LaCNet : real-time end-to-end arbitrary-shaped lane and curb detection with instance segmentation network

Zhou, Hui; Wang, Han; Zhang, Handuo; Hasith, Karunasekera

2020

Zhou, H., Wang, H., Zhang, H., & Hasith, K. (2020). LaCNet : real-time end-to-end arbitrary-shaped lane and curb detection with instance segmentation network. Proceedings of International Conference on Control, Automation, Robotics and Vision (ICARCV), 184-189. doi:10.1109/ICARCV50220.2020.9305341

<https://hdl.handle.net/10356/146087>

<https://doi.org/10.1109/ICARCV50220.2020.9305341>

© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at:
<https://doi.org/10.1109/ICARCV50220.2020.9305341>

Downloaded on 26 Apr 2023 21:02:48 SGT

LaCNet: Real-time End-to-End Arbitrary-shaped Lane and Curb Detection with Instance Segmentation Network

Hui Zhou¹, Han Wang², Handuo Zhang³ and Karunasekera Hasith⁴

Abstract—Accurate and robust detection of lane and curb in urban areas is essential to many real-world intelligent vehicle applications. Existing vision-based researches treat lane detection and curb detection separately due to the nature of curb detection problem relying on 3D features, which is not efficient when driving in a real world. This paper presents a unified network to incorporate these two tasks together by taking advantage of the powerful feature learning ability brought by deep convolutional neural networks. The resulting unified network provides valuable road boundary information by curb detection even when lane markings are not visible during vehicle navigation. Another significant capability coming with the proposed method is able to accurately differentiate various lane and curb instances with tiny gaps or complex spatial relationships which are thought as the biggest challenge in real driving situations. To achieve this, the network is specially designed to guide lane and curb instances to learnable kernel through pixel grouping. This learnable kernel is capable to handle any number of arbitrary-shaped lanes and curbs no matter which angle the vehicle is heading at. In the end, the presented approach is evaluated on two datasets (BDD100K and self-collected dataset) scoring 32 FPS processing speed. Results are very encouraging with more than 98% F_1 measure for both lane and curb detection on the self-collected dataset.

I. INTRODUCTION

Localizing lanes in real-time in urban areas has been active research over the last few decades. It is an important component for various intelligent vehicle applications, such as lane level localization [1], [2] and navigation [3], [4], Lane Departure Warning Systems (LDWS) as well as Advanced Driver Assistance Systems (ADAS). Curb detection is also a necessary component since it can provide road boundary information when lane markings can not be found. It can be integrated to vehicle localization [5], navigation [6] and lateral control. However, existing vision-based researches take these two components as two separate tasks because the feature extraction module is completely different between them and the detection of curb usually require additional 3D information [7].

With the fast development of deep learning approaches, the conventional lane detection methods [8]–[11] using hand-crafted features are outperformed significantly due to lack of robustness considering the real driving situations (e.g. emerging, ending, spitting and merging of lane markings, faded lanes, heavily curved lanes, zig-zag lanes, on/off ramp lanes and disturbance of other road writings). These end-to-end deep learning lane detection methods [12]–[17] usually

Fig. 1. An example result of lane and curb detection on the BDD100K dataset. Red dots are curb and green dots are lane. These dots are drew with 5-pixel vertical spacing.

adopt semantic segmentation framework to predict the label of each pixel to be lane or not. And their research paradigm mainly focus on capturing richer context information to improve lane detection accuracy through multi-task learning and message passing [17]. For example, geometric constraints are introduced in [18] to improve the overall lane detection performance. Multi-task including the lane boundary segmentation and road area segmentation are combined in this framework. [14] performs sequential message passing between the outputs of high-level layers to better exploit the structural information of lane. Generative Adversarial Network (GAN) is integrated into the framework in [15] to resemble more information of labels. Only the work of [16] follows the instance segmentation framework and the lane pixels are classified into different lane instances. However, random number and arbitrary-shaped lane detection, one of the most challenging aspects in real driving urban scenarios is seldomly considered. This situation also exists in curb detection.

Curb detection is more challenging than lane detection because curb feature is not constant regarding its texture, geometry, or gradient information. The typical solutions using cameras for curb detection are to combine multi-cue (e.g. image intensity and texture) information with 3D height elevation constraints [19]–[21]. However, height elevation becomes indistinct with increasing distance to the vehicle. Thus, low height and distant curbs are difficult to be detected and sensitive to noise even without considering illumination changes and occlusions. In addition, curbs in urban streets often have complex geometry and orientation with respect to vehicle at roundabouts, junctions, parking space, traffic isles and etc. Existing approaches in the literature can not handle these arbitrary number and geometric shape of curbs. In addition, there is only one CNN-based deep learning research on curb detection [22], which selects SegNet [23],

All authors are with School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798. {¹hzhou005, ³hzhang032, ⁴karu0009}@e.ntu.edu.sg, ²hw@ntu.edu.sg.

Fig. 2. LaCNet network architecture designed for lane and curb detection. The network predicts lane and curb regions, similarity vectors and kernels to generate instances. After masking lane and curb regions, the similarity vectors are grouped and assigned to their respective kernels. L_c , L_g , and L_{dis} comprise the total loss. Red dots are curb and green dots are lane in the output result.

DeepLabv2 [24] and DeepLabv3+ [25] for curb detection to compare with the proposed method in [22]. Due to the limited 88 KITTI training images, the performance of SegNet, DeepLabv2, and DeepLabv3+ based curb detection is not satisfied. Therefore, how to design an efficient, unified end-to-end random number and arbitrary-shaped lane and curb detection deep neural network remains to be solved.

In this paper, we seek to present solutions to an arbitrary-shaped and arbitrary number lane and curb detector, which should achieve a good balance between speed and performance. The challenge originates from the observation that curb and lane sometimes only have tiny gaps to each other and the geometric relationship between them are very sophisticated. Simple post-processing on the semantic segmentation prediction results can hardly differentiate them. We draw inspiration from widely used deep metric learning [16], [26], [27] and map lane and curb pixels onto an embedding space where pixels inside the same instance have similar embedding vectors and pixels belonging to different lane or curb are enforced to be different. Hence, the designed network named LaCNet following the pipeline as: i) Predicting the lane and curb regions by segmentation network; ii) Reconstructing lane and curb instances from predicted kernels on embedding space. In this paper, LaCNet is built upon FCN-8s [28] to predict lane and curb regions. To reconstruct and differentiate each lane and curb instance accurately, we propose to group the lane and curb pixels on embedding space to correct kernels through similarity vectors.

To demonstrate the effectiveness of the proposed LaCNet, we conduct extensive experiments on BDD100K [29], which is the only dataset that contains road curb annotations, and self-collected real driving dataset in Singapore with labels generated by Scalabel [30] open-sourced annotation tool brought by Berkeley DeepDrive.

In summary, our contributions are three-fold: (1) we propose a unified end-to-end arbitrary-shaped lane and curb detection deep neural network named LaCNet that is applicable to the sophisticated real driving urban scenarios. (2) The pixel grouping mechanism by similarity vector is proposed which can be learned by the network. This mechanism can

selectively group pixels around the kernel and thus facilitate the differentiation between long, thin lanes and curbs with complex geometric relationships. (3) The proposed method achieves satisfactory performance on the two datasets with the real-time processing speed of 32 FPS at the same time. To the best of our knowledge, this is the first network that can simultaneously detect random number and arbitrary-shaped lane and curb in real-time.

II. METHOD

A. Overall Architecture

We address the multi-lane and multi-curb detection problem under the fully convolutional network architecture. LaCNet's network architecture consists of typical encoder-decoder network [28], and FCN-8s is adopted to predict lane and curb regions, and then followed by a two branches which are class segmentation and pixel grouping as shown in Figure 2. The output of class segmentation is a 3 channel image which indicates the probability score of lane, curb and background. The output of pixel grouping is an N channel image. N is dimension of the similarity vector, and we set it as 4 in all experiments. The pixel grouping are obtained by passing the last layer output of the upsampling process of FCN-8s through consecutive 1×1 convolution layer and ReLU layer with 4 output channels.

B. Class Segmentation

The class segmentation is trained to produce a three class segmentation map predicting which pixels are lane and curb and the remaining pixels belong to background. The loss function L_c of class segmentation is the sparse softmax cross entropy loss function defined as Equ. 1:

$$L_c = - \sum_i^n y_i \ln (S(\logits_i)) \quad (1)$$

$$S(\logits_i) = \frac{e^{\logits_i}}{\sum_i^n e^{\logits_i}} \quad (2)$$

where n is the class number. y_i is the i th class label. $\logits_i$ is the last layer output of the class segmentation branch.

$S(\logits_i)$ is to produce the probability of assigning to i class by using softmax layer.

Since the lane, curb and background pixel numbers are highly unbalanced, we used a custom class weighting scheme defined as Equ. 3:

$$W_i = \frac{1}{\ln(c + P_i)} \quad (3)$$

$$P_i = \frac{C_i}{\sum_i^n C_i} \quad (4)$$

where i is the i th class, n is the class number, c is a hyper-parameter, which we set to 1.02. Thus, the weights W_i are bounded in the interval of $[1, 50]$. P_i is the occurring probability of i th class. C_i is the number of pixels in i th class.

C. Pixel Grouping

To rebuild the lane and curb instances, the lane and curb prediction pixels should be merged into kernels which can be automatically learned during training phase. *Pixel Grouping* is proposed to merge the lane and curb pixels towards correct kernels through similarity vector. The kernels can be considered as instance centers. Thus, to guide the lane and curb pixels to the corresponding kernels, the distance between similarity vector of lane and curb pixel and kernel of the same instance should be minimized. In the training phase, the grouping loss L_g is adopted to force intra instance distance minimized, as shown in Equ. 5.

$$L_g = \frac{1}{C} \sum_{c=1}^C \frac{1}{N_c} \sum_{p \in N_c} D(p, K_c) \quad (5)$$

$$D(p, K_c) = \max(0, \|S(p) - S(K_c)\| - r_g)^2 \quad (6)$$

where C is the number of lanes and curbs (clusters), N_c is the pixel number in instance c , $D(p, K_c)$ defines the similarity vector distance between pixel p and kernel K_c of instance c . $S(p)$ is the similarity vector of pixel p . $S(K_c)$ is the similarity vector of kernel K_c , which is the mean vector of pixel embedding and can be obtained by calculating $\sum_{p \in N_c} \frac{S(p)}{|N_c|}$. $\max(\cdot)$ means that the grouping force will only be activated when the similarity vector distance is further from r_g .

In order to differentiate different lane and curb instances, the similarity vector distance of different kernels should be maximized. We use discriminative loss L_{dis} as shown in Equ. 7 to describe this rule during training.

$$L_{dis} = \frac{1}{C(C-1)} \sum_{c_a=1}^C \sum_{c_b=1, c_b \neq c_a}^C D(K_{c_a}, K_{c_b}) \quad (7)$$

$$D(K_{c_a}, K_{c_b}) = \max(0, r_d - \|S(K_{c_a}) - S(K_{c_b})\|)^2 \quad (8)$$

where $D(K_{c_a}, K_{c_b})$ defines the similarity vector distance between kernel K_{c_a} and kernel K_{c_b} . $\max(\cdot)$ means that the

the similarity vector distance among the kernels should be not less than r_d .

With the grouping loss function L_g and discriminative loss L_{dis} , the network is trained so that the each lane and curb instance will be grouped together (similarity vector distance smaller than r_g), and different lanes and curbs are lay further than r_d from each other.

In the testing phase, the class segmentation output is adopted as a mask to obtain predicted similarity vector firstly rather than directly perform similarity vector clustering. This can ensure the distinguish between lane and curb. If we set $r_d > 6r_g$ in the above training, then during inference, we can randomly select an unlabeled predicted similarity vector and apply threshold around its value with radius of $2r_g$ to group all lane and curb pixels belonging to the same instance. Then we update the mean pixel similarity vector and use the new mean to threshold again by applying mean-shift algorithm [31] until mean convergence. Another pixel without assigning label will be selected to repeat the whole process until all pixels are labeled.

D. Loss Function

The overall loss function L can be formulated as Equ. 9:

$$L = \alpha L_c + \beta L_g + \gamma L_{dis} \quad (9)$$

where α, β, γ are used to assign proper importance among L_c, L_g, L_{dis} , and we set them as equally weighted $\alpha = \beta = \gamma = 0.5$ in all experiments.

III. EXPERIMENTS

A. Generated Labels

To construct the lane and curb ground truth labels for the self-collected dataset, we use the open-sourced Scalabel annotation tool [30]. The ground-truth lane and curb points needs only a few points to form a connected line per lane or per curb, thus can improve the labeling efficiency. Since lane and curb are long and thin, the annotated pixels is far fewer than the background pixels. Therefore, the width of lane and curb labels is then increased with thickness 5 on each lane and curb line. The training samples consist of three components: the original image, a class segmentation label and a instance segmentation label. The class segmentation use 255 to represent the curb region, 127 to represent the lane region and 0 for the rest. The instance use different pixel value to represent different lane or curb region and 0 for the rest.

B. Dataset

1) BDD 100K

Berkeley DeepDrive (BDD) dataset [29] is the only dataset which provides both lane and curb anotations for training LaCNet, which was released in 2018. The dataset consists of 70K training images, 10K validation images and 20K testing images, under diverse weather conditions and at different daytimes in real traffic urban streets. BDD100K is very challenging considering the large amount of images and

Fig. 3. Qualitative results of LaCNet. (a)-(d) are examples of lane and curb detection on BDD100K dataset (left: lane and curb ground truth; right: detection results). Red dots are curb and green dots are lane. These dots are drew with 5-pixel vertical spacing on the fitted second order polynomial line.

complex road scenarios. The annotations is a json format and we can directly extract images with lane and road curb attributes as training and testing dataset. Similar to the self-collected dataset, the instance ground truth for BDD100K uses different grayscale values to represent different lane and curb region and 0 for the rest, and the thickness is increased to 5.

2) Self-collected data

To further test the performance of LaCNet in the real driving world, we also conducted lane and curb evaluation on the self-collected sequences with the vehicle equipped with cameras as shown in Figure 4. The collected ROS bag sequence 01-07 contain various road situations as described in Table I. How to efficiently label these large amount of successive frames and at the same time keep high detection accuracy on the whole ROS bag after training around 1000 selected frames is very tricky. Due to the differentiation of

Fig. 4. Our experimental platform equipped with cameras to collect data.

complex lane or curb geometry is difficult, we specifically increased the number of images of challenging road scenarios. Finally, we extracted 964 images for training and 164 images for validation, and the whole ROS bag as the testing data.

C. Evaluation Metrics

To facilitate evaluation of the LaCNet performance, we follow the literature and use the following evaluation metrics

TABLE I. Test image sequence description

Sequence	Description	Bag time
seq01	To detect lane and curb on multiple lanes	59s
seq02	To detect lane and curb when approaching traffic junction	76s
seq03	To detect lane and curb on curvy roads	136s
seq04	To detect curb at roundabout and lane after exit roundabout	102s
seq05	To detect lane and curb on/off slope	383s
seq06	To detect lane and curb on curvy roads and junctions	63s
seq07	To detect curb at roundabout	34s

for each dataset.

1) BDD 100K

We follow the official metric (*Accuracy*) of TuSimple [32] which is the largest lane detection dataset before 2018. *Accuracy* is defined as the average correct lane and curb pixel number compared to the ground truth. A lane and curb pixel is correct when the pixel is within the threshold (30 pixels) of the ground-truth labels.

$$Accuracy = \frac{N_p}{N_g} \quad (10)$$

where N_p is the number of predicted lane and curb points, N_g is the number of ground-truth points in the image.

2) Self-collected data

Since it is impossible to evaluate every frame detection result in the self-collected ROS bag image data, the lane and curb detection results are saved every 5 frames of sequence seq01 to seq07 automatically when running the ROS bag. Predictions whose fitted line align with more than 90% within 30 pixels width of the real lane or curb are treated as true positives (TP). F_1 measure is used as the evaluation metric, which is defined as Equ. 11.

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (11)$$

$$Precision = \frac{TP}{TP + FP} \quad (12)$$

$$Recall = \frac{TP}{TP + FN} \quad (13)$$

TP(True Positives)- Detected in both ground truth and results using our method.

FN(False Negatives) - Detected in ground truth but not detected in results.

FP (False Positives) - Detected in results but not in ground truth.

D. Implementation Details

We use the VGG16 [33] pretrained on ImageNet [34] as our base encoder stage. The image resolution of BDD100K and self-collected data are resized to 512×256 . In the pixel grouping process, $r_g = 0.5$, $r_d = 3$ in the experiments. LaCNet is trained with hyper parameters batch size = 8, initialized learning rate is 0.0005 and decrease by multiply 0.1 every 100000 epochs. The total number of training episodes is set as 200000. The loss is minimized by stochastic gradient descent (SGD). The two dataset training strategies adopted are both training from scratch.

E. Results

We have tested our lane and curb detection algorithm on the two dataset. The platform is on a normal PC with Nvidia GeForce GTX 1080, Intel(R) Core i7-4770 CPU @ 3.40GHz $\times$ 8 and 8 GB RAM. The processing speed is fast and can reach around 32 frames per second (fps), including the pre-processing time for image resizing.

1) BDD 100K

The accuracy on BDD 100K is 35.56%. Since this is the first work that simultaneously considers lane and curb instance detection, there is no similar work at this moment that can be compared to the best of our knowledge.

Since BDD 100K images are not consecutive between adjacent frames, we only show the qualitatively visualization of the lane and curb ground truth and its LaCNet output in Figure 3. As we can see, the network can differentiate adjacent curb and lane, and also it can handle the arbitrary orientation with respective to vehicle. To our surprise, the network can even differentiate nearby lines which is actually one lane that is labeled by two lines (leftside and rightside of one lane) in BDD100K.

2) Self-collected data

We conducted the evaluation of 1677 frames in total. The evaluation results of lane and curb detection on the self-collected dataset are shown in Table II and Table III. The average F_1 measure can reach 98.43% for lane detection and 98.65% for curb detection, which is really encouraging when the network needs to be deployed on new real-driving data.

Figure 5 give some results visualization on Nanyang Technological University (NTU) campus, which is challenging scenarios. From these figures, we can see the network can perform well on arbitrary-shaped lane and curb detection. A video of the results is available at <https://youtu.be/vZonvviCd60>.

TABLE II. Lane evaluation results

Sequence	Total frames	Precision	Recall	F_1 measure
seq01	193	100.00%	97.97%	98.97%
seq02	214	99.51%	97.18%	98.33%
seq03	498	99.79%	99.80%	99.79%
seq04	289	99.64%	97.22%	98.41%
seq05	181	100.00%	97.24%	98.59%
seq06	177	98.30%	100.00%	99.14%
seq07	125	99.13%	92.74%	95.83%
Average	1677	99.48%	97.45%	98.43%

TABLE III. Curb evaluation results

Sequence	Total frames	Precision	Recall	F_1 measure
seq01	193	100.00%	99.48%	99.74%
seq02	214	99.53%	100.00%	99.76%
seq03	498	100.00%	100.00%	100.00%
seq04	289	100.00%	99.65%	99.82%
seq05	181	96.13%	100.00%	98.02%
seq06	177	94.15%	96.41%	95.26%
seq07	125	99.17%	96.77%	97.95%
Average	1677	98.42%	98.90%	98.65%

IV. CONCLUSION

We have proposed a real-time instance segmentation deep neural network for end-to-end arbitrary-shaped lane and curb detection in this paper. This is the first work that can simultaneously detect lane and curb. The proposed method does not have the limitation of fixed number of lanes and curbs, and it can cope with lanes and curbs of complex geometry, orientation and even in the distance with respect to the vehicle. The processing speed is also fast and accuracy is really encouraging when the network needs to be deployed on new real-driving data. We are planning to investigate how to improve the neural network performance especially at nighttime in the future.

ACKNOWLEDGMENT

This research work was partially supported by ST Engineering NTU Corporate Lab in Nanyang Technological University funded by the National Research Foundation (NRF) Singapore.

REFERENCES

- [1] M. Schreiber, C. Knöppel, and U. Franke, "Laneloc: Lane marking based localization using highly accurate maps," in *Intelligent Vehicles Symposium (IV)*, 2013 IEEE, pp. 449–454, IEEE, 2013.
- [2] D. Cui, J. Xue, and N. Zheng, "Real-time global localization of robotic cars in lane level via lane marking detection and shape registration," *IEEE Trans. Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1039–1050, 2016.
- [3] C. Rose, J. Britt, J. Allen, and D. Bevely, "An integrated vehicle navigation system utilizing lane-detection and lateral position estimation systems in difficult environments for gps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 6, pp. 2615–2629, 2014.
- [4] T. Song, N. Capurso, X. Cheng, J. Yu, B. Chen, and W. Zhao, "Enhancing gps with lane-level navigation to facilitate highway driving," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 4579–4591, 2017.
- [5] H. Lee, J. Park, and W. Chung, "Curb feature based localization of a mobile robot in urban road environments," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2794–2799, IEEE, 2015.

Fig. 5. Examples of lane and curb detection on NTU campus (left: original image; right: detection result). Red dots are curb and green dots are lane. These dots are drawn with 5-pixel vertical spacing on the fitted second order polynomial line.

- [6] Q. Li, L. Chen, M. Li, S.-L. Shaw, and A. Nüchter, "A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 2, pp. 540–555, 2014.
- [7] C. Fernández, R. Izquierdo, D. F. Llorca, and M. Sotelo, "Road curb and lanes detection for autonomous driving on urban scenarios," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1964–1969, IEEE, 2014.
- [8] H. Yoo, U. Yang, and K. Sohn, "Gradient-enhancing conversion for illumination-robust lane detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1083–1094, 2013.
- [9] R. Gopalan, T. Hong, M. Shneier, and R. Chellappa, "A learning approach towards detection and tracking of lane markings," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1088–1098, 2012.
- [10] J. Hur, S.-N. Kang, and S.-W. Seo, "Multi-lane detection in urban driving environments using conditional random fields," in *Intelligent Vehicles Symposium (IV)*, 2013 IEEE, pp. 1297–1302, IEEE, 2013.
- [11] M. Revilloud, D. Gruyer, and M.-C. Rahal, "A new multi-agent approach for lane detection and tracking," in *Robotics and Automation (ICRA)*, 2016 IEEE International Conference on, pp. 3147–3153, IEEE, 2016.
- [12] Z. Chen and Z. Chen, "Rbnet: A deep neural network for unified road and road boundary detection," in *International Conference on Neural Information Processing*, pp. 677–687, Springer, 2017.
- [13] S. Lee, J. Kim, J. Shin Yoon, S. Shin, O. Bailo, N. Kim, T.-H. Lee, H. Seok Hong, S.-H. Han, and I. So Kweon, "Vpgnet: Vanishing point guided network for lane and road marking detection and recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1947–1955, 2017.
- [14] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: Spatial cnn for traffic scene understanding," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [15] M. Ghafoorian, C. Nugteren, N. Baka, O. Booi, and M. Hofmann, "Elgan: embedding loss driven generative adversarial networks for lane detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 0–0, 2018.
- [16] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards end-to-end lane detection: an instance segmentation approach," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 286–291, IEEE, 2018.
- [17] Y. Hou, Z. Ma, C. Liu, and C. C. Loy, "Learning lightweight lane detection cnns by self attention distillation," *arXiv preprint arXiv:1908.00821*, 2019.
- [18] J. Zhang, Y. Xu, B. Ni, and Z. Duan, "Geometric constrained joint lane segmentation and lane boundary detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 486–502, 2018.
- [19] J. Siegemund, D. Pfeiffer, U. Franke, and W. Förstner, "Curb reconstruction using conditional random fields," in *2010 IEEE Intelligent Vehicles Symposium*, pp. 203–210, IEEE, 2010.
- [20] J. Siegemund, U. Franke, and W. Förstner, "A temporal filter approach for detection and reconstruction of curbs and road surfaces based on conditional random fields," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, pp. 637–642, IEEE, 2011.
- [21] M. Enzweiler, P. Greiner, C. Knöppel, and U. Franke, "Towards multi-cue urban curb recognition," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 902–907, IEEE, 2013.
- [22] M. Cheng, Y. Zhang, Y. Su, J. M. Alvarez, and H. Kong, "Curb detection for road and sidewalk detection," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10330–10342, 2018.
- [23] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
- [25] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.
- [26] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy, "Semantic instance segmentation via deep metric learning," *arXiv preprint arXiv:1703.10277*, 2017.
- [27] B. De Brabandere, D. Neven, and L. Van Gool, "Semantic instance segmentation with a discriminative loss function," *arXiv preprint arXiv:1708.02551*, 2017.
- [28] 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*, pp. 3431–3440, 2015.
- [29] "Berkeley deepdrive." <https://bair.berkeley.edu/blog/2018/05/30/bdd/>.
- [30] "Scalabel: A scalable open-sourced annotation web tool brought by berkeley deepdrive." <https://www.scalabel.ai/>.
- [31] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on information theory*, vol. 21, no. 1, pp. 32–40, 1975.
- [32] "Tusimple." <http://benchmark.tusimple.ai/#/t/1/dataset>.
- [33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.