Contrastive Predictive Coding-based LaneNet CNN Model for Lane Detection

Rahil Modi, Zhimin Chen, Minni Bin, Bing Li
Department of Automotive Engineering, Clemson University, Greenville, SC, USA 29607

Abstract

With an increasing number of intelligent vehicles with automated driving advanced driver-assistance systems (ADAS) features plying on the road, the chance that they get involved in an accident also increases. Lane detection is one of the critical aspects of lane-related ADAS which ensures safe driving. To improve the data-efficiency of the convolutional neural network (CNN) model, we propose a Contrastive Predictive Coding (CPC) approach based on the LaneNet. The LaneNet model is made as an instance segmentation problem. Every lane forms its own instance which helps in recognizing lanes in the various scenarios. Furthermore, the output of CPC which consists of the extracted features from the images is used as the initial input to the LaneNet, this helps the LaneNet model recognize and learn classifying lanes faster and give improved results. After training the LaneNet with the labeled TuSimple dataset, an accuracy of around 0.671 mIOU is achieved. CPC is based on unsupervised training. Our proposed network trains the CPC model with a large amount of unlabeled lane data, as a result, the model will understand and recognize the lanes with higher accuracy considering a large amount of unlabeled data available, it makes this process less complex. After training the CPC with unlabeled data images from various datasets such as CU-Lane, Berkeley Deep Drive, NEXET, and Apollo, an accuracy of around 94% was achieved in classifying lanes by testing it on labeled test images. Our proposed approach can achieve competitive detection accuracy compared with the state-of-the-art models using less labeled data.

Introduction

After the introduction of the ADAS systems there is reduction in the safety problems that are caused by normal driving of human beings, as these electronically smart systems intervene or assist the driver in avoiding the mistake they would have attempted unknowingly without the assistance from these systems. According to the National Highway Traffic Safety Administration (NHTSA) in 2018 the number of people dying in United States in road accidents reduced by 2.4% from 2017 due introduction of ADAS systems and other safety programs, though there was a reduction in percentage, $36,560$ people died that year [1]. There is a constant requirement of improving these systems to reduce any sort of failure of these systems and reduce any casualty.

With the arrival of Autonomous Vehicles, the goal of the manufacturers and automotive industry remains consistent to reduce safety problems caused due to a human being. According to the NHTSA in 2017 around $91,000$ motor vehicle crashes occurred due to drowsy driving or improper lane change [2]. This depicts the importance of Lane Detection as a very safety critical component of ADAS systems and upcoming Autonomous Vehicles.

The traditional Lane detection methods relied on methods that were computationally very costly, required post processing and were not adaptive to changes in the lanes or road and also required hand crafted heuristics. These methods were generally combined with Kalman filter or Hough transform. Post processing involved removing misdetections with the help of these filters. Hough transform is widely used for detecting the lanes as it is not very sensitive to noise, but it is not suitable for real time detection due to its complex nature and storage issues. Hough transform is generally used to detect lanes which are straight, for curved lanes models such as Bezier Splines and parabola are considered, these models are robust if the complexity of these models is low but simple models do not fit the lanes accurately and a complex model can fit lanes accurately but are prone to be sensitive to the noise.

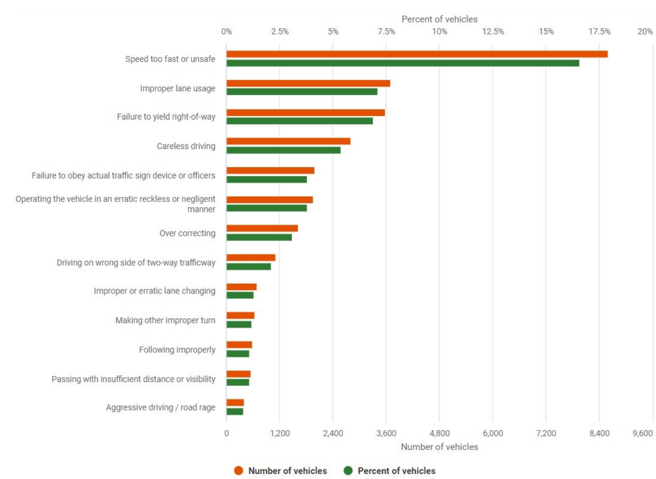

Figure 1: NHTSA 2018 National Motor Vehicle Crash Causation Survey [3]

It can be observed in Figure (1), the National Motor Vehicle Crash Causation Survey carried out by NHTSA in the year 2018 indicating improper lane usage as the second largest cause of road accidents which results in around $3,707$ vehicles getting involved in accidents.

The existing research carried out till date in the area of learning-based lane detection mostly involved using supervised training which requires data with labels and labelling data is a tedious task and requires expert people to complete the task. Due to these reasons labeled data is limited. On the other hand, unsupervised training requires unlabeled data which is available in abundance.

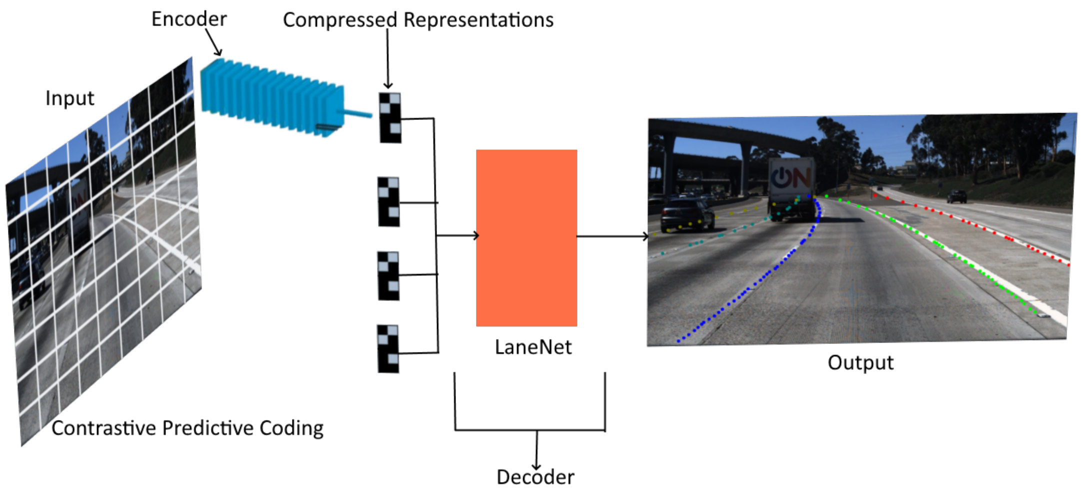

Figure 2: Proposed Method architecture: (From Left) The input is divided into overlapping patches. Each patch is run through an encoder which extracts the features in the patch area and stores them as compressed Representations. These representations are sent into LaneNet which acts as the decoder. With the help of the representations the lanes are predicted in the image and are plotted into the image which is the output

This research focuses on the improving of data-efficiency of LaneNet which is a convolutional neural network. It is trained end to end for lane detection and is capable of compensating change in the number of lanes and change in ground plane used as the base model. We further implement a semi-supervised training for our proposed network, and the transfer learning is performed from the Contrastive Predictive Coding (CPC) which is based on unsupervised training. The features extracted from the images by the CPC model are transferred to the LaneNet in transfer learning to make the predictions.

Related Works

Traditional lane detection methods used consisted of color-based features Kuo-Yu Chiu et al [4] have proposed a method which involved determining the lanes based on the light levels of the pixels, if certain pixels had a light value within a specified range it would detect it as lane. Zhu Teng et al. [5] have proposed a method which is bar filter, it is another one of the traditional lane detection method which involves lane detection by detecting lane by the bar type shape they have. This is not a very effective way of detecting lanes because if the lane is curved it will have difficulty in detecting lanes.These methods are computationally costly, not ideal for real time detection as they would require some post processing.

Mu Chunyang et al [6] have used OTSU segmentation [7] to identify different classes in the image and then uses Sobel edge detector to detect the lane marker. The major disadvantage of Sobel edge detector is signal to noise ratio. With an increase in the noise there is degradation in the gradient magnitude of the edges which lead to inaccurate results. There are high chances of encountering roads in real life where there will be a lot of noise, so the Sobel edge detector is not advised to be used in real life scenarios. OTSU segmentation also has limitations such as if the image has lot of noise the deep and sharp valley in the bimodal histogram of the image degrades which leads to segmentation error. A good bimodal histogram is necessary for good OTSU segmentation.

Seungwoo Yoo et al [8] have used row wise classification segmentation in this paper, it divides an image in different sections and looks for lanes in those sections. The limitation observed in this method is, as the image is converted into rows for representation and identification of the lanes, the lanes need to be vertical to get accurate results, if the lanes are curved like on turn they would extend in two sections and the algorithm might detect a single lane as 2 different lanes. This approach has showed good results for straight or vertical lanes on the road.

Various convolutional neural network (CNN) approaches [9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22] have been invented for lane detection in recent years. For accurate and robust lane detection, Bei He et al [9] proposed a dual-view CNN framework. It used a weighted hat-like filter to recalls potential lane candidates and reduce the disturbance from gradual textures. An efficient PointLaneNet [13] CNN model was also proposed for lane departure warning (LDW) and lane keeping assist (LKA) features. It has been deployed in edge computing AI platform NVIDIA PX2 for real-time ADAS scenarios. Davy Neven et al [22] have trained a state-of-the-art neural network with end-to-end training for lane detection. They have proposed lane detection which is based on fully supervised training which involves using labeled data for training. The drawback of LaneNet model is it requires large labeled dataset to achieve good accuracy, however, labeled lane data is limited.

Olivier J. Hénaff et al [23] have proposed a novel feature extraction method which is based on unsupervised training, namely Contrastive Predictive Coding (CPC). The CPC network generates representations from images given as input. It uses these representations to predict the features in images. As it is a unsupervised training, it uses unlabeled data which is easily available. Using the representation generated by CPC as input to supervised training, it has great potential of reducing the need of labeled data to achieve good accuracy.

Proposed Methodology

We have proposed a network which is based on semi-supervised training. LaneNet is used as the base model for lane detection model training. With the help of transfer learning as seen in Figure (2) CPC feature vectors are used for transfer learning. The generation of masked ConvNets and prediction parts from CPC are not implemented in this method. The generated feature vectors are used as the input to the LaneNet model and LaneNet supervised training is executed above this.

First the LaneNet is trained with fully supervised training and TuSimple Dataset is used for this training. The results for this training are discussed in the Results section. After completion of the fully supervised LaneNet training CPC is trained for two combinations of datasets:

CPC Configuration 1:
For this configuration mixed dataset is used which included unlabeled images from datasets such as CUlane, Nexet and Berkeley Deep Drive.

CPC Configuration 2:
For this configuration 50% of dataset consisted of unlabeled TuSimple dataset and rest 50% dataset consisted of the mixed dataset used in the first configuration. The results of the CPC training are discussed in the Results section.

The extracted features are saved in the form of tensors and these tensors are used as input to the LaneNet model and training is performed. LaneNet is trained for three different configurations. First configuration is trained with CPC configuration 1 with 100% labeled TuSimple dataset. Second configuration is trained with CPC configuration 1 with 50% of labeled TuSimple dataset and the third configuration is trained with CPC configuration 2 and 50% of labeled TuSimple dataset.

LaneNet Detection: is deployed as instance segmentation approach and divided this task into two parts, Lane Instance segmentation and Lane Embedding branch. Lane Instance segmentation focuses on generating a binary segmentation map highlighting the pixels belonging to lane, with the help of ground truth points it is trained to connect all the ground truth points and draw a connected line for every lane it also draws the line over the obstacle to train the model to predict lanes when they are not visible due to some occlusion is depicted as decoder in Figure (2).

Lane Embedding branch focuses on disentangling the lane pixel from the binary segmentation map and give every pixel a lane embedding and pixels belonging to same lane, their embeddings will pulled closer and embeddings belonging to different pixels will be pushed away. LaneNet can accommodate change in ground plane and change in number of lanes. With these features it cannot use normal polynomial to fit the lanes back onto the image it would require using higher order polynomial to fit these lanes.

To solve this type of problem image in most cases is converted to bird's eye view, lanes become parallel to each other in this view and 2nd or 3rd order polynomial is used to fit lanes onto the image. LaneNet is designed to compensate for change in ground plane, this causes the transformation matrix which is used to convert an image into bird's eye view to change. In addition to LaneNet a H-Net is trained to output a transformation matrix for every image.

Contrastive Predictive Coding (CPC): is an un-supervised training network model and it is trained to learn predictable representations. In this method the image given as the input is divided into overlapping patches as feature encoders as shown in the left side of Figure (2).

Each patch is running through a feature extractor and a representation vector is generated which is depicted with the help of the blue line in the image. The local feature vectors generated by feature extractor are combined together and a masked ConvNets are created. For the next process, the image is divided into two parts, it can be divided in left and right or top and bottom parts. One part is used for training and the second part is used for the prediction.

The generated context networks are used to predict features in the other half part of the image. CPC is stated to give higher accuracy than the model trained with only labeled data when trained on full dataset. The authors have claimed to achieve around 80% accuracy when trained on 1% of the ImageNet dataset with CPC representations.

Experimental Results

LaneNet Training and Validation

Dataset: LaneNet training is performed with TuSimple dataset. This dataset contains 3,626 video clips of 1 sec duration each. Each of these video clips contains 20 frames of which, the last frame is annotated in the TuSimple dataset [24].

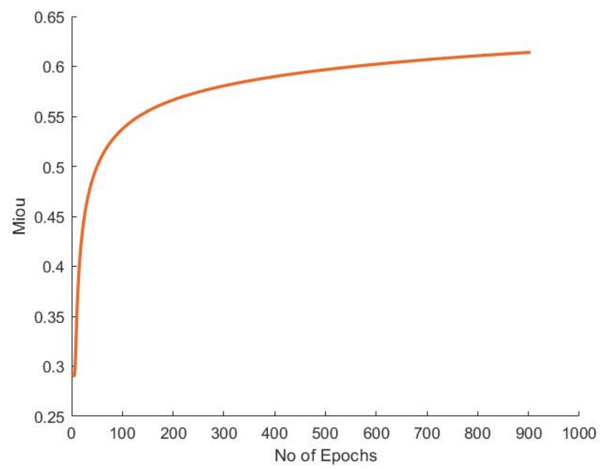

Figure 3: Training mIOU of LaneNet

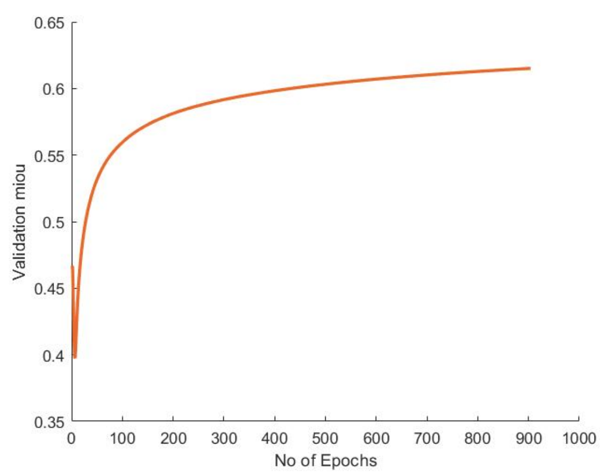

Figure 4: Validation mIOU of LaneNet

For the LaneNet training, the training mIOU and Validation mIOU results are shown in Figures (3) and (4). This training is a fully supervised training. The code we referred to for training LaneNet belongs to this repository [25]. The training results can be observed below and a training mIOU of 0.615 is achieved. With the validation results it shows that the training was done smoothly and correctly. The model is able to detect lanes and no overfitting or improper lane detection is observed.

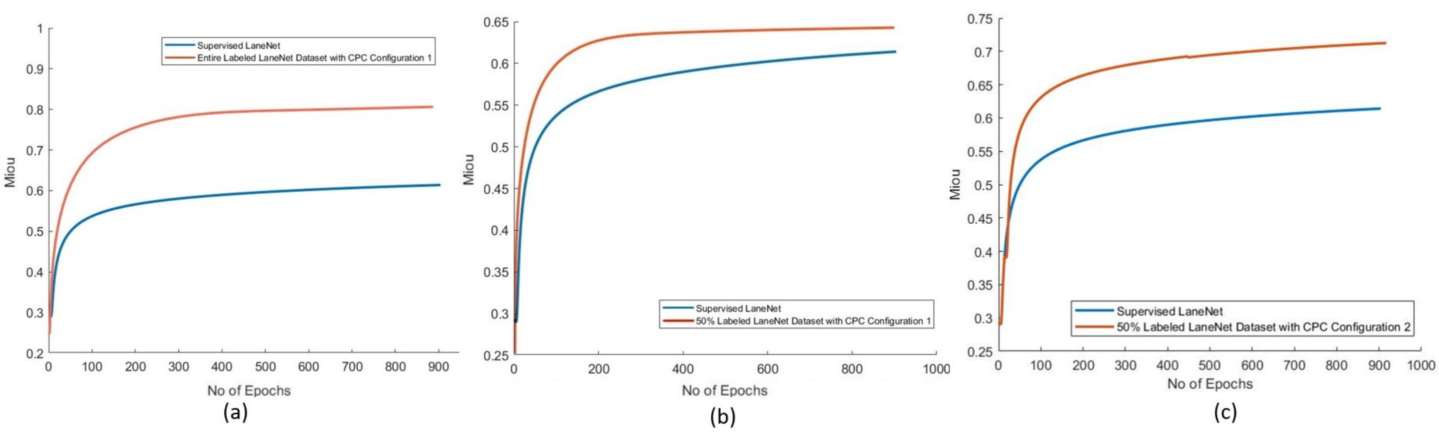

Figure 5: LaneNet with CPC training accuracy results.
(a) Configuration 1: 100% labeled TuSimple dataset with CPC configuration 1 (mixed dataset),
(b) Configuration 2: 50% of labeled TuSimple dataset with CPC configuration 1,
(c) Configuration 3: 50% of labeled TuSimple dataset with CPC configuration 2 (50% unlabeled TuSimple images + mixed dataset)

CPC Network Training and Validation

CULane: CULane is a large-scale challenging dataset for academic research on traffic lane detection. It is collected by cameras mounted on six different vehicles driven by different drivers in Beijing. More than 55 hours of videos were collected, and 133,235 frames were extracted. The dataset is divided into 88,880 for training set, 9,675 for validation set, and 34680 for test dataset [26].

NEXET: NEXET is a dataset with 50,000 training images with bounding boxes and 41,190 images for testing. It contains images from 77 countries which makes this dataset very diverse. It consists of 49.8% images from daylight, 46.4% from nightlight and 3.7% in twilight [27].

Berkeley Deep Drive: Berkeley Deep Drive consists of 100,000 HD video sequences with 1,100 hours driving video. Annotation is done in the form of 2D bounding boxes on around 100,000 images [28].

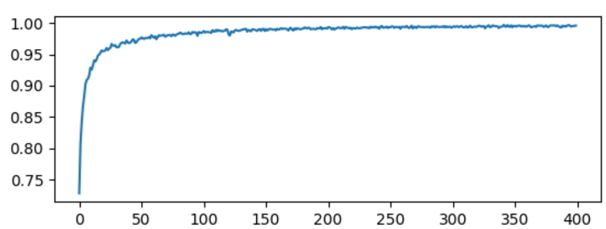

Figure 6: Training accuracy of CPC Configuration 1

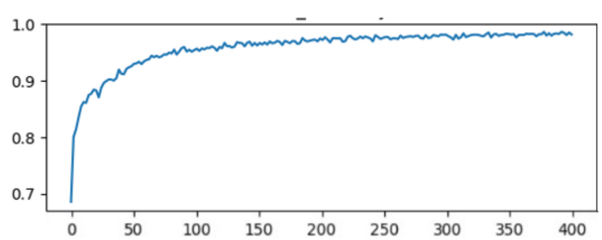

Figure 7: Validation accuracy of CPC Configuration 1

For the training of CPC, two configurations of CPC were trained. The images were resized to 1204x720 resolution to make them compatible with the CPC code we have used by David Tellez [29]. The training results of both the configurations are given below.

CPC Configuration 1:
The CPC Configuration 1 consisted of unlabeled lane images from various datasets mentioned earlier. The results can be observed in Figure (6) and Figure (7). The training and validation accuracy confirm that the training was done properly and the model was able to achieve good validation accuracy when it was evaluated on a part of dataset left for validation from the training dataset.

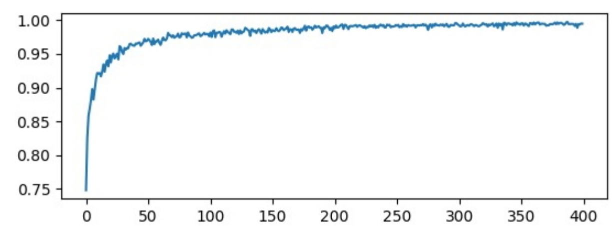

Figure 8: Training accuracy of CPC Configuration 2

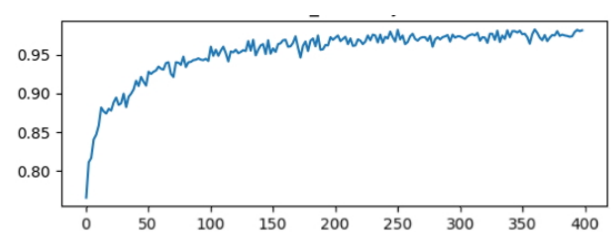

Figure 9: Validation accuracy of CPC Configuration 2

CPC Configuration 2:
The CPC Configuration 2 consisted of 50% unlabeled images from TuSimple dataset and 50% from the mixed dataset used for CPC configuration 1. The results can be observed in Figures (8) and (9). This configuration CPC also achieved good training and validation accuracy. The validation accuracy took some time to get good results but eventually reached to a good result.

LaneNet with CPC Training and Validation

This part corresponds to our proposed network in this paper. The features are extracted from the CPC training and stored in the form of Keras model. CPC training acts as the encoder model of the semi supervised training we have proposed and LaneNet acts as the decoder and base model of our training. We modified the LaneNet code to incorporate the CPC model which involved making the output of the CPC model compatible with the input size of the LaneNet. The training of LaneNet with CPC was performed for three configurations. The training results can be observed in Figure (5).

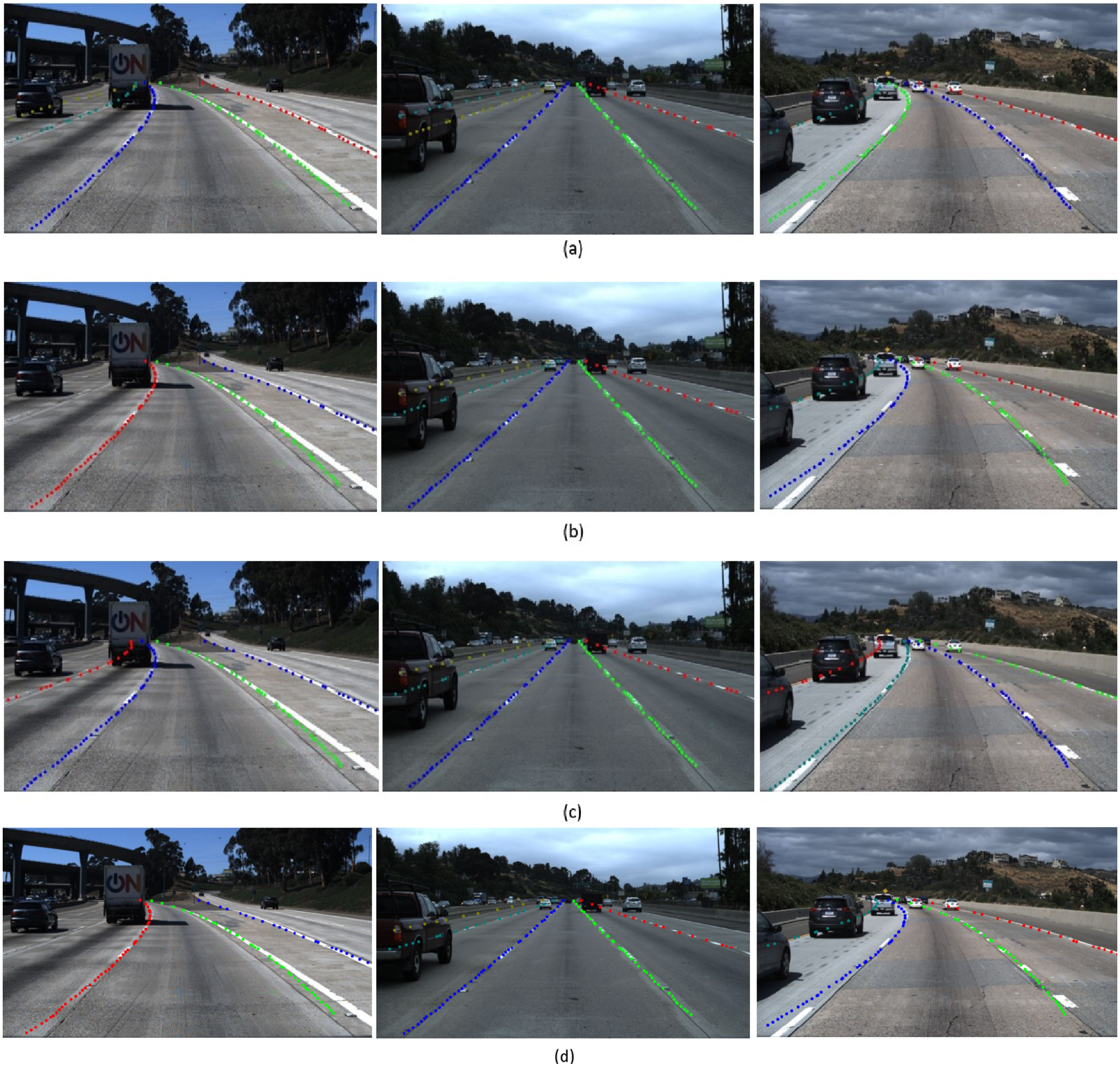

Figure 10: LaneNet with CPC network inference results.
(a) Configuration 1: 100% labeled TuSimple dataset with CPC configuration 1 (mixed dataset),
(b) Configuration 2: 50% of labeled TuSimple dataset with CPC configuration 1,
(c) Configuration 3: 50% of labeled TuSimple dataset with CPC configuration 2 (50% unlabeled TuSimple images + mixed dataset),
(d) Fully supervised LaneNet results with labeled TuSimple dataset.

CPC-LaneNet Configuration 1:
It is trained with CPC configuration 1 (which used mixed dataset) with entire labeled TuSimple dataset. As shown in Figure (10), the results yielded from this training are better than fully supervised training of LaneNet. Training mIOU of 0.82 has been achieved.

CPC-LaneNet Configuration 2:
It is trained with CPC configuration 1 and only 50% of the labeled TuSimple dataset. The results of this training were similar to the fully supervised training which validated the results of CPC claimed by its author which stated requirement of lesser labels for similar accuracy, training mIOU of 0.642 has been achieved.

CPC-LaneNet Configuration 3:
It is trained with CPC configuration 2 (which consisted of 50% TuSimple unlabeled images and 50% mixed dataset) with 50% labeled TuSimple dataset, this training results showed a balance between Configuration 1 and Configuration 2 of LaneNet with CPC, it can be observed in Figure (10) as it achieved a training mIOU of 0.723.

It can be observed from the results in Figure (10), the difference of different models is noticeable in image 1 where the lanes are difficult to detect, Configuration 1 was able to detect all the lanes where as Configuration 2 and fully supervised training LaneNet could not perform very good in image 1. Configuration 3 is improvement over Configuration

2 still cannot achieve the accuracy of Configuration 1.

Conclusion

In this paper we have presented a network which is based on semi supervised training and involves transfer learning of feature vectors from a unsupervised training method Contrastive Predictive Coding to a supervised training LaneNet which is a CNN developed for lane detection and it is used as the base model for our presented method.

After training CPC with different types of unlabeled data of lanes and using its feature vectors as input to the LaneNet CNN for semi supervised training. The results of the presented method were compared with the supervised training of LaneNet with labeled TuSimple dataset. The results indicate that the presented network showed improvement over fully supervised training and highlighted the benefits of choosing semi supervised training. With this method the requirement of labeled data is reduced, and similar accuracy can be achieved with lesser labeled data compared to supervised training.

References

1. NHTSA, "2018 fatal motor vehicle crashes: Overview," *Traffic Safety Facts: Research Note*, 2018.
2. NHTSA, "Drowsy driving 2015," *Traffic Safety Facts: Crash Stats*, 2017.
3. N. S. Council, "Nsc analysis of nhtsa fars data," 2018.
4. Kuo-Yu Chiu and Sheng-Fuu Lin, "Lane detection using color-based segmentation," in *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, pp. 706–711, 2005.
5. Z. Teng, J.-H. Kim, and D. Kang, "Real-time lane detection by using multiple cues," 01 2010.
6. M. Chunyang and X. Ma, "Lane detection based on object segmentation and piecewise fitting," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 12, 05 2014.
7. H. J. Vala and A. Baxi, "A review on otsu image segmentation algorithm," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 2, no. 2, pp. 387–389, 2013.
8. S. Yoo, H. Lee, H. Myeong, S. Yun, H. Park, J. Cho, and D. H. Kim, "End-to-end lane marker detection via row-wise classification," 2020.
9. B. He, R. Ai, Y. Yan, and X. Lang, "Accurate and robust lane detection based on dual-view convolutional neutral network," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1041–1046, IEEE, 2016.
10. J. Li, X. Mei, D. Prokhorov, and D. Tao, "Deep neural network for structural prediction and lane detection in traffic scene," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 690–703, 2016.
11. J. Kim, J. Kim, G.-J. Jang, and M. Lee, "Fast learning method for convolutional neural networks using extreme learning machine and its application to lane detection," *Neural Networks*, vol. 87, pp. 109–121, 2017.
12. Y. Hou, Z. Ma, C. Liu, and C. C. Loy, "Learning lightweight lane detection cnns by self attention distillation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.
13. Z. Chen, Q. Liu, and C. Lian, "Pointlanenet: Efficient end-to-end cnns for accurate real-time lane detection," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 2563–2568, IEEE, 2019.
14. Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang, "Robust lane detection from continuous driving scenes using deep neural networks," *IEEE transactions on vehicular technology*, vol. 69, no. 1, pp. 41–54, 2019.
15. S. Chougule, N. Koznek, A. Ismail, G. Adam, V. Narayan, and M. Schulze, "Reliable multilane detection and classification by utilizing cnn as a regression network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 0–0, 2018.
16. Y. Y. Ye, X. L. Hao, and H. J. Chen, "Lane detection method based on lane structural analysis and cnns," *IET Intelligent Transport Systems*, vol. 12, no. 6, pp. 513–520, 2018.
17. Z. Chen and X. Huang, "End-to-end learning for lane keeping of self-driving cars," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1856–1860, IEEE, 2017.
18. W. Song, Y. Yang, M. Fu, Y. Li, and M. Wang, "Lane detection and classification for forward collision warning system based on stereo vision," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5151–5163, 2018.
19. N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3d-lanenet: End-to-end 3d multiple lane detection," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2921–2930, 2019.
20. W. Wang, H. Lin, and J. Wang, "Cnn based lane detection with instance segmentation in edge-cloud computing," *Journal of Cloud Computing*, vol. 9, pp. 1–10, 2020.
21. J. Kim, D. Lee, and M. Lee, "Lane detection system using cnn," *IEMEK Journal of Embedded Systems and Applications*, vol. 11, no. 3, pp. 163–171, 2016.
22. D. Neven, B. D. Brabandere, S. Georgoulis, M. Proesmans, and L. V. Gool, "Towards end-to-end lane detection: an instance segmentation approach," 2018.
23. O. J. Hénaff, A. Srinivas, J. D. Fauw, A. Razavi, C. Doersch, S. M. A. Eslami, and A. van den Oord, "Data-efficient image recognition with contrastive predictive coding," *CoRR*, vol. abs/1905.09272, 2019.
24. TuSimple, "Tusimple lane detection challenge."
25. MaybeShewill-CV, "Lanenet lane detection." `https://github.com/MaybeShewill-CV/lanenet-lane-detection`.
26. P. L. X. W. Xingang Pan, Jianping Shi and X. Tang, "Spatial as deep: Spatial cnn for traffic scene understanding," in *AAAI Conference on Artificial Intelligence (AAAI)*, February 2018.
27. Nexar, "Nexet dataset."
28. F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," 2020.
29. D. Tellez, "Contrastive predictive coding." `https://github.com/davidtellez/contrastive-predictive-coding`.

Contact Information

Bing Li, Ph.D.
Assistant Professor
Department of Automotive Engineering, Clemson University
International Center for Automotive Research (CU-ICAR)
4 Research Drive, Greenville, SC 29607
Phone: (864)365-0649
E-mail: bli4@clemson.edu