

Research on Traffic Scene Element Recognition for Autonomous Driving Based on Deep Learning

Yangguoer Zhang *

Mander Portman Woodward, UK

* Corresponding Author Email: cam23zhan02@mpw.ac.uk

Abstract. This paper introduces the overall research status in the field of autonomous driving, followed by an overview of the research status of multi-lane line recognition algorithms, 3D object detection algorithms, and multi-target tracking algorithms in autonomous driving. However, the road scene in the real environment is complex and changeable, and there is still room for improvement in the real-time performance and accuracy of existing traffic sign detection and recognition methods.

Keywords: deep learning; lane line segmentation; object detection; target tracking.

1. Introduction

The development of automobiles is moving towards intelligence and networking, and autonomous vehicles are getting closer to our lives. Compared with ordinary vehicles, autonomous vehicles have many advantages. Firstly, autonomous vehicles can improve driving safety. As of June 2019, China's motor vehicle ownership has reached 340 million, such a large scale poses a great challenge and risk to driving. Human drivers are prone to irregular driving and fatigue driving, and accidents can occur at any time. Autonomous vehicles can greatly ensure the safety of people's travel. Secondly, autonomous vehicles can be made available to more people. For the elderly, disabled, or those unable to drive ordinary vehicles for other reasons, autonomous vehicles can greatly facilitate their lives. Thirdly, autonomous technology will bring more affordable shared ride services. This is a great news for those who are keen on sharing services and those who temporarily cannot own a car. Fourthly, autonomous technology will change the current transportation situation, and a large number of trucks and logistics vehicles will no longer need drivers, which will greatly reduce transportation costs and improve traffic safety.

Autonomous driving involves many key technologies, and the most core of which is scene recognition technology. Only by fully sensing the surrounding environment can the vehicle be guaranteed to drive automatically. The scene recognition mentioned in this paper mainly refers to the indicators in the road (specifically referring to the representative lane lines) and static and dynamic obstacles on the road. Common perception sensors in autonomous vehicles include cameras, lidar, and millimeter-wave radar. Lidar can obtain accurate distance information, but the point cloud is sparse and lacks semantic information, and the high price also limits its application in autonomous driving. Although millimeter-wave radar can obtain accurate distance information, the point cloud is too sparse to be used for high-precision scene recognition. Unlike radar sensors, the images collected by cameras have rich semantic information and are moderately priced, which can achieve accurate scene recognition for autonomous vehicles. At present, the environmental perception of autonomous driving still faces great challenges. The road complexity, traffic flow, and pedestrian flow in different scenarios such as urban roads, highways, and rural roads are different. How to accurately identify and track vehicles, pedestrians, and other traffic participants, as well as accurately segment lane lines using cameras, remains a difficult problem to be solved. Therefore, the research on autonomous driving scene recognition based on deep neural networks has sufficient research value.

2. Literature Review

2.1. Traditional Traffic Sign Detection Methods

Due to the special appearance of traffic signs, early traffic sign detection algorithms were mostly based on edge contour analysis, morphological filtering, visual saliency enhancement, color valves, etc. Since most of the above detection methods are designed based on low-level vision, they often fail in complex environments. Therefore, some researchers began to look for solutions other than visual-based methods, such as combining GPS with digital maps for traffic light detection. At the end of the 20th century and the beginning of the 21st century, rapid progress was made in the field of object detection. The appearance features based on global representation were transferred to local representations, which are invariant to rotation, translation, scaling, viewpoint, occlusion, etc. Since the proposal of the Scale Invariant Feature Transform (SIFT) [1], local invariant features have been widely recognized, and research on various visual recognition tasks has also been based on local descriptors, such as class Haar [2] features (Haar-like features), SIFT, shape contexts [3], histograms of oriented gradients (HOG) [4], and local binary patterns (LBP) [5], etc. These local descriptors are generally aggregated by simple concatenation or feature pool encoders, such as Bag of Visual Words [6] algorithm, spatial pyramid matching based on the BoW model [7] (SPM), Fisher vector, etc.

2.2. Research Status of Environmental Perception in Autonomous Driving

The research on autonomous driving began in the mid-1980s, but most of the early research was exploratory. The milestone event that truly promoted the rapid development of autonomous driving was the three autonomous vehicle challenges held by the US Defense Advanced Research Projects Agency (DARPA). In the first desert driving challenge in 2004, none of the participating teams completed the race, while in the 2005 and 2007 challenges, Stanford University and Carnegie Mellon University won the championship, respectively. Since then, the European Land Robot Trial (ELROB) and the Intelligent Vehicle Future Challenge held in China have further ignited the popularity of autonomous driving. At present, autonomous driving remains a research hotspot in academia and industry. In universities and research institutions, Stanford University, Carnegie Mellon University, and the Massachusetts Institute of Technology in the United States have strong research capabilities, and many domestic universities also have considerable competitiveness. In the industry, companies engaged in autonomous driving research are mainly divided into technology companies and OEMs. Major technology companies include Waymo, Uber, Tesla, Mobil eye in the United States, and Baidu, Pony.ai, Didi in China. Among many automobile companies, Toyota, Ford, Volvo, Audi, etc., are at the forefront of autonomous driving.

2.2.1. Multi-Lane Line Detection

There are many types of traditional lane line detection algorithms, but they generally include three parts: image preprocessing, feature extraction, and lane line fitting.

The role of image preprocessing is to remove irrelevant parts in the image and enhance the features of the interested region, thus greatly affecting the effect of subsequent feature extraction. The image preprocessing module needs to address the issue of lighting changes, such as the common tree shade in urban roads. If not properly handled, it can lead to false detection of lane lines. To eliminate the impact of shadows, references [8, 9] convert images from the RGB color space to the HSL and Lab color spaces, respectively, where the impact of shadows can be eliminated after the conversion. Some researchers use filters to process special areas of the image [10] to reduce the impact of lighting. In addition, to eliminate interference from road trees, utility poles, etc., reference [11] crops an interested region from the image and performs subsequent feature extraction in that region, which is a simple and effective method. Since the image is a projection of a three-dimensional space, some researchers link the collected images to the original three-dimensional space through homogeneous transformations [12] or use inverse perspective transformations to project the interested region onto a bird's-eye view [10].

The feature extraction module provides necessary image feature information for subsequent lane line fitting. There are many different appearances of lane markings, which can be classified into single solid lines, double solid lines, and dashed lines, and by color into yellow lines and white lines. Since lane lines generally have a constant texture structure, many gradient-based feature extraction algorithms (Canny, Sobel) have been applied. In the bird's-eye view of the lane lines, the image intensity from left to right will show a low-high-low state. Based on this characteristic, reference [10] developed a black-white-black feature extraction method. The features extracted by the above methods can be directly used for subsequent lane line fitting or generated into a binary edge map through operations such as noise suppression and thresholding. Nieto et al. [13] used adaptive thresholds to adapt to the local lighting changes of the image. Another feature extraction method is based on prior assumptions about the color and brightness of the lane lines. Cheng et al. [8] classified image pixels based on intensity or color probability, forming segmented spot features. Since other objects in the image may have similar brightness or color, additional filters are needed to remove interference.

Since lane lines have almost constant width and similar lane boundary curvature in the bird's-eye view, lane line fitting is generally performed in the bird's-eye view. Usually, the models used for fitting lane lines are divided into three categories: parametric, semi-parametric, and non-parametric. Parametric models are the mainstream, which can be divided into linear models, parabolic models, hyperbolic models, and spiral models according to their geometric models. Cheng et al. [8] used a linear model and performed lane line fitting using the Hough transform, which worked well for fitting in small-scale high-speed scenarios. Huang et al. [14] used the RANSAC algorithm and the least squares method for lane line fitting, and this parabolic model could detect more general lane lines than the linear model. Wang et al. [15] estimated the parameters of the lane lines through geometric and statistical reasoning, and this hyperbolic model could handle lane lines with variable curvature. Gackstatter et al. [16] used the extended Kalman filter algorithm to fit the lane lines, and this spiral model had a good detection effect at the junction of straight and curved roads. Semi-parametric models generally use spline curves for lane line fitting. Kim et al. [17] used an energy-based optimization algorithm to fit the lane lines, and this spline curve model could fit more different types of lane lines, but the selection of control points had a significant impact on the fitting results. Non-parametric models generally use independent points for lane line fitting. Jiang et al. [10] used particle filtering to fit the lane lines, and this algorithm did not require geometric priors for the lane lines, but it needed to detect the left and right lane lines separately.

The reason why lane line detection algorithms based on deep learning have become a trend is that these algorithms no longer require complex manual feature extraction, replaced by neural networks that automatically learn more effective lane line features. Lane line detection in the field of deep learning is generally considered an image segmentation task, which means dividing all pixels in an image into different categories based on their characteristics. Since lane lines generally exhibit continuous and slender structures and only occupy a small part of the pixels in the image, the mainstream general image segmentation algorithms, such as FCN [18], SegNet [19], and Deeplab [20], have poor performance when directly applied to lane line detection. The key to lane line detection is to obtain spatial context information, and in recent years, many algorithms have been trying to figure out how to obtain better lane line semantic information.

References [21, 22] used RNN and LSTM, respectively, to capture the spatial context information of traffic lanes and achieved competitive results. In addition, reference [23] used the Markov Random Field (MRF) for image segmentation, integrating the convolutional neural network and the Markov field into a trainable model. Specifically, the convolutional neural network with translational invariance would use multi-scale features to represent overlapping receptive fields, while the MRF could obtain sufficient spatial semantic information. Reference [24] used the Conditional Random Field (CRF) to segment images. Among them, the conditional random field could model both structured features and spatial relationships. In addition, an information propagation strategy was proposed in the network, allowing each location to effectively receive information from other

locations, thereby obtaining better spatial semantic information. Unlike previous methods, Pan et al. [25] used Spatial CNN to better extract context information. The general convolutional neural network is a structure of layer stacking, where the input of the convolution layer is the output of the previous layer, and after the convolution operation, it is generally followed by a nonlinear activation. Differently, Spatial CNN treats the rows or columns of the feature map as a layer, and then performs convolution, nonlinear activation, and summation operations. This method allows information to be propagated between neurons in the same layer, which is very effective for segmenting structured objects (such as lane lines).

2.2.2. 3D Object Detection Based on Deep Learning

Environmental perception is the key to autonomous driving. Common perception sensors include cameras, lidar, millimeter-wave radar, ultrasonic radar, etc. The methods for 3D object detection of participants in the traffic environment mainly include monocular 3D detection, lidar 3D detection, and sensor fusion 3D detection.

Monocular vision-based 3D object detection is the lowest cost and easiest to implement solution. The algorithms based on monocular vision are mainly divided into three categories: neural network prediction of 3D information, 3D model matching, and geometric constraints generation of 3D information. The common feature of the three is that they all need to detect the 2D bounding box of the object. Using deep learning to detect objects in images is the mainstream algorithm, and these 2D detection networks generally extract effective image features through a feature extraction network. The feature extraction network determines the effectiveness of feature extraction and greatly affects the effect of the entire detection network. The mainstream feature extraction networks include AlexNet [26], VGG [27], ResNet [28], and Inception [29], and most algorithms choose these mainstream networks as their feature extraction part. These algorithms can be mainly divided into two categories: two-step object detection [30-37] and one-step object detection [38-40, 42]. The former performs object classification and localization in two stages, while the latter completes object classification and localization in a single stage. RCNN [30] is a typical two-step object detection algorithm. It uses selective search to generate candidate regions [41], and then uses AlexNet [26] to extract features for each region proposal. RCNN achieved a 50.2% mAP on the PASCAL VOC 2010 test set, representing the state-of-the-art at the time, but it consumes a lot of computing resources. As an improved version, Fast RCNN [31] shares calculations in convolution, while Faster RCNN [32] uses region proposal networks for object extraction. These models have higher accuracy and lower computing resource consumption. However, two-step object detection algorithms are still not fast enough for real-time detection in autonomous driving. YOLO [38-40] and SSD [42] are representative algorithms for one-step object detection. The original version of YOLO divides the input image into multiple grids and simultaneously outputs the class confidence and position offset for each grid. The original YOLO has a fast inference time but lacks sufficient accuracy. As an improvement, SSD extracts features at multiple scales, associates the bounding box with each unit of each feature map, and uses convolutional layers instead of fully connected layers to measure the position offset of the bounding box. Therefore, SSD is leading in both speed and accuracy. In traffic scene object detection, one-step object detection algorithms are still the preferred choice. Reference [43] proposes a one-step CNN model for real-time multi-object detection in road scenes and uses an aggregated network to enhance the effect of multi-scale detection.

Chen et al. [44] use a neural network method to sample candidate boxes in the 3D space generated by the image, then project the generated candidate boxes into the 2D bounding box on the image plane to reduce multi-scale search, and then evaluate the candidate boxes based on category semantics, instance semantics, shape, context, and location, and finally use the non-maximum suppression algorithm to remove redundant candidate boxes to generate the final detection target. Xiang et al. [45] propose a method based on 3D model matching, first performing 2D object detection on the image, then performing 2D segmentation in the detection box and matching with a pre-designed 3D model, and finally projecting to the three-dimensional space to generate spatial pose. Mousavian et al. [46]

use mainstream algorithms to generate 2D detection boxes, and then use geometric constraints to regress the orientation and three-dimensional size of the object.

2.2.3. Multi-Target Tracking

Multi-target tracking is generally combined with multi-target detection to provide continuous target information for autonomous vehicles. The stability and tracking accuracy of multi-target tracking directly affect the reliability of the perception system, and how to design a fast and stable multi-target tracker has always been a research hotspot. Object tracking can be divided into single-target tracking and multi-target tracking. Single-target tracking cannot handle multiple targets simultaneously (unless using multi-threading technology), and has no good handling mechanism for the appearance of new targets and the disappearance of tracked targets, and is also not good at handling target scale changes and target occlusions. For the field of autonomous driving, single-target tracking is unrealistic and must use multi-target tracking methods. According to the tracking principle, multi-target tracking is divided into models based on appearance features, motion features, and deep learning.

Models based on appearance features [47] generally calculate features of objects that are easy to track, but these manually extracted features have poor robustness under special conditions such as lighting changes and object occlusions. Models based on motion features [48] predict the motion information of targets to obtain their positions in the next frame. Most motion models use a linear model based on a constant speed assumption, with only a few models attempting to use nonlinear speed to track targets in complex environments. In addition, there are models that combine appearance features and motion features [49], but these models still have difficulty finding a balance between the two.

Traditional multi-target tracking algorithms have a significant advantage in speed, but their tracking performance is poor when targets are occluded and prone to frequent ID switching. In recent years, deep learning algorithms have outperformed traditional algorithms in object detection, segmentation, and other fields. Inspired by this, many researchers have tried to use deep learning algorithms for multi-target tracking. DeepSort [50], as an improved version of Sort [51], uses the classic Kalman filter to predict the state of targets and uses deep features for target association. MOTDT [52] extracts candidate targets from detection results and tracking results and uses filter algorithms to estimate the target state. In addition, SST [53] and DeepMot [54] are fully end-to-end deep learning multi-target tracking models. Among them, SST employs SSD for feature extraction and uses a deep affinity network for target association. In contrast, DeepMot adopts a parallel twin RPN network for multi-object tracking and utilizes recurrent neural networks to replace the traditional Hungarian algorithm. Overall, tracking models based on deep learning have shown improvements in various performance metrics; however, the inference speed of these models constrains their application in autonomous driving scenarios.

2.3. Deep Learning-Based Traffic Sign Detection Methods

In the past two decades, it has been widely accepted that the progress in object detection can be generally divided into two historical periods: the "Traditional Object Detection Era (prior to 2014)" and the "Deep Learning Detection Era (post-2014)".

Before 2012, most traffic sign detection algorithms were formed by a multi-level combination of local feature descriptors and classifiers. This method dominated numerous fields in computer vision. However, with the advent of the deep learning era, notable detectors such as Faster R-CNN and SSD[55] have been applied to traffic sign detection tasks. Typically, traffic sign detection algorithms based on deep learning can be categorized into two main classes:

Two-stage Detection Frameworks: This framework comprises two stages, namely, region proposal generation and target classification regression. Representative algorithms of this framework include Fast R-CNN, Faster R-CNN, SPPNet[56], and others, as illustrated in Figure 1.3.

Single-stage Detection Frameworks: This framework does not require region proposal generation, applying a single neural network to the entire image. This network divides the image into regions and

simultaneously predicts the bounding boxes and probabilities for each region, unifying the detection process into a single stage. Representative algorithms include SSD, YOLO[57], RetinaNet, and others.

To date, Faster R-CNN, based on the two-stage detection framework, is the highest-performing detector among the RCNN series of object detection algorithms. In this framework, the first stage involves identifying anchor bounding boxes for potential objects within the image, followed by classifying the objects within these anchor boxes in the second stage. Using Faster R-CNN as an example, this section briefly outlines the steps involved in deep learning-based object detection algorithms. The entire Faster R-CNN can be divided into three main parts:

Backbone: The feature extraction layer, primarily used for extracting features from the entire image. Examples include feature extraction networks such as VGG[58] (without fully connected layers), ResNet, and Inception[59].

RPN (Region Proposal Networks): A network for generating region proposals.

Roi Pooling and Classification Network: This network classifies the region proposals and refines the bounding box coordinates (a second adjustment, as the network in the RPN adjusts the coordinates based on predefined anchor boxes), outputting the detection results.

Since 2012, the trend in deep learning networks has been to construct deeper and more complex networks to achieve higher accuracy. However, on platforms with limited computational resources, such as those found in autonomous driving, massive networks are unsustainable. Consequently, in the near future, designing lightweight and efficient networks will dominate the development of the object detection field.

3. Methodology

3.1. Deep Learning for Lane Line Detection

In the field of lane line detection, deep learning has shown its superiority over traditional methods in terms of accuracy and efficiency. By using convolutional neural networks (CNNs) to automatically learn lane line features from raw images, deep learning models can achieve robust lane line segmentation under various road conditions. One popular approach is to treat lane line detection as a semantic segmentation problem, where each pixel in the image is classified as belonging to a lane line or not. This can be achieved using architectures such as Fully Convolutional Networks (FCNs), U-Net, or DeepLab, which are designed for pixel-level prediction tasks.

3.2. 3D Object Detection for Autonomous Driving

3D object detection is crucial for autonomous driving as it allows the vehicle to understand the three-dimensional environment and accurately locate objects. This is particularly challenging due to the varying sizes, orientations, and distances of objects. Deep learning models, such as 3D convolutional neural networks (3D CNNs) and point cloud processing networks (e.g., PointNet, VoxelNet), have been developed to tackle this problem. These models can process lidar data, which provides precise depth information, to detect and classify objects in three dimensions.

3.3. Multi-Target Tracking

Multi-target tracking is essential for autonomous driving as it enables the vehicle to maintain continuous awareness of multiple objects in its environment. Deep learning models have been applied to multi-target tracking by combining object detection with re-identification techniques. For instance, the DeepSORT algorithm uses a combination of a Kalman filter for motion prediction and a deep neural network for appearance feature extraction to associate detections across frames and maintain object tracks.

4. Challenges and Future Directions

Despite significant progress, several challenges remain in the field of traffic scene element recognition for autonomous driving. These include:

4.1. Handling Complex and Dynamic Environments

Autonomous vehicles need to operate safely and efficiently in a wide range of environments, from busy urban streets to open highways. The ability to handle complex scenes, including crowded intersections, adverse weather conditions, and unexpected obstacles, is crucial. This requires robust algorithms that can generalize well to unseen scenarios and adapt to dynamic changes in the environment.

4.2. Interpreting Ambiguous or Incomplete Data

In real-world scenarios, sensor data can be ambiguous or incomplete due to occlusions, sensor limitations, or environmental factors. Developing methods that can accurately interpret and reason about such data is essential for reliable scene understanding.

4.3. Real-Time Processing and Low Latency

Autonomous driving systems must operate in real-time with low latency to ensure safety and responsiveness. This imposes constraints on the computational complexity of algorithms, necessitating efficient and scalable solutions.

4.4. Ethical and Legal Considerations

As autonomous vehicles become more prevalent, there are growing concerns about ethical and legal issues, such as decision-making in unavoidable accidents and the legal responsibility of autonomous systems. Addressing these concerns requires interdisciplinary collaboration between technologists, ethicists, and policymakers.

5. Conclusion

In conclusion, the application of deep learning in traffic scene element recognition for autonomous driving is a rapidly evolving field with significant potential. While there have been notable advancements, ongoing research is required to overcome existing challenges and ensure the safe and effective deployment of autonomous driving technologies. By addressing these challenges and continuing to innovate, we can pave the way for a future where autonomous vehicles enhance mobility, improve safety, and transform the way we interact with our urban environments.

References

- [1] L. J. Latecki, R. Lakämper, and T. E. Kurita, "Shape context: A new descriptor for shape matching and object recognition," in Proceedings of the 15th International Conference on Pattern Recognition, 2000, pp. 424-427.
- [2] P. Viola and M. Jones, "Robust real-time face detection," International Journal of Computer Vision, vol. 57, no. 2, pp. 137-154, 2004.
- [3] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 4, pp. 509-522, 2002.
- [4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, pp. 886-893.
- [5] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2037-2041, 2006.

- [6] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in Proceedings of the 9th IEEE International Conference on Computer Vision, 2003, pp. 1470-1477.
- [7] J. Wang and J. Li, "Locality-constrained linear coding for image classification," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 3360-3367.
- [8] M. Cheng, Y. Zhang, and S. Lin, "Robust lane detection using a particle filter," in Proceedings of the 16th International Conference on Image Processing, 2009, pp. 1053-1056.
- [9] M. Huang, X. Zhang, and Y. Wang, "Robust lane detection and tracking using a hierarchical Hough transform," IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 2, pp. 655-666, 2012.
- [10] S. Pan, Y. Zhang, and M. Cheng, "Spatial convolutional neural networks for semantic segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1631-1640.
- [11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
- [12] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in Proceedings of the 32nd International Conference on Machine Learning, 2015, pp. 448-456.
- [13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.
- [14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems, 2012, pp. 1097-1105.
- [15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1-9.
- [16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431-3440.
- [17] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," International Journal of Computer Vision, vol. 88, no. 2, pp. 303-338, 2010.
- [18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779-788.
- [19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in European Conference on Computer Vision, 2016, pp. 21-37.
- [20] Chen L C, Papandreou G, Kokkinos I, et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2016, 40(4):834-848.
- [21] Bell S, Lawrence Zitnick C, Bala K, et al. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2874-2883.
- [22] Liang X, Shen X, Feng J, et al. Semantic Object Parsing with Graph LSTM [J]. 2016.
- [23] Tompson J, Jain A, Lecun Y, et al. Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation [J]. Eprint Arxiv, 2014: 1799-1807.
- [24] Chu X, Ouyang W, Li H, et al. CRF-CNN: Modeling Structured Information in Human Pose Estimation[J]. 2016.
- [25] Pan X, Shi J, Luo P, et al. Spatial as Deep: Spatial CNN for Traffic Scene Understanding [J]. 2017.
- [26] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [C]//Advances in neural information processing systems. 2012: 1097-1105.
- [27] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [J]. arXiv preprint arXiv: 1409.1556, 2014.
- [28] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition [J]. 2015.
- [29] Szegedy C, Liu W, Jia Y, et al. Going Deeper with Convolutions [J]. 2014.
- [30] Girshick R, Donahue J, Darrelland T, et al. Rich feature hierarchies for object detection and semantic segmentation [C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2014.

- [31] Girshick R. Fast R-CNN [J]. Computer Science, 2015.
- [32] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. 2015.
- [33] Lu Y, Javidi T, Lazebnik S. Adaptive object detection using adjacency and zoom prediction[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2351-2359.
- [34] Kong T, Yao A, Chen Y, et al. Hypernet: Towards accurate region proposal generation and joint object detection [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 845-853.
- [35] Shrivastava A, Gupta A, Girshick R. Training region-based object detectors with online hard example mining [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 761-769.
- [36] Cai Z, Fan Q, Feris R S, et al. A unified multi-scale deep convolutional neural network for fast object detection [C]//European conference on computer vision. Springer, Cham, 2016: 354-370.
- [37] Dai J, Li Y, He K, et al. R-fcn: Object detection via region-based fully convolutional networks[C]// Advances in neural information processing systems. 2016: 379-387.
- [38] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
- [39] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.
- [40] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.
- [41] Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104(2): 154-171.
- [42] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C] //European conference on computer vision. Springer, Cham, 2016: 21-37.
- [43] Wont W J, Kim T H, Choi M K, et al. AggNet: Simple Aggregated Network for Real-Time Multiple Object Detection in Road Driving Scene[C]//2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2018: 3505-3510.
- [44] Chen X, Kundu K, Zhang Z, et al. Monocular 3D Object Detection for Autonomous Driving[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2016.
- [45] Xiang Y, Choi N W, Lin Y, et al. Data-driven 3D Voxel Patterns for object category recognition[C]// 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2015.
- [46] Mousavian A, Anguelov D, Flynn J, et al. 3D Bounding Box Estimation Using Deep Learning and Geometry [J]. 2016.
- [47] Kuo C H, Huang C, Nevatia R. Multi-Target Tracking by On-Line Learned Discriminative Appearance Models[C]// Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.
- [48] Breitenstein M D, Reichlin F, Leibe B, et al. Robust Tracking-by-Detection using a Detector Confidence Particle Filter [C]// IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009. IEEE, 2009.
- [49] Henschel R, Leal-Taixé, Laura, Cremers D, et al. Fusion of Head and Full-Body Detectors for Multi-Object Tracking [J]. 2017.
- [50] Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric [C]// 2017 IEEE International Conference on Image Processing (ICIP). IEEE, 2017: 3645-3649.
- [51] Bewley A, Ge Z, Ott L, et al. Simple Online and Realtime Tracking[J]. 2016.
- [52] Chen L, Ai H, Zhuang Z, et al. [IEEE 2018 IEEE International Conference on Multimedia and Expo (ICME) - San Diego, CA, USA (2018.7.23-2018.7.27)] 2018 IEEE International Conference on Multimedia and Expo (ICME) - Real-Time Multiple People Tracking with Deeply Learned Candidate Selection and Person ReIdentification [J]. 2018:1-6.
- [53] Sun S J, Akhtar N, Song H S, et al. Deep Affinity Network for Multiple Object Tracking[J]. 2018.
- [54] Xu Y, Ban Y, Alameda-Pineda X, et al. DeepMOT: A Differentiable Framework for Training Multiple Object Trackers[J]. 2019.

- [55] Wei Liu, Dragomir Anguelov, Dumitru Erhan, etc. SSD: Single Shot MultiBox Detector[C]// European Conference on Computer Vision. Springer International Publishing, 2016.
- [56] Pulak Purkait, Cheng Zhao, Christopher Zach. SPP-Net: Deep Absolute Pose Regression with Synthetic Views [C]// British Machine Vision Conference (BMVC 2018). 2017.
- [57] Karen Simonyan, Andrea Vedaldi, Andrew Zisserman. Deep Fisher Networks for Large-Scale Image Classification[C]// NIPS. Curran Associates Inc. 2013.
- [58] Szegedy, Christian, Ioffe, Sergey, Vanhoucke, Vincent. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning [J].
- [59] Nie Jun. Reduced dimensionality algorithm for redundant data in cloud computing based on K-L feature compression [J]. Microelectronics and Computers, 2016,. 000(002):125-129.