From 2d to 3d: Evolution of Object Detection Algorithms and Their Impact on Traffic Systems

Sicheng Lin

School of software & Internet of Things Engineering, Jiangxi University of Finance and Economics, Nanchang, Jiangxi, China

Abstract. Target detection is a fundamental research area in computer vision, with wide applications in pedestrian detection, vehicle recognition, and autonomous driving. For autonomous systems to navigate and engage with their surroundings safely, they must be able to detect and classify objects accurately in real-time. The development of deep learning technology has led to a rapid evolution in object detection techniques. This paper reviews the evolution of 2D object detection algorithms, ranging from deep learning-based techniques to conventional machine vision approaches. Additionally, it talks about how 3D object identification systems have advanced and how they are used in traffic situations. The study also examines the difficulties that object detection algorithms are currently facing and suggests possible avenues for further investigation. This study seeks to deliver an extensive review of the development and present landscape of object detection technologies, helping researchers better understand the key technical pathways, identify existing challenges, and explore innovative approaches for advancing the field further. It is hoped that this review can serve as a reference and inspiration for subsequent research and practical applications.

1 Introduction

As one of the important research directions in the field of computer vision, the main task of target detection is to efficiently and accurately find all the target regions of interest in a specified scene, and determine the geometric and semantic information of the target through a rectangular candidate box, which is widely used in the fields of pedestrian recognition, vehicle tracking, and unmanned driving. In the process of target detection, due to the different appearance, material and morphology of various types of objects in the image, but also by the light, weather, shooting angle and other image acquisition conditions, resulting in target heterogeneity, greatly increasing the detection difficulty, triggering a large number of researchers to explore the detection algorithm.

A target visual detection algorithm must not only recognize the category of objects within an image, but also accurately locate their positions, which is usually realized by drawing a candidate box. Its basic process can be roughly divided into three steps: region suggestion, feature extraction and region classification [1], where potential bounding boxes are proposed through heuristic algorithms, from which visual information is captured, a classifier is used

978882657x@gmail.com

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

to identify whether or not the candidate region contains the target object, and finally, the final candidate box is obtained through post-processing methods such as non-extremely large value suppression.

Since AlexNet was first proposed [2], the field of target detection has shifted from traditional manual feature extraction methods to end-to-end deep learning methods, resulting in a great improvement in both detection accuracy and efficiency.

This paper provides a comprehensive overview of the evolution of object detection algorithms, focusing on the transition from traditional 2D methods to the advanced 3D techniques enabled by deep learning. The significant strides made in both 2D and 3D object detection, particularly in the context of autonomous driving and traffic scenarios, are explored. The strengths and limitations of current approaches are analyzed, highlighting the existing challenges such as computational efficiency, robustness under dynamic conditions, and real-time processing. Finally, potential research directions are proposed to address these issues, with an emphasis on integrating sensor fusion and improving detection accuracy for future applications in intelligent transportation systems.

2 2D Target Detection Methods

The basic process of target detection consists of three core steps: region suggestion, feature representation and region classification [1].

2.1 Regional recommendations

Region suggestion is the first step in target detection, which first generates candidate regions that may contain targets, and then uses a classifier to determine whether these regions contain specific classes of targets. Traditional region suggestion strategies can be broadly categorized into three types: region suggestion based on a sliding window, region suggestion based on a voting mechanism and region suggestion based on image segmentation.

In 2005, Dalal et al. proposed the HOG feature extraction method [3], which calculates the gradient of the normalized detection window, then divides the image into cellular units, and traverses all scales of the image by fixing the window, and finally puts the extracted HOG features into the SVM classifier for classification.

Later generations have successively proposed methods to reduce the dimensionality of the features, Sermanet et al. proposed the OverFeat feature extraction operator [4], which combines the dense prediction properties of traditional sliding windows in a first attempt to unify the classification, localization, and detection tasks into a single convolutional neural network for learning. The authors tested the model with different scales and the 6-scale model achieved 13.6% in top-5 error rate.

In order to create a structured, spatially limited, multilevel voting system, Felzenszwalb et al. proposed the classical deformable part model (DPM) [5]. The basic idea of this model is to represent the target as a hierarchical structure of "root filters + part filters" by identifying latent variables and updating model parameters. The core idea is to represent the target as a hierarchical structure of “root filter + component filter”, and through the alternating training of identifying potential variables and updating model parameters, the system essentially learns the optimal voting scheme and determines the overall position of the target.

Image segmentation generates candidate regions through hyper pixel segmentation or edge detection. Zitnick et al. proposed the concept of "edge group"[6], which uses a structured edge detector to evaluate whether a box contains a target object using edge information. The Faster R-CNN framework was proposed by Ren et al. [7], which improves detection efficiency by directly learning to create region recommendations using a deep convolutional network (CNN) for the first time.

2.2 Characterization

Feature representation is an important step in target recognition, aiming at extracting meaningful representations from the original image by means of appropriate feature models and training classifiers to classify the feature vectors. Earlier feature representations can be broadly categorized into point of interest based, dense extraction based and multi-feature fusion based algorithms.

Point of interest based algorithms detect significant changing regions in an image as points of interest and extract low level feature descriptors from these localized regions. Canny first proposed Canny edge detector [8], which reduces the effect of noise by Gaussian filter, performs non-maximal value suppression by gradient vectorization, and then establishes high and low dual thresholds with hysteresis to retain the weak edge points to achieve coherent edge extraction.

Dense extraction based algorithms perform systematic feature extraction for each pixel or fixed region in an image without relying on salient points, and are therefore suitable for local information characterization. Lowe proposed Scale Invariant Feature Transformation (SIFT) [9], and the literature proposes a method capable of extracting invariant features from images to maximize the underlying description of the image. Local Binary Pattern (LBP) is an operator for describing the local texture features of an image by quantifying the grayscale relationship through the conversion of the progression [10]. Maenpaa et al. extended it [11], and based on the original grayscale-invariant operator, they proposed LBP operator with rotational invariance, and at the same time, they proposed the LBP Equivalent Pattern (Uniform Pattern) to combine the At the same time, the LBP Uniform Pattern is proposed, which defines the codes whose binary bit values do not change more than twice as "equivalent", in order to downscale their pattern types.

The hand-extracted features usually have good expandability, such as DPM combining multiple features [5], which has been widely used in subsequent work and achieved excellent results, combining the capture of Haar edge features and LBP texture features, which can form a more comprehensive feature representation, and improve the robustness and accuracy of target detection.

2.3 Regional classification

Region classification utilizes a training dataset to train a classifier through supervised learning, which then predicts the target category based on the feature vectors of the candidate regions. Commonly used classifiers include Support Vector Machine (SVM), Adaptive Boost (AdaBoost), etc. The former achieves efficient binary classification or multi-classification by maximizing the classification boundaries, while the latter iteratively adjusts the sample weights to allow subsequent weak learners to pay more attention to the samples that have been misclassified by the previous round, and ultimately combines multiple simple models in a weighted manner to form a strong classifier.

Currently, the vast majority of image recognition tasks use one-to-many classifier training [12], where an equal number of classifiers are trained for a number of categories that need to be recognized, and when a new image needs to be recognized, its features are input to all classifiers, and the category corresponding to the one with the highest response value is selected as the recognition result.

3 Deep learning driven 2D target detection

The deep learning model is a machine learning method that imitates the structure and function of the neural network of the human brain, and it is widely used in feature extraction

architectures by constructing multi-layer neural network models with powerful feature extraction and modeling capabilities. In the following section, the 2D detection algorithm based on deep learning is briefly described from the feature extraction architecture.

3.1 Feature Extraction Architecture

Deep learning in target detection mainly relies on Convolutional Neural Networks (CNNs), a deep learning model designed for processing grid-like topological data, which uses one or more sets of learnable filters to perform convolutional operations on the input image to generate a feature map, and thus capture local features in the image.

As deep learning has advanced, numerous CNN-based convolutional neural network models have been applied to problems involving target recognition. Krizhevsky et al. proposed AlexNet network, which introduces deep networks and ReLU activation functions, it outperformed other approaches with a Top-5 error rate of 16.4% on the ILSVRC 2012 picture classification assignment [2]. Subsequently, other researchers have proposed deep learning models without special connections such as VGG, which uses standard convolution and pooling operations and is suitable for basic feature extraction.

A residual network called ResNet was introduced by He et al. to address the issue of gradient vanishing in deep networks caused by the overstacking of convolutional and pooling layers [13]. ResNet introduces residual connection.Convolutional layers and a short-circuit connection method make up the residual block, which greatly increases the model's stability and training depth while lowering network parameters and computation.

Tan et al. proposed an efficient convolutional neural network, EfficientNet [14], in 2019, which introduces the concept of Compound Scaling to optimize resource utilization by simultaneously scaling the depth, width, and input resolution of the network proportionally. The EfficientNet-B7 suggested in this literature achieves the highest accuracy of the year, 84.3%, according to experiments on the dataset, on the ImageNet Top-1 dataset, with a parameter count of only 1/8.4 and an inference speedup of nearly 6.1 times compared to GPipe, which had the highest accuracy before.

3.2 Target detection algorithms for Anchor

Anchor is in essence a set of predefined bounding a priori boxes, which are localized by the network to extract the points of the feature map, used to construct the offset of the true border position relative to the predefined boundaries during training, his role is to solve the problem of region allocation. Algorithms concerning anchors can be categorized into anchor-base and anchor-free approaches, the essential difference being the difference in the solution space, how the positive and negative samples are defined and the way the regression is performed [15].

The Anchor-base algorithm can be roughly divided into three steps, firstly, a large number of anchors are preset in the image or point cloud space, and then four offsets of the target with respect to the anchors are regressed, and then the true position of the target is calibrated based on this offset. This method is divided into one-stage and two-stage algorithms based on whether or not it generates candidate regions, the former mainly includes SSD, RetinaNet, YOLOv3 and earlier versions, which generates candidate frames by means of sliding windows, and then classifies the frames directly; the latter mainly includes the RCNN series such as Faster R-CNN, Mask R-CNN, and so on. These methods mainly realize the detection process through convolutional neural networks. During network training, region proposals are created by extracting features (RPN) using CNN networks or heuristic techniques. The detection network is then trained to classify and regress based on the region proposals.

Anchor-free algorithms are used in two ways, one is to limit the detection region by locating the key points of the detected target, and the other is by determining the center point of the target and then estimating the distance from the target boundary to the center point. Keypoint-based methods include CornerNet, Grid R-CNN, ExtremeNet, etc. Taking CornerNet as an example, it extracts features by concatenating multiple Hourglass modules, and then outputs two branches of the upper-left and lower-right points, each of which is then used to generate prediction frames by outputting three branches through pooling. The centroid-based algorithm treats the model as a centroid, and the detector captures its relevant attributes, such as early YOLO, when returning to the centroid.

4 3D target detection algorithm classification and traffic scene adaptation analysis

4.1 Point cloud based approach

Conventional 3D object identification techniques identify 3D objects from 2D pictures and use the geometric relationship between the dimensional boundaries to ascertain the object's pose [16], but this method usually relies on predefined anchor frames and subsequent region corrections.LiDAR sensors are typically utilized in unmanned situations to create 3D point clouds in order to record 3D structures in the driving environment, and most of the point cloud-based 3D detection methods utilize 2D CNNs to learn the point cloud features used to generate 3D frames or 3D CNNs to learn the features of voxels for generating 3D candidate frames [17], thus applying the well-established 2D detection framework to 3D detection.

Qi et al. proposed PointNet, a method that does not need to convert the point cloud into other data structures for feature learning, but they used PointNet to 3D target detection [18], predicting 3D bounding boxes based on the cropped out point cloud based on 2D RGB color value detection results. This method learns 3D representations straight from the point cloud.However, this approach cannot utilize the 3D information to produce strong bounding box candidate recommendations and is overly reliant on the 2D detection performance.

Based on this, Shi et al. suggested PointRCNN, a point cloud-based 3D target identification architecture [19], which is used to detect 3D targets from point clouds, and the network specifically consists of two phases(Figure 1, Figure 2), The first stage uses point cloud network structures, including multi-scale grouped PointNet++, to learn point cloud representations and creates 3D candidate regions from the original point cloud in a bottom-up fashion, and after the The point-by-point feature vectors are generated after encoding and decoding, and the two steps of 3D boundary candidate region generation and boundary candidate region segmentation are carried out at the same time; the second stage is used to refine the 3D candidate regions in canonical coordinates. For precise bounding box refinement and confidence prediction, the convergence points of each candidate region are converted to coordinates and combined with the global semantic features of each point that were learned in the first stage.

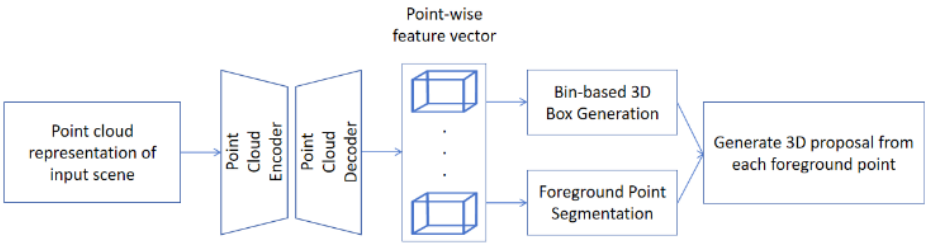

Figure 1 Phase I Schematic [19]

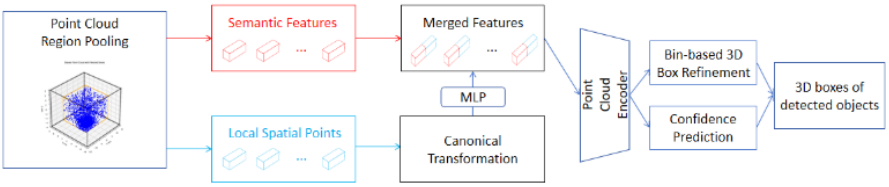

Figure 2 Phase II Schematic [19]

PointRCNN was evaluated on the car class of the KITTI validation segmentation set, with its performance compared to that of previous approaches. Average Precision (AP) served as the evaluation metric. An Intersection over Union (IoU) threshold of 0.7 was applied for cars, while thresholds of 0.5 were used for pedestrians and cyclists,it can be seen that the PointRCNN method is robust to val splitting and improves 8.28% over the previous best AP in Hard conditions.

4.2 Voxel-based approach

The main idea of this method is to extract features from the original sparse 3D point cloud data collected by the sensors, discretize it into a uniform 3D voxel grid, encode features within each voxel, and then aggregate the voxel features using 3D convolution or other efficient operations, and finally complete the target localization and classification by a region-proposed network or a single-stage detection network. The following section will briefly introduce two major point cloud extraction backbone networks, Voxel-Based [20] and Pollar-Based [21], as well as methods combining point-level and voxel-level features (e.g., PV-RCNN).

The overall idea of the Voxel-Based network is to voxelize the point cloud data extracted by LIDAR to obtain a form that facilitates the extraction of point cloud features, and subsequently, after feature encoding (VFE) of the voxel features, 3D feature extraction is performed on the encoded features by using 3D convolutional backbone network, and subsequently, the extracted features are projected into BEV space to obtain the BEV features, and finally, a 2D convolutional backbone network is utilized to obtain the BEV features. The 2D convolutional backbone network is used to complete further feature extraction on the BEV features.

The main difference between the Pillar-Based and Voxel-Based point cloud feature extraction methods is that, compared to the latter, the former eliminates the feature extraction process of the 3D convolutional backbone network and directly processes the point cloud data into BEV features, which provides faster inference speed and is easy to deploy. This

method replaces the 3D convolutional backbone network by point cloud enhancement as well as point cloud feature extraction operations for voxel feature encoding.

4.3 Network Architecture for Integrated Point-Based and Voxel-Based Approaches

The PV-RCNN algorithm belongs to the Pre-BEV Feature Extraction (extract features before generating BEV representations) method [22], which combines the advantages of voxel- and point-level features with flexible sensory field ranges and accurate localization information, while significantly improving the 3D detection performance with limited memory consumption.

It can be divided into two phases as a whole, with the first phase being the feature encoding and proposal generation phase, i.e., voxel-based branching. The original input point cloud P is first divided into a small voxel grid containing feature values, then gradually downsampled into BEV views using a series of sparse convolutional layers, and finally one-stage classification and regression are performed to obtain 3D candidate frames.

The second stage is proposal refinement and confidence prediction, i.e., point-based branching. Firstly, a small number of key points that are uniformly distributed and representative of the scene are extracted, and then the voxel features and point features are combined using the VSA (Voxel Set Abstraction) module, and the key points are used as the center of the spherical receptive field, and local feature extraction is performed in the voxel feature space output from the 3D sparse convolutional network for each key point, and then the VSA extension module E- VSA uses a bilinear interpolation method to further integrate the information.

4.4 Multimodal fusion methods

Traditional multimodal fusion improves the accuracy and robustness of detection by capturing rich texture, color, and other appearance details based on optical sensing cameras, and laser ranging radar provides geometrical structure information in precise depth. Traditional methods are typically classified into three categories: early fusion, intermediate fusion, and late fusion. Early fusion involves combining raw data or low-level features from multiple modalities directly at the input stage, intermediate fusion fuses the extracted features of each modality at the middle layer, and late fusion consists of independent detection of each modality and then fuses the results through post-processing strategies, such as confidence and position.

Generally speaking, traditional methods are devoted to elaborate fusion strategies. Liu et al. formulated multimodal fusion as a multilinear function, with the goal of constituting the input vectors of each modality into a high-dimensional tensor by outer product operation, and then obtaining the final output representation by linear mapping, while proposing a low-rank weight decomposition algorithm, which regards the weight tensor corresponding to each output dimension as a multimodal tensor, and the number of parameters is greatly reduced [23].

Dong et al. proposed an innovative cross-modal target detection method, which significantly improves the detection performance by fusing features in the hidden state space; the method designs a Fusion-Mamba Module (FMB), which combines the Mamba architecture, a deep learning architecture based on the Selective State Space Model (SSM), and a gating mechanism to make the feature fusion more effective [24].

The core of the Fusion-Mamba approach is the Fusion-Mamba Block (FMB), which solves the modal difference problem by processing cross-modal features in the hidden state space, he includes the State Space Channel Swapping module (SSCS) and the Dual State

Space Fusion (DSSF) module, the former enhances cross-modal interactions by aligning shallow features from different modules and processing them through the VSS module, while the latter is responsible for deep feature fusion, which models cross-modal correlations by using Manba's SSM state space model to map features into the hidden state space.

5 Conclusion

This paper describes the development history and research status of target detection, from the traditional target detection algorithms, two-dimensional target detection to three-dimensional advancement, also describes based on the point cloud and voxel and multimodal fusion methods, and combined with deep learning algorithms on the past and recent popular detection algorithms for the review, introduces part of the experimental data and evaluation criteria, the study shows that these methods significantly improve the detection accuracy and real-time performance, particularly in autonomous driving, but small object detection, occlusion processing and data labeling cost are still challenges. While deep learning drives technological advances, challenges include handling complex environments, ensuring real-time performance and reducing labeling costs. These issues may be addressed in the future through Transformer architectures, lightweight models, and unsupervised learning, and ethical and privacy concerns also need to be addressed. It is hoped that this paper will provide new information and ideas for researchers in the field to choose their methods.

References

1. H. Zhang, K. Wang and F. Wang, "Deep Learning in Target Vision Detection: Progress and Outlook," Acta Automatica Sinica, vol. 43, no. 8, pp. 1289-1305, 2017.
2. A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems, vol. 25, pp. 1097-1105, 2012.
3. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893 vol. 1.
4. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," arXiv:1312.6229.
5. P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1627-1645, Sept. 2010.
6. C. L. Zitnick and P. Dollár, "Edge Boxes: Locating Object Proposals from Edges," in Computer Vision - ECCV 2014, D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars, Eds. Cham: Springer, 2014, pp. 391-405.
7. S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," arXiv:1506.01497.
8. J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986.
9. D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.

10. T. Ojala, M. Pietikäinen, D. Harwood, "A comparative study of texture measures with classification based on featured distributions," Pattern Recognition, vol. 29, pp. 51-59, 1996.
11. T. Ojala, M. Pietikäinen, T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, July 2002.
12. S. Lazebnik, C. Schmid, J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), New York, NY, USA: IEEE, 2006, pp. 2169–2178.
13. K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition," arXiv:1512.03385.
14. M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," arXiv:1905.11946.
15. S. Zhang, C. Chi, Y. Yao, Z. Lei, S. Z. Li, "Bridging the Gap Between Anchor-based and Anchor-free Detection via Adaptive Training Sample Selection," arXiv:1912.02424.
16. A. Mousavian, D. Anguelov, J. Flynn, and J. Košecká, "3D bounding box estimation using deep learning and geometry," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5632-5640, 2017.
17. S. Song and J. Xiao, "Deep sliding shapes for amodal 3D object detection in RGB-D images," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 808–816, 2016.
18. C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D Object Detection from RGB-D Data," arXiv:1711.08488.
19. S. Shi, X. Wang, H. Li, "PointRCNN: 3D Object Proposal Generation and Detection from Point Cloud," arXiv:1812.04244 [cs.CV].
20. Y. Zhou, O. Tuzel, "VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection," arXiv:1711.06396.
21. A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, O. Beijbom, "PointPillars: Fast Encoders for Object Detection from Point Clouds," arXiv:1812.05784 [cs.LG].
22. S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, H. Li, "PV-RCNN: Point-Voxel Feature Set Abstraction for 3D Object Detection," arXiv:1912.13192 [cs.CV].
23. Z. Liu, Y. Shen, V. B. Lakshminarasimhan, P. P. Liang, A. Zadeh, L.-P. Morency, "Efficient Low-rank Multimodal Fusion with Modality-Specific Factors," arXiv:1806.00064.
24. W. Dong, H. Zhu, S. Lin, X. Luo, Y. Shen, X. Liu, J. Zhang, G. Guo, B. Zhang, "Fusion-Mamba for Cross-modality Object Detection," arXiv:2404.09146.