From one to many: unsupervised traversable area segmentation in off-road environment

Li Tang, Xiaqing Ding, Huan Yin, Yue Wang, *Member, IEEE*, and Rong Xiong *Member, IEEE*

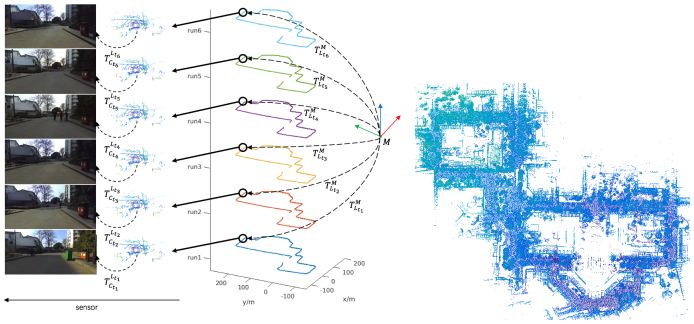
***Abstract*—Traversable area segmentation is important for safe navigation of mobile robot in outdoor environment. To address this problem, we propose a unified framework to register data across sessions, on which an unsupervised method is presented for traversable area segmentation intended for unstructured environments. With data collected on a vehicle equipped with camera and laser, the proposed method can generate massive label images for traversable and obstacle area without any human intervention, which are fed as training samples of a pixel-wise semantic neural network. In deployment, only a monocular camera is needed to work with the trained network, without structured assumption of the road such as lanes and traffic signs. The proposed method is validated on 4 datasets to demonstrate performance on traversable area segmentation. Moreover, it is shown that our method can be generalized to varied appearance at different location and time with distinct sensors.**

I. Introduction

Traversable area segmentation in unstructured environments is a vital topic for autonomous vehicle. Precise segmentation of roads is necessary for safe navigation and correct behavior controlling. Active sensors like LiDAR suffer from sparsity of data which makes it hard to avoid small obstacle avoidances. Furthermore high cost prevents them from being widely used. Low-cost cameras are gaining more and more popularities in road understanding. Recent success of deep neural network, especially CNNs for semantic segmentation, shows potential of machine vision in this field.

At present most available ADAS(Advanced Driver Assistance Systems) are restricted in on-road environment, mainly aiming at lane-keeping task. These systems rely on lane extraction and traffic sign recognition (e.g. [1], [2]), which cannot be applied in unstructured environment directly. Semantic segmentation using supervised learning like [3] and [4] shows notable results but depends heavily on labeled data, which is precious and labor-intensive, making it hard to leverage massive raw data. To avoid manual labor, [5] proposed a self-learning method that assumes an initial guess and grows the traversable area gradually, but the on-line learning framework cannot make use of collected data, failing to further improve the performance through running. In [6] a weak supervised method was proposed to generate labeled data, but it just gave 'path proposals', which limited space for traversable area. More importantly, it confused the model by providing the inconsistent labeling at the same place because of the different route across sessions.

Li Tang, Xiaqing Ding, Yue Wang, Huan Yin and Rong Xiong are with the State Key Laboratory of Industrial Control and Technology, Zhejiang University, Hangzhou, P.R.China. Yue Wang is with iPlus Robotics Hangzhou, P.R.China. Yue Wang is the corresponding author wangyue@iipc.zju.edu.cn. Rong Xiong is the co-corresponding author.



(a) Proposed framework. (b) Global consistent map.

Fig. 1: After repeating the same route for times, data is organized in a unified coordinate by building a global map. Poses of laser can be obtained by registering against the map, on which poses of other sensors (e.g. camera) are chained.

In this paper we propose a unified framework to fuse information from different sensors across multiple sessions, so as to automatically generate labeled images for training of traversable area segmentation model. We first use LiDAR to build a globally consistent map, in which data from any sensor at any time can be assigned with a global pose by localization. Therefore, the routes in the different sessions can be registered together to provide consistent labeling. The method does not make use of road features such as lanes and signs so that it is pretty suitable for unstructured environment. So massive data across many sessions, which includes lots of variations, can be leveraged to train a traversable area segmentation model without any manual labeling or intervention. The model employed in our experiments is deep neural network (DeepLab [7]), which only requires equipment of a laser and a monocular camera on the vehicle in training phase but only a monocular camera in testing phase.

Our contribution is presented as following:

- a unified framework is proposed to register data across many sessions with different spatial, temporal and sensory properties for data mining.
- Based on the framework, an unsupervised sensor fusion method is presented to generate consistent labeling for training of the traversable area segmentation model.
- The proposed algorithm is validated extensively on both internal and external data, illustrating its generalization.

The remainder of the paper is organized as following:

978-1-5386-3742-5/17/$31.00 © 2017 IEEE

Related work is discussed in Session II. Methods including framework and label generation are introduced in Session III, while experiments to validate our methods are presented in Session IV. At last we draw conclusion in Session V.

II. Related work

3D sensors such as laser or LiDAR provides good geometry information but the sparsity of them brings difficulty to traversable area segmentation. Thus they often work with cameras. For example, [8] jointly estimated the MAP of a model which integrated Radar and camera to detect road boundary, while [9] used laser for curb lines detection followed by fusion with road paints detected by cameras. [10] uses laser to find out nearby traversable area, which was later used as seeds to train a traversable pixel classifier with RGB camera. Requirement of high cost sensors prevents these solutions from being practical deployed.

Lots of the purely camera-based traversable area detection methods rely on road lanes, typically including lane features extraction [11][12][13] and road model fitting [14][15], optionally followed by time integration [16][17]. These handcrafted algorithms are interpretable and work well in structured environment, but meet problems on roads where no lane exists or the road structure is broken. To overcome it, [5] adopted self-learning strategy, which assumed an initial gauss of traversable patch and gradually expanded the traversable region on-line. However it cannot make use of collected data to improve the classifier.

Classical semantic segmentation has ever been used for detecting traversable area. Those methods can be classified as random forest(e.g. [18]) and CRF (e.g. [19]). However they are difficult to achieve high accuracy and far from practical.

Recent trend in computer vision to use deep convolutional network as a feature extraction tools is also seen in traversable area segmentation. It is formulated as the pixel-wise semantic segmentation problem, where one class of the label is road surface or traversable area. [20] used only convolution layers followed by fractionally strided convolution layers to achieve pixel-wise prediction, while [3] employed an encode-decode architecture with skip connections. However they rely heavily on manually labeled data (e.g. [4][21]), bringing difficulty to obtaining large scale datasets, which is necessary to capture enough variation. [22] used a virtual world to automatically generate synthetic images with pixel-level annotations, but there is no desirable solution to adapt it to realistic data. Some neural network approaches also leverage high definition maps to improve segmentation performance ([23]), such as landmark based maps and semantic point cloud maps. But such maps are expensive thus not widely used in research.

Our work is mostly inspired by [6], which leverages motion information and additional sensors of a data collection vehicle to automatically generate labeled data. However, [6] only generates 'path proposal', a small part of actual traversable area on which the vehicle has driven. This limits the space for navigation, and confuses the model training as inconsistent labeling is provided due to the route changing across sessions.

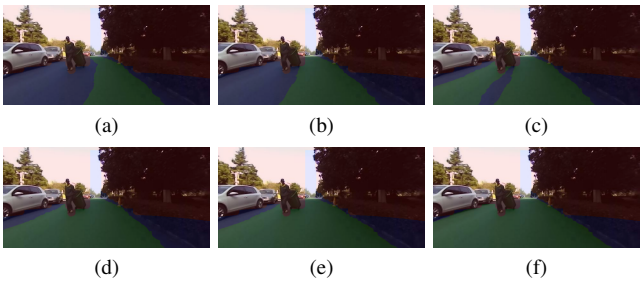


Fig. 2: (a)∼(f) present the fusion process. The labeled traversable area is getting wider when fusing more sessions.

III. Methods

We propose a unified framework to register data in different dimensions, e.g. spatial, temporal and sensory. The key is to represent data in a global coordinate. Our method firstly building a global map M with data from a specific sensor L in a single session, then assigning a global pose $T^W_{L_{t_i}}$ to L_{t_i} by performing corresponding localization algorithm. L_{t_i} is data acquired at some time t_i from sensor L, where time t_i contains two parts, namely, number of session it lays on and the running time on that session. With the above definitions relative transform between two frames of data, L_{t_i} and L_{t_j}, is denoted by $T^{L_{t_i}}_{L_{t_j}} = (T^W_{L_{t_i}})^{-1} T^W_{L_{t_j}}$.

Our framework supports multiple sensors setting. For some sensor C other than L, it is assumed that relative transform between L and C is fixed, being denoted by T^L_C, which is obtained by calibration. It is important to know transform between C_{t_i} and C_{t_j}, namely, $T^{C_{t_i}}_{C_{t_j}} = (T^L_C)^{-1} T^{L_{t_i}}_{L_{t_j}} T^L_C$, which depends on localization results of L_{t_i} and L_{t_j}.

In our case, L is laser and C is monocular camera. The kth point from laser at time t_i is denoted by $p^{t_i}_k$. $w^{t_j}_{\{l,r\}}$ is contact points between wheels and ground plane at time t_j in camera coordinate, with l and r corresponding to left and right wheel, which is assumed to be obtained by calibration or simple measurement.

A. Label generation

We define 3 kinds of semantic classes, that is, traversable, obstacle and unknown. To automatically generate labeled images, we leverage sensory data collected from a mobile robot on the same route for several times repeatedly driven in a varied campus environment manually. With assumption that area once covered by the vehicle is traversable and laser point above ground plane is part of obstacle, the label generation process is as following:

Traversable area The proposed traversable area is based on 'path proposal' P_s in [6], which is the future path of vehicle in a specific session s projected onto the current image C_{t_i}. The path is determined by trajectory of contact points between wheels and the ground relative to the camera C_{t_i}, whose position on the image plane is:

$$ {}^{C_{t_i}}w_{\{l,r\}}^{t_{i+j}} = K \cdot T_{C_{t_{i+j}}}^{C_{t_i}} \cdot w_{\{l,r\}}^{t_{i+j}} = K \cdot (T_C^L)^{-1} \cdot T_{L_{t_{i+j}}}^{L_{t_i}} \cdot T_C^L \cdot w_{\{l,r\}}^{t_{i+j}} $$

where $T_{L_{t_{i+j}}}^{L_{t_i}}$ is localization result as mentioned above and T_C^L is calibration between laser and camera. K is the intrinsic matrix of camera. Path proposal P_s is a polygon with vertexes:

$$ P_s = \{ {}^{C_{t_i}}w_l^{t_i}, {}^{C_{t_i}}w_l^{t_{i+1}}, \cdots, {}^{C_{t_i}}w_l^{t_{i+h-1}}, {}^{C_{t_i}}w_l^{t_{i+h}}, \\ {}^{C_{t_i}}w_r^{t_{i+h}}, {}^{C_{t_i}}w_r^{t_{i+h-1}}, \cdots, {}^{C_{t_i}}w_r^{t_{i+1}}, {}^{C_{t_i}}w_r^{t_i} \} $$

where h is the length of trajectory projected on the image plane. Similar to [6], it is chosen based on the criterion that $\left\| T_{C_{t_{i+h}}}^{C_{t_i}} \cdot w_l^{t_{i+h}} - w_l^{t_i} \right\| > 60$.

From each session s, one path proposal P_s is generated. In multiple sessions case, the generated path proposals can be fused together to become a wider traversable area. In fact, it is 'union' operation on the image plane, namely $\bigcup P_s$. Fig. 2 is an example of the fusion process, in which 1, 2, 3, 4, 5, 13 sessions are gradually fused, leading to wider traversable area. The fusion is benefit from the unified framework, in which path proposals are generated under the same coordinate. Methods like [6] relaying on locality(e.g. visual odometry) cannot share information across sessions.

Obstacle area Obstacle area represents region that the vehicle should not driven on. The traversable area generated above is in fact not perfect, missing some pixels on the border. Thus it is important to find out the actual obstacle area, instead of treating the non-traversable region as obstacle.

In our sensor settings, obstacle can be detected by laser, which returns 3D points of the surrounding structure. Laser point $p_k^{t_i}$ at time t_i are projected onto image plane of C_{t_i}:

$$ {}^{C_{t_i}}p_k^{t_i} = K(T_C^L)^{-1}p_k^{t_i} $$

The pixels above the 2D pixel ${}^{C_{t_i}}p_k^{t_i}$ are marked as obstacle. To avoid mistakenly seeing points on ground as obstacle, we also fit a ground plane in current laser scan using [24] and ignore point 0.25m bellow the plane. Dilatation is also performed on the obstacle mask to fill holds caused by sparsity of laser sensor. Opposite to traversable area, only laser scan of current time is used. For safety, obstacle area is prior to the traversable one, thus if one pixel is marked as traversable and obstacle simultaneously, it is regarded as obstacle.

Unknown area As the label generation procedure is automatic, some area may be neither traversable nor obstacle, which is regarded as unknown area. such region is in fact place that have not been traveled by the experimental vehicle and no obstacle point is found on it. Pixel classifier used in this paper(DeepLab [7]) outputs probabilities of every pixels, which is useful for further unknown area refinement.

B. Network training

Method proposed in III-A can generate vast of labeled images, depending on the distance traveled by the data vehicle. In theory, the labeled data generated by the framework can be generalized to any pixel-wise semantic segmentation neural network. During deployment, the network can be fed with images from a monocular camera to predict traversable area. In this paper DeepLab [7] is chosen as pixel classifier for its trade off between accuracy and efficiency. Similar to [6], we build a histogram on turning angle of the vehicle and sample uniformly among the bins, to avoid unbalance data among roads with different curvature.

TABLE I: Datasets Summary

Datasets	YQ21	YQ-South	Shadow Road	Variational Road
Camera Sensor	ZED Stereo Camera	ZED Stereo Camera	GoPro Camera	Bumblebee Stereo Camera
Input Resolution	672x376	672x376	640x480	640x480
Laser Sensor	Velodyne VLP-16	Velodyne VLP-16	None	None
#Sessions	21	1	1	1
#Images	232627	39567	137	254

IV. Experiments

4 experiments are conducted to validate the framework.

A. Experimental platforms and datasets

3 experimental platforms are involved in experiments. The vehicle is a four-wheeled mobile robot equipped with a ZED stereo Camera, a Velodyne VLP-16 laser scanner and a D-GPS(Fig. 3a). Only images from the left camera of ZED are used. The laser with 16 beams is leveraged to detect obstacles as in Session III-A and is used to build a global map along with D-GPS using [25]. The second platform is bicycle mounted with a GoPro camera. The third one is a car-like platform equipped with a Bumblebee Stereo Camera, of which only the left camera is used.

4 datasets are used for evaluation(summarized in Table I):

- YQ21: recorded by driving the robot in Fig. 3a on a 1km route(blue line in Fig. 4) for 21 times manually. Captures appearance variation at different time in three days.
- YQ-South: collected by piloting robot in Fig. 3a on a longer route(red line in Fig. 4, 4.9km) manually.
- Shadow Road: captured by GoPro Camera mounted on a bicycle as in Fig. 3b with shadows caused by trees, whose instability may add blur to the image(Fig. 5a).
- Variational Road: gathered by a Bumblebee Stereo Camera on robot in Fig. 3c with varied conditions such as blur, barriers, changing illumination and significant texture-color changes(Fig. 5b).

Most scenes do not have road lanes and traffic signs. In following the rth session of a specific dataset in the dth day is denoted by $s_{d,r}$.

B. Preprocessing and training

Experimental setups including dataset usage and network assignment are presented in Table II.

(a) (b) (c)

Fig. 3: Experimental platforms. The YQ21 and YQ-South datasets are collected on (a), with the Shadow Road and Variational Road dataset being recorded on (b) and (c) respectively. (b) is the sketch of experimental setup for experiment 3, in which a GoPro camera is mounted on a bicycle.

Fig. 4: Blue line represents route of YQ dataset, over 1km. The red line indicates route of experiment 3, which is much longer than the former(4.8km).

(a) Shadow Road

(b) Variational Road

Fig. 5: Sample images for dataset in [5].

The first two experiments are conducted on YQ21 datasets. In experiment 1, 8 sessions are chosen as validation set while the rest are randomly split as training and testing set with ratio of 7:3(See Table I for detail). To illustrate necessary of fusion across sessions, 6 networks are trained with different labels. Precisely, for the same image, 6 label images are generated by fusing 1, 2, 3, 4, 5, 13 sessions respectively, because it is found that trajectories from 5 sessions almost cover most of the traversable area. Those networks are trained with exactly the same settings, except for the input labels. Experiment 2 has no difference with the first one except that the validation set are selected from sessions after noon(13:00), while training/testing only capture variation before noon(Table I). Another slight difference is that only 12 sessions are in training set, 12 sessions are fused at most in this experiment.

Experiment 3 and 4 leverage N_6 trained in experiment 1 to validate generalization ability. YQ-South, Shadow Road and Variational Road are fed into N_6 to make prediction.

We use Deeplab([7]) as pixel-wise segmentation classifier. In all experiments, images are resized to 321x153 as input of the neural network. For Shadow Road and Variational Road, images are first cropped vertically to a proper size then resized resizing to 321x153. Training phrase last for 40000 iteration with batch size of 10.

C. Evaluation metrics

Two kinds of ground truths to evaluate network performance, fusing 1 and 21 sessions receptively. They are referred as SGT(Single-session Ground Truth) and MGT(Multi-session Ground Truth) in the following. In other words, SGT $= P_i$, while MGT $= \bigcup_{i=1}^{21} P_i$.

Three evaluation metrics are used in experiment 1 and 2, that is, precision $PRE = \frac{N_{TP}}{N_{TP}+N_{FN}}$, recall $REC = \frac{N_{TP}}{N_{TP}+N_{FP}}$ and Intersection Of Union $IOU = \frac{N_{TP}}{N_{FP}+N_{TP}+N_{FN}}$, where T/F indicates that the prediction result is true or false, while P/N means that the pixel belongs to a specific class or not. Take traversable area for example, T_{TP}/T_{FP} is number of traversable pixels predicted as traversable/non-traversable, while T_{TN}/T_{FN} is number of non-traversable pixels predicted non-traversable/traversable.

To compare with [5], two metrics in [5] , FPR and FNR, are used. In fact, FPR $= 1 -$ (PRE of traversable area) while FNR $= 1 -$ (REC of obstacle area).

D. Experiment 1: basic performance

Experiment 1 leverages YQ21 to evaluate basic performance. Results can be seen from Fig. 6, where precision, recall and IOU are used as evaluation metrics, with MGT being ground truth. Recall of traversable area grows rapidly as fusing more sessions, with the best path over twice as wide as the worst case, while recall of unknown and obstacle remains above 90%. It means that our method actually improves performance of detecting traversable area without satisfying capability to distinguish non-traversable region. Similar conclusion can be obtain from precision. The overall performance metric, IOU, get steady growth with the best performance of over 70% in traversable and unknown area, showing that the proposed algorithm is practical to be used on non-structure environment. All metrics for obstacle area remain high in all cases because fusion mostly changes boundary of traversable and unknown area, making little effect on obstacle, which ensures that our method enables safe navigation.

[6] using single session as ground truth, which is the same as using one session in our algorithm(row 1 in Table III). Thus compared to [6], our method has slight improvement of over 6%, meaning that more traversable area is found after fusion.

E. Experiment 2: appearance variation

Experiment 2 is intended to illustrate generalization of the proposed method to varied appearance. Training samples of

TABLE II: Experimental setups

Experiment	Image Source	#Training Images	#Testing Images	#Validation Images	Sessions for Validation	Trained Networks[1]	Networks for Validation
1	YQ21	60232	25813	88440	$s_{1,2}$, $s_{1,5}$, $s_{1,8}$, $s_{2,1}$, $s_{2,4}$, $s_{2,7}$, $s_{3,2}$, $s_{3,5}$	$N_1(1)$, $N_2(2)$, $N_3(3)$, $N_4(4)$, $N_5(5)$, $N_6(13)$	N_1, N_2, N_3, N_4, N_5, N_6
2	YQ21	54662	23426	101331	$s_{1,6}$, $s_{1,7}$, $s_{1,8}$, $s_{2,5}$, $s_{2,6}$, $s_{2,7}$, $s_{2,8}$, $s_{3,4}$, $s_{3,5}$	$N_1^{'}(1)$, $N_2^{'}(2)$, $N_3^{'}(3)$, $N_4^{'}(4)$, $N_5^{'}(5)$, $N_6^{'}(12)$	$N_1^{'}$, $N_2^{'}$, $N_3^{'}$, $N_4^{'}$, $N_5^{'}$, $N_6^{'}$
3	YQ-South	-	-	39567	$s_{1,1}$(YQ-South)	-	N_6
4	Shadow Road + Variational Road	-	-	391	$s_{1,1}$(Shadow Road) + $s_{1,1}$(Variational Road)	-	N_6

[1] In format of $N_i(f)$ where N_i is the network ID and f is the number of fused sessions. For example N_6 represent network fused with 13 sessions.

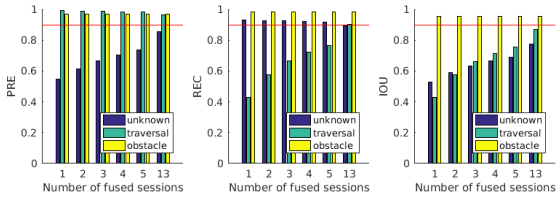


Fig. 6: Precision, recall and IOU of unknown, traversable and obstacle area in experiment 1. The horizontal red line indicates value of 0.9.

TABLE III: Comparison of recall: experiment 1 vs 2

#Fused sessions	Traversable[1]	Obstacle[1]
1	**0.8848** / 0.86786	**0.9816** / 0.9780
2	**0.9187** / 0.90242	**0.9814** / 0.9778
3	**0.9301** / 0.91629	**0.9819** / 0.9797
4	**0.9369** / 0.92393	**0.9818** / 0.9793
5	**0.9398** / 0.92701	**0.9829** / 0.9774
13/12	**0.9511** / 0.94359	**0.9828** / 0.9775

[1] Values represented for experiment 1/ experiment 2.

$N_1 \sim N_6$ and $N_1^{'} \sim N_6^{'}$ are slightly different, thus domains of them are not the same, which may reduce the performance of neural network. Table III is the comparison of networks trained in experiment 1 and 2, with precision as evaluation metric and SGT as ground truth. As can be seen, performance decreases slight without much appearance variation, thus our temporal fusion is helpful to overcome environment changes.

F. Experiment 3: generalization to locations

Experiment 3 is intended to validate generalization ability to appearance variation. Images from YQ21 and YQ-South are collected two routes with nearly 20% overlapping, meaning that N_6 has not witnessed most of the scenes in YQ-South before. Although the same camera is used, it captures more appearance variation. It is found that the network works pretty well on most case, especially on the straight roads(row 1 in Fig. 7). Obstacle like cars, pedestrian and people on bicycles can be segmented out(row 2 in Fig. 7). Thus the proposed method can be generalized to most similar scenes.

There also exist failure cases like difficult corners, shadows and rare road markers(e.g. zebra crossing), showing weakness of neural network in domain adaption(row 3 in Fig. 7). This can be solved by collecting more data with more variation.

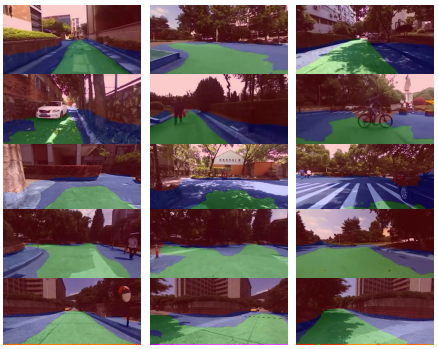


Fig. 7: Some prediction results of experiment 3. Row 1: Straight roads or corners. Row 2: Roads with obstacles like cars, pedestrian and people on bicycle. Row 3: Some failure cases, such as puzzling corners, contrasting shadows and zebra crossing. Row 4: Corner ambiguity. Row 5: The straight way is found out at first(left), while additional way is observed just before entering the corner(middle). The original way is ignore after turning to a new direction(right).

As reported in [6], the classifier may output multiple predicted traversable area at intersection of roads(row 4 in Fig. 7), showing generalization of our method. Sometimes the method may give ambiguous prediction at corners(row 5 in Fig. IV-F). One way is pointed out before entering the corner by the network, while another way is found after driving on the corner. It is because in training data only single traversable area is given as ground truth, leading to ambiguity at corners.

G. Experiment 4: generalization to sensors and locations

This experiment is intended to validate generalization to different sensors. Similar to IV-F, N_6 is used for inference. Image sequence of Shadow Road and Variational Road are used as input, where appearance may varied a lot because of different sensors. Only traversable and obstacle area are considered because no unknown area exists in this dataset.

Results can be viewed in Table IV. Precisions of both datasets are high for both traversable and obstacle area. Recall and IOU of traversable area is not high because traversable area masked by the proposed algorithm is still not complete. Our method shows generalization to different sensors.

TABLE IV: Performance on outer dataset with different sensor

Metrics	Type	Traversable	Obstacle
PRE	Shadow	0.999	0.941
	Variational	0.987	0.880
REC	Shadow	0.701	0.931
	Variational	0.614	0.857
IOU	Shadow	0.701	0.879
	Variational	0.609	0.767

TABLE V: Comparison with LFTD

	Metrics	LFTD	Ours
Shadow	FPR	0.85%	**0.10%**
	FNR	7.87%	**5.93%**
Variational	FPR	2.10%	**1.32%**
	FNR	14.92%	**12.03%**

Table V presents comparison between LFTD[5] and our method. The proposed algorithm gets lower FPR and FNR than LFTD in all appearance cases, demonstrating that our method works better than the self-learning LFTD.

V. Conclusion

In this paper we demonstrated a unified framework to register data across sessions. Leveraging a global consistent map, every frame of data from different location, time and sensor is assigned with a global pose. Within it, we propose a method to fuse sensor data across sessions to improve performance of traversable area segmentation. Our segmentation is based on the state of art pixel-wise semantic segmentation algorithm, namely, deep convolutional neural network, which requires pixel-wise labels. In our method, the labels for training samples are generated automatically, without any human intervention, thus it can create massive data conveniently. Experimental results validate that our framework largely promotes performance of traversable area detection. What's more, our method can generalize to varied appearance of different time, location and sensors. Up to now, the classifier in our method may be confused by corners, such as giving multiple ways or switching between different directions unstably. Thus our future work is trying to add semantic information into the neural network as additional input, which may lead to classifier to the proper direction.

Acknowledgment

This work was supported by the National Nature Science Foundation of China(Grant No.U1609210 and Grand No.61473258).

References

[1] F. Zhang, H. Stähle, C. Chen, C. Buckl, and A. Knoll, "A lane marking extraction approach based on random finite set statistics," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 1143–1148.

[2] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2110–2118.

[3] V. Badrinarayanan, A. Handa, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," *arXiv preprint arXiv:1505.07293*, 2015.

[4] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[5] Y. Wang, Y. Liu, Y. Liao, and R. Xiong, "Scalable learning framework for traversable region detection fusing with appearance and geometrical information," *IEEE Transactions on Intelligent Transportation Systems*, 2017.

[6] D. Barnes, W. Maddern, and I. Posner, "Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy," *arXiv preprint arXiv:1610.01238*, 2016.

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *arXiv preprint arXiv:1606.00915*, 2016.

[8] B. Ma, S. Lakshmanan, and A. O. Hero, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 3, pp. 135–147, 2000.

[9] A. S. Huang, D. Moore, M. Antone, E. Olson, and S. Teller, "Finding multiple lanes in urban road networks with vision and lidar," *Autonomous Robots*, vol. 26, no. 2, pp. 103–122, 2009.

[10] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski, "Self-supervised monocular road detection in desert terrain." in *Robotics: science and systems*, vol. 38. Philadelphia, 2006.

[11] A. Borkar, M. Hayes, and M. T. Smith, "A novel lane detection system with efficient ground truth generation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 365–374, 2012.

[12] H. Zhou, A. M. Wallace, and P. R. Green, "A multistage filtering technique to detect hazards on the ground plane," *Pattern Recognition Letters*, vol. 24, no. 9, pp. 1453–1461, 2003.

[13] H. Hu and M. Brady, "Dynamic global path planning with uncertainty for mobile robots in manufacturing," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 5, pp. 760–767, 1997.

[14] M. Aly, "Real time detection of lane markers in urban streets," in *Intelligent Vehicles Symposium, 2008 IEEE*. IEEE, 2008, pp. 7–12.

[15] A. S. Huang and S. Teller, "Probabilistic lane estimation for autonomous driving using basis curves," *Autonomous Robots*, vol. 31, no. 2-3, pp. 269–283, 2011.

[16] H. Sawano and M. Okada, "A road extraction method by an active contour model with inertia and differential features," *IEICE transactions on information and systems*, vol. 89, no. 7, pp. 2257–2267, 2006.

[17] Z. Kim, "Robust lane detection and tracking in challenging scenarios," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 1, pp. 16–26, 2008.

[18] J. Shotton, M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *Computer vision and pattern recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[19] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," *Computer Vision–ECCV 2008*, pp. 44–57, 2008.

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

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

[22] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3234–3243.

[23] M. Siam, S. Elkerdawy, M. Jagersand, and S. Yogamani, "Deep Semantic Segmentation for Automated Driving: Taxonomy, Roadmap and Challenges," *ArXiv e-prints*, Jul. 2017.

[24] P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.

[25] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing icp variants on real-world data sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, 2013.