The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114

VISION SENSE AI: Enhanced Object Detection and Recognition Using Deep Learning

Ms.A.Anusha[1], M.Deepika[2], P.Durga prasad[3], S.Venkatesh[4],Y.Abhiram Sai Manikanta [5]

[1, 2,3, 4, 5] Student of Department of CSE, NSRIT, Vishakhapatnam, India

Corresponding Author: 22nu1a0588@nsrit.edu.in

Abstract

Arti One other way of developing an AI-based recognition system of supermarket goods could use Vision transformers (ViTs) which have actually developed a lot of promise in visual image recognition tasks. Unlike the conventional CNN networks, the transformer models transform the visual input images with the self-attention mechanism focusing on more important parts of the image, making them robust to distinguishing features especially in cluttered environments such as supermarket shelves. To develop the model in the furtherence of this method. A py torch environment is configured for model management and model training purposes. In the first instance, a large labeled dataset of different supermarket products is collected where data augmentation techniques (like random cropping and rotation) could be added to help model generalization.

In order to develop the model, a pretrained Vision transformer which considers various animal images from the image net database is retrained to fit exactly to mediating recognition of super market products. Using this transfer learning, the model is able to take advantage of general extract features and only subject the new data set to specific features. As it has high expectation to identify different animals and it has the capability of sensing complex features, the Vision transformer is trained and it learns to identify objects little error. After training is completed, the model performance is tested on real-life supermarkets and then compared with the public datasets of supermarkets, which have been ableness of the model.

Keywords

Artificial Intelligence (AI), Deep Learning, Vision Transformers (ViTs), PyTorch Environment, PostPandemic Era, Computer Vision

1. Introduction

The application of object detection is controversial in the sense that it can be argued that it addresses the area of computer vision which itself is broad. In the beginning, there was a

The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114

dominance of two-stage object detectors which relied on machine learning (ML) and deep learning (DL) Models for performance. However, following years of research efforts with ML/DL led algorithms, a new competitor surfaced in the form of single-stage object detectors

which in some cases out stripped two-stage models. Significantly, the development of new models which incorporate revised logic particularly the YOLO models have changed the landscape for object detection. This has made it possible to write YOLO-specific algorithms in the sense that they include revisions of the design philosophy, which are further optimized by successor models that compete with two-stage detectors. This section addresses briefly the theoretical aspects of deep learning and computer vision, explains relevant terminology, and highlights challenges, steps and roles of detection algorithms. It also advances the history of the most common object detection algorithms, datasets often referenced in other works and the contribution of this synthesis.

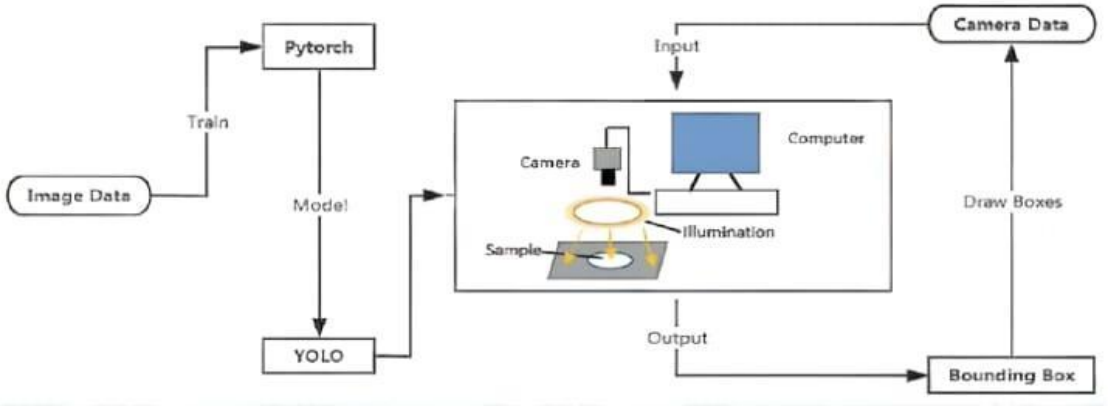
2. Methodology

2.1. Deep learning and computer vision

Deep learning made its debut around 2000 as support vector machines, multilayer perceptions, and artificial neural networks were the precursors to deep learning. All these methods together formed a partial branch of this novel AI termed Machine learning. It is imperative to note that it garnered very little interest before due to the computational power it required. However, after 2006, deep learning gained a lot of interest and surpassed many other algorithms due to two reasons. Available data to train models increased drastically, and on the other hand powerful computing technologies surfaced. Deep Learning can now further be of assistance to weather forecasting, stock market prediction as also to speech, object and character recognition, intrusion detection, time series forecasting, biological data analysis and a whole lot more models.

Vision as a part of artificial intelligence has rapidly gained traction as it has various applications. In basic literature, this is the area which allows vision-based approaches to be realized with high accuracy.Sub Neural Networks (CNN), Deep Belief Networks (DBN), Deep Boltzmann Machines (DBM), Restricted Boltzmann Machines (RBM), and Stacked Autoencoders.

The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114

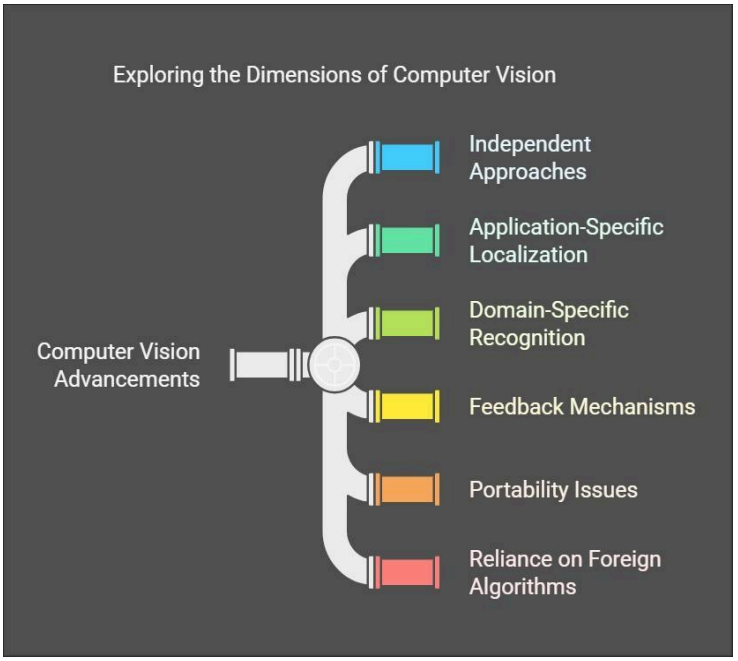


2.2. Object classification and localization

The process of image classification as the name suggests involves classifying an image or its constituent objects into one out of several predefined classes. This problem is usually solved using supervised machine learning or deep learning, where the model learns from large amounts of labeled data. Popular algorithms employed in machine learning for image categorization include the Artificial Neural Networks (ANN), Support Vector Machine (SVM), Decision trees, and k-nearest neighbours algorithm(s). In contrast, convolutional neural networks and their various architectures/versions are the most popular models that are being used for image classification within deep learning. Besides basic ML and DL methods, some other methods, such as fuzzy logic and genetic algorithm have also been implemented in some image classification tasks.

The term object localization refers to the process of determining the location of one or multiple objects in an image, as depicted by the rectangular bounding box around the object. Image segmentation, on the other hand, is the process of partitioning an image into different regions or segments, with each segment representing either a whole object or a portion of an object. Segmentation mainly helps to determine the edges, lines and curves within the image. Pixels within the segmented regions often are characterize.

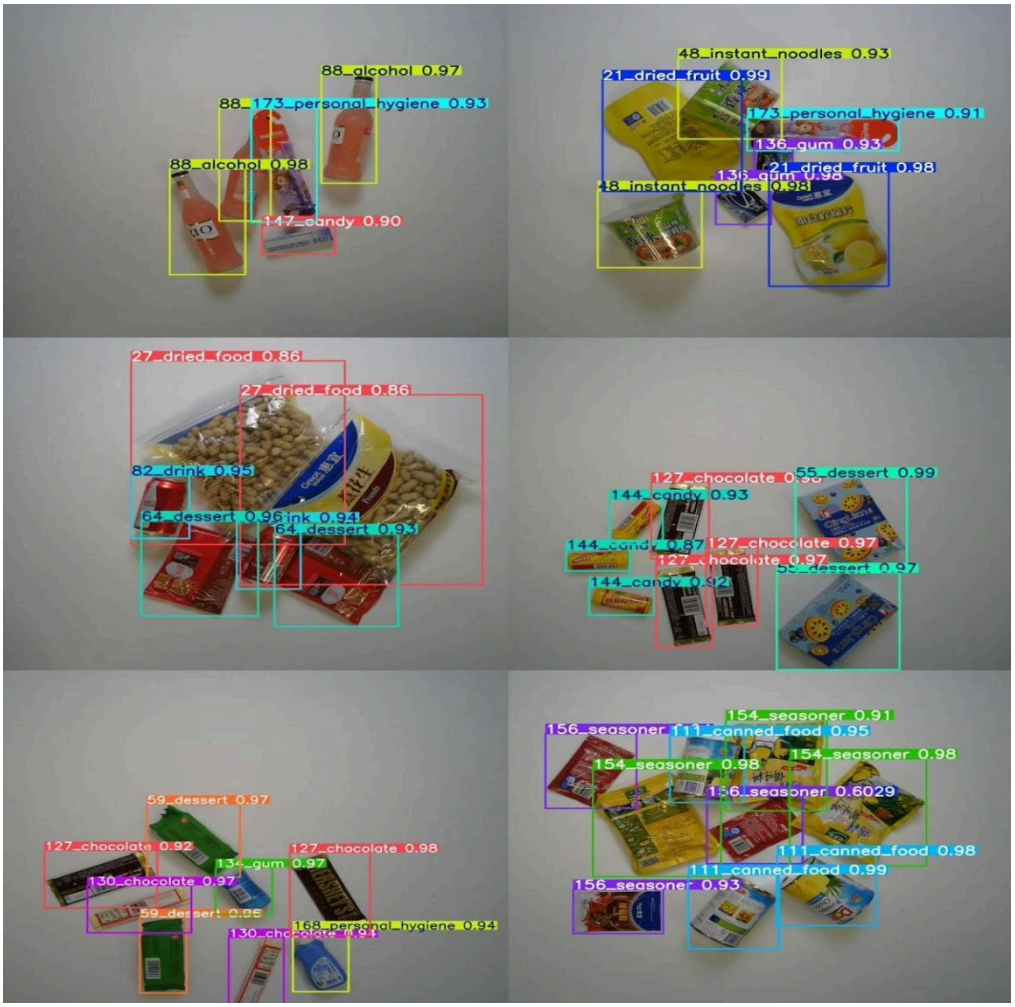
The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114



2.3. Growth of Unmanned Supermarkets

Computer vision can be called as an overlapping domain which takes information from biology, psychology, engineering, and mathematics. Models formulated in this area depend a lot on the development and ideas stemming from these areas. Starting from early 21st century, China turned into an economy where the Internet had a dominant role, which took place parallel to the large-scale construction of high-speed network infrastructure. This advancement has laid a firm groundwork for the achievement of a range of a variety of online services and features. Aside from the direct improvements, the forward momentum of the Internet has seen China make further progress as well in developing a society that is cashless, so that a great number of citizens instead of using a regular credit card have directly used transactions through virtual money - a transformation which has been enabled by widespread societal acceptance and collective effort.This transformation has spurred the development of various innovations, including unmanned supermarkets, which have seen rapid growth. These supermarkets typically occupy small spaces and operate through self-service, with no need for cashiers, as the entire shopping and checkout process is automated. Such shopping models are expected to further evolve and mature in the coming years, and there is widespread societal optimism about their future development.

The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114



2.4. Advancements and Challenges in Computer Vision: OpenCV's Role

The rapid advancement of information science has propelled computer vision within the broader field of artificial intelligence, demonstrating significant growth and potential. Key developments in this area include:

1.Independent Approaches in Target Vision: At present, machine learning-based target vision and 3D spatial vision continue to be largely independent. Deep learning, while highly effective, is not yet poised to fully replace geometric vision techniques in the short term.

2.Shift Toward Application-Specific Target Localization: The focus of target localization in computer vision is shifting toward "application research," with an increasing emphasis on multi-sensor fusion technologies for more precise localization.

3.Domain-Specific Object Recognition: Object recognition through deep learning is evolving from general recognition methods to more specific domain-based recognition. By leveraging prior knowledge tailored to specific domains, these methods can significantly improve both the accuracy and efficiency of object detection.

4.Incorporating Feedback Mechanisms in Deep Networks: Future research is expected to focus on enhancing deep network architectures with feedback mechanisms, allowing for continuous improvement and adaptation of the network.

Currently, many popular Computer vision programs prioritize the acceleration of image processing speeds. However, several challenges remain:

1.Portability Issues: While portability is crucial for future technical advancements, many domestic software solutionsstill lack the support for this feature, limiting their adaptability.

2.Reliance on Foreign Algorithms: Most core algorithms used in existing computer vision software have been developed by international researchers, and there is limited indigenous development of proprietary algorithms.

3.Infrastructure Constraints: Many existing network servers are unable to meet the high computational demands of computer vision applications, restricting many functions to local systems. Since computer vision requires processing large volumes of pixel data, it also demands powerful processors and high bandwidth.

To address these challenges, Intel Corporation introduced **OpenCV,** a versatile and portable computer vision library compatible with various platforms. OpenCV supports development across mainstream operating systems and offers programming interfaces for languages like Python and Matlab. It also includes standardized algorithms for pixel processing in various scenarios, making it a valuable tool for overcoming current limitations in the field.

3.System architecture and validation

We trained the model based on the collected data, obtained the training file of the model using YOLO in the PyTorch environment, and used the model for commodity detection and identification in the actual production environment.

1. Enhancing Object Detection Accuracy through Dataset Expansion and Augmentation Techniques

The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114

The performance of an object detection model during training is dependent on the size of the training dataset. In cases where the dataset is too small, models are highly likely to overfit: presenting low errors in training but higher errors in testing as a result of limited variety in data. To address this, we started with a base dataset of 1,000 items, each photographed in 12 different angles. To expand the dataset and improve its generalizability, we applied augmentations via a simple Python script.Four operations were used for augmentation: 90°, 180°, and 270° rotations along with horizontal flipping, thus allowing to increase the original dataset size five times. That resulted in 60,000 training images linearly computed as;

1. 1,000 items×12 images per item×5 transformations=60,000 images.

2. Alternatively, along with rotating fixed into angle rotations and flips, more complex augmentations such as random cropping, color jittering, and scaling would yield further diversity in the dataset. Other than general data augmentation, methods for synthetic data generation or the use of models already trained could also help boost accuracy if there is little initial data available.

2. Automated Labeling and Barcode Tagging for Training Dataset Management

After creating the training set, it was necessary to generate label tags for each item. During dataset preparation, barcode labels for each item were collected and associated with the corresponding images. Using Python scripts, the 60,000 images were named according to their commodity codes. A TXT file was generated, with each line listing the file path of an image alongside the barcode label for the corresponding item.

3. Automated Labeling and Barcode Tagging for Training Dataset Management

Our test set aims to accurately assess the recognition performance of the trained model. Therefore, a valid test implies that the test set images are unique, not overlapping with those in the training set. Using a consistent python script, images are first shuffled and put into a training and a test set at a ratio of 70:30 between each other. This system of distribution is done using built-in python randomization functions, which yield a substantially fair amount for a proper testing process.

4. Testing process

The whole testing procedure is trivial: an environment has been synthesized in preparation for a run; all that needs to be done is run the test program after training produces the weights file. Training Process Parameters-The iteration number indicated how often the training program updates the weights file. This measure indicated the number of examined objects of a certain class that were correctly detected as opposed to the total objects of the kind in the test set. The calculation is as follows:

{Recall} = {tp} / {tp + fn},

where "tp" in the equation stands for True Positives, whereas "fn" describes False Negatives

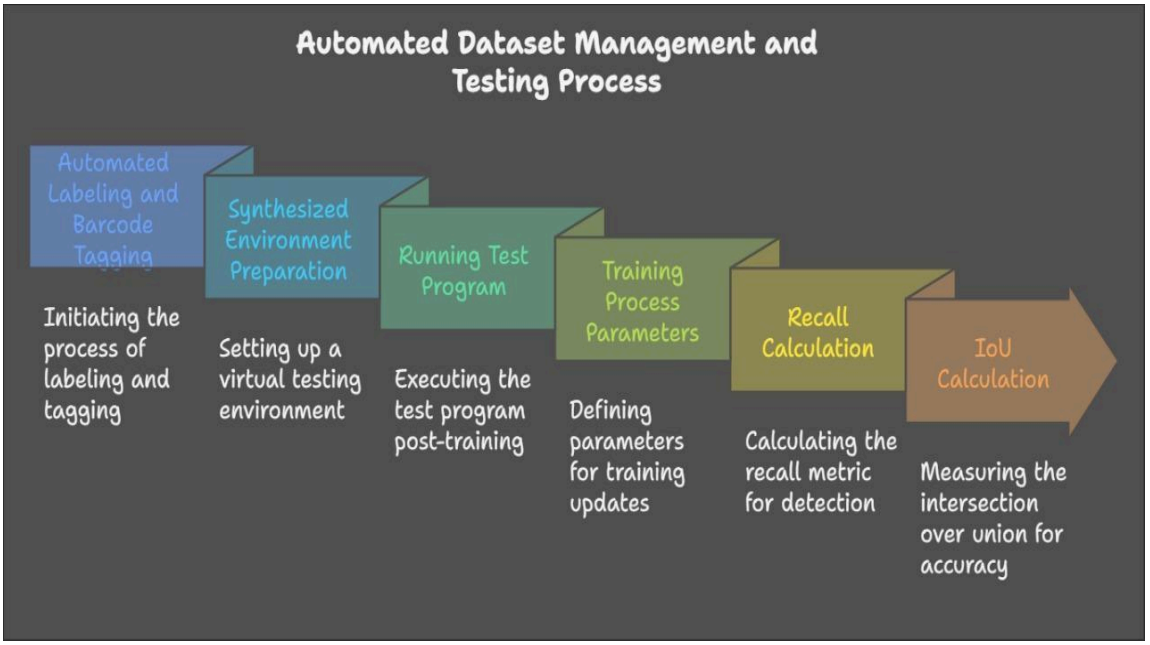
The Journal of Computational Science and Engineering (TJCSE)
ISSN 2583-9055 (Media Online)
Vol 3, No 2, February 2024
PP 104-114

corresponding to the total number of a certain type of object.

*IoU***: This metric quantifies the overlap between the predicted object location produced from the network and the manually labeled object position. The IoU is then computed as the set operation of intersection and union of Detection Result and Ground Truth:

{IoU}={DetectionResult} intersection {GroundTruth}/{DetectionResult} union {G roundTruth}:

where the intersection represents the predicted true and false object positions and the denominator is the union.



5. Conclusion

It is a ratio of True Positives (tp) to total number of predictions (True Positives+False Positives), respectively, saying the accuracy of predicted object locations. Mathematically it can be written as:

{Precision}=tp/(tp+fp)

The number of good boxes predicted in truth. After processing the image data, IMGnet predicts bounding boxes for different kinds of objects. That prediction is compared to the GT for the object, and then the IoU will be computed. If that max IoU value surpasses a preset threshold, this counts as a correct prediction. This means Region Proposals per Image, which indicates how many average frames are predicted per map. The threshold set in YOLO_RECALL function is 0.001. This takes care of more boxes being called and, hence,

boxes that aren't really objects. So, the recall of the positive samples increases and thereby improves recognition and hence detection of the object in an image. This makes the test set really a check of the performance of the model, like IoU, Recall, and Precision: three upright measures needed for object detection performance assessment.

In supermarket systems, self-service shopping occupies the central part of commodity detection tasks. The present article concentrates on target detection by employing computer vision and the YOLO object recognition algorithm. Developed in Python, this system lends itself to simplicity and readability, and while Python is slower in terms of program execution than lower-level languages, it is no longer of that much consequence in this age of rapidly changing computational power. Besides, the ease of editing and understanding Python continues to carry lot of goodwill. Yet another important aspect of this system is the pragmatic deployment of technologies. Computer vision, integral to AI, is more and more dovetailing with natural language processing (NLP). This conjunction is supposed to artificially increase productivity in comparison with humans across orchids of industries. The flip side may however be automation of the human routine of doing work, causing them to cease from the work at all in the future. Today, some of the largest challenges facing computer vision are trade-offs between recognition accuracy and speed. The recognition processes may be accurate, but improving the speed of recognition processes is a challenging endeavor. However, it is expected future advancements will dramatically impact recognition speed allowing for change-detection in human facial expressions and deducing emotions from psychological models. Underneath the priority given to information technology by China, information technology is bound to play a more increased role in the economic development across all sectors in the future.

6. Acknowledgment

We deeply express our gratitude to all who contributed to the successful development of this project We, first of all, thank Ms. A. Anusha, Assistant Professor, for her invaluable guidance and support throughout the project. The completion of the project would not have been possible without her knowledge and constructive feedback in shaping the system into a stable and functional model. We also thank the team members M. Deepika, P. Durga Prasad, S. Venkatesh, and V. Abhiram Sai Manikanta for their contributions and support in accomplishing the objectives. The results that came out extremely well in making this complex system were achieved through teamwork and effort.

This should also be an opportunity to express our gratitude to the Department of Computer Science for supporting the completion of this project by providing generous facilities and resources.

7. References

Mask R-CNN (2017). This paper added a branch for predicting segmentation masks beside box regression and binary segmentation for instance segmentation. Single Shot MultiBox

Detector (SSD) (2016). It is an efficient object detection method that generates several bounding boxes for every object and performs detection in a single pass through the network, hence having a good balance between speed and accuracy. Focal Loss for Dense Object Detection by RetinaNet (2017). RetinaNet introduces a new function, called focal loss, which helps mitigate the class imbalance problem so frequent in dense object detection tasks. This was a significant advancement toward improving the performance of object detectors.

YOLOv4: Optimal Speed and Accuracy of Object Detection (2020). YOLOv4 brings several improvements over YOLOv3 using optimization techniques and clever architectures, with a better trade-off between speed and accuracy. SqueezeDet: Low Power Network for Real-Time Object Detection (2017). The lightweight deep learning model that SqueezeDet is designed for real-time object detection for edge or mobile devices that operate with limited computational resources. Faster R-CNN with R-FCN: A Benchmark for Object Detection (2016). The study describes RFCN (Region-based Fully Convolutional Network) method that enhances the use of positionsensitive score maps for efficient object detection against Faster R-CNN. DeepLabv3+: Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. DeepLabv3+ brings state-of-the-art semantic segmentation due to atrous convolutions and an encoder-decoder structure that integrates rich contextual information to capture distinguished features.

References

1. YOLOv5: The New Era in Object Detection (2020). While not officially from the original authors of YOLO, YOLOv5 gained skiing popularity, mainly due to its speed and easy approach. This implementation has decent performance in real-time, offering numerous optimizations and smaller models in favor of edge devices.
2. DenseNet:Densely Connected Convolutional Networks (2017). DenseNet presents DenseNet, which has dense connections, enhancing the information flow between layers, which can be implemented in object detection networks for enhanced performance with fewer parameters.
3. EfficientDet: Scalable and Efficient Object Detection (2020). EfficientDet is an object detection framework that employs a compound scaling method to balance network depth, width, and resolution, allowing it to achieve a high accuracy-to-complexity ratio while requiring very little computational resources.
4. D Object Detection for Autonomous Driving: A Review (2020). The review paper considers the 3D object detection techniques for autonomous driving, consisting of both LiDAR-based and RGB-based methods, pointed out specific challenges in 3D detection.
5. Deep Learning for Computer Vision: A Brief Review (2019). A review of the foundational principles of deep learning and its application in computer vision tasks, including object detection, image consideration, and segmentation.
6. 6Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.
7. .He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
8. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
9. .Zhang S, Chi C, Yao Y, Lei Z, Li S Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 9759-9768.

10. Dai J, Li Y, He K, Sun J. R-FCN: Object detection via region-based fully convolutional networks[C]//Advances in neural information processing systems. 2016: 379-387.
11. Wang J, Sun K, Cheng T, Jiang B, Deng C, Zhao Y, Liu D, Mu Y, Tan M, Wang X, Liu W. Deep High-Resolution Representation Learning for Visual Recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2020, 43(10): 3349-3364.
12. .Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N. An image is worth 16x16 words: Transformers for image recognition at scale[C]//International Conference on Learning Representations. 2021.
13. Chen K, Wang J, Pang J, Cao Y, Xiong Y, Li X, Sun B, Feng W, Liu Z, Xu J, Zhang Z, Cheng D, Zhu C, Cheng T, Zhao Q, Li B, Lu R, Zhu W, Wu Y, Dai J, Wang J, Shi J, Ouyang W, Loy C C, Lin D. MMDetection: Open mmlab detection toolbox and benchmark[J]. arXiv preprint arXiv:1906.07155, 2019.
14. Everingham M, Gool L V, Williams C K, Winn J, Zisserman A. The Pascal visual object classes (VOC) challenge[J]. International journal of computer vision, 2010, 88(2): 303-338.
15. Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2012: 3354-3361.
16. Cholke, D. R., Kakade, P., Nandini, K., Kapse, S., & Agwan, P. (2024). Solar panel cleaning robot. The Journal of Computational Science and Engineering, 2(3).
17. Londhe, R. N., Rohini, K., Shravani, K., Nikita, M., & Shivani, N. (2024). Anti-theft pressure sensing floor mat. The Journal of Computational Science and Engineering, 2(3).
18. Chimbalkar, S., Halnor, R., Doifode, K., & Lokhand, S. (2024). Anti-radar missile system. The Journal of Computational Science and Engineering, 2(3).