Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/43/20230840

Exploring key modules in lane detection through comparing two updated models

Haoxiang Xu[1,4,†], Yushuo Xu[2,†], Skye Shen[3,†]

[1]Beihang University, Beijing, 100191, China
[2]Changzhou Senior High School of Jiangsu Province, Changzhou, 213004, China
[3]Harrow International School Shanghai, Shanghai, 200137, China

[4]1747045175@qq.com
[†]These authors contributed equally to this work and should be considered co-first authors

Abstract. Lane detection is significant in the digital navigation system of modern vehicles. It is vita in lane switching and route planning. We want to explore what modules are critical in lane detection. In this work, we choose two representative models, Cross Layer Refinement Network (CLRNet) and Ultra Fast Lane Detector version 2 (UFLD-v2), for comparison, aiming at evaluating their performance and finding out what determines their performance or why they perform well. The first one is a top-down model while another is a bottom-top model, which represents they are built through different ideas. We use two common datasets, CULane and TuSimple, to train, validate, and test them. The experiment results are divided into quantitative results and qualitative results, by which we can conclude that global learning capability and immunity to interference are indispensable.

Keywords: lane detection, CLRNet, UFLD-v2, model comparison, key module.

1. Introduction

Currently, driving is a moderately dangerous activity due to various factors like poor designs of roads or reckless driving. This has caused it to turn into one of the most common causes of mortality around the world, regardless of the country's development level. However, we believe that implementation of autonomous driving could decrease at least a portion of these unnecessary accidents, save numerous lives, and reduce damages.

Autonomous driving, if brought to the masses properly and successfully, will have a big influence on our lives. Autonomous driving will free our hands from the torture of long drives and provide us with a safe environment so that we no longer have to worry about car accidents. What's more, it gives us more free time.

A fundamental aspect of autonomous driving is road detection and recognition. Lane detection can determine the direction between the current car position and the road lanes. If the vehicle deviates from the lane, it can be adjusted or alerted to avoid collisions and accidents. At the same time, by detecting the lane, the vehicle can be continuously controlled in the center of the lane, which will help vehicles become fully autonomous and allow our vehicles to drive for us.

© 2024 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/43/20230840

The problem we are trying to solve is to detect lanes in different scenarios to ensure the security of traffic. We investigated the current mainstream algorithms and selected two representative state-of-the-art models for detecting road lanes, CLRNet and UFLD-v2, which have completely different structural frameworks and can each represent some similar models. We reproduced their modeling algorithms and compared the advantages and disadvantages of the two models through experiments. We also extracted the core modules of lane detection, pointed out their problems, and explained some of our research ideas on these problems.

2. Related Works

Throughout the past decades, different approaches have been used to identify what is part of the road and the different roadside objects, most of which use machine learning. Traditionally, methods usually revolve around hand-crafted algorithms that leverage visual cues, such as HSI color models or edge extraction algorithms [1-3], or tracking when visual information is not sufficient [4,5]. Deep learning has been applied to road lane detection in recent years, showing improvements compared to the traditional methods. For autonomous driving, we explored six different models with moderate details, two of which are the topic of our report. Based on top view processing and Gaussian kernel filtering of road images, a model of lane detection in the street is presented by line detection and RANSAC spline fitting; it is efficient, robust, and capable of real-time lane detection at 50Hz (on an Intel Core2 2.4GHz machine), but false detection could occur due to road signs, passersby, right side curb, other vehicles, and some interfering lines on the streets [6,7] proposed a model that combines LaneNet, which add together binary segmentation and a loss function using a clustering way that is designed for one-shot instance segmentation, and H-Net, which estimates an ideal perspective that allows the lanes to be fitted with low-order polynomials. It is capable of detecting a huge number of lanes and coping with lane change strategies and changes in the ground-plane's slope. It managed to achieve 4th place in the TuSimple Competition (one of the ways the researcher compared their model to others') at the time of introduction. A model based on a variance of CNN called SCNN, allows it to identify both thin and long lane lines and larger objects while outperforming other models like ReNet and ResNet-101[8]. However, it slightly underperforms under conditions with dazzle light, curves, or crossroads compared to other models; this paper also put forward a new dataset — CULane, one of the datasets we're using. A new way of estimating road lanes through the way of solving the problem of weighted least-squares[9], and the weights used here are generated from a deep network based on the input image; it outperforms the researchers' baseline and runs at a much higher fps when compared to other models (70+ on a Nvidia 1080Ti), albeit at a slightly lower accuracy.

3. Method

The two lane detection models we investigated and selected as our topic are CLRNet and UFLD-v2 [10, 11]. We replicated their algorithms and built the models to validate their performance and explore the factors affecting lane detection. These two models were chosen for two reasons. Firstly, they each represent two different mainstream lane detection ideas: the first is a top-down structure that learns the picture as a whole, and the second is a bottom-up structure that extracts and fuses the picture features; secondly, they are the latest models of these two structures and represent the highest current performance level. We aim to train, validate, and test the two models using two different datasets to compare the advantages and disadvantages of the two structures and to figure out optimal solutions for lane detection.

3.1. CLRNet

CLRNet, or Cross Layer Refinement Network, entirely utilizes both high-level and low-level characteristics. In detail, it first employed high-level semantic features to detect lanes and then refined them to join low-level features together. In such a way, more contextual information can be exploited for lane detection and taking advantage of local detailed lane features to enhance the accuracy of localization. In addition, this model introduces two unique modules-RoIGather and Line IoU loss. RoIGather module is used to collect global information, which further improves the performance of the lane's feature representation. Line IoU loss is used as the loss function, which sees the lane line as a whole part to enhance the accuracy of localization. It proposes a coarse-to-fine mechanism like the feature pyramid extractor to detect lane lines using high-level features and adjust lane line positions using low-level features. It proposes RoIGather module to obtain global context information and Line IoU loss to optimize lane lines as a whole.

3.2. UFLD-v2

UFLD-v2, or Ultra Fast Lane Detection with Hybrid Anchor Driven Ordinal Classification, is a model that treats the road lane detection process as an anchor-driven ordered classification problem using global characteristics. Firstly, lanes are represented as discrete coordinates defined by horizontal or vertical anchors, named the anchor-driven method. Then, the representation is used to restate the detection task as an ordered classification problem, and the lane coordinates are obtained. The ULFD-v2 is a model of the top-down structure. The whole image is sent to the backbone network for overall processing. Then flattened results are sent to the MLP. Unlike wise-pixel segmentation, the MLP classifies the features and separates them into two branches. One is the existence branch, which is used to determine the existence of lanes. The other one is the localization branch. It uses the anchor-based driven network to localize which box is the lane box. Then, the two features are fused together to determine the lane location. Due to its anchor-based driven approach, it is very fast and able to process more than three hundred images per second.

4. Experiments

4.1. Dataset

In the experiment, we utilized two benchmark datasets — TuSimple dataset and CULane dataset — to judge the performance and robustness of the two models[12,13]. The TuSimple dataset includes about 6,400 road images on highways in the United States, whose resolution is 1280×720. There are about 3,600 training images, 360 validating images, and 2,700 testing images in the TuSimple dataset. These images are under various illumination conditions, such as the middle of the night. The CULane dataset is gathered by six cameras installed on vehicles driven by various drivers. There are over 55 hours of videos and about 133,000 frames in this dataset. They have separated the dataset into 3 parts — 88880 images divided for training, 9675 images divided for validation, and 34680 images divided for testing. The characteristic of the CULane dataset is that there are nine scenarios of the dataset — "normal," "crowd" (the line is obscured), "hlight" (very bright light on the road), "shadow" (where the road is covered in shadows), "noline"(where the line is not visible), "curve," "cross" (crossing or intersection), "arrow" (pictures with directional arrows on the road), "night," so it is more generalized than the previous dataset and is more suitable for training models. Although we have selected 2 separate datasets, they all have a few common categories missing or lacking in numbers; this includes situations such as poor weather conditions (i.e., heavy rain), snow, and fog, which may lead to a lack of ability to adapt to new scenarios that are not in the dataset.

4.2. Performance of the Model

4.2.1. Quantitative Results. We trained two models using 18-layer ResNet as the backbone and CULane as the dataset, and we used three metrics for both models in nine different scenarios of the dataset, based on which we plot in Table 1.

Table 1. The three evaluation metrics for the two models in different scenarios based on the CULane dataset are listed, and the model that performs better in the three metrics is labeled using bold. The IoU threshold is set to be 0.5.

Scenario	model	normal	Crowd	hlight	shadow	noline	arrow	curve	cross	night
Precision(%)	CLRNet	94.66	86.84	84.01	88.87	76.16	93.45	87.75	0	86.22
	UFLD	92.20	75.52	65.20	73.16	52.05	89.45	75.38	0	71.34
Recall (%)	CLRNet	91.89	71.75	66.11	71.35	39.74	86.61	57.85	0	66.21
	UFLD	91.75	72.60	61.37	71.56	44.52	86.05	63.49	0	68.27
F1-score(%)	CLRNet	93.30	78.58	74.00	79.15	52.23	89.90	69.73	0	74.90
	UFLD	91.97	74.04	63.22	72.35	47.99	87.71	68.93	0	69.77

We utilized rigor and thought of F1-score to measure the generalization ability of the models, and formulas for three metrics are shown below.

$$precision = \frac{tp}{tp+fp} \tag{1}$$

$$recall = \frac{tp}{tp+fn} \tag{2}$$

$$f1-score = \frac{2\times recall\times precision}{recall+precision} \tag{3}$$

It can be seen that the precision rate measures the degree of false alarms of the model, the higher the precision rate of a model, the lower the probability of its false alarms. Recall rate measures the degree of under-reporting of a model, the higher the recall rate, the lower the probability of its under-reporting. These two are contradictory and cannot be improved at the same time. F1-score combines precision and recall together and measures the overall capability of the model, the higher score it gets, the better the model works. Through the figure, we can know that CLRNet has a better precision rate while UFLD-v2 has a better recall rate, which means that CLRNet may choose not to mark some uncertain lanes and miss some results, while UFLD-v2 may mark some not-lanes as lanes, which may produce misidentification although the probability of missing results is smaller. F1 score of CLRNet is higher than that of UFLD-v2's, and therefore, it is considered to be a superior model. In addition to this, through our tests, corresponding to the same dataset, the processing speed of UFLD-v2 is much faster than that of CLRNet, and the processing time of UFLD-v2 is much smaller than that of CLRNet for the same number of frames, which should not be neglected in quantitative evaluation.

4.2.2. Qualitative Results. We selected two representative frames to visualize this difference between precision and recall rates. The results are shown in Fig 1. The two models were labeled for two identical frames, the two figures are the results of UFLD-v2, and the bottom two are the results of CLRNet. The frame on the left was selected from the night scene, and as can be seen, UFLD-v2 marked lanes where there should be no lanes, which is a false alarm. The frame on the right was taken from a shadowed scene, and you can see that CLRNet missed two lanes. In addition to this, since UFLD-v2 uses an anchor-driven approach, it is more susceptible to interference from redundant information on the image. For example, in the right-side image, there are some curves in the dotted lines of the predicted lanes due to the influence of the passers-by.

DOI: 10.54254/2755-2721/43/20230840

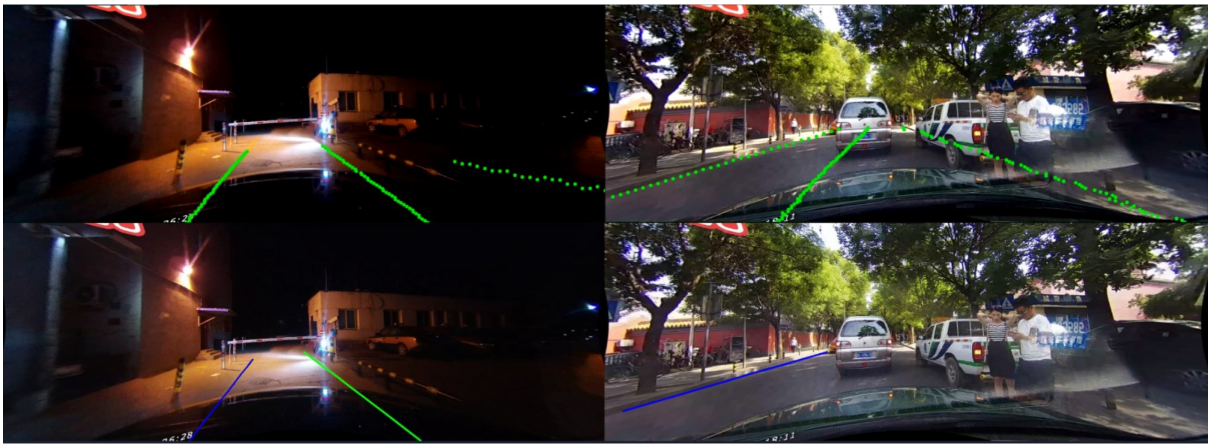

Figure 1. The top two are UFLD-v2 detections, and the bottom two are CLRNet detections. The two on the left are the same frame, and the two on the right are the same.

In addition to this difference, the two models are characterized differently when dealing with complex road sections. In a congested roadway with high traffic flow, the lanes will become very inconspicuous and the model will rely solely on the alignment of the traffic to determine the virtual lanes. The results of UFLD-v2 and CLRNet for processing two similar frames are shown in Fig 2. The two frames are separated by only a few frames, but the lane markings of CLRNet are very different, which is obviously disturbed by the position of the vehicles more seriously, and thus focuses more on the obvious vehicle gaps. UFLD-v2, on the other hand, does not have an overly obvious difference between the two frames, and the lane markings are more consistent with lanes marked where there are no gaps in the vehicles, despite a bit of fluctuations in the lane lines. Therefore, UFLD-v2 has better stability than CLRNet.

Figure 2. The two on the left are UFLD-v2 detections and the two on the right are CLRNet detections. The top two are the same frame, and the bottom two are the same frame, with only a few frames separating the two frames.

We believe that the reason for this phenomenon is because CLRNet adopts a bottom-up structure, which itself has a stronger ability for feature extraction, so the probability of false alarms will be lower, although the ROIGather module is used to obtain global context information and propose Line IoU loss to optimize the lane lines as a whole, its ability to learn the overall learning ability is still not as good as the top-down structure, so it will generate a certain degree of under-reporting, and this incomplete learning can also lead to labeled; while UFLD-v2 adopts the top-down structure, which is more capable of grasping the overall information, but is not detailed enough for feature processing, so the probability of false alarms will be higher instead of under-reporting due to the neglect of some global features, and this comprehensive learning capability will also lead to better continuity in complex scenarios. Since CLRNet exploits a coarse-to-fine mechanism that uses high-level features to detect lane lines and then uses low-level features to adjust the lane line positions, its overall generalization ability is better than that of UFLD-v2. However, the rapidity of UFLD-v2 makes them also very promising for practical applications.

5. Conclusion

Through experimentation, we can conclude that top-down structure has fast speed and can learn information from the whole image, while bottom-up structure is better suited to extracting features. We can assume that regardless of the structure of the model, a good attention mechanism and the ability to eliminate interference are essential in lane detection, and subsequent advances in models will reflect this aspect. In addition to this, because of the complexity of the road scene, a well-developed dataset is also indispensable, which will determine the generalization ability of a model. Considering that lane detection is mainly applied to regular roadways to assist in planning vehicle travel, CULane is considered a good dataset because it contains road surfaces under various conditions, but some extreme weather scenarios are still not included, and the definition of lane lines is too homogeneous and does not take into account more complex lane scenarios, such as lane lines at intersections and two side-by-side lane lines, which are points for improvement. Finally, the current lane detection is unable to recognize solid and dashed lines, which is very important in lane changing, so we conceived a way to solve this problem by adding a length computation module to distinguish them, which is to be done in the future.

Acknowledge

Haoxiang Xu, Yushuo Xu and Tiantian Shen contributed equally to this work and should be considered co-first authors.

References

[1] Tsung-Ying Sun, Shang-Jeng Tsai and V. Chan, "HSI color model based lane-marking detection," 2006 IEEE Intelligent Transportation Systems Conference, Toronto, ON, Canada, 2006, pp. 1168-1172, doi: 10.1109/ITSC.2006.1707380.

[2] B. Yu and A. K. Jain, "Lane boundary detection using a multiresolution Hough transform," Proceedings of International Conference on Image Processing, Santa Barbara, CA, USA, 1997, pp. 748-751 vol.2, doi: 10.1109/ICIP.1997.638604.

[3] Wang, Y., Shen, D., & Teoh, E. K.(2000). Lane detection using spline model. Pattern Recognition Letters, 21(8), 677-689. https://doi.org/10.1016/S0167-8655(00)00021-0

[4] Wang, Y., Teoh, E.K., Shen, D.(2004) Lane detection and tracking using b-snake. Image and Vision Computing 22(4), 269–280.

[5] Kim, Z.: (2008) Robust lane detection and tracking in challenging scenarios. IEEE Transactions on Intelligent Transportation Systems 9(1), 16–26.

[6] Aly, M. (2008). Real time detection of lane markers in urban streets. *2008 IEEE Intelligent Vehicles Symposium*, 7-12.

[7] Neven, D., Brabandere, B.D., Georgoulis, S., Proesmans, M., & Gool, L.V. (2018). Towards End-to-End Lane Detection: an Instance Segmentation Approach. *2018 IEEE Intelligent Vehicles Symposium (IV)*, 286-291.

[8] Pan, X., Shi, J., Luo, P., Wang, X., & Tang, X. (2017). Spatial As Deep: Spatial CNN for Traffic Scene Understanding. AAAI Conference on Artificial Intelligence.

DOI: 10.54254/2755-2721/43/20230840

[9] Brabandere, B.D., Van Gansbeke, W., Neven, D., Proesmans, M., & Gool, L.V. (2019). End-to-end Lane Detection through Differentiable Least-Squares Fitting. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 905-913.

[10] Zheng, T., Huang, Y., Liu, Y., Tang, W., Yang, Z., Cai, D., & He, X. (2022). CLRNet: Cross Layer Refinement Network for Lane Detection. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 888-897.

[11] Qin, Z., Zhang, P., & Li, X. (2022). Ultra Fast Deep Lane Detection with Hybrid Anchor Driven Ordinal Classification. *IEEE transactions on pattern analysis and machine intelligence, PP*.

[12] TuSimple, "Tusimple benchmark," https://github.com/TuSimple/tusimple-benchmark.

[13] CULane, https://xingangpan.github.io/projects/CULane.html.