russakovsky et al., 2015). A quick, accurate, and versatile method of general-purpose item recognition is required. Since the introduction of neural networks, detection frameworks have improved in speed and accuracy. However, the bulk of detection methods are still restricted to a few types of objects (Redmon & Farhadi, 2017). Deep learning has become the most talked-about technology due to its success in applications including language processing, object identification, and picture classification (Srivastava et al., 2021). One of the study's sections covered a review and history of deep learning and its uses for object detection (Zhao et al., 2019) (L. Liu et al., 2020).

### 1.2 Research Objectives and Outline

This study provides an overview of the research that has been done on the existing object detection models and their applications, particularly in blind movements, which is motivated by the current advances and numerous significant studies in the field of real-time object detection models.

The rest of the article is organized as follows: In section 2, Materials and methods are discussed; section 3 discusses the results and section 4 discusses the conclusions and recommendations.

### 1.3 Significance

Due to the rapid advancements in machine learning, computing power, smart phone adoption, and highly dynamic real-world environments, it is essential to assess the efficiency of heterogeneous real-time object-detection models by considering their capabilities and how they can be used to address current problems.

## 2. Methodology

### 2.1 Materials and Methods

According to several review studies (Xie et al., 2019), (Hsu et al., 2012), (Hwang & Tsai, 2011), it is crucial to review publications from reliable data sources. In this work, a thorough keyword-based search for publications on object identification techniques was carried out in the top scientific databases, including Google Scholar, Wiley, Science Direct, Springer, IEEE, Scopus, Nature, Elsevier, and PubMed. This study also included relevant postgraduate theses.

### 2.2 Inclusion criteria

This study took into account studies that offered deep learning-based approaches to object detection. The PRISMA technique and flow diagram (Bakator, 2018) was used to find the pertinent research publications. The four steps of this approach are as follows: (i) the Identification Phase, which involved gathering content from diverse sources; (ii) the Screening Process. Duplicate articles were disregarded throughout this phase, and subpar articles were also eliminated. Phase of eligibility; (iii) To decide which articles could be subjected to more evaluation, an examination was carried out. The excluded articles were not eligible; (iv) The included phase is the last stage. This stage involved the analysis of the articles that were used in the study.

### 2.3 Exclusion criteria

Studies that involved object detection approaches other than deep learning methods have been excluded from this study.

## 3. Discussion of Results

### 3.1 Object Detection Models

The basic objective of object detection, one of computer vision's most effective applications, is to recognize and classify the items in an image (Kaur & Singh, 2022). The object detection framework divides detectors into two groups: conventional detectors and deep learning-based detectors. The two-stage detector and the one-stage detector are the two types of deep learning object detectors. Unified detectors (Chen et al., 2019) (Lin et al., 2020) (Tian et al., 2019) (Zhang et al., 2018) (C. Zhu et al., 2019) When performing classification and regression without first creating a sparse region of interest (RoI) collection, employ dense anchor boxes, whereas in two-stage detectors (He et al., 2020) (X. Li et al., 2019) (Lu et al., 2019) (Ren et al., 2017), Sparse region proposals are developed in the first stage of two-stage detectors, following which they are regressed and classified (Lu et al., 2020). One-stage detectors are more effective due to their straightforward architecture, although two-stage detectors still exceed them in terms of accuracy.
The accuracy gap still exists despite recent attempts to enhance one-stage detectors by copying the structural design of two-stage detectors (Lu et al., 2020). On the other hand, single-stage detectors treat object detection as a simple regression problem that accepts the entire image as input and simultaneously generates multiple bounding boxes and class probabilities. The model is hence considerably quicker than the two-stage object detectors (Junos et al., 2021). Two-stage detectors provide adequate accuracy, but their computation times are considerable. Therefore, one-stage detectors are advised to process in less time while maintaining adequate accuracy (Adarsh et al.,

2020). While RCNN (Zhou et al., 2021), Fast RCNN (Girshick, 2015), and Faster RCNN (Xiao et al., 2020) (Ren et al., 2017) algorithms are instances of two-stage detector algorithms, SSD and YOLO with its variants are examples of one-stage model algorithms. According to the study (Lu et al., 2020), two-stage detectors are superior to one-stage ones in the following ways: 1) One-stage detectors have a class imbalance issue if no additional design is added because they immediately face every region on the image. On the other hand, two-stage detectors eliminate the majority of the unfavorable recommendations by selecting a small number of region proposals. 2) The head of the network (used for proposal classification and regression) can be larger with two-stage detectors because they examine fewer proposals than one-stage detectors do. This makes it possible to extract richer features. Three) Two-stage detectors employ the RoIAlign operation to extract the location consistent feature from each sampled proposal, as opposed to one-stage detectors that might permit different region proposals to share the same feature and could lead to severe feature misalignment because of the coarse and spatially implicit representation of the proposals. 4) In comparison to single-stage methods, two-stage detectors do a double regression of the object location (once on each step). The fundamental issue that all one-stage detectors face, the mismatch between anchor boxes and convolutional features, significantly reduces one-stage detector performance (Chen et al., 2019). One-stage object detectors' inability to outperform top-performing, two-stage algorithms is primarily due to class imbalance, claims the study (Lin et al., 2020). The study suggested the focused loss, which adjusts the cross-entropy loss and concentrates learning on difficult negative examples, as a remedy. The creation of a fully convolutional one-stage detector and detailed experimental analysis proving that it reaches cutting-edge accuracy and speed were used to demonstrate the study's effectiveness. Figure 1. shows object detection techniques.

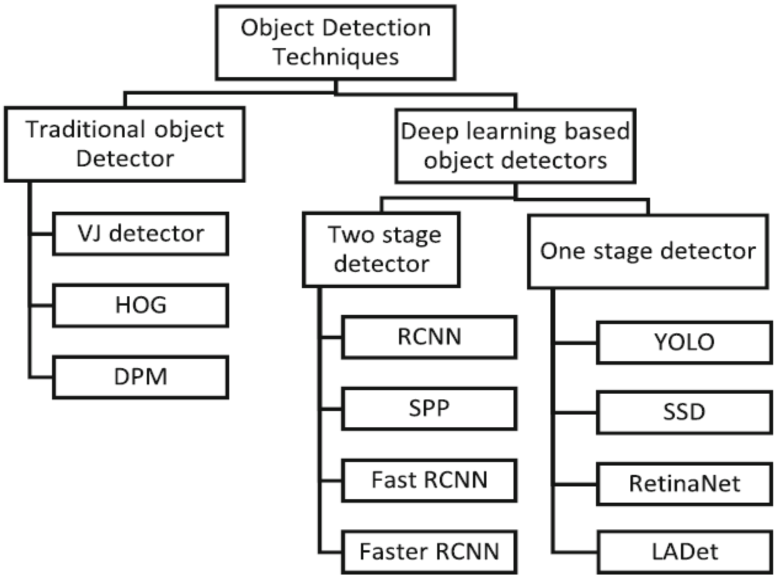

*Figure 1: Object Detection Techniques Adapted from (Kaur & Singh, 2022).*

## 3.2 Review of Object Detection Models

### 3.2.1 One shot detection models

### a) YOLO (You Only Look Once) and its variants

YOLO is a single object detection model (Redmon et al., 2016). The model is easy to construct and can be directly trained on whole images. Unlike classifier-based approaches, YOLO is trained on a loss function that directly relates to detection performance, and the entire model is learned at once. Yolo pushes the limits of real-time object detection as the fastest general-purpose object detector in the world. YOLO is suitable for applications that call for speedy and accurate object identification because it can be tailored to new domains. YOLO functions as a tracking system when coupled with a webcam, identifying objects as they move and change their appearance. However, YOLO imposes strong geographic constraints on bounding box predictions because each grid cell can only predict two boxes and one class. This spatial restriction restricts the number of nearby items that our model can anticipate. Bird flocks, for example, are a good example of a little object that emerges in a group yet the model struggles to handle. Additionally, while the model learns to estimate bounding boxes from data, it has trouble generalizing to objects with novel or odd aspect ratios or configurations. The model also employs fairly coarse characteristics for predicting bounding boxes because our architecture incorporates numerous downsampling layers from the input image. The model was trained using a loss function that roughly represents detection performance, although our loss function treats errors in small and large bounding boxes equally. A small inaccuracy will typically go unnoticed in a large box, while a small error in a small box will have a far bigger impact on intersection over union (IOU).

The image is divided into SS grid cells with similar size using YOLOv1 (Redmon et al., 2016). A grid cell is in charge of object detection if the centroid of the object is located inside of that grid cell. Each cell may forecast a predetermined B number of bounding boxes with a confidence score. Each bounding box is composed of five values of x, y, w, h, and confidence score. In the paper, a modified YOLOv1-based neural network is suggested for object detection (Ahmad et al., 2020). Yolo-LITE, a real-time object identification model developed for mobile devices without a GPU, like a laptop or a smartphone, was proposed in the paper (R. Huang et al., 2019). (GPU). Real-time object identification is now more accessible to a wider range of devices thanks to YOLOLITE, a model that was created to offer a more compact, rapid, and efficient version of the original object detection algorithm YOLOV2. A state-of-the-art, real-time object identification system called YOLOv2 (YOLO9000) (Redmon & Farhadi, 2017) can identify more than 9000 different item types. In YOLOV2, batch normalization and convolution layers were merged to improve accuracy and reduce the overfitting problem. It is flexible enough to operate at different image sizes and enables a seamless trade-off between speed and accuracy. In order to address the problems with the YOLOv2 object identification model's excessive number of model parameters and poor performance on small-size objects, the study (R. Li & Yang, 2018) suggested an improved YOLOv2 object detection model. First, it improves the YOLOv2 by using depth-wise separable convolution instead of the YOLOv2's conventional convolution. There are 78.83% fewer parameters in the convolution layer. The feature extraction engine of Darknet19, which has problems identifying small objects, was upgraded to Darknet 53 in YOLOv3 (Redmon & Farhadi, 2018) to address the problem. The accuracy of the method was significantly improved in that study by the addition of residual block, skip connections, and up-sampling. The core of the feature extractors was again upgraded to CSPDarknet53 in YOLOv4 (Bochkovskiy et al., 2020), which significantly improved the accuracy and efficiency of the method. The most recent and effective version of the YOLO algorithm is called YOLOv5, and instead of using Darknet as its framework, it uses PyTorch (X. Zhu et al., 2021). Since the depth of the convolutional layer was decreased, another variation of YOLO v3 is known as YOLO v3-Tiny (Redmon & Farhadi, 2018). As a result, even though the running speed is significantly higher (around 442% quicker than the prior YOLO versions), the detection accuracy is reduced (Adarsh et al., 2020). An enhanced version of the YOLOv3 tiny network called YOLO-P (Junos et al., 2021) features a lightweight backbone made of densely coupled neural networks, a multi-scale detection architecture, and an anchor box size that has been optimized. The proposed YOLO-P model had a satisfactory mean average precision and F1 score of 98.68% and 0.97, respectively, based on the experimental results. The study recommended PP-YOLOv2, which performs better than other well-known detectors like YOLOv4 and YOLOv5 in terms of speed and accuracy (X. Huang et al., 2021).

### b) SSD

SSD (W. Liu et al., 2016) is a single-shot detector. It excels at striking a perfect balance between speed and precision of results. To create the feature map, the model computes a CNN-based model once on the input image. Additionally, it uses anchor boxes that are comparable to faster RCNNs at various aspect ratios and learns the offset rather than identifying the box. There are numerous layers in CNN, each of which uses distinct feature maps and processes data at varying sizes. As a result, it can find targets of various sizes. Even when input photos are of a small size, SSD beats other single-stage techniques in studies in terms of accuracy across a variety of datasets (Kumar et al., 2020).

### c) Comparison Between YOLO and SSD

The image is not divided into random-sized grids by SSD as it is by YOLO. For each location on the feature map, it projects the offset of predefined anchor boxes (default boxes). Each box is a certain size, proportion, and location in relation to the relevant cell. Convolutionally, the entire feature map is covered by all of the anchor boxes. SSD and YOLO's anchors are slightly different from one another. Because YOLO uses anchors that can range in size from a single grid cell to the entire image, all of its predictions are based on a single grid. The SSD's anchors place a lot of emphasis on various pragmatic perspectives and dimensional ratios of its target shapes, but not nearly enough on goal size. Unlike the anchors of SSD, which are created using a simple method, the anchors of YOLO are calculated using k-means clustering on the training data. Although YOLO determines the confidence score to show confidence in the anticipated results, SSD does not use it. SSD performs this task via a unique backdrop class. A poor YOLO confidence score matches the anticipated SSD background class result. Both illustrate the impossibility of the detector ever discovering a target (Tan et al., 2021).

**d) FCOS**

FCOS (Tian et al., 2019) is a fully convolutional one-stage object detector that resolves object detection in a similar way to how semantic segmentation does so: by making predictions about individual pixels. Modern object detectors like RetinaNet, SSD, YOLOv3, and Faster R-CNN largely rely on pre-defined anchor boxes. Contrarily, FCOS is a proposal that is free of anchor boxes. By removing the predefined set of anchor boxes, FCOS completely eliminates the complex calculations related to anchor boxes, such as computing overlaps during training (M. Zhu et al., 2022).

**e) FSAF: Feature Selective Anchor-Free**

In comparison to its competitors that use anchors, the FSAF module (C. Zhu et al., 2019) operates faster and performs better. While adding the least amount of computational overhead when used in conjunction with anchor-based branches, the FSAF module can consistently beat the strong baselines over a range of backbone networks. FSAF outperforms the most cutting-edge single-shot detectors now available and significantly enhances strong baselines with little inference overhead.

**f) RefineDet**

RefineDet (Zhang et al., 2018) maintains efficiency comparable to one-stage approaches while beating two-stage methods in terms of accuracy. The two interconnected parts that make up RefineDet are the anchor refinement module and the object detection module. The former especially aims to (1) filter out negative anchors to reduce the search space for the classifier and (2) coarsely adjust the sizes and positions of anchors to enhance initialization for the next regressor. RefineDet needs an attention mechanism in order to improve performance even more.

### 3.2.2 Two-shot detection models

**a) Region-Based Convolutional Neural Networks (R-CNN)**

Region-based convolutional neural networks are referred to as R-CNN (Girshick et al., 2014). The model combines potent CNNs for object detection with region suggestions for object segmentation. This approach had a number of issues. Because it needs to categorize 2000 area suggestions, CNN training takes a while. Real-time implementation is not practical because it would take 47 seconds to execute each test image. One object detection approach that solves some of R-drawbacks CNN's is Fast R-CNN (Girshick, 2015). Similar to its predecessor, it uses region proposals, but CNN first creates a convolutional feature map from the image, which is then used to pick and warp region proposals from. In order for a fully linked layer to accept the distorted squares, they must be reformed using a RoI (Region of Interest) pooling layer to a preset size. Using a SoftMax layer and the RoI vector, the region class is then predicted. Fast R-CNN is quicker than its predecessor since it does not require the CNN to get 2,000 ideas for each execution. To create a feature map for each image, only one convolution process is used. The drawback of RCNN is that it processes many areas using CNN and uses three separate models to detect targets, which lengthens prediction time. Fast-RCNN uses a lengthy and time-consuming method. The outcome is that computation times are still rather long. For quicker RCNN, the object region proposal takes a long time. Different sorts of systems are operating one after the other.

Therefore, the success of the operations that came before is necessary for the proper completion of the entire treatment (Adarsh et al., 2020).

**b) Grid R-CNN**

An novel framework for object detection called Grid R-CNN (Lu et al., 2019) accurately recognizes items by using a grid-guided localization technique. In contrast to standard regression-based techniques, the Grid R-CNN explicitly collects spatial information and benefits from the position-sensitive characteristic of a fully convolutional architecture. Grid R-CNN replaces the traditional box offset regression model in object detection by employing a grid-guided method for high-quality localization. Numerous tests show that Grid R-CNN performs at the cutting edge, particularly on challenging assessment criteria like AP at IoU=0.8 and IoU=0.9. Grid R-CNN also contributes to significant and ongoing advancement.

**c) Mask R-CNN**

Mask R-CNN (He et al., 2020) extends Faster R-CNN by incorporating an additional branch for object mask prediction on top of the existing branch for bounding box identification. Faster than Faster R-CNN by only a small amount, Mask R-CNN operates at 5 frames per second and is simple to train. Mask R-CNN is very straightforward to generalize to other issues; for instance, we can estimate human postures within the same framework.

### 3.3 Applications of Real-time Object Detection Models

An improved YOLO-based object recognition model for agricultural harvesting is presented in the study (Junos et al., 2021). The proposed model had a 98.91% accuracy rating when it came to identifying fresh fruit bunches of various ages. The extensive testing results show that the suggested YOLO-P model is effective in carrying out accurate and reliable detection at the palm oil plantation. The study (Kumar et al., 2019) suggested an object detector model for blind people to employ that is based on deep learning neural networks. With this method, objects can be located in webcam feeds, movies, and even still images. The accuracy of the model is more than 75%. Approximately 5 to 6 hours must be spent training this model. For real-time object recognition for a blind individual, the single-shot multi-box detector (SSD) technique was used in the model to achieve high accuracy and IOU. In order to observe people on the road, object detection is essential, according to the study (Kaur & Singh, 2022). Numerous application disciplines, such as robots, autonomous driving, and video surveillance, have engaged many researchers. The study also showed that face detection and identification, which has been the subject of much research, is one of the oldest applications of computer vision. Additionally, text detection, target detection for remote sensing, and traffic sign detection and classification all require object detection (Kaur & Singh, 2022). The study (Kaur & Singh, 2022) presents an object detection approach using deep learning neural networks to recognize objects in the photos. The study uses a multilayer convolutional network and an improved SSD approach to quickly and accurately recognize objects. A real-time object detection approach for blind people to use on any device running this model was suggested by the study (Kumar et al., 2019). A convolutional neural network and a single-shot multi-box detection method were used to create the suggested model. TPH-YOLOv5, which is highly effective in object detection in drone-captured situations, was proposed in the study (X. Zhu et al., 2021). For the visually challenged, a neural network model was suggested in the study (Potdar et al., 2018). Blind or visually impaired people rely heavily on their other senses, such touch and aural cues, to interpret their surroundings.

## 4. Discussions and Conclusion

Computers can now perceive, identify, and evaluate things in still photos and moving images thanks to the study of computer vision. Face detection, face identification, pedestrian counting, security systems, vehicle detection, self-driving cars, and other computer vision applications have all been used. The object detection framework divides detectors into two groups: conventional detectors and deep learning-based detectors. The two-stage detector and the one-stage detector are the two types of deep learning object detectors. Examples of one-stage model algorithms are SSD and YOLO in all of its versions, whereas examples of two-stage detector algorithms include RCNN, Fast RCNN, and Faster RCNN. Crop harvesting, object detector models for the blind, pedestrian identification on the road, traffic sign recognition and classification, text detection, and remote sensing target detection are some examples of applications for object detection. In our future work, it is proposed to optimize a YOLO object detection model that may help in guiding blind movements.

### References

Adarsh, P., Rathi, P., & Kumar, M. (2020). YOLO v3-Tiny: Object Detection and Recognition using one stage improved model. *2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, 687–694. https://doi.org/10.1109/ICACCS48705.2020.9074315

Ahishakiye, E., Van Gijzen, M. B., Tumwiine, J., Wario, R., & Obungoloch, J. (2021). A survey on deep learning in medical image reconstruction. *Intelligent Medicine*. https://doi.org/10.1016/j.imed.2021.03.003

Ahmad, T., Ma, Y., Yahya, M., Ahmad, B., Nazir, S., Haq, A. U., & Ali, R. (2020). Object Detection through Modified YOLO Neural Network. *Scientific Programming*, *2020*, 1–10. https://doi.org/10.1155/2020/8403262

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. In *Journal of Big Data* (Vol. 8, Issue 1). Springer International Publishing. https://doi.org/10.1186/s40537-021-00444-8

Bakator, M. (2018). Deep Learning and Medical Diagnosis : A Review of Literature. *Multimodal Technologies and Interaction*. https://doi.org/10.3390/mti2030047

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. http://arxiv.org/abs/2004.10934

Chen, Y., Han, C., Wang, N., & Zhang, Z. (2019). *Revisiting Feature Alignment for One-stage Object Detection*. 1–11. http://arxiv.org/abs/1908.01570

Girshick, R. (2015). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 580–587. https://doi.org/10.1109/CVPR.2014.81

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2020). Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 386–397. https://doi.org/10.1109/TPAMI.2018.2844175

Hsu, Y. C., Ho, H. N. J., Tsai, C. C., Hwang, G. J., Chu, H. C., Wang, C. Y., & Chen, N. S. (2012). Research trends in technology-based learning from 2000 to 2009: A content analysis of publications in selected journals. *Educational Technology and Society*, *15*(2), 354–370.

Huang, R., Pedoeem, J., & Chen, C. (2019). YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers. *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, 2503–2510. https://doi.org/10.1109/BigData.2018.8621865

Huang, X., Wang, X., Lv, W., Bai, X., Long, X., Deng, K., Dang, Q., Han, S., Liu, Q., Hu, X., Yu, D., Ma, Y., & Yoshie, O. (2021). *PP-YOLOv2: A Practical Object Detector*. 1–7. http://arxiv.org/abs/2104.10419

Hwang, G. J., & Tsai, C. C. (2011). Research trends in mobile and ubiquitous learning: A review of publications in selected journals from 2001 to 2010. *British Journal of Educational Technology*, *42*(4), 65–70. https://doi.org/10.1111/j.1467-8535.2011.01183.x

Junos, M. H., Mohd Khairuddin, A. S., Thannirmalai, S., & Dahari, M. (2021). An optimized YOLO-based object detection model for crop harvesting system. *IET Image Processing*, *15*(9), 2112–2125. https://doi.org/10.1049/ipr2.12181

Kaur, J., & Singh, W. (2022). Tools, techniques, datasets and application areas for object detection in an image: a review. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-022-13153-y

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Neural Information Processing Systems Conference*, *4*(25), 1097–1105. https://doi.org/10.1201/9781420010749

Kumar, A., Reddy, S. S. S. S., & Kulkarni, V. (2019). An Object Detection Technique For Blind People in Real-Time Using Deep Neural Network. *Proceedings of the IEEE International Conference Image Information Processing*, *2019-Novem*, 292–297. https://doi.org/10.1109/ICIIP47207.2019.8985965

Kumar, A., Zhang, Z. J., & Lyu, H. (2020). Object detection in real time based on improved single shot multi-box detector algorithm. *Eurasip Journal on Wireless Communications and Networking*, *2020*(1). https://doi.org/10.1186/s13638-020-01826-x

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. In *Nature* (Vol. 521, Issue 7553, pp. 436–444). Nature Publishing Group. https://doi.org/10.1038/nature14539

Li, R., & Yang, J. (2018). Improved YOLOv2 Object Detection Model. *International Conference on Multimedia Computing and Systems -Proceedings*, *2018-May*, 1–6. https://doi.org/10.1109/ICMCS.2018.8525895

Li, X., Lai, T., Wang, S., Chen, Q., Yang, C., & Chen, R. (2019). Feature Pyramid Networks for Object Detection. *Proceedings - 2019 IEEE Intl Conf on Parallel and Distributed Processing with*

*Applications, Big Data and Cloud Computing, Sustainable Computing and Communications, Social Computing and Networking, ISPA/BDCloud/SustainCom/SocialCom 2019*, 1500–1504. https://doi.org/10.1109/ISPA-BDCloud-SustainCom-SocialCom48970.2019.00217

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2020). Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 318–327. https://doi.org/10.1109/TPAMI.2018.2858826

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2020). Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, *128*(2), 261–318. https://doi.org/10.1007/s11263-019-01247-4

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9905 LNCS*, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

Lu, X., Li, B., Yue, Y., Li, Q., & Yan, J. (2019). Grid R-CNN. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2019-June*, 7355–7364. https://doi.org/10.1109/CVPR.2019.00754

Lu, X., Li, Q., Li, B., & Yan, J. (2020). MimicDet: Bridging the Gap Between One-Stage and Two-Stage Object Detection. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12359 LNCS*, 541–557. https://doi.org/10.1007/978-3-030-58568-6_32

Pathak, A. R., Pandey, M., & Rautaray, S. (2018). Application of Deep Learning for Object Detection. *Procedia Computer Science*, *132*(Iccids), 1706–1717. https://doi.org/10.1016/j.procs.2018.05.144

Potdar, K., Pai, C. D., & Akolkar, S. (2018). *A Convolutional Neural Network based Live Object Recognition System as Blind Aid*. http://arxiv.org/abs/1811.10399

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 779–788. https://doi.org/10.1109/CVPR.2016.91

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-Janua*, 6517–6525. https://doi.org/10.1109/CVPR.2017.690

Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. http://arxiv.org/abs/1804.02767

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, *115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, *2*(6), 1–20.

https://doi.org/10.1007/s42979-021-00815-1

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0197-0

Srivastava, S., Divekar, A. V., Anilkumar, C., Naik, I., Kulkarni, V., & Pattabiraman, V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00434-w

Szegedy, C., Toshev, A., & Erhan, D. (2013). Deep Neural Networks for object detection. *Advances in Neural Information Processing Systems*, 1–9.

Tan, L., Huangfu, T., Wu, L., & Chen, W. (2021). Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. *BMC Medical Informatics and Decision Making*, *21*(1), 1–11. https://doi.org/10.1186/s12911-021-01691-8

Tian, Z., Shen, C., Chen, H., & He, T. (2019). FCOS: Fully convolutional one-stage object detection. *Proceedings of the IEEE International Conference on Computer Vision*, *2019-Octob*, 9626–9635. https://doi.org/10.1109/ICCV.2019.00972

Xiao, Y., Wang, X., Zhang, P., Meng, F., & Shao, F. (2020). Object detection based on faster r-cnn algorithm with skip pooling and fusion of contextual information. *Sensors (Switzerland)*, *20*(19), 1–20. https://doi.org/10.3390/s20195490

Xie, H., Chu, H. C., Hwang, G. J., & Wang, C. C. (2019). Trends and development in technology-enhanced adaptive/personalized learning: A systematic review of journal publications from 2007 to 2017. *Computers and Education*, *140*(June), 103599. https://doi.org/10.1016/j.compedu.2019.103599

Zhang, S., Wen, L., Bian, X., Lei, Z., & Li, S. Z. (2018). Single-Shot Refinement Neural Network for Object Detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 4203–4212. https://doi.org/10.1109/CVPR.2018.00442

Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object Detection with Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212–3232. https://doi.org/10.1109/TNNLS.2018.2876865

Zhou, Z., Lai, Q., Ding, S., & Liu, S. (2021). Novel Joint Object Detection Algorithm Using Cascading Parallel Detectors. *Symmetry*.

Zhu, C., He, Y., & Savvides, M. (2019). Feature selective anchor-free module for single-shot object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2019-June*, 840–849. https://doi.org/10.1109/CVPR.2019.00093

Zhu, M., Hu, G., Li, S., Zhou, H., Wang, S., & Feng, Z. (2022). A Novel Anchor-Free Method Based on FCOS + ATSS for Ship Detection in SAR Images. *Remote Sensing*, *14*(9). https://doi.org/10.3390/rs14092034

Zhu, X., Lyu, S., Wang, X., & Zhao, Q. (2021). TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. *Proceedings of the IEEE International Conference on Computer Vision*, *2021-Octob*, 2778–2788. https://doi.org/10.1109/ICCVW54120.2021.00312