Embedding Structured Contour and Location Prior in Siamesed Fully Convolutional Networks for Road Detection

Junyu Gao, Qi Wang*, Yuan Yuan

***Abstract*— Road detection from the perspective of moving vehicles is a challenging issue in autonomous driving. Recently, many deep learning methods spring up for this task because they can extract high-level local features to find road regions from raw RGB data, such as Convolutional Neural Networks (CNN) and Fully Convolutional Networks (FCN). However, how to detect the boundary of road accurately is still an intractable problem. In this paper, we propose a siamesed fully convolutional network (named as "s-FCN-loc") based on VGG-net architecture, which is able to consider RGB-channel, semantic contour and location prior simultaneously to segment road region elaborately. To be specific, the s-FCN-loc has two streams to process original RGB images and contour maps respectively. At the same time, the location prior is directly appended to the last feature map to promote the final detection performance. Experiments demonstrate that the proposed s-FCN-loc can learn more discriminative features of road boundaries and converge 30% faster than the original FCN during the training stage. Finally, the proposed approach is evaluated on KITTI road detection benchmark, and achieves a competitive result.**

I. INTRODUCTION

Recently, autonomous driving has drawn great attention with the popularity of intelligent vehicles. Autonomous driving system aims at avoiding accidents during the driving process. Since the most traffic accidents happen on the road region, road detection becomes a fundamental task in the field of autonomous driving. To be specific, an accurate road detection can not only make the vehicle navigate in the correct region but also prompt driving system to focus on the specific conditions on the road, such as the lane markings, pedestrians and other anomalous events. In other words, road information is also viewed as the region of interest (ROI) for lane detection [1], vehicle detection [2], pedestrian detection [3] in the street scenes. Most traditional methods exploit 3D point clouds or location information by some extra sensor such as laser scanner and GPS. In the real world, nevertheless, a human can drive vehicle safely under the complex traffic environment without the above extra information. Thus, how to dig out deeper vision information is still an important issue, which is our focus in this paper.

With the rise of deep learning, the Convolutional Neural Networks (CNN) improve the image comprehension by learning more discriminative and richer features. Fully convolutional networks (FCN) [4] is a variant of traditional CNN, which makes huge progress in many applications, especially in object detection and image semantic segmentation. It focuses on what an object is but ignores the essential spatial structure and location information in images. In view of this, we introduce spatial structure and location prior information into traditional FCN.

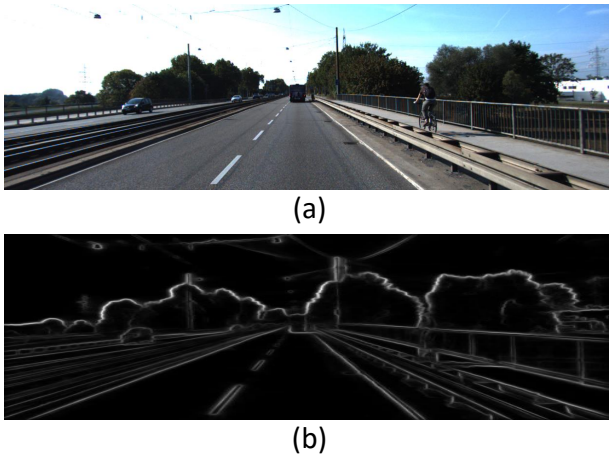

Fig. 1. The exemplar display of an original image and the corresponding contour map.

As we all know, the contours of images represent essential edge information of objects. Given a contour image of a street scene, we are capable of recognizing important objects and their boundaries, such as road, vehicle and so on. Fig. 1 illustrates the exemplar of an original image and the corresponding contour map. Based on the above observation, we regard a CNN classification model as the human's vision system to recognize objects from contour information. After fine-tuning it, the new model will be sensitive to spatial structure and sketch information. Our specific approach is adding the contour convolutional stream to traditional FCN, which shares all parameters with the RGB convolutional stream.

In summary, the overview of our method is described below. Given an input image, the semantic contour map is firstly generated by Structured Forests (SF) [5]. Then, the RGB image and contour map are fed into the s-FCN-loc simultaneously, and the location information is appended to the last feature map. Finally, the road region is output by s-FCN-loc. The concrete flowchart is shown as Fig. 2.

Contributions: The main contributions of this paper are:

1) An s-FCN-loc is proposed that learns more discrimina-

School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, PR China.

Qi Wang is the corresponding author (e-mail: crabwq@nwpu.edu.cn).

This work is supported by the National Natural Science Foundation of China under Grant 61379094 and Natural Science Foundation Research Project of Shaanxi Province under Grant 2015JM6264.

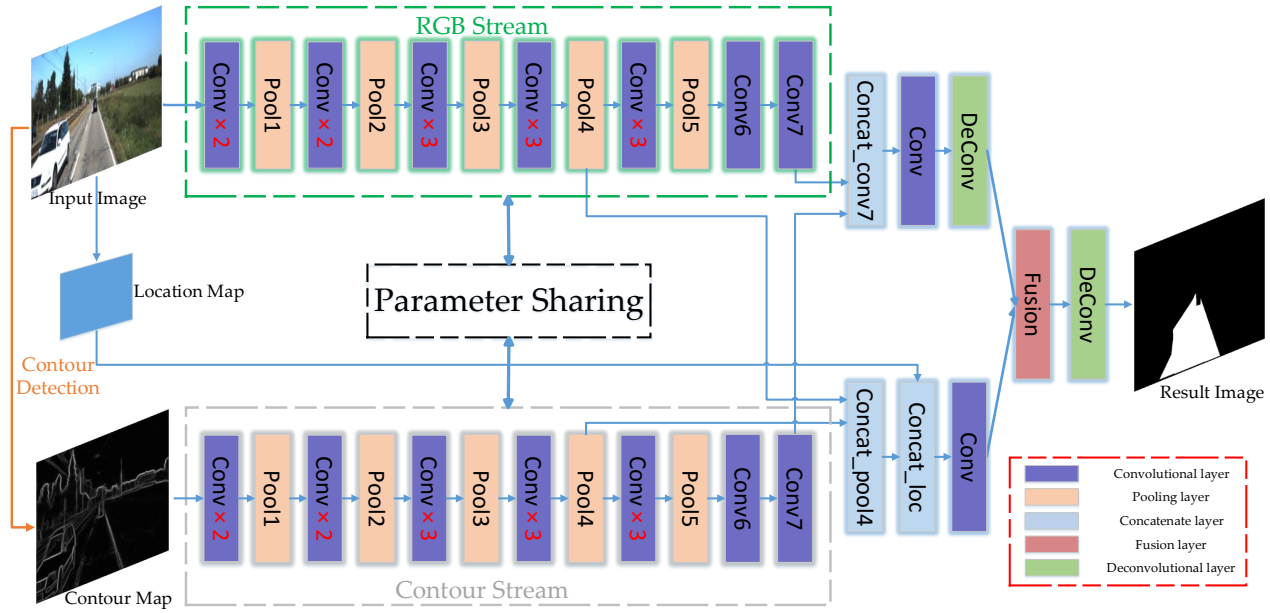

Fig. 2. The flowchart of our proposed siamesed FCN with location prior ("s-FCN-loc" for short). Firstly, given an input image, the semantic contour map is generated by a fast contour detection. Then, the RGB-channel image and contour map are fed into s-FCN-loc, which makes s-FCN-loc focus on learning discriminative features of road boundaries and spatial structure information in street images. At the same time, the location prior is appended to concat_pool4 layer for alleviating false detection. Finally, the feature map is mapped to each pixel by deconvolution operation.

tive features of road boundaries than the original FCN to detect more accurate road regions;

2) Location prior is viewed as a type of feature map and directly appended to the final feature map in s-FCN-loc to promote the detection performance effectively, which is easier than other traditional methods, namely different priors for different inputs (image patches);
3) The convergent speed of training s-FCN-loc model is 30% faster than the original FCN during the whole training process, because highly structured contours prompt the model to converge quickly.

II. RELATED WORK

Before the popularity of deep learning, many approaches [6], [7], [8], [9] about road detection are usually comprised of hand-craft feature extraction, per-pixel (superpixel, or block) classification and contextual information refinement. Alvarez *et al.* [6] propose illumination invariant features to improve the performance in shadowed street scenes. Mendes *et al.* [7] present a block scheme that classifies small images patches using features (RGB, gray-scale, entropy, LBP and Leung-Malik filters responses) to efficiently incorporate contextual cues. Lu [8] proposes a self-supervised method without priori knowledge of the road structure. Wang *et al.* [9] design a novel superpixel based context-aware descriptor by using depth map and transfer label in a nearest neighbor search set. Yuan *et al.* [10] propose an on-line structural learning method for efficiently exacting the drivable road region from video sequences.

Because of the powerful feature learning ability of CNN, the methods exploiting it emerge in recent years. Alvarez *et al.* [11] train a CNN model from noisy labels to recover the 3D layout of a street image. Brust *et al.* [12] propose convolutional patch networks and incorporate location information into the learning process. Mendes *et al.* [13] train a FCN model based on Network-in-Network (NiN) architecture, which utilizes large amounts of contextual information. Mohan [14] proposes a deep deconvolutional networks in combination with traditional CNNs for feature learning to road detection.

Contour detection is a basic task in image processing, which is traced to the last century such as Sobel, Canny edge detectors and so on. Current methods (e.g. [5]) focus on detecting semantic edges, which represent essential contours of a whole image. Contour information usually serves for high-level tasks like object detection and semantic segmentation task. Zitnick and Dollar [15] locate object proposals based on contour information in image. Liu *et al.* [16] combine CNNs and simple edge map via Conditional Random Field for semantic face segmentation.

III. APPROACH

In this Section, we explain the core components of the original FCN [4] and describe the architecture of our proposed s-FCN-loc which combines the RGB, contour and spatial prior information.

A. Fully Convolutional Network(FCN)

In the traditional CNN, the convolutional ("conv" for short) layers focus on extracting local feature in a image, and on the top of multiple conv layers, the fully connected ("fc" for short) layers integrate those high-level local feature maps into a n-D vector by the inner product operation to predict the image's label. Nevertheless, the architecture of

Fig. 3. The manually annotated results of the trial in Section III-B. The first image is the original RGB image; the second is the contour map to subjects; the last two are the results of two subjects.

this network does not predict the label for each pixel. Until 2015, Long *et al.* [4] propose Fully Convolutional Networks (FCN) to tackle the dense prediction problem, which replaces all fc layers with conv layers to produce arbitrary-size output. However, since the deep layer's output loses a lot of location and edge information, the authors of FCN combine deep and shallow layers' feature maps to obtain finer results, which is called as "FCN-xs". The x denotes that the fused feature maps need to be x times upsampled to predict the input-image per-pixel label. In this paper, we adopt the FCN-16s architecture of VGG16-net [17], which fuses the pool4 layer and conv7 layer (convolutionalized fc7 of the original network) by a summing operation. It should be noted that pool4's output is cropped and the conv7's is 2 times upsampled before the fusion for consistent dimensions. VGG16-net can recognize more than 1,000 categories objects from images, which consists of 13 conv and 3 fc layers. In 2014, it wins the second prize in ImageNet Large Scale Visual Recognition Challenge (ILSVRC).

B. Semantic Contour Map

Although FCN fuses the deep and shallow layers' information for alleviating imprecise boundary segmentation to some extent, it is difficult to learn the spatial structure and contour information of an image. Fortunately, semantic contour maps represent them more effectively than traditional edges, such as Sobel, Canny, Roberts and so on. In addition, contour map is a gray-scale not binary image so that the intensity of contour is quantified.

We review the proposed viewpoint in Section I that people is able to recognize important objects and their boundaries from a semantic contour map in the case of knowing a specific scene. In order to validate this thought, we design a simple trial that let some subjects segment each objects from a semantic contour map of a street scene. And they do not go through some special training to recognize objects from contour images. Fig. 3 illustrates the results of this trial. The first image is the original RGB image for comparison; the second is the contour map corresponding to the first; the last two are the results of two randomly selected subjects. From manual segmentation results, we find human brains are capable of understanding scene roughly just using the semantic contour map. Although there are some recognition errors contrast with the original image, it is undeniable that the boundary segmentation is elaborate.

The above trail confirms our thought in a way. Furthermore, we think image classification CNN models can also learn similar ability by supervised training. In this paper, the semantic contour map is generated by SF[1] [5] and a new stream is added to traditional neural network to process contour information. The concrete description is reported in the next section.

C. Siamesed FCN (s-FCN)

Our proposed siamesed network is based on FCN-16s [4], which is shown in Fig. 2. It consists of two streams that handle RGB image and semantic contour map. For integrating the two streams' features, the output of pool4 and conv7 layer are concatenated together (the sizes are $\mathrm{n}\times 1024\times 44\times 44$ and $\mathrm{n}\times 8192\times 16\times 16$ respectively, where n denotes the size of each mini-batch). Considering the correspondence of RGB image and contour map per-pixel, two streams should interact with each other. Thus, at the training stage, the convolutional parameters of two streams are shared with each other and updated simultaneously. However, parameter sharing causes another problem that the number of channels of RGB image is not equal to that of the contour map, namely a single channel gray-scale image. For solving it, the single channel is replicated to three channels.

During the training process, we fine-tune the proposed s-FCN based on the original VGG16-net according to the thought of previous section, and minimize the sum of unnormalized soft-max loss for each pixel by SGD.

D. Incorporating Location Priors in s-FCN

In the street scene, the location prior is important: the objects' spatial distributions are regular. For example, road region is usually located at the bottom of images, and the buildings and trees are on both sides of the roads. Thus, utilizing this location prior is essential to remove the false detection. However, the traditional FCN is only sensitive to local appearance features instead of location prior information, which causes some unreasonable results, such as some building regions are mistakenly recognized as road. In order to reduce the problem, location prior is appended to feature map directly. Compared with the previous methods [12] (the location prior needs to be generated and enter into CNN at the time of each patch inference), the location prior is generated only once and for all images.

To be specific, location prior (the coordinate values of x and y axis in the input image) is represented as a 2-channel feature map, which is appended to the last feature map in

[1]The source code is provided by Piotr Dollár in https://github.com/pdollar/edges

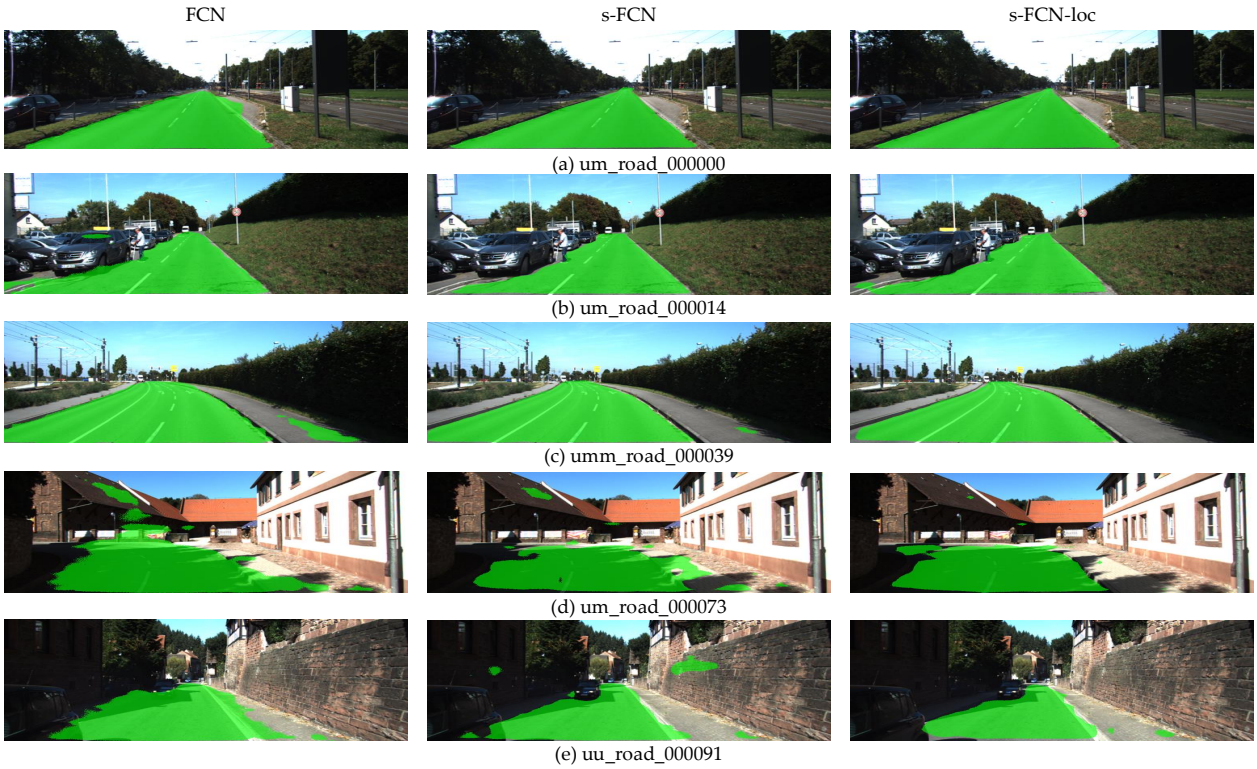

Fig. 4. Exemplar results of different models (the original FCN, s-FCN and s-FCN-loc) on KITTI benchmark testing set.

s-FCN. Since the height and width of the feature map are smaller than the input's, location maps should be resized to the size of the last feature map for concatenating them. It's important to note that there are two final feature maps to be fused in s-FCN: the outputs of concat_pool4 layer and concat_conv7 layer (the height $\times$ width are 44×44 and 16×16, respectively). We choose to add location prior to concat_pool4 layer, because of larger output and more accurate location prior.

IV. EXPERIMENT

A. Dataset and Settings

In order to evaluate the proposed approach, we select the road detection dataset in KITTI Vision Benchmark Suite [18], which consists 579 images (289 training images and 290 testing images respectively) with a resolution of 375×1242 pixel. The entire data set is divided into three categories, the concrete descriptions of which are shown in Table I. The evaluation server of the benchmark ranks all submitted methods according to their max F-measure on the Bird's-eye view (BEV). The benchmark features color stereo, GPS and LIDAR data for each scene. As for this dataset, we only exploit monocular color data to detect road region in the experiment.

In the experiment, we report the three comparative results: the original FCN-16s, s-FCN and the full version (s-FCN-loc), respectively. For showing the effect of each component, the training set is randomly divided into two classes (272 images for training and 17 images for validation), and all of the stage results are evaluated on the validation set. Moreover, we also list the result of our s-FCN-loc on the benchmark server to compare with other popular methods.

TABLE I

THE DETAIL INFORMATION OF KITTI DATASET IS SHOWN AS BELOW

Scene category	Train no.	Test no.
UU (urban unmarked)	98	100
UM (urban marked two-way road)	95	96
UMM (urban marked multi-lane road)	96	94
URBAN(All)	289	290

The experimental environment is equipped with Intel(R) CPU Xeon(R) E5-2697 v2 @ 2.70GHz, 128GB RAM, and four NVIDIA Tesla K80 GPUs. As for the software environment, we modify the standard Caffe[2] by merging the #2016[3] pull request (PR) of Caffe for saving memory during training process.

B. Implementation Details

In the entire experiment, original images are resized to 500×500 to enter into s-FCN-loc. Contour maps are generated by default parameters (the number of decision trees is 1) of SE-SS in SF [5]. We use two fixed learning rates of 10^{-10} for weights and 2×10^{-10} for biases, a mini-batch size of 4 images, momentum of 0.99 and decay of 0.0005 (in the training process, we find the models are only sensitive to the learning rate). We also set dropout ratio of 0.5 in conv6 and

[2]http://caffe.berkeleyvision.org/

[3]https://github.com/BVLC/caffe/pull/2016

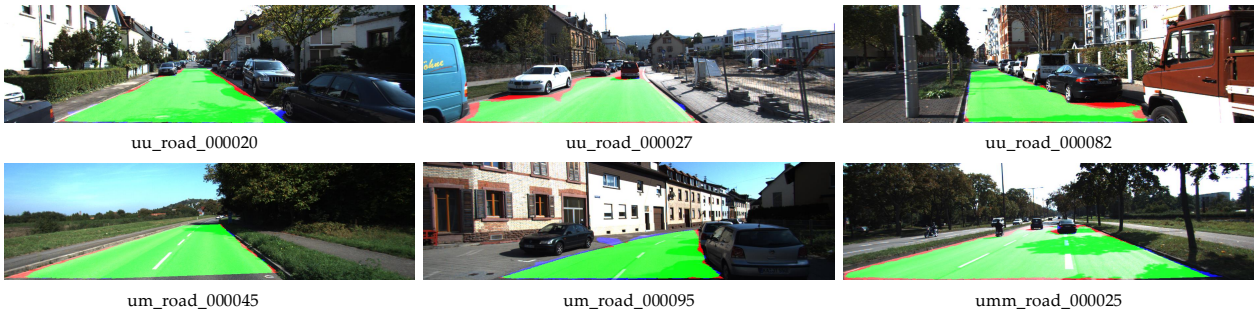

Fig. 5. Exemplar results on the KITTI server. The green, blue and red regions denote respectively true positives, false positives and false negatives.

conv7 layers. Besides, the size of location map is 44×44 for correspondence of concat_pool4's output.

C. Performance on Our Validation Set

Table II presents the four metrics (F1-measure, accuracy, precision and recall) of different stage models on the validation set. Through quantitative results, we find each criterion has improved to some extent except the recall rate, which demonstrates the effectiveness of our proposed siamesed FCN and location prior incorporation.

TABLE II
COMPARISON OF DIFFERENT STAGES ON OUR VALIDATION SET (IN %)

Methods	F1-measure	Accuracy	Precision	Recall
Baseline(FCN)	92.53	97.58	89.40	95.90
s-FCN	94.60	98.31	93.64	95.60
s-FCN-loc	**95.38**	**98.56**	**94.29**	**96.48**

In order to analyze the detection performance further and intuitively, Fig. 4 displays the visualization results of road detection from UM, UMM and UU category. From the first three rows, FCN's results are unclean, especially at the road boundary. For example, the distant sidewalk is mistakenly recognized as road in the third row. By comparison, however, s-FCN and s-FCN-loc are sensitive to the contours of objects, which segment road region accurately. In the last two sets of exemplars, some building regions are mistaken for road in FCN and s-FCN, the positions of which are rarely where the road locates. As we can see from the results of the third column, s-FCN-loc model alleviates this problem.

D. Performance on KITTI Benchmark

For comparing our proposed s-FCN-loc with other popular methods, we submitted the final results to the KITTI server. Table III shows the results of the first ten real-name submissions[4] and ours in the Urban Road category. Our method achieves a competitive result of Max F-measure 93.26%, which does not differ much from the best 93.43% of DDN [14]. In the listed methods, DDN [14], FTP [19], FCN_LC [13], StixelNet [20] and MAP [19] are deep learning methods and only take advantage of RGB information; NNP [21], FusedCRF [22] and ProbBoost [23] exploit 3D information such as stereo vision and LIDAR data; HIM [24] and CB [7] make use of hand-craft features to detect road region. As for runtime, the proposed method is the 4-th place in all 11 real-name submissions. Compared with the faster methods, the proposed method is superior to them according to the max F-measure. In general, our method is competitive in terms of detection and time performance.

[4]The leader boards on the server includes some anonymous submissions. As for these anonymous submissions, because without their detail information, we do not list them in this paper. It is noted that the proposed model obtains the 8-th prize in all 52 submissions.

TABLE III
COMPARISON OF DIFFERENT APPROACHES ON URBAN ROAD (IN %)

Methods	MaxF	Pre.	Rec.	FPR	FNR	Runtime
DDN [14]	93.43	95.09	91.82	2.61	8.18	2s
Ours methods	93.26	94.16	92.39	3.16	7.61	0.4s
FTP [19]	91.61	91.04	92.20	5.00	7.80	0.28s
FCN_LC [13]	90.64	90.87	90.72	5.02	9.28	0.03s
HIM [24]	90.07	91.62	89.68	4.52	10.32	7s
NNP [21]	89.68	89.67	89.68	5.69	10.32	5s
StixelNet [20]	89.12	85.80	92.71	8.45	7.29	1s
CB [7]	88.97	89.50	88.44	5.71	11.56	2s
FusedCRF [22]	88.25	83.62	93.44	10.08	6.56	2s
MAP [19]	87.80	86.01	89.66	8.04	10.34	0.28s
ProbBoost [23]	87.78	86.59	89.01	7.60	10.99	150s

Fig. 5 shows our final results on the KITTI benchmark server. The green, blue and red regions denote respectively true positives, false positives and false negatives. As we can see from the displayed exemplars, the proposed model generalizes from the training set to the testing set. Furthermore, some erroneous and missed detections usually occur in the boundaries of road regions.

E. Analysis of Convergent Speed

Fig. 6 illustrates the trends of convergence for different models. From it, we find the convergent speeds of s-FCN and s-FCN-loc are faster than the original FCN, and the curve lines of s-FCN and s-FCN-loc are very close during the training process. To be specific, the original FCN converges after $240,000$ iterations, but the proposed s-FCN and s-FCN-loc only need $80,000$ iterations to converge. In terms of iteration number, the convergent speeds of the latter two are about 70% faster than that of the original FCN. As a matter of fact, it is unfair to measure the convergent speed of each model by iteration number, because the computation time of each iteration is not equable for different models.

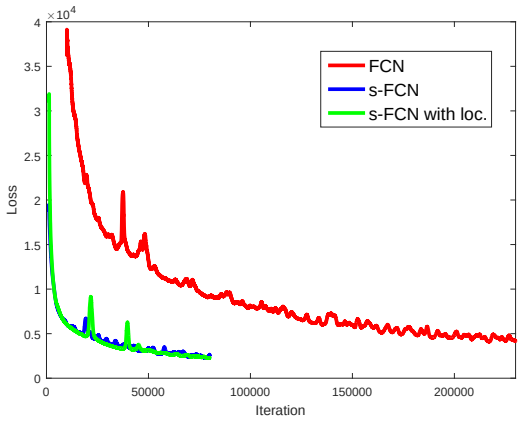

Fig. 6. The convergence trends of FCN (red), our proposed s-FCN (blue) and s-FCN-loc (green).

Since the original FCN has only one stream, the time of its one iteration is only half of s-FCN and s-FCN-loc. Even so, the convergent speeds of s-FCN and s-FCN-loc are still 30% faster than the original FCN according to the overall training time. It's worth mentioning that the difference of convergent speeds is more larger in the initial stage (the first 1,000 iterations) of training.

The above phenomenon demonstrates that effective spatial structure and contour information speed up the training models. In essence, the semantic contour maps are regarded as high-level feature more than raw RGB images. The neural network can easily learn more effective semantic representation from highly structured contour maps, which guides model to convergence more quickly. So it saves more training time than traditional single-stream network. Nevertheless, the neural network can not only extract features from semantic contour maps because it loses a lot of detailed and colorful information. Thus, it needs to have two streams to handle RGB images and semantic contour maps, respectively.

V. CONCLUSIONS

This paper presents an s-FCN-loc model based on VGG-net for road detection which is able to learn discriminative features of road boundaries to detect more accurate road regions by exploiting the RGB-channel image, semantic contour and location prior simultaneously. Stepwise experimental results verify the effectiveness of each component in the proposed method. We also find that s-FCN-loc converges faster than the original FCN at the training stage, which saves more training time. In the future, we plan to transform the siamesed architecture to other neural network and explore the changes of conv layers brought by adding contour stream through visualization operation, which will prompt us to understand the effects of contour map in depth.

REFERENCES

[1] M. Revilloud, D. Gruyer, and M.-C. Rahal, "A new multi-agent approach for lane detection and tracking," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3147–3153.

[2] J. Cheng, Z. Xiang, T. Cao, and J. Liu, "Robust vehicle detection using 3d lidar under complex urban environment," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 691–696.

[3] A. Angelova, A. Krizhevsky, and V. Vanhoucke, "Pedestrian detection with a large-field-of-view deep network," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 704–711.

[4] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[5] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1841–1848.

[6] J. M. Á. Alvarez and A. M. Lopez, "Road detection based on illuminant invariance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 184–193, 2011.

[7] C. C. T. Mendes, V. Frémont, and D. F. Wolf, "Vision-based road detection using contextual blocks," *arXiv preprint arXiv:1509.01122*, 2015.

[8] X. Lu, "Self-supervised road detection from a single image," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2989–2993.

[9] Q. Wang, J. Fang, and Y. Yuan, "Adaptive road detection via context-aware label transfer," *Neurocomputing*, vol. 158, pp. 174–183, 2015.

[10] Y. Yuan, Z. Jiang, and Q. Wang, "Video-based road detection via online structural learning," *Neurocomputing*, vol. 168, pp. 336–347, 2015.

[11] J. M. Alvarez, T. Gevers, Y. LeCun, and A. M. Lopez, "Road scene segmentation from a single image," in *European Conference on Computer Vision*. Springer, 2012, pp. 376–389.

[12] 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," *arXiv preprint arXiv:1502.06344*, 2015.

[13] C. C. T. Mendes, V. Frémont, and D. F. Wolf, "Exploiting fully convolutional neural networks for fast road detection," in *2016 IEEE International Conference on Robotics and Automation, ICRA 2016, Stockholm, Sweden, May 16-21, 2016*, 2016, pp. 3174–3179.

[14] R. Mohan, "Deep deconvolutional networks for scene parsing," *arXiv preprint arXiv:1411.4101*, 2014.

[15] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European Conference on Computer Vision*. Springer, 2014, pp. 391–405.

[16] S. Liu, J. Yang, C. Huang, and M.-H. Yang, "Multi-objective convolutional learning for face labeling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3451–3459.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[18] J. Fritsch, T. Kuehnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 1693–1700.

[19] A. Laddha, M. K. Kocamaz, L. E. Navarro-Serment, and M. Hebert, "Map-supervised road detection," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 118–123.

[20] D. Levi, N. Garnett, E. Fetaya, and I. Herzlyia, "Stixelnet: A deep convolutional network for obstacle detection and road segmentation." BMVC, 2015.

[21] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *Advances in Neural Information Processing Systems*, 2015, pp. 424–432.

[22] L. Xiao, B. Dai, D. Liu, T. Hu, and T. Wu, "Crf based road detection with multi-sensor fusion," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 192–198.

[23] G. B. Vitor, A. C. Victorino, and J. V. Ferreira, "A probabilistic distribution approach for the classification of urban roads in complex environments," in *Workshop on IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[24] D. Munoz, J. A. Bagnell, and M. Hebert, "Stacked hierarchical labeling," in *European Conference on Computer Vision*. Springer, 2010, pp. 57–70.