Smart Camera Parking System With Auto Parking Spot Detection

Tuan T. Nguyen[1] and Mina Sartipi[1]

University of Tennessee at Chattanooga, USA, 37402 xwz778@mocs.utc.edu
mina-sartipi@utc.edu

Abstract. The proliferation of urban centers has exacerbated traffic congestion, underscoring the critical need for intelligent parking solutions. While computer vision approaches have gained traction, their reliance on manual spot labeling poses practical challenges. This study introduces `PakLoc`, a novel framework for automated parking spot detection, complemented by `PakSke`, a module that refines bounding box orientation and dimensions. Empirical evaluation on the PKLot dataset demonstrates a remarkable 94.25% reduction in manual labor requirements. Furthermore, we present `PakSta`, an innovative method leveraging `PakLoc`'s object detector to concurrently assess occupancy across all parking spaces within a given frame. `PakSta` achieves an impressive AP75 of 93.6% on the PKLot dataset, surpassing the performance of the benchmark Yolo SPS (93.3% AP75) which relies on manually labeled data, and significantly outperforming other methods such as POD (61.8% AP75). These advancements offer a promising avenue for efficient, label-free smart parking systems, potentially revolutionizing urban parking management.

Keywords: Smart Parking System, · SPS · Parking Spots Localization

1 Introduction

Urban population growth presents unprecedented challenges for city planners and residents alike. The United Nations projects that by 2050, a staggering 68% of the global population will reside in urban areas [3]. This demographic shift exacerbates existing issues, particularly in transportation and parking management. While recent research has focused on traffic simulation to mitigate congestion [24, 30, 32], the parking problem remains a significant concern. An INRIX survey reveals that American drivers spend an average of 17 hours annually searching for parking, with this figure soaring to 107 hours in densely populated cities like New York [2]. As autonomous vehicles become more prevalent, the need for accurate, real-time parking information becomes increasingly critical.

Smart parking systems (SPS) offer a promising solution, linking drivers and parking operators to optimize space utilization and reduce emissions. However, current sensor-based systems, while precise, are prohibitively expensive for large-scale deployment. The San Francisco SF Park program, for instance, spent

Fig. 1: Parking spot detection challenges: varying angles and orientations (PKLot Dataset [10])

$27 million to equip 19,250 parking spaces with sensors, at a cost of $1,400 per unit [1]. Computer vision (CV) approaches present a cost-effective alternative [7, 14]. CV systems can monitor multiple spaces with a single camera, reducing installation and maintenance costs while offering additional benefits such as detecting improper parking and suspicious activities. Despite their promise, existing CV algorithms face three main challenges:

1. Time-consuming manual labeling of parking spots
2. Poor scalability due to the need for re-annotation in new environments
3. Slow real-time performance for large parking lots due to multiple forward passes in classification methods [17, 27]

To address these issues, we propose three novel contributions:

– **PakLoc**: An automated parking space localization algorithm that reduces manual labeling effort by 94.25% (AR75) when deploying an SPS in new environments.
– **PakSke**: A module that automatically adjusts bounding box rotation and size, ensuring alignment with actual parking spot angles (Figure 1).
– **PakSta**: A framework that simultaneously detects and monitors the status of all parking spots, utilizing the PakLoc detector to eliminate the need for additional model training.

Our comprehensive experiments on the PKLot dataset demonstrate the efficacy of these approaches, offering a competitive and scalable solution for smart parking systems. The remainder of this paper is organized as follows: Section 2 discusses related work, Section 3 details our proposed methodology, Section 4 presents our experimental setup and results, and Section 5 offers concluding remarks and future directions.

2 Related Work

2.1 Automatic Parking Spots Localization

Parking spot localization, crucial during system installation or camera perspective changes, has evolved through three main approaches:

Traditional Image Processing: Early methods employed classical computer vision techniques like perspective transformation and edge detection [8,31]. While effective in controlled environments, these struggled with real-world variability.

Chessboard Approaches: These methods use homography transformations to generate bird's-eye views of parking lots [16, 28]. However, they can produce false positives for passing vehicles.

Deep Learning Approaches: Recent advancements include CNN-based methods using Mask R-CNN [4,9] and transformer-based models [19,29]. Despite their sophistication, most generate perpendicular bounding boxes, misaligning with angled parking spots.

Our proposed PakSke module addresses this limitation by automatically aligning bounding boxes with actual parking spot angles. We employ deformable DETR [33] as our detection backbone for multi-scale object detection.

2.2 Parking Spots Status Identification

Parking spot status identification employs two main approaches: classification and detection.

Classification methods treat each spot independently, using either feature extraction or deep learning techniques. Feature extraction methods, like those by Almeida et al. [5] and Suwignyo et al. [25], pre-process images to extract salient features before classification. Deep learning approaches, exemplified by Nyambal and Klein [17], integrate feature extraction and classification within a representation learning framework. While effective, these methods often rely on manually defined spots or perspective transformations, limiting scalability.

Detection approaches unify localization and classification tasks, offering increased flexibility and faster inference. Two-stage detectors, like Faster R-CNN [23], first propose regions of interest before classification. One-stage detectors, such as RetinaNet adaptations [18], perform both tasks simultaneously. Recent advancements include attention mechanisms and custom network backbones [11] to improve accuracy.

Our proposed methodology innovates within the two-stage detector framework, applying the initial detector to ROI-filtered frames and using a mapping schema for efficient status determination across all parking slots.

3 Methodology

As describe in Section 1, our proposed method is divided into two modules: (1) PakLoc for automatic parking spots localization task and (2) PakSta for parking spots status identification task. The detail architecture is visualized in Figure 2. The automobile detector plays a vital role in our proposed concept. The component in question holds significant importance in both the PakLoc and PakSta modules, since its performance directly influences the overall consequences of the architecture. Consequently, we have partitioned this section into three distinct components, namely the Vehicle Detector, PakLoc, and PakSta.

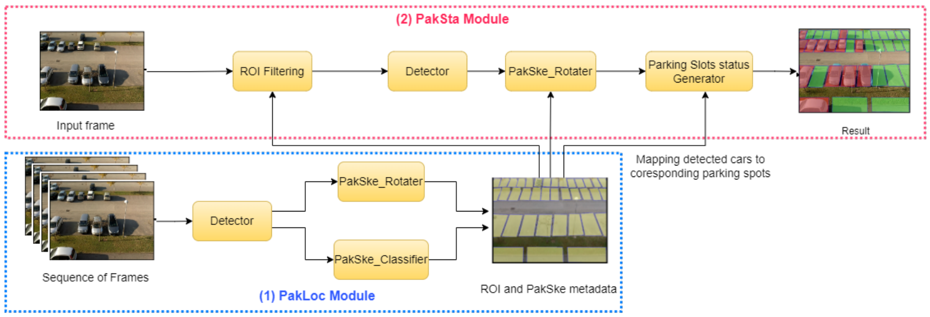

Fig. 2: Proposed Architecture. It includes two main modules: (1) PakLoc for automatic parking spots localization task and (2) PakSta for parking spots status identification task.

3.1 Vehicle Detector

There exist two methodologies for deep learning detection frameworks: the CNN-based approach and the Transformer-based model. The efficacy of transformer-based models in extracting various diversified discriminative parts of information and fine-grained features has been demonstrated in [12]. Furthermore, the PKlot dataset [10] includes variations in car scale and viewpoint. As a result, in this study, the we employ a transformer-based object detection model known as deformable DETR [33] that demonstrates effective performance in detecting objects of varying scales. This detector model has been pre-trained using the VeRi-776 [15] and CityFlow [26] datasets. The VeRi-776 dataset has a total of 49,357 images depicting 776 distinct vehicles. These images were captured using 20 different cameras. On the other hand, the CityFlow dataset consists of over 229,680 labeled bounding boxes of 666 distinct cars. These bounding boxes were obtained from 40 cameras positioned at 10 different junctions. The detector was pre-trained for 70 epochs using the same set of parameters as described in the original research. The detector model is fine-tuned on the training set of the PKlot dataset for a total of 40 epochs.

3.2 PakLoc - Automatic Parking Spots Localization

PakLoc tackles the challenge of automatic parking spot localization through an innovative approach of vehicle movement tracking across consecutive frames. This method capitalizes on the temporal nature of parking lot surveillance to differentiate between moving vehicles and parked cars.

At its core, PakLoc operates by processing successive frames to detect cars and generate corresponding bounding boxes. These newly identified bounding boxes are then compared against an existing inventory of tracked vehicles using the Intersection over Union (IoU) metric. A vehicle is deemed stationary when its IoU value consistently exceeds a predefined threshold θ for a specified number of frames γ. Consequently, locations where vehicles remain stationary are designated as potential parking spots. The effectiveness of this approach is contingent upon the judicious selection of the IoU threshold θ and frame threshold γ. Our empirical studies, elaborated in Section 4.2, reveal optimal performance with $\theta = 0.75$ and $\gamma = 4$ for the PKLot dataset, which captures images at 5-minute intervals. These parameters, however, may require adjustment when applied to datasets with different temporal characteristics.

To address the prevalent issue of non-perpendicular parking spots (as illustrated in Figure 1), we introduce the novel PakSke layers. During the training phase, the PakSke_Rotater layer determines optimal triplet hyperparameters $[angle, width\_scaling, height\_scaling]$ to align detected spots with ground truth labels, as depicted in Figure 3.

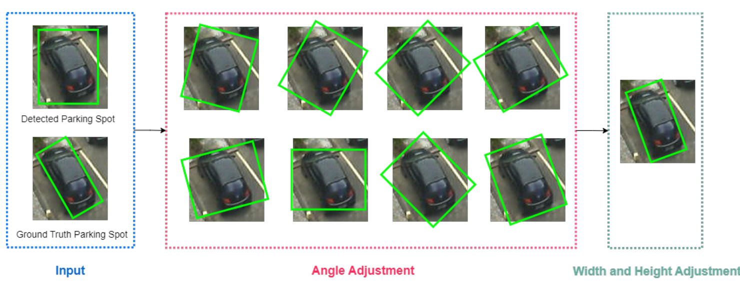

Fig. 3: PakSke_Rotater: Automatically adjusting parking slot angle, width, and height by maximizing IoU with ground truth.

For inference and deployment on new datasets where ground truth labels are unavailable, we employ the PakSke_classifier. This CNN-based classifier, derived from the CmAlexnet architecture [21], predicts the optimal triplet for newly detected parking spots. The PakSke_classifier is trained using the output produced by the PakSke_Rotater layer during the training phase, ensuring adaptability to various parking lot configurations. The culmination of PakLoc's processing is the generation of comprehensive Parking Spots metadata, encapsulated as [CameraId, x, y, w, h, angle, $width\_scaling$, $height\_scaling$]. This metadata provides a rich description of each identified parking spot, including its location, dimensions, and orientation adjustments. Through this multifaceted approach, PakLoc achieves accurate and adaptive parking spot localization across diverse parking lot layouts, overcoming the limitations of traditional perpendicular bounding box methods.

3.3 PakSta - Parking Spots Status Identification

The objective of PakSta is to forecast the state of parking spaces identified from the PakLoc. The PakSta system receives camera video input in a sequential manner, processing each frame individually. These frames are then sent via a region of interest (ROI) filtering layer. The ROI in question corresponds to the coordinates of parking spots, which are obtained from the parking spot metadata extracted in PakLoc. Subsequently, the filtered image is inputted into the detector in order to identify the presence of a car within the image. In the parking slots status generator layer, every identified vehicle is assigned a parking spot in the parking spot metadata and its bounding box is adjusted using the appropriate triplet parameter $[angle, width\_scaling, height\_scaling]$. Ultimately, the parking spaces that are mapped to any identified vehicle are categorized as occupied, whereas the remaining spaces are categorized as vacant.

4 Experiments and Results

4.1 Datasets, Evaluation Metrics and Baseline

Dataset:, Our study utilizes three datasets, each serving a distinct purpose in our model's development and evaluation. *PKlot Dataset* [10]: Our primary testing dataset, PKlot comprises 12,417 images, capturing 695,851 manually annotated spaces under various weather conditions. Images are taken at 5-minute intervals, providing a comprehensive view of parking dynamics across different times and environments. *CityFlow Dataset* [26]: Used for detector pre-training, CityFlow offers 229,680 labeled bounding boxes of 666 vehicles from 40 cameras across 10 intersections. This dataset provides diverse urban traffic scenarios, enhancing our model's ability to detect vehicles in complex environments. *Veri-776 Dataset* [15]: Also used in detector pre-training, VeRi-776 contains 49,357 images of 776 vehicles captured from multiple angles and under various conditions. Its real-world traffic scenarios and detailed vehicle annotations contribute to our model's robustness in diverse settings.

Evaluation metric:, for PakLoc, we primarily use recall-based metrics: AR75 (average recall at IoU threshold 0.75), mAR40_90 (mean average recall for IoU thresholds 0.4 to 0.9), and AP50 (average precision at IoU threshold 0.5) for comparison with related work. For parking space status identification, we employ AP75 (average precision at IoU threshold 0.75) to assess accuracy.

Baseline:, given the lack of standardized benchmarks for automatic parking spot localization, we selected representative works [6, 13, 18, 20] with comparable metrics for evaluation. For parking spot status identification, PakSta is benchmarked against relevant studies [11, 18, 22] as discussed in Section 2.

4.2 Experimental Results

This section is divided into two subsections to evaluate the performance of PakLoc and PakSta.

PakLoc Performance The evaluation of PakLoc is conducted using the test set of the PKLot dataset. Firstly, as discussed in Section 3.2, an ablation study is conducted to determine the best parameter IoU threshold θ. In this ablation study, we assess the results (AR) of PakLoc by varying the parameter θ throughout the range of 0.4 to 0.9, with a step size of 0.05. The experimental result indicates that the ideal value of θ is determined to be 0.75. Then, we select the θ as 0.75 for all next experiment. It is worth noting that in this ablation investigation, we adjusted the frame threshold γ to 4. To demonstrate the impact of the PakSke layer, we present outcomes obtained from both scenarios: one with the inclusion of the PakSke layer and the other without it using these three metrics. The results in Table 1 demonstrates the efficiency of our proposed PakSke layers. Using PakSke layers increases all three metrics by at least 6%.

	AR75	**mAR40_90**	**AP50**
Without PakSke	88.31	74.38	80.23
With PakSke	**94.25**	**82.11**	**86.37**

Table 1: PakLoc result on testset of PKLot with and without PakSke layer

Lastly, we compare PakLoc's performance to other baselines described above. The results presented in Table 2 indicate that our proposed method outperforms all prior work using the same dataset with 86.4% AP50. It even achieve a better result with 92.7% AP75.

Method/Ref	**Backbone**	**Test Set**	**Metric**	**Result**
Faster PSP [13]	Faster-RCNN	CNRPark-EXT	AP50	83.1
Auto PSP [20]	Yolo4	CNRPark-EXT	AP50	**97.6**
Realtime PSP [18]	Resnet and faster RCNN	PKLot	AP50	63.6
Cascade PSP [6]	Cascade Mask R-CNN	PKLot	AP50	59.1
PakLoc (ours)	Deformable DETR	PKLot	AP50	**86.4**
PakLoc (ours)	Deformable DETR	PKLot	AP75	**92.7**

Table 2: PakLoc result on testset of PKLot and other baselines method

PakSta Performance A comparison is made between the results of PakStat and three additional baseline models [11, 18, 22]. The findings are presented in Table 3. The approach presented in [11] achieved the highest AP75 score of 98%.

However, it should be noted that this system relied on the manual annotation of parking places during the training phase. This factor restricts the practical implementation of the paradigm in real-world scenarios. In contrast, our suggested solution, PakSta, does not necessitate human labeling of parking slots for implementation in a fresh dataset or a real parking lot environment. Furthermore, PakSta was able to obtain a notable outcome of 93.6% AP75, positioning it as the second best performer. This even surpasses the strategy employed in the study [22], where manually labeled parking spaces data was utilized. In addition, the results table further demonstrates the effectiveness of PakSke layers by indicating that they contribute a positive impact (improve 6%) on the ultimate outcome of PakSta.

Method/Ref	**Backbone**	**Data**	**Use Manually Label Of Parking Spots**	**AP75**
POD [18]	RetinaNet	PKLot	No	61.8
Yolo SPS [22]	Yolo3	PKLot	Yes	93.3
OcpDept [11]	MBN-FPN	PKLot	Yes	**98.0**
PakSta without PakSke (ours)	Deformable DETR	PKLot	No	87.3
PakSta with PakSke (ours)	Deformable DETR	PKLot	No	**93.6**

Table 3: PakSta result on testset of PKLot and other baselines method

5 Conclusion

In this paper, two new approaches, PakLoc and PakSta, are proposed to address the problems of automatic parking spot localization and parking spot status identification, respectively. Both of these methods demonstrate superior performance compared to the existing approaches in the given context. Additionally, we propose the incorporation of PakSke layers as a means to enhance the performance of these methods. The utilization of PakSke layers as a plug-in module is applicable in a comparable manner. In the forthcoming period, our objective is to construct a comprehensive smart parking system utilizing the way we have put forth.

Acknowledgement

This material is based upon work partially supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Award Number DE-EE0009208. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

1. Demand-responsive parking pricing. https://www.sfmta.com/demand-responsive-parking-pricing (2017), Accessed: 07/31/2023 2
2. Smart parking – a silver bullet for parking pain. https://inrix.com/blog/parkingsurvey/ (2017), Accessed: 07/31/2023 1
3. United nation. https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html (2018), Accessed: 07/31/2023 1
4. Agrawal, T., Urolagin, S.: Multi-angle parking detection system using mask r-cnn. In: Proceedings of the 2020 2nd international conference on big data engineering and technology. pp. 76–80 (2020) 3
5. Almeida, P., Oliveira, L.S., Silva, E., Britto, A., Koerich, A.: Parking space detection using textural descriptors. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics. pp. 3603–3608. IEEE (2013) 3
6. de Almeida, P.R., Alves, J.H., Oliveira, L.S., Hochuli, A.G., Fröhlich, J.V., Krauel, R.A.: Vehicle occurrence-based parking space detection. arXiv preprint arXiv:2306.09940 (2023) 6, 7
7. de Almeida, P.R.L., Alves, J.H., Parpinelli, R.S., Barddal, J.P.: A systematic review on computer vision-based parking lot management applied on public datasets. Expert Systems with Applications **198**, 116731 (2022) 2
8. Bohush, R., Yarashevich, P., Ablameyko, S., Kalganova, T.: Extraction of image parking spaces in intelligent video surveillance systems. Machine Graphics and Vision **27** (2018) 3
9. Coleiro, A., Scerri, D., Briffa, I.: Car parking detection in a typical village core street using public camera feeds. In: 2020 IEEE 10th international conference on consumer electronics (ICCE-Berlin). pp. 1–6. IEEE (2020) 3
10. De Almeida, P.R., Oliveira, L.S., Britto Jr, A.S., Silva Jr, E.J., Koerich, A.L.: Pklot–a robust dataset for parking lot classification. Expert Systems with Applications **42**, 4937–4949 (2015) 2, 4, 6
11. Duong, T.L., Le, V.D., Bui, T.C., To, H.T.: Towards an error-free deep occupancy detector for smart camera parking system. In: European Conference on Computer Vision. pp. 163–178. Springer (2022) 3, 6, 7, 8
12. He, S., Luo, H., Wang, P., Wang, F., Li, H., Jiang, W.: Transreid: Transformer-based object re-identification. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 15013–15022 (2021) 4
13. Kirtibhai Patel, R., Meduri, P.: Faster r-cnn based automatic parking space detection. In: Proceedings of the 2020 3rd International Conference on Machine Learning and Machine Intelligence. pp. 105–109 (2020) 6, 7
14. Lin, T., Rivano, H., Le Mouël, F.: A survey of smart parking solutions. IEEE Transactions on Intelligent Transportation Systems **18**(12), 3229–3253 (2017) 2
15. Liu, X., Liu, W., Mei, T., Ma, H.: A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14. pp. 869–884. Springer (2016) 4, 6
16. Nieto, R.M., Garcia-Martin, A., Hauptmann, A.G., Martinez, J.M.: Automatic vacant parking places management system using multicamera vehicle detection. IEEE Transactions on Intelligent Transportation Systems **20**(3), 1069–1080 (2018) 3
17. Nyambal, J., Klein, R.: Automated parking space detection using convolutional neural networks. In: 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech). pp. 1–6. IEEE (2017) 2, 3

18. Padmasiri, H., Madurawe, R., Abeysinghe, C., Meedeniya, D.: Automated vehicle parking occupancy detection in real-time. In: 2020 Moratuwa engineering research conference (MERCon). pp. 1–6. IEEE (2020) 3, 6, 7, 8
19. Pannerselvam, K.: Adaptive parking slot occupancy detection using vision transformer and llie. In: 2021 IEEE International Smart Cities Conference (ISC2). pp. 1–7. IEEE (2021) 3
20. Patel, R., Meduri, P.: Car detection based algorithm for automatic parking space detection. In: 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA). pp. 1418–1423. IEEE (2020) 6, 7
21. Rahman, S., Ramli, M., Arnia, F., Sembiring, A., Muharar, R.: Convolutional neural network customization for parking occupancy detection. In: 2020 International Conference on Electrical Engineering and Informatics (ICELTICs). pp. 1–6. IEEE (2020) 5
22. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018) 6, 7, 8
23. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015) 3
24. Sen, R., Tran, T., Khaleghian, S., Pugliese, P., Sartipi, M., Neema, H., Dubey, A.: Bte-sim: Fast simulation environment for public transportation. In: 2022 IEEE International Conference on Big Data (Big Data). pp. 2886–2894. IEEE (2022) 1
25. Suwignyo, M.A., Setyawan, I., Yohanes, B.W.: Parking space detection using quaternionic local ranking binary pattern. In: 2018 International Seminar on Application for Technology of Information and Communication. pp. 351–355. IEEE (2018) 3
26. Tang, Z., Naphade, M., Liu, M.Y., Yang, X., Birchfield, S., Wang, S., Kumar, R., Anastasiu, D., Hwang, J.N.: Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8797–8806 (2019) 4, 6
27. Valipour, S., Siam, M., Stroulia, E., Jagersand, M.: Parking-stall vacancy indicator system, based on deep convolutional neural networks. In: 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT). pp. 655–660. IEEE (2016) 2
28. Vítek, S., Melničuk, P.: A distributed wireless camera system for the management of parking spaces. Sensors **18**(1), 69 (2017) 3
29. Wang, L., Zhang, X., Zeng, W., Liu, W., Yang, L., Li, J., Liu, H.: Global perception-based robust parking space detection using a low-cost camera. IEEE Transactions on Intelligent Vehicles **8**(2), 1439–1448 (2022) 3
30. Xu, D., Chen, Y., Ivanovic, B., Pavone, M.: Bits: Bi-level imitation for traffic simulation. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 2929–2936. IEEE (2023) 1
31. Zhang, W., Yan, J., Yu, C.: Smart parking system based on convolutional neural network models. In: 2019 6th International Conference on Information Science and Control Engineering (ICISCE). pp. 561–566. IEEE (2019) 3
32. Zhong, Z., Rempe, D., Xu, D., Chen, Y., Veer, S., Che, T., Ray, B., Pavone, M.: Guided conditional diffusion for controllable traffic simulation. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 3560–3566. IEEE (2023) 1
33. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020) 3, 4