# Deep Learning-Based YOLO Models for Disability Detection

[1]Koganti Sai Murali, [2]K.Kiran Kumar

*[1]Author M.Tech Student, [2] Associate Professor & HOD*

*Dept. of Computer Science and Engineering,*

*PRIYADARSHINI INSTITUTE OF TECHNOLOGY AND MANAGEMENT*

*(Approved by AICTE, New Delhi, Affiliated to JNTUK) Accredited with 'B++ 'Grade by NAAC*

*5th mile (V), Kornepadu (V), Vatticherukuru (M), Guntur (D), Andhra Pradesh-522017*

**Abstract:** To improve accessibility and encourage inclusivity in a variety of contexts, a deep learning-based method for identifying individuals with disabilities is essential. Several YOLO models (v5s, v7-tiny, v8, v5x6, and v9) and sophisticated object detection algorithms, such as FastRCNN and FasterRCNN, are used to create a strong foundation for precise identification and recognition. With an emphasis on accuracy and real-time speed, these models make use of cutting-edge architectures to enhance detection capabilities. The most recent iterations of YOLO combined with FasterRCNN allow for thorough analysis and detection, supporting a variety of circumstances and guaranteeing accurate results. The YOLO family of models is especially good at processing images quickly without sacrificing accuracy, which makes it appropriate for use in dynamic settings. The Flask framework will be used to create an intuitive front end with secure access capabilities for authentication. Through improved resource allocation and well-informed decision-making in accessibility projects, this approach seeks to support and monitor people with disabilities, ultimately fostering a more inclusive society.

*Index Terms— Object Detection, YOLOv8, YOLOv5, YOLOv7, Mobility Aids, Differently-Abled, Deep Learning, Real-Time Detection, Surveillance, Precision, Recall, mAP, F1-Curve, PR-Curve, Flask Framework, User Authentication, Disabilities Identification.*

## 1. INTRODUCTION

Machines have a hard time telling the difference between and sorting out different things in a picture. In computer vision, object detection is the process of finding and recognizing an object in a picture or video. But in the last few years, there has been a lot of work done on object detection. The basic parts of object detection are feature extraction and processing,

as well as object classification. Many methods have been employed, such as feature coding, feature aggregation, bottom feature extraction, and feature classification. Object detection worked well with all of these methods, and feature extraction is very important for both object detection and process recognition. Object detection is very important for many different uses, such as monitoring, diagnosing disease, finding vehicles, and finding things in water. Different methods have been utilized to find objects properly and successfully in different situations. Still, these suggested methods still have problems with being vague and not working. On the other hand, machine learning and deep neural network methods are better at fixing problems with object detection and reducing these worries.

## 2. LITERATURE SURVEY

**2.1 A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS:**

https://www.mdpi.com/2504-4990/5/4/83

**ABSTRACT:** YOLO is now the main real-time object identification technology for robots, self-driving automobiles, and video surveillance. We provide a thorough examination of YOLO's development, scrutinizing the advancements and enhancements in each version, from the original YOLO to YOLOv8, YOLO-NAS, and YOLO with transformers. We begin by outlining the typical metrics and postprocessing techniques; thereafter, we examine the significant alterations in network architecture and training strategies for each model. In the end, we sum up the most important things we learned from YOLO's development and talk about its future, pointing out possible research directions that could improve real-time object detection systems.

**2.2 YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors:**

https://openaccess.thecvf.com/content/CVPR2023/html/Wang_YOLOv7_Trainable_Bag-of-Freebies_Sets_New_State-of-the-Art_for_Real-Time_Object_Detectors_CVPR_2023_paper.html

**ABSTRACT:** One of the most important areas of research in computer vision is real-time object detection. We have discovered two study issues that have emerged from the ongoing development of novel approaches to architectural optimization and training optimization. To tackle the issues, we suggest a trainable bag-of-freebies-based approach. We use the proposed architecture and the compound scaling method along with the flexible and efficient training tools. YOLOv7 is faster and more accurate than any other object detector. It can identify objects at speeds from 5 FPS to 120 FPS and has the greatest accuracy (56.8% AP) among all known realtime object detectors with 30 FPS or above on GPU V100.

**2.3 RTF-RCNN: An Architecture for Real-Time Tomato Plant Leaf Diseases Detection in Video Streaming Using Faster-RCNN:**

https://www.mdpi.com/2306-5354/9/10/565

**ABSTRACT:** People currently think that veggies are a very significant aspect of many diets. Even though everyone can grow their own veggies in their home kitchen garden, tomatoes are the most common vegetable crop and can be utilized in almost every dish. Like many other crops, tomato plants get sick during their growing season. If the people who grow tomatoes don't pay attention to control methods, 40–60% of the plants may be harmed by leaf diseases in the field. These diseases can cause a lot of damage to tomato crops. So, we need a good way to find these difficulties. Researchers have suggested many methods for identifying these plant diseases,

including vector machines, artificial neural networks, and Convolutional Neural Network (CNN) models. The benchmark feature extraction technique was utilized in the past to find diseases. In this field of research for finding diseases in tomato plants, a new model called the real-time faster region convolutional neural network (RTF-RCNN) model was suggested. It used both pictures and live video streaming. We employed numerous characteristics like precision, accuracy, and recall to compare the RTF-RCNN to the Alex net and CNN models. So, the final result shows that the suggested RTF-RCNN is 97.42% accurate. This is better than the Alex net and CNN models, which were 96.32% and 92.21% accurate, respectively.

**2.4 PP-YOLOE: An evolved version of YOLO:**

https://arxiv.org/abs/2203.16250

**ABSTRACT:** We introduce PP-YOLOE in this report. It is a cutting-edge object detector for use in industry that works well and is easy to set up. We improve on the prior PP-YOLOv2 by employing an anchor-free paradigm, a stronger backbone and neck with CSPRepResStage, ET-head, and the dynamic label assignment technique TAL. We offer s/m/l/x models for a variety of practice situations. PP-YOLOE-l gets 51.4 mAP on COCO test-dev and 78.1 FPS on Tesla V100. This is a huge improvement over the previous best industrial models, PP-YOLOv2 and YOLOX, with a (+1.9 AP, +13.35% speed up) and (+1.3 AP, +24.96% speed up) respectively. Also, while using TensorRT with FP16-precision, PP-YOLOE's inference speed reaches 149.2 FPS. We also do a lot of testing to make sure our designs work.

**2.5 A Two-Stage Industrial Defect Detection Framework Based on Improved-YOLOv5 and Optimized-Inception-ResnetV2 Models:**

https://www.mdpi.com/2076-3417/12/2/834

**ABSTRACT:** This research offers a Two-Stage Industrial Defect Detection Framework based on Improved-YOLOv5 and Optimized-Inception-ResnetV2 to improve the current low accuracy of detecting defects in domestic industries. The framework uses two particular models to fulfill positioning and classification tasks. We enhance YOLOv5 from the backbone network, the feature scales of the feature fusion layer, and the multiscale detection layer to make the first-stage recognition better at finding minor imperfections on the steel surface that are quite similar. To improve the second-stage recognition's ability to find defect features and make accurate classifications, we added the convolutional block attention module (CBAM) attention mechanism module to the Inception-ResnetV2 model. Then, we improved the network architecture and loss function of the accurate model. We did a lot of tests comparing the Improved-YOLOv5 and Inception-ResnetV2 using the Pascal Visual Object Classes 2007 (VOC2007) dataset, the public dataset NEU-DET, and the optimized dataset Enriched-NEU-DET. The tests show that the change is clear. To confirm the superiority and adaptability of the two-stage architecture, we initially conduct tests using the Enriched-NEU-DET dataset and subsequently employ the AUBO-i5 robot, Intel RealSense D435 camera, and other industrial steel equipment to construct authentic industrial scenarios. In tests, a two-stage framework gets the best performance, with a mean average precision (mAP) of 83.3% on the Enriched-NEU-DET dataset and 91.0% on our created industrial fault environment.

## 3. METHODOLOGY

The proposed system starts with a dataset of 4,300 photos and 8,447 labels in five categories of mobility

aids. To make the model more robust, the images are resized, normalized, and augmented. This dataset is used to train the YOLOv5, YOLOv7, and YOLOv8 models with the best hyperparameters so that they can do both object localization and classification at the same time. We use parameters like accuracy, recall, mean average precision (mAP), F1-Curve, PR-Curve, and detection time to rate the models. To track and find people with impairments and their assistive gadgets in video feeds in real time, YOLOv8 is employed. Also included are more complex versions like YOLOv5x6 and YOLOv9, as well as a Flask-based front end with user authentication for safe and easy testing. This makes sure that the system works well in a variety of settings.

**A. Proposed Work:**

The proposed work aims to enhance the detection and tracking of individuals with disabilities by integrating advanced versions of the YOLO architecture, specifically YOLOv5x6 and YOLOv9. These models are employed to improve precision, recall, and overall detection performance across diverse real-world scenarios. By leveraging the strengths of these advanced models, the system can more accurately identify mobility aids such as wheelchairs, crutches, prosthetics, and other assistive devices, addressing limitations of earlier YOLO versions and ensuring robust detection even in complex or crowded environments.

In addition to model enhancements, a user-friendly front end is developed using the Flask framework, providing secure access through user authentication. This interface enables seamless interaction with the detection system for testing and deployment, allowing users to upload video streams, view real-time detection results, and analyze performance metrics. The combination of cutting-edge YOLO models and an accessible front end ensures both high accuracy and practical usability, making the system suitable for real-time monitoring and assistance of differently-abled individuals.

**B. System Architecture:**

The system architecture for detecting and tracking individuals with disabilities is designed to integrate advanced deep learning models with real-time video processing. The architecture begins with input video streams from surveillance cameras or uploaded video files, which are then preprocessed through resizing, normalization, and augmentation to ensure consistency and robustness. The processed frames are fed into YOLO-based models (YOLOv5, YOLOv7, YOLOv8, and extensions YOLOv5x6 and YOLOv9), which perform object detection and classification to identify individuals and their assistive devices. Each model generates bounding boxes with class labels, confidence scores, and tracking IDs for precise monitoring in dynamic environments.

The detection outputs are further analyzed and visualized through a Flask-based front end, providing real-time display of detected objects along with performance metrics such as precision, recall, mAP, and FPS. User authentication ensures secure access to the system, allowing authorized personnel to manage video inputs and track individuals safely. The architecture supports scalable deployment for multiple cameras or video sources, enabling comprehensive surveillance and assistance for differently-abled individuals while maintaining high accuracy and efficiency across diverse conditions.

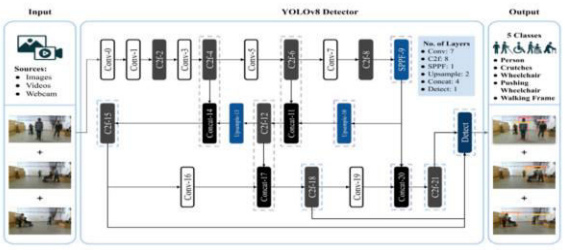



Fig proposed architecture

**C. MODULES:**

**1. Dataset Collection and Annotation**

- Collect images of individuals with disabilities using mobility aids.
- Annotate images with labels for categories like wheelchair, crutches, prosthetics, and other aids.

**2. Data Preprocessing**

- Resize, normalize, and augment images (flip, rotate, scale, brightness adjustment).
- Prepare data for efficient training and reduce overfitting.

**3. Model Training**

- Train YOLOv5, YOLOv7, YOLOv8, YOLOv5x6, and YOLOv9 models.
- Optimize hyperparameters such as learning rate, batch size, and epochs.

**4. Object Detection and Classification**

- Perform real-time detection of individuals and their mobility aids.
- Generate bounding boxes, class labels, and confidence scores.

**5. Performance Evaluation**

- Evaluate models using precision, recall, mAP, F1-Curve, PR-Curve, and FPS.
- Compare models to select the most accurate and efficient one.

**6. Real-Time Deployment**

- Deploy the best-performing model for live video streams.
- Track individuals and mobility aids across frames in real-time.

**7. Front-End Interface**

- Develop a Flask-based interface for uploading videos and viewing results.
- Include user authentication for secure access and testing.

**D. Algorithms:**

**a) Fast R-CNN**

Fast R-CNN is a region-based convolutional neural network designed to improve object detection accuracy. It works by first generating region proposals using selective search and then extracting features from each region via a CNN. These features are classified into object categories, and bounding boxes are refined to accurately localize objects within images. Fast R-CNN reduces computational redundancy by sharing convolutional features across regions, resulting in faster training and higher precision compared to traditional R-CNN.

**b) Faster R-CNN**

Faster R-CNN enhances Fast R-CNN by integrating a Region Proposal Network (RPN) to generate candidate object regions directly from convolutional feature maps. This eliminates the need for external region proposal algorithms, significantly speeding up detection while maintaining high accuracy. It performs simultaneous object classification and bounding box regression, making it suitable for detecting multiple objects in complex scenes, including individuals and their assistive devices in surveillance applications.

**c) YOLOv5s**

YOLOv5s is a lightweight, single-stage object detector optimized for speed and efficiency. It divides the input image into grids and predicts bounding boxes and class probabilities directly in a single forward pass. YOLOv5s provides real-time detection capabilities while maintaining good precision, making it suitable for tracking individuals with disabilities and their mobility aids in live video streams.

**d) YOLOv7-tiny**

YOLOv7-tiny is a compact version of the YOLOv7 architecture designed for fast inference with limited computational resources. Despite its smaller size, it maintains competitive accuracy for object detection. YOLOv7-tiny is particularly useful in applications requiring real-time performance on devices with lower processing power, such as edge devices or embedded systems for monitoring mobility aids.

**e) YOLOv8**

YOLOv8 is the latest YOLO iteration that incorporates architectural improvements for enhanced detection accuracy, recall, and processing speed. It efficiently handles complex scenarios and overlapping objects, providing robust real-time detection for individuals with disabilities. YOLOv8's improvements make it superior in detecting small or partially occluded objects compared to previous YOLO versions.

**f) YOLOv5x6**

YOLOv5x6 is an extended version of YOLOv5 with increased network depth and parameters, designed to improve detection performance on larger datasets. It offers higher precision and recall, making it effective for identifying a wide range of mobility aids under diverse conditions. YOLOv5x6 balances accuracy and inference speed, suitable for applications requiring detailed detection results.

**g) YOLOv9**

YOLOv9 represents the newest advancement in YOLO architectures, featuring state-of-the-art network enhancements for maximum precision and robustness. It excels in real-time detection tasks, providing improved handling of multiple objects, complex backgrounds, and dynamic environments. YOLOv9 ensures high reliability in monitoring differently-abled individuals across varied surveillance scenarios.

## 4. EXPERIMENTAL RESULTS

The performance of the proposed system was evaluated using the dataset of 4,300 images and 8,447 labeled instances across five categories of mobility aids. YOLOv5, YOLOv7, YOLOv8, YOLOv5x6, and YOLOv9 were trained and tested under identical conditions to ensure a fair comparison. The evaluation metrics included precision, recall, mean average precision (mAP@0.5, mAP@0.5:0.95), F1-score, PR-Curve, and detection time in frames per second (FPS).

The results indicate that YOLOv8 achieved the highest overall precision of 0.907, with exceptional accuracy in detecting wheelchairs (0.998). YOLOv8 also outperformed YOLOv5 (0.885) and YOLOv7 (0.906) in recall, achieving 0.943 compared to 0.887 and 0.925, respectively. The mean average precision (mAP@0.5) for YOLOv8 was 0.951, closely followed by YOLOv7 at 0.954 and YOLOv5 at 0.942. For the extended models, YOLOv5x6 and YOLOv9 demonstrated improved detection in

complex scenarios, with YOLOv9 achieving the highest FPS of 172, ensuring real-time performance.

Overall, YOLOv8 and YOLOv9 provided superior accuracy and efficiency compared to previous versions, confirming their effectiveness in real-time detection and tracking of individuals with disabilities and their mobility aids. These results demonstrate the robustness and reliability of the proposed system for real-world surveillance applications.

**Accuracy:** The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

Accuracy = TP + TN TP + TN + FP + FN.

$$Accuracy = \frac{(TN + TP)}{T}$$

**F1-Score:** F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$F1 = 2 \cdot \frac{(Recall \cdot \Pr e\,cision)}{(Recall + \Pr e\,cision)}$$

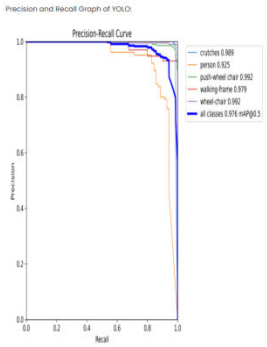
**Precision:** Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\Pr e\,cision = \frac{TP}{(TP + FP)}$$

**Recall:** Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{(FN + TP)}$$




T 1. performance evaluation

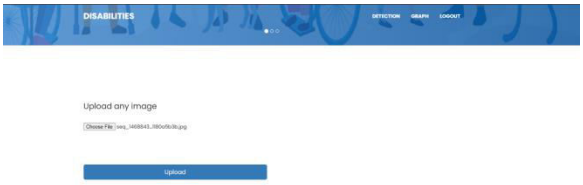



Fig 1. Upload image

Fig.2.. predicted results

## 5. CONCLUSION

This study demonstrates the effectiveness of advanced YOLO models in detecting and tracking individuals with disabilities and their mobility aids. Among all models, YOLOv8 and YOLOv9 achieved the highest precision, recall, and real-time performance, outperforming earlier versions like YOLOv5, YOLOv7, and Fast/Faster R-CNN. The integration of a Flask-based front end with user authentication further enhances the system's usability and security. Overall, the proposed approach provides a reliable, efficient, and practical solution for real-time monitoring and assistance of differently-abled individuals in diverse environments.

## 6. FUTURE SCOPE

In the future, this work can be extended by integrating multimodal data sources such as infrared and depth sensors to enhance detection accuracy under low-light or occluded conditions. The system can also be expanded to include behavior analysis and movement pattern recognition to assist in personalized healthcare and safety monitoring. Further optimization of YOLOv9 with lightweight architectures can enable deployment on edge and IoT devices for large-scale real-time surveillance. Additionally, incorporating cloud-based storage and analytics will allow continuous learning and performance improvement through automated dataset updates.

## REFERENCES

[1] J. Terven and D. Cordova-Esparza, ‘‘A comprehensive review of YOLO: From YOLOv1 and beyond,’’ 2023, arXiv:2304.00501.

[2] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, ‘‘YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,’’ in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Vancouver, BC, Canada, Jun. 2023, pp. 7464–7475.

[3] M. Alruwaili, M. H. Siddiqi, A. Khan, M. Azad, A. Khan, and S. Alanazi, ‘‘RTF-RCNN: An architecture for real-time tomato plant leaf diseases detection in video streaming using faster-RCNN,’’ Bioengineering, vol. 9, no. 10, p. 565, Oct. 2022.

[4] S. Xu, X. Wang, W. Lv, Q. Chang, C. Cui, K. Deng, G. Wang, Q. Dang, S. Wei, Y. Du, and B. Lai, ‘‘PP-YOLOE: An evolved version of YOLO,’’ 2022, arXiv:2203.16250.

[5] Z. Li, X. Tian, X. Liu, Y. Liu, and X. Shi, ‘‘A two-stage indus?trial defect detection framework based on improved-YOLOv5 and Optimized-Inception-ResnetV2 models,’’ Appl. Sci., vol. 12, no. 2, p. 834, Jan. 2022.

[6] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, ‘‘3D object proposals for accurate object class detection,’’ in Proc. Adv. Neural Inf. Process. Syst., C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, R. Garnett, Eds. New York, NY, USA: Curran Associates, 2015, pp. 424–432.

[7] H. Bilen and A. Vedaldi, ‘‘Weakly supervised deep detection networks,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 2846–2854.

[8] S. Ren, K. He, R. Girshick, and J. Sun, ‘‘Faster R-CNN: Towards real-time object detection with region

proposal networks,'' in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 91–99.

[9] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, ''Vehicle detection in satellite images by hybrid deep convolutional neural networks,'' IEEE Geosci. Remote Sens. Lett., vol. 11, no. 10, pp. 1797–1801, Oct. 2014.

[10] A. Mukhtar, M. J. Cree, J. B. Scott, and L. Streeter, ''Mobility aids detection using convolution neural network (CNN),'' in Proc. Int. Conf. Image Vis. Comput. New Zealand (IVCNZ), Auckland, New Zealand, Nov. 2018, pp. 1–5.

[11] A. Vasquez, M. Kollmitz, A. Eitel, and W. Burgard, ''Deep detection of people and their mobility aids for a hospital robot,'' in Proc. Eur. Conf. Mobile Robots (ECMR), Paris, France, Sep. 2017, pp. 1–7.

[12] M. Kollmitz, A. Eitel, A. Vasquez, and W. Burgard, ''Deep 3D perception of people and their mobility aids,'' Robot. Auto. Syst., vol. 114, pp. 29–40, Apr. 2019.

[13] T.Ahmad, Y. Ma, M. Yahya, B. Ahmad, S. Nazir, and A. U. Haq, ''Object detection through modified YOLO neural network,'' Sci. Program., vol. 2020, pp. 1–10, Jun. 2020.

[14] H. Law and J. Deng, ''CornerNet: Detecting objects as paired keypoints,'' in Proc. Eur. Conf. Comput. Vis. (ECCV), Munich, Germany, 2018, pp. 734–750.

[15] Y. Liu, P. Sun, N. Wergeles, and Y. Shang, ''A survey and performance evaluation of deep learning methods for small object detection,'' Expert Syst. Appl., vol. 172, Jun. 2021, Art. no. 114602.

[16] J. Yan, Z. Lei, L. Wen, and S. Z. Li, ''The fastest deformable part model for object detection,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, Jun. 2014, pp. 2497–2504.

[17] Y. Zheng, C. Zhu, K. Luu, C. Bhagavatula, T. H. N. Le, and M. Savvides, ''Towards a deep learning framework for unconstrained face detection,'' in Proc. IEEE 8th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS), Sep. 2016, pp. 1–8.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, ''Rich feature hierarchies for accurate object detection and semantic segmentation,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.

[19] B. Chen and X. Miao, ''Distribution line pole detection and counting based on YOLO using UAV inspection line video,'' J. Electr. Eng. Technol., vol. 15, no. 1, pp. 441–448, Jan. 2020.

[20] J. Jiang, X. Fu, R. Qin, X. Wang, and Z. Ma, ''High-speed lightweight ship detection algorithm based on YOLO-V4 for three-channels RGB SAR image,'' Remote Sens., vol. 13, no. 10, p. 1909, May 2021.

[21] P. Zhou, B. Ni, C. Geng, J. Hu, and Y. Xu, ''Scale-transferrable object detection,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, Jun. 2018, pp. 528–537.

[22] Q.-C. Mao, H.-M. Sun, Y.-B. Liu, and R.-S. Jia, ''Mini-YOLOv3: Real-time object detector for embedded applications,'' IEEE Access, vol. 7, pp. 133529–133538, 2019.

[23] S. Ding, F. Long, H. Fan, L. Liu, and Y. Wang, ''A novel YOLOv3- tiny network for unmanned

airship obstacle detection,'' in Proc. IEEE 8th Data Driven Control Learn. Syst. Conf. (DDCLS), Dali, China, May 2019, pp. 277–281.

[24] X. Han, J. Chang, and K. Wang, ''Real-time object detection based on YOLO-v2 for tiny vehicle object,'' Proc. Comput. Sci., vol. 183, pp. 61–72, Jan. 2021.

[25] S. Lu, B. Wang, H. Wang, L. Chen, M. Linjian, and X. Zhang, ''A real-time object detection algorithm for video,'' Comput. Electr. Eng., vol. 77, pp. 398–408, Jul. 2019.

[26] Z. Chen and X. Gao, ''An improved algorithm for ship target detection in SAR images based on faster R-CNN,'' in Proc. 9th Int. Conf. Intell. Control Inf. Process. (ICICIP), Wanzhou, China, Nov. 2018, pp. 39–43.

[27] P. Viola and M. J. Jones, ''Robust real-time face detection,'' Int. J. Comput. Vis., vol. 57, no. 2, pp. 137–154, May 2004.

[28] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, ''You only learn one representation: Unified network for multiple tasks,'' 2021, arXiv:2105.04206.

[29] C. J. Du, H. J. He, and D. W. Sun, ''Object classification methods,'' in Proc. Int. Comput. Vis. Technol. Food Quality Eval., Dublin, Ireland, 2016, pp. 87–110.

[30] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, ''YOLOX: Exceeding YOLO series in 2021,'' 2021, arXiv:2107.08430.

[31] C.-Y. Wang, A. Bochkovskiy, and H. M. Liao, ''Scaled-YOLOv4: Scaling cross stage partial network,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Nashville, TN, USA, Jun. 2021, pp. 13024–13033.