Symbolic Road Marking Recognition Using Convolutional Neural Networks

Touqeer Ahmad*, David Ilstrup†, Ebrahim Emami*, George Bebis*
†Nissan Research Center, Silicon Valley, USA, dave0mi@gmail.com
*Department of Computer Science and Engineering, University of Nevada, Reno, USA
{tahmad, ebrahim}@nevada.unr.edu, bebis@cse.unr.edu

***Abstract*—This paper investigates the use of Convolutional Neural Networks for classification of painted symbolic road markings. Previous work on road marking recognition is mostly based on either template matching or on classical feature extraction followed by classifier training which is not always effective and based on feature engineering. However, with the rise of deep neural networks and their success in ADAS systems, it is natural to investigate the suitability of CNN for road marking recognition. Unlike others, our focus is solely on road marking recognition and not detection; which has been extensively explored and conventionally based on MSER feature extraction of the IPM images. We train five different CNN architectures with variable number of convolution/max-pooling and fully connected layers, and different resolution of road mark patches. We use a publicly available road marking data set and incorporate data augmentation to enhance the size of this data set which is required for training deep nets. The augmented data set is randomly partitioned in 70% and 30% for training and testing. The best CNN network results in an average recognition rate of 99.05% for 10 classes of road markings on the test set.**

I. Introduction

Symbolic road markings (SRMs) are the marks/symbols and numbers/characters painted on the roads for driver assistance and safely maneuvering the vehicles. Conventionally road markings are painted in bright colors specifically in white or yellow. These include but not limited to arrow marks, merge signs, zebra crossings, speed limits, railway, Ped and Xing signs etc. The detection and recognition of such road markings could be of essential use for Advanced Driving Assistant Systems (ADAS) and Autonomous Driving (AD) and has already been shown to be of importance for vehicle localization and navigation [33]. Recently significant progress has been made towards various aspects of autonomous driving and vision based detection, segmentation and recognition tasks. With increasing availability of annotated urban road scene data sets [19], [17], [20] and emerging deep networks [18], [21], [16], [22], [23]; significant focus has been on the detection and segmentation of majorly occurring classes on urban scenes e.g. pedestrians[30], roads [26], [27], [28], vehicles/cars, road signs [29], lanes [32], [31] etc. Considerably less attention has been on segmentation and recognition of other small classes such as road markings.

This is due to the fact that most of the data sets do not provide ground truth annotations for such small classes e.g. CityScapes [19] is probably one of the best (5000 images with pixel labeling) data sets focused on urban scene-understanding yet it lacks annotations for road markings and lanes which are assumed to be part of the road surface. The unavailability of annotations for such small classes is a significant hurdle towards their semantic level segmentation since deep networks or other classical classifier cannot be trained. Due to ground truth unavailability; often samples of two different classes are considered instances of a super-class e.g. SegNet [21] is trained on CamVid data set [17] where road markings (turn arrows) and lanes are considered to be members of the same class i.e. paint on road – whereas if distinguished/segmented correctly they can be used for different aspects of autonomous driving.

Most of the work on symbolic road marking detection and recognition follows the well known two steps of object recognition pipeline i.e. detection and classification. First, candidate regions are conventionally generated using Maximally Stable Extremal Regions [9] of the Inverse Perspective Mapping (IPM) images or some other region proposal method. Then, these regions are classified into different classes of road markings based on the extracted features. Mostly, HOG is used as the feature choice [1], [5], [11] and SVM or its variants as the classifier choice [3], [11]. This paper serves as a step forward towards semantic segmentation of symbolic road markings which should soon be possible as ground truth annotations for small road classes become available.

In this paper we explore the effectiveness of Convolutional Neural Networks (CNNs) for recognition of symbolic road markings which can be used for vehicle localization and navigation. We train five different CNN networks which have different number of convolution, max-pooling and fully connected layers. The networks are trained and tested on a publicly available data set of ten frequently occurring SRMs. We incorporate in-plane rotation as data augmentation technique and train our networks at three different resolutions of road mark patches. The results on the augmented test set demonstrate how CNN can be affective for road marking recognition and a comparable/better recognition rate can be achieved than those based on feature engineering and classifier training. We anticipate that the ground truth annotations of such small classes is just a matter of time as new trends for generating ground truths [24], [25] are emerging and becoming a reality.

The rest of the paper is organized as follows: In section II, we review some of the recent work on road marking

detection and recognition. Section III lists the architectural details of CNN networks being considered in this paper. Section IV presents the experimental details and results. The paper is then concluded in the subsequent section V with directions to future work.

II. Related Work

Wu and Raganathan [1] presented a template-matching approach for identifying various classes of road markings. They used MSERs [9] to generate regions of interest (ROIs) for a rectified (using inverse perspective mapping (IPM)) test image. FAST corners [10] are generated within the ROIs which are then described using HOG features [12] at three different scales and one orientation. These extracted features are matched with the already saved templates for each of the SRM category and this matching is further verified by structural shape matching. The data set used in this work has annotation for 1,443 images and road markings belonging to various categories, and have been made available freely. Chen et al. [2] proposed a machine learning based road marking detection and recognition system. They employed BING feature [4] for objectness estimation of the image windows i.e. the candidates for potential road markings which are then classified into 9 classes (+1 negative class) using PCANet classifier [3]. The PCANet emulates a Convolutional Neural Network where first multiple stages of PCA are used as a feature extractor and subsequently resultant histograms/features are used to train a multi-class SVM [6]. The proposed pipeline is evaluated on the images from [1].

Suhr and Jung [5] based their SRM detection and recognition system on a lane detector which finds the lane markings using a top-hat filter [7] and RANSAC-based line estimator. The SRM candidate regions/blobs are then generated within this reduced search space identified by the two lane markings using projection histograms of top-hat filter response. The SRM candidate regions are converted to fixed size IPM image patches, HOG features [12] are generated for such patches which are then classified. They employ a two-stage classification cascade where the first stage rejects the non-SRM regions and the second classifies the remaining patches into 9 (+1 negative) classes of SRMs. They used a total-error based classifier [8] and compared its performance against SVM. The performance of [5] is bounded by the accuracy of lane detector. It should be noted the 9 SRMs classes considered in [2] and [5] although have some overlap (straight, left, right classes) but are not the same (35, 40, PED, STOP, RAIL and XING in [2] straight-left, straight-right, no-straight, no-left, no-right, diamond in [5]).

Greenhalgh and Mirmehdi [11] also apply MSER on IPM transformed images and then use HOG features to discriminate between various classes of SRMs using SVMs with linear kernels. The linear SVM is chosen as a tradeoff between good accuracy and real-time processing requirement among various classifiers compared (SVM-RBF, SVM-Linear, MLP and Random Forest). The classifier is trained on synthetic data and is demonstrated to outperform base line (template matching) and another earlier approach on a real data set using F-measure. It should be noted that they [11] exploit temporal information and also address the painted text recognition where each character is recognized instead of recognizing a group of characters as a single SRM. This paper is focused on SRM recognition and hence that aspect of their paper [11] is out of scope.

III. Network Architectures

Deep learning has been investigated lately for various applications including but not limited to detection, classification, recognition and semantic segmentation etc. With the ever increasing availability of urban street data and emerging deep networks, ADAS and autonomous driving systems are benefitting from these networks. The core of these networks are the cascades of convolutional layers which act as feature extractors and replace the conventional hand engineered feature descriptors e.g. SIFT [14], HOG [12], LBP [13] etc. In CNN networks, each of the convolutional layer is followed by a sub-sampling or max-pooling layer which downsamples the resolution of the input feature maps. The cascades of convolution-maxpooling (CP) modules are followed by number of fully connected layers which are in turn followed by a multi-class classifier or softmax. Back propagation is typically used to train the network.

The CNN networks explored in this paper are based and inspired from the popular LeNet [15] which has been shown to perform extremely well on real world problems such as digit classification (MNIST) and general object recognition (CIFAR-10/100). In this work, we have explored five different variations of this architecture. The network architectures differ from each other based on the input patch size and the number of convolution-maxpooling (CP) modules. Each of the convolution layers is followed by a max-pooling layer which collectively constitute to one CP module. The considered networks either contain two or three of these modules which are then followed by three or four fully connected layers. Before applying max-pooling, the output of each of the convolution layer is passed through a Rectified Linear Unit (ReLU) which works as the required non-linearity and used for learning complex mappings. The ReLU has been demonstrated to outperform other activation functions e.g. tanh [16]. Each neuron's output in the fully connected layers is also passed through ReLU. Regardless of the input size, each of the convolution mask is of size 5×5 while the max-pooling mask of 2×2 with a horizontal and vertical stride of 2. We include a zero-padding to the max-pooling if the original input size happens to be an odd number. Given the input patch size, the output size for the convolutional and max-pooling layer can be determined by the following equation:

$$O_r = \frac{(I_r - F + 1 + P_x + P_y)}{S} \tag{1}$$

where, O_r and I_r are the output and input resolutions, F is the mask/filter size, P_x and P_y are the amount of zero-padding in x and y directions and S is the stride.

Next, we list the details of each of the individual networks

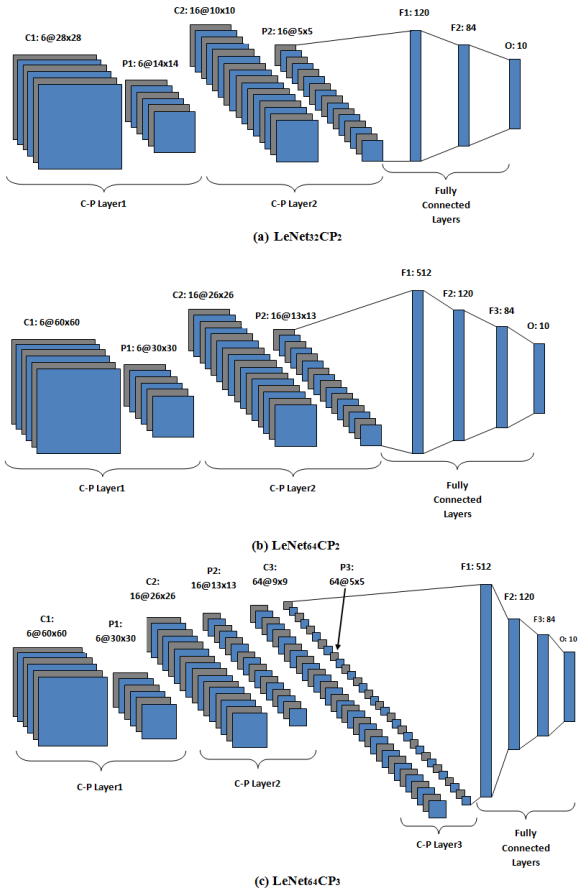

Fig. 1. Three of the five CNN architectures: (a) $\text{LeNet}_{32}\text{CP}_2$, (b) $\text{LeNet}_{64}\text{CP}_2$ and (c) $\text{LeNet}_{64}\text{CP}_3$. $\text{LeNet}_{96}\text{CP}_2$ and $\text{LeNet}_{96}\text{CP}_3$ follows from $\text{LeNet}_{64}\text{CP}_2$ and $\text{LeNet}_{64}\text{CP}_3$ respectively with the resolution differences for input and intermediate resultant feature maps.

- $\text{LeNet}_{32}\text{CP}_2$ – This network is applied on a patch size of 32×32 and has two CP modules. The first CP module contains six convolution filters while the second one is comprised of sixteen. The second CP module is followed by three fully connected/hidden layers which have 120, 84 and 10 neurons respectively. As stated earlier the output of the last fully connected layer is passed through soft-max which generates the class label for the input patch.
- $\text{LeNet}_{64}\text{CP}_2$ – This network follows from $\text{LeNet}_{32}\text{CP}_2$, the only difference being the size of input patch (i.e. 64×64 instead of 32×32) and an additional fully connected layer comprised of 512 neurons.
- $\text{LeNet}_{64}\text{CP}_3$ – This network is also based on an input size of 64×64, however unlike the earlier ones; it is comprised of three CP modules. The first two CP modules are same as in the previous two networks, while the third module contains 64 convolution filters. Additionally, the max-pooling layer in this module enforces a zero-padding of 1 pixel on the input. The number of fully connected layers and their respective number of neurons are same as that of $\text{LeNet}_{64}\text{CP}_2$.
- $\text{LeNet}_{96}\text{CP}_2$ – This network has same number of CP modules and fully connected layers as that of $\text{LeNet}_{64}\text{CP}_2$, only difference being the input resolution (96×96).
- $\text{LeNet}_{96}\text{CP}_3$ – This network follows from $\text{LeNet}_{64}\text{CP}_3$ and differs only due to its input resolution (96×96).

Figure 1 shows the visual layout for the first three networks, highlighting the number of filter banks in each CP module and the resolution resulting from each of the convolution and max-pooling layers. The number of neurons in each of the fully connected layer is also listed, while ReLU and softmax are not visualized to keep the diagram less cumbersome. Figure 2 shows the resultant feature maps for one of the query images for first and second convolution layers before passing through the ReLU.

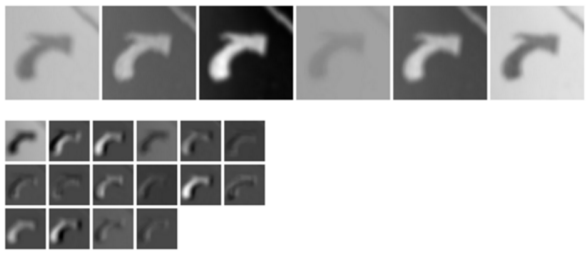

Fig. 2. Resultant feature maps for one of the test images when passed through the convolution masks of first (top) and second (bottom) CP modules of $\text{LeNet}_{64}\text{CP}_2$. The feature maps are shown before the application of subsequent ReLUs.

IV. Experiments and Results

A. Data Set and Augmentation

We evaluate the performance of the CNN networks on the publicly available road marking data set [1]. The original data set is comprised of 1,443 images of 800×600 resolution with ground truth annotations of 27 classes of road markings. However, we retain the first ten classes of road markings for our experiments as number of instances for rest of the classes are fewer than 20 (per each class) in the original data set and not good enough for training CNNs. We introduce data augmentation using in-plane rotation of the original images. Since, the images are captured by a forward facing camera on a moving vehicle; each of the road mark has been captured at various scales and hence we choose not to use extra scaling as an augmentation. The augmentation results in 20,449 road mark patches which are re-scaled to fixed sizes of 32×32, 64×64 and 96×96 for training of respective networks. The resultant data set is re-ordered randomly and randomly partitioned into 70% (14,479 image patches) and 30% (6,000 image patches) chunks for training and testing respectively.

Table I shows the number of examples belonging to each of the road markings class in the resultant augmented

set while figure 3 shows how the relative frequency of road markings is maintained across the training and test sets. Figure 4 shows the examples for each class of the road markings from the augmented set and demonstrates the intra-class variations due to scale, orientation, shadows and road conditions etc. These variations highlight the underlying challenges of road marking recognition which make the feature engineered methods less suitable for such applications. It should be noted that the STOP road markings contribute towards half of the data set in the original set and this distribution is maintained in the resulting augmented set.

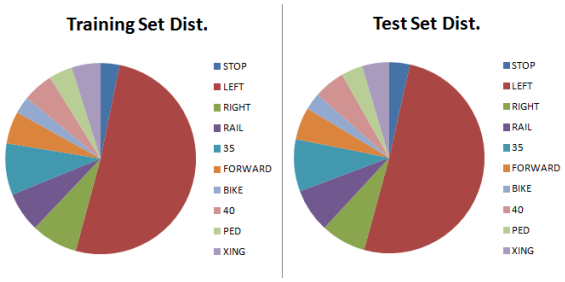

Fig. 3. Frequency distributions of road markings across test (right) and training (left) sets.

TABLE I. NUMBER OF INSTANCES FOR EACH CLASS OF ROAD MARKINGS.

Road Marking Class	Number of instances
STOP	674
LEFT	10433
RIGHT	1600
RAIL	1419
35	1793
FORWARD	1118
BIKE	590
40	1067
PED	799
XING	986
Total	20479

B. Experimental Details

We train each of the five networks on the respective training set using a fixed learning rate of 0.001. The performance of each network is then evaluated on the respective test set. To evaluate the performance of each network quantitatively, we report an average recognition accuracy across the considered ten classes of road markings (table II). Additionally, table III shows the confusion matrices computed for each of the networks. We perform the conventional normalization on both training and test sets: the mean and standard deviation for each of the color channels is computed for the training set. Each of the training images is then subtracted and divided by the respective mean and standard deviation for each color channel. The means and standard deviations computed from training set are used for normalization of the test set as well.

C. Results & Discussion

As demonstrated by table II, all the considered networks overall perform really well on the test set ($> 98\%$ recognition rate) while $\text{LeNet}_{96}\text{CP}_2$ has proven to be the best

Fig. 4. Sample patches from augmented data set for each of the road marking class (stop, left arrow, right arrow, rail, 35, forward arrow, bike, 40, ped and xing) demonstrating intra-class variations due to viewpoint (scale and orientation), road conditions, and shadows caused by buildings and trees.

performing configuration. Further details regarding which road marking has been confused by a specific network as a particular different road marking can be seen from table III. In figure 5 we show the examples of instances for each network which have been most wrongfully classified as another particular road marking e.g. in case of $\text{LeNet}_{32}\text{CP}_2$, PED class has the least recognition rate and has mostly been confused as STOP road marking. The caption of figure 5 provides such details for each of the networks and highlights the class which has the least recognition rate for that particular network.

A quick look at figure 5 reveals some of the obvious confusions encountered by the CNN architectures e.g. the FORWARD arrow mark has been confused as speed limit 40 and vice versa (rows 2 and 3). This can easily be explained due to the geometric similarity between the arrow of the FORWARD sign and the triangular part of the digit four in the speed limit marking. Similarly, the textual road markings e.g. PED, STOP etc. (rows 1, 4 and 5) have been confused as instances of another textual class (BIKE, XING). It should also be noted that several of these instances are essentially the transformed version of each other (e.g. overexposed STOP road markings in row 5 of figure 5).

Next, we would like to compare the recognition rate achieved by CNN networks in this paper against those reported in [2] and [5]. Both of these methods are instances of classical feature extraction and classifier training framework; and report an average recognition accuracy of 96.8% [2] and 99.2% [5] respectively. Comparatively, our best performing network ($\textbf{LeNet}_{96}\textbf{CP}_2$) is able to achieve a recognition accuracy of 99.05% without extra effort required for feature extraction and tuning. It should further be noted that the road markings considered in [5] are all arrow-based which are comparatively easy to recognize compared to textual road

TABLE II. AVERAGE RECOGNITION RATE FOR EACH OF THE NETWORKS.

CNN Architecture	Average % Recognition on Test Set
$\text{LeNet}_{32}\text{CP}_2$	98.6500%
$\text{LeNet}_{64}\text{CP}_2$	98.6500%
$\text{LeNet}_{64}\text{CP}_3$	98.6833%
$\text{LeNet}_{96}\text{CP}_2$	**99.05%**
$\text{LeNet}_{96}\text{CP}_3$	98.2833%

markings. Hence, our direct comparison is against [2] which is also based on the same data set [1] and does consider textual road markings.

Fig. 5. Examples of road markings (row 1 through 5) which have been wrongly classified by each network. row1: **PED** road markings classified as **STOP** by $\text{LeNet}_{32}\text{CP}_2$. row2: **FORWARD** arrow being classified as **40** speed limit by $\text{LeNet}_{64}\text{CP}_2$. row3: **40** speed limit classified as **FORWARD** arrow mark by $\text{LeNet}_{64}\text{CP}_3$. row4: **PED** classified as **XING** by $\text{LeNet}_{96}\text{CP}_2$. row5: **STOP** classified as **BIKE** road markings by $\text{LeNet}_{96}\text{CP}_3$.

V. CONCLUSIONS

In this paper, we have investigated the use of Convolutional Neural Networks for recognition of painted road markings. Earlier work for road marking recognition is based on feature engineering and classifier training, hence involving extra steps and efforts compared to CNNs. Features specific to the training data arise naturally as a by-product of training when the CNN approach is used. We explored five different networks for this problem which have different number of convolutional/max-pooling and fully connected layers and applied at different patch resolutions. Our experimental results on an augmented data set demonstrate the superiority of CNN for ten classes of road markings over the previously reported recognition rates based on classical feature extraction and classifier training. For future, we plan to investigate semantic segmentation of road markings and explore the suitability of networks (e.g. FCN [18], SegNet [21] etc.) for such small road classes which have been proposed for general semantic segmentation.

ACKNOWLEDGEMENTS

This work is supported by NASA EPSCoR (cooperative agreement No. NNX11AM09A) and, in part by NSF PFI.

REFERENCES

[1] T. Wu, A. Raganathan: A Practical System for Road Marking Detection and Recognition. In IEEE Intelligent Vehicles Symposium (IV), 2012. 1, 2, 3, 5

[2] T. Chen, Z. Chen, Q. Shi, X. Huang: Road Marking Detection and Classification Using Machine Learning Algorithms. In IEEE Intelligent Vehicles Symposium (IV), 2015. 2, 4, 5

[3] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, Y. Ma: PCANet: A Simple Deep Learning Baseline for Image Classification? IEEE Transactions on Image Processing, **24**(12):5017–5032, 2015. 1, 2

[4] M.-M. Cheng, Z. Zhang, W.-Y. Lin, P. H. S. Torr: BING: Binarized Normed Gradients for Objectness Estimation at 300fps. In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2014. 2

[5] J. K. Suhr, H. G. Jung: Fast Symbolic Road Marking and Stopline Detection for Vehicle Localization. In IEEE Intelligent Vehicles Symposium (IV), 2015. 1, 2, 4

[6] C. Cortes, V. Vapnik: Support-vector networks. Machine Learning, **20**(3):273–297, 1995. 2

[7] T. Veit, J.-P. Tarel, P. Nicolle, P. Charbonnier: Evaluation of Road Marking Feature Extraction. In IEEE International Conference on Intelligent Transportation Systems, 2015. 2

[8] K.-A. Toh, H.-L. Eng: Between Classification-Error Approximation and Weighted Least-Squares Learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, **30**(4):658–669, 2008. 2

[9] J. Matas, O. Chum, M. Urban, T. Pajdla: Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In British Machine Vision Conference, 2002. 1, 2

[10] E. Rosten, R. Porter, and T. Drummound: Faster and Better: A Machine Learning Approach to Corner Detection . IEEE Transactions on Pattern Analysis and Machine Intelligence, **32**(1):105–119, 2010. 2

[11] J. Greenhalgh and M. Mirmehdi Detection and Recognition of Painted Road Surface Markings. In International Conference on Pattern Recognition Applications and Methods, 2015. 1, 2

[12] N. Dalal and Triggs: Histograms of Oriented Gradients for Human Detection. In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2005. 2

[13] T. Ojala, M. Pietikainen, and T. Maenpaa: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, **24**(7):971 - 987, 2002. 2

[14] D. G. Lowe: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, **60**(2):91–110, 2004. 2

[15] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner: Gradient based learning applied to document recognition. PIEEE, **86**(11):2278–2324, 1998. 2

[16] A. Krizhevsky, I. Sutskever and G. E. Hinton: Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (NIPS), 2012. 1, 2

[17] G. J. Brostow, J. Fauqueur, and R. Cipolla: Semantic object classes in video: A high-definition ground truth database. Pattern Recognition Letters, **30**(2):88–97, 2009. 1

[18] J. Long, E. Shelhamer, and T. Darrell: Fully Convolutional Networks for Semantic Segmentation. In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2015. 1, 5

[19] M. Cordts et al.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2016. 1

[20] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun: Vision meets Robotics: The KITTI Dataset. International Journal of Robotics Research, **32**(11), 2013. 1

TABLE III. CONFUSION MATRIX FOR EACH OF THE TRAINED CNN ARCHITECTURE.

Network		STOP	LEFT	RIGHT	RAIL	35	FORWARD	BIKE	40	PED	XING
$LeNet_{32}CP_2$	**STOP**	**100**									
	LEFT		**99.24**	0.10		0.03	0.32	0.06	0.03	0.06	0.13
	RIGHT		1.30	**98.04**					0.65		
	RAIL		0.45	0.45	**98.42**	0.45			0.22		
	35		0.94	0.38	0.19	**97.73**		0.19	0.38		0.19
	FORWARD		0.61	0.30			**99.08**				
	BIKE		0.58					**99.42**			
	40					1.90	0.63		**96.83**	0.32	0.32
	PED	1.82	0.91			0.91				**96.36**	
	XING		1.10	0.36		0.73			0.73		**97.09**
$LeNet_{64}CP_2$	**STOP**	**100**									
	LEFT		**99.31**	0.26		0.09		0.03	0.13	0.03	0.13
	RIGHT		0.87	**98.04**	0.22	0.22			0.65		
	RAIL		0.22		**99.55**	0.22					
	35	0.19	0.94			**98.50**				0.19	0.19
	FORWARD		2.13				**91.46**		6.40		
	BIKE	0.58	0.58					**98.84**			
	40					0.95			**99.05**		
	PED		0.45							**98.18**	1.36
	XING		0.36			0.73			0.36		**98.54**
$LeNet_{64}CP_3$	**STOP**	**97.62**				2.38					
	LEFT		**99.67**	0.33		0.10	0.13	0.03			0.03
	RIGHT		1.1	**98.47**			0.22				0.22
	RAIL			0.22	**99.55**	0.22					
	35		0.57		0.38	**99.06**					
	FORWARD		0.91				**99.08**				
	BIKE		1.16			1.73		**97.11**			
	40		0.63	0.32		2.53	1.90		**94.30**		0.32
	PED		0.45			1.82				**97.27**	0.45
	XING		1.45	0.36		4.73					**93.45**
$LeNet_{96}CP_2$	**STOP**	**100**									
	LEFT		**99.90**	0.03		0.03			0.03		
	RIGHT		1.09	**98.26**			0.43		0.22		
	RAIL	0.22	0.90		**98.65**	0.22					
	35		1.89		0.19	**97.73**			0.19		
	FORWARD		1.22				**98.78**				
	BIKE		0.58	0.58	0.58			**98.26**			
	40		0.63				1.58		**97.78**		
	PED	0.45	0.91			0.45				**96.82**	1.36
	XING		2.18							0.36	**97.45**
$LeNet_{96}CP_3$	**STOP**	**75.71**	1.90	0.95		7.14		14.28			
	LEFT		**99.77**	0.03		0.06	0.03		0.06		0.03
	RIGHT		0.87	**98.26**		0.22			0.22		0.43
	RAIL	0.22			**99.55**	0.22					
	35		0.57	0.38		**98.68**		0.19			0.19
	FORWARD		0.61				**99.39**				
	BIKE		0.58		0.58			**98.26**			0.58
	40		0.63			0.95			**98.10**		0.32
	PED		0.45			0.45				**98.64**	0.45
	XING		4.00	0.36		0.73					**94.91**

[21] V. Badrinarayanan, A. Kendall, and Roberto Cipolla : SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. arXiv:1511.00561v2 [cs.CV], 2015. 1, 5

[22] K. Simonyan and A. Zisserman: Very Deep Convolutional Networks for Large-scale Image Recognition. arXiv:1409.1556[cs.CV], 2014. 1

[23] C. Szegedy et al.: Going Deeper with Convolutions. arXiv:1409.4842[cs.CV], 2014. 1

[24] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. Lopez: The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In IEEE International Conference on Computer Vision and Pattern Recognition, 2016. 1

[25] A. Shafaei, J. L. Little, and M. Schmidt: Play and Learn: Using Video Games to Train Computer Vision Models. In British Machine Vision Conference (BMVC), 2016. 1

[26] J. M. Alvarez, Y. LeCun, T. Gevers, and A. M. Lopez: Semantic Road Segmentation via Multi-scale Ensembles of Learned Features. In European Conference on Computer Vision (ECCV), 2012. 1

[27] G. Oliveira, W. Burgard, and T. Brox: Efficient Deep Models for Monocular Road Segmentation. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016. 1

[28] C-A. Brust, S. Sickert, M. Simon, E. Rodner, and J. Denzler: Convolutional Patch Networks with Spatial Prior for Road Detection and Urban Scene Understanding. In International Conference on Computer Vision Theory and Applications (VISAPP), 2015. 1

[29] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu: Traffic-Sign Detection and Classification in the Wild. In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2016. 1

[30] P. Dollar, C. Wojek, B. Schiele, and P. Perona: Pedestrian Detection: An Evaluation of the State of the Art. IEEE Transactions on Pattern Analysis and Machine Intelligence, **34**(4):743–761, 2012. 1

[31] X. Du and K. K. Tan: Vision-based approach towards lane line detection and vehicle localization. Machine Vision and Applications (MVA), **27**(2):175–191 , 2016. 1

[32] A. Gurghian, T. Koduri, S. V. Bailur, K. J. Carey, and V. N. Murali: DeepLanes: End-To-End Lane Position Estimation using Deep Neural Networks. In IEEE International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2016. 1

[33] A. Raganathan, D. Ilstrup, and T. Wu, : Light-weight Localization for Vehicles using Road Markings. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013. 1