

Driving Lane Detection based on Recognition of Road Boundary Situation

Hiroyuki Komori

Hirosaki University

3 Bunkyo-cho, Hirosaki, Japan 036-8561

Kazunori Onoguchi

Hirosaki University

3 Bunkyo-cho, Hirosaki, Japan 036-8561

Abstract—This paper presents the method that recognizes the road boundary situation from a single image and detects a driving lane based on the recognition result. Driving lane detection is important for lateral motion control of the vehicle and it usually realized based on lane mark detection. However, there are some roads where lane marks such as white lines are not drawn. Also, when the road is covered with snow, lane marks cannot be seen. In these cases, it's necessary to detect the boundary line between the roadside object and the road surfaces. Since traffic lanes are divided by various roadside objects, such as curbs, grass, walls and so on, it's difficult to detect all kinds of road boundary including lane marks by a single algorithm. Therefore, we propose the method which changes the driving lane detection method according to the road boundary situation. At first, the situation of the road boundary is identified as some classes, such as white line, curb, grass and so on, by the Convolutional Neural Network (CNN). Then, based on this result, the lane mark or the boundary between the road surface and the roadside object is detected as the lane boundary. When a clear lane mark is drawn on a road, this situation is identified as a class of “White line” and a lane mark is detected as a lane boundary. On the other hand, when a lane mark is not present, this situation is identified as the other class and the boundary of the roadside object corresponding to the identified class is detected as the lane boundary. Experimental results using the KITTI dataset and our own dataset show the effectiveness of the proposed method. In addition, the result of the proposed method is compared with the boundary of the road area extracted by some semantic segmentation method.

Index Terms—Inverse perspective mapping (IPM), CNN, lane boundary, lane mark

I. INTRODUCTION

For the past several decades, many vision-based lane detection method has been proposed for advanced driver assistance system or autonomous driving system [1] [2]. Most of these methods detect lane marks such as white lines from an image and estimate a traffic lane [3] [4] [5] [6]. However, these methods cannot be applied to roads where lane marks are not drawn or they are too blurred to see. In these cases, it's necessary to detect the boundary line between the road surface and the roadside object such as curbs, grass and so on, instead of lane marks. Especially, when the road is covered with snow, the boundary with the snow sidewall (Figs. 1b, 1c, 1d) needs to be detected since other roadside objects are hidden by snow.

Recently, a lot of road detection method based on the semantic segmentation using deep neural networks have been put forward. Alvarez et al. [7] [8] proposed the method which detects road area by Convolutional Neural Networks.

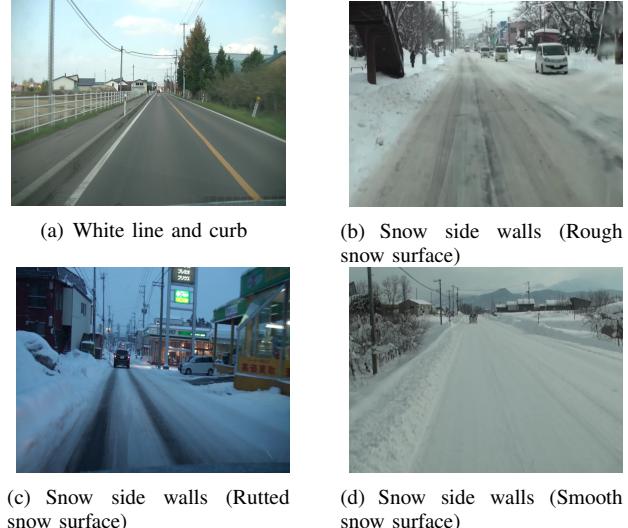


Fig. 1: Road boundary situation

The StixelNet whose input is Stixel instead of images was proposed by [9]. Convolutional Patch Network whose input is a single image patch extracted around a pixel to be labelled was proposed for road area detection [10]. Road detection method which combines deep deconvolutional and convolutional neural networks was proposed by [11] and [12]. Laddha et al. [13] presented a boosting based method for semantic segmentation of road scenes. Costea et al. [14] proposed road detection method which reduces human labeling effort by a map-supervised approach. Fast algorithm for road segmentation using CNN and gated recurrent units (GRU) was presented in [15]. Liu et al. [16] proposed a RPP model for monocular vision-based road detection based on the combination of fully convolutional network, residual learning, and pyramid pooling. Even for small datasets, some techniques such as data augmentation and so on enable high-performance road detection. Wang et al. [17] proposed s-FCN-loc, which is able to consider RGB-channel images, semantic contours and location priors simultaneously to segment the road region elaborately. Learning of these models converges more quickly than the conventional one. These methods show considerably good results in various road scenes but to our best knowledge, results applied to the snow road are not shown.

This paper proposes the method that recognizes the situation of the road boundary from an image taken by a monocular

TABLE I: road boundary classification class

Class number	Road boundary situation
0	white line
1	blurred white line
2	curb - white line
3	curb - blurred white line
4	curb
5	curb - grass
6	grass - white line
7	grass - blurred white line
8	grass
9	grass - curb - white line
10	grass - curb
11	grass - gutter - white line
12	gutter - white line
13	traffic rail - white line
14	traffic rail - curb - white line
15	traffic rail - grass
16	sidewall - white line
17	sidewall
18	sidewall - grass - white line
19	sidewall - grass
20	parked vehicle
21	snow side wall - rough snow surface
22	snow side wall - rutted snow surface
23	snow side wall - smooth snow surface
24	snow side wall - white line

camera and detects a driving lane based on this recognition result. The situation of the road boundary is identified into 25 classes as shown TABLE I. The left column of Table I is the class number and the right column shows roadside objects existing around the road boundary. Since multiple kinds of roadside objects may exist around the road boundary as shown in Fig. 1a, some classes contain multiple roadside objects. The right column of Table I describes the kind of roadside object from turned left to right. For examples, when grass, a curb and a white line appear from turned left to right around the road boundary, it's represented as grass - curb - white line. VGG16 [18] and GoogLeNet [19] which are the Convolutional Neural Network (CNN) architectures are employed to this identification. When the situation of the road boundary is identified as the class in which a white line exists closest to the vehicle, the proposed method detects the position of the white line as the lane boundary. When the recognition model identifies the situation of the road boundary as the other class, the boundary line of the road is detected according to the identified class.

This paper is organized as follows. Section II shows the outline of the proposed method. Section III explains how to create the Inverse Perspective Mapping (IPM) image from an input image. Section IV describes the detail for recognizing the road boundary situation. Section V describes the detail of driving lane detection. Section VI discusses experimental results. Conclusions are presented in Sect VII.

II. THE OUTLINE OF THE PROPOSED METHOD

Figure 2 shows in the procedure of the proposed method. At first, an input image is projected to a road plane and the inverse perspective mapping (IPM) image is created (Fig. 3). Next, the left side of the IPM image is cut out as a processing region. A red rectangle in Fig. 4 shows the example of this

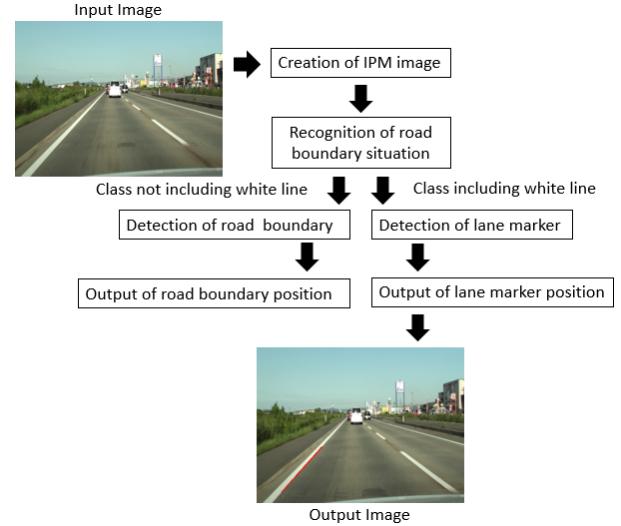
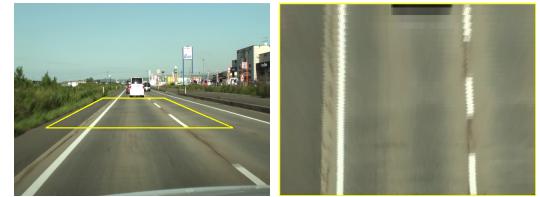


Fig. 2: Outline of the proposed method



(a) Input image (b) IPM image
Fig. 3: Inverse perspective mapping



Fig. 4: Region for recognizing road boundary situation

processing region. This processing region is input to the CNN which identifies the road boundary situation into 25 classes. A lane mark or a road boundary is detected according to the result of this identification. According to the recognized class, the roadside object closest to the vehicle, that is, the rightmost roadside object in Japan is detected. For examples, when the recognition result is class 2 (curb - white line), a white line is detected by the lane mark detection method. On the other hands, when the recognition result is class 10 (grass - curb), the boundary between the curb and the road surface is detected by the road boundary detection method.

III. CREATION OF IPM IMAGE

Inverse perspective mapping (IPM) image, which overlooks a road surface, is created by inverse perspective transform. A point (x, y) in the image coordinate system and a point (u, v) in the IPM image coordinate system satisfy

$$(u, v) = \left(RX \frac{x - V_{px}}{y - V_{py}}, \frac{RY^2}{y - V_{py}} - \frac{RY^2}{y_{lim}} \right), \quad (1)$$

where (V_{px}, V_{py}) is the position of a vanishing point in the image coordinate system, RX and RY are compression or expansion rates for direction x and y , and y_{lim} is the lower limit of y-coordinate value in the image [22]. Figure 3b shows the IPM image created from Fig. 3a. In experiments, we used rigid values as (V_{px}, V_{py}) , RX , RY and y_{lim} . In experiments, the IPM image whose size is 640×480 is created from an input image whose size is 640×480 .

IV. RECOGNITION OF ROAD BOUNDARY SITUATION

A. Dataset

The situation of the road boundary in front of a vehicle is recognized from the IPM image. The road boundary is located on the left side of the IPM image since vehicles drive on the left side of the road in Japan. Therefore, the region whose size is 256×256 is cut out of the left side of the IPM image. The cutout position is fixed in every image. This region is input to Network for recognizing the situation of the road boundary. We used two datasets for experiments. One is our own dataset created from images taken from the single camera mounted in a vehicle running in urban and suburban areas. The other is the KITTI dataset [23]. Since vehicles drive on the left side in Japan, mirror images are created and resized to 640×480 pixels in the KITTI dataset. Main roadside objects are classified into 7 types of white lines, curbs, grass, traffic rails, sidewalls, parked vehicles, snow sidewalls. The proposed method classifies the situation of the road boundary into 25 classes consisting of a single or a combination of these seven types of roadside objects as shown in TABLE I. The proposed method classifies white lines into two classes "white line" and "blurred white line" depending on the degree of blurring. It's difficult to detect a blurred white line by parameters for detecting a clear white line. For this reason, the class of the white line is divided into 2 classes and the white line is detected by parameters suitable for each class. Our own dataset has 99,500 training images and 18,457 test images. The KITTI dataset has 1,161 training images and 609 test images. Our dataset includes dry road images taken from sunny and cloudy daytime, wet road images taken at rainy daytime and various kinds of snowy road images. Figure 5 shows examples of images contained in each dataset. In order to improve the accuracy of the CNN classifier, data augmentation was performed using ImageMagick [21] with the following processing. As a result of data augmentation, our own dataset has 282,000 training images and 70,499 of the training images are used for validation. Also, KITTI dataset has 7,118 training images and 1,779 of the training images are used for validation.

Data Augmentation

Addition of noise

Gaussian noise, Poisson noise and multiplicative noise are in addition to original images respectively.

Smoothing

Median filter and the Gaussian filter are applied for

smoothing. A linearly fading blur is applied in one direction.

Conversion of brightness and saturation

The brightness and saturation of the original image were changed respectively. The brightness was changed between 0.8 and 1.5 times of the original image and the saturation was changed between 0.5 and 1.5 times of the original image.

Contrast adjustment

The contrast of the image is raised or decreased.



Fig. 5: Example of road boundary image included in datasets

B. Model and train

We used VGG16 for our dataset and GoogLeNet for KITTI dataset since learning of VGG16 did not converge well in the KITTI dataset where the number of images was poor. We used same structures as described in original papers [18] [19]. The intensity of each pixel is normalized by subtracting the average value from the intensity of each channel. Batch sizes of VGG16 for training and validation are 64 and 32. The epoch attribute and the first learning rate are set to 25 and 0.01. Learning rate is set to 0.001 when one third of learning is completed and 0.0001 when two-thirds of learning is completed. Initial weight of all layers is set from the Xavier distribution [20] with a standard deviation of 0.01 as opposed to VGG16 [18] which used a Gaussian distribution with 0.01 standard deviation. On the other hand, batch sizes of GoogLeNet for training and validation are 32 and 16. The epoch attributes and the initial learning rate is set to 45 and 0.01. Like VGG16, the Xavier distribution is also applied to the initial weight of each layer. The learning rate was changed in the same way as VGG16. Stochastic gradient descent (SGD) was utilized for both networks to optimize learning.

V. DETECTION OF DRIVING LANE

According to the recognition result of road boundary situation, lane marks and the boundary line between the road

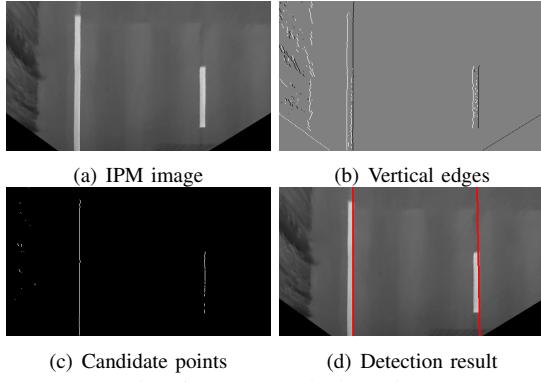


Fig. 6: Lane mark detection

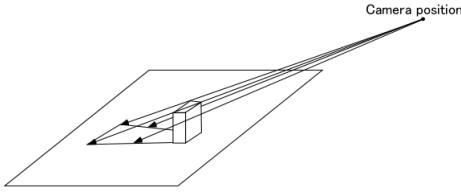


Fig. 7: Shape of projected area

surface and the roadside object are detected.

A. Detection of lane mark

A lane mark is detected by the method described in [22]. This method detects a lane mark in the IPM image shown in Fig. 6a. At first, vertical edges are detected in the IPM image by a vertical component of the Sobel operator. Then, edge points whose strength are the local maximum or the local minimum are extracted by scanning each horizontal line. Finally, in every scanning lines, a pair of an edge point whose left is the local maximum and whose right is the local minimum is extracted because the left edge of a lane mark has positive strength and the right edge of a lane mark has negative strength. Negative edges are detected as candidate points of a lane mark. Figure 6b shows positive edges (white point) and negative edges (black point). Figure 6c shows candidate points of lane marks. Lane marks are approximated as straight lines by Hough transform [24]. Figure 6d shows the final result of lane mark detection. Two kinds of parameter sets are prepared for detecting a clear white line and a blurred white line. When the class of the road boundary is class 0, 2, 6, 9, 11, 12, 13, 14, 16, 18 or 24, parameter set for a clear white line is used. When the class is 1, 3 or 7, parameter set for a blurred white line is used. These parameters are determined by experiments conducted in advance.

B. Detection of road boundary

A road boundary is detected in the IPM image by the method referred to in [25]. In the IPM image, the patterns existing on the road surface are projected in the shape viewed from the right overhead. On the other hand, as shown in Fig. 7, road side objects or obstacles whose height is different from the road surface are projected to the shape falling backward from the location where the obstacles touch the road surface.

Therefore, this method detects the boundary between dense areas of slant edges and sparse areas of slant edges as the road boundary since a lot of slant edges appear around a road side in the IPM image as shown in Fig. 10b. First, the IPM edge image that emphasizes slant edges on road side objects is created since there are some roadside objects whose texture are weak. The IPM edge image is created by the following procedure.

- 1) Vertical edges in an input image are transformed into the IPM image.
- 2) Slant edges are detected in the IPM image converted from an input image.
- 3) The AND image of these two edge images is created as the IPM edge image.

Vertical edges on a roadside object are transformed into slant edges in the IPM image. On the other hand, there do not be too many vertical edges on the road surface converted into slant edges. Therefore, in the IPM edge image, slant edges around a road side remains but slant edges on a road surface are suppressed. Next, the degree of road boundary whose value increases at the boundary with the dense areas of slant edges is calculated in each pixel of the IPM edge image. A parallelogram shaped mask is set along a straight line connecting the camera position C and each pixel P, as shown in Fig. 8. An enlarged view of a parallelogram shaped mask is illustrated in Fig. 9. Let the length of the left and right sides of a mask be H, the width between the left and right sides be W, the region on the left side of the point P be R_W and the region on the right side of the point P be R_B . The road boundary is located on the left side of the IPM image since vehicles drive on the left side of the road in Japan. Since slant edges usually appear densely on road side objects in the IPM edge image, the number of edges in R_W is large and the number of edges in R_B is small if the pixel P is around the road boundary. For this reason, at each pixel P on the left half of the IPM edge image, the degree of road boundary BD is calculated by

$$BD = \frac{(N_W + (S_B - N_B))}{S_W + S_B}, \quad (2)$$

where N_W is the number of edges in R_W , N_B is the number of edges in R_B , S_W is the total number of pixels of R_W and S_B is the total number of pixels of R_B . The degree of road boundary BD increases around the road boundary since it shows large value when N_W is large and N_B is small.

Figure 10c shows in the result of the degree of road boundary BD calculated from the IPM edge image shown in Fig. 10b. Finally, the road boundary is detected and tracked by the Snakes whose image force is the degree of road boundary. In the BD image as shown in Fig. 10c, the intensity is accumulated vertically and the vertical line passing through the peak is used as the initial position of the Snakes. The number of control points is 61 and the number of updates is 10 per frame. Figure 10d shows the convergence result of Snakes. Since textures on the snow sidewall are weaker

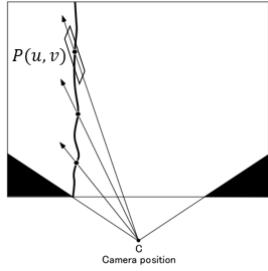


Fig. 8: Projection of vertical edges

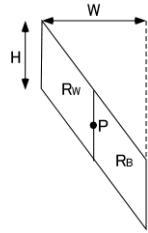


Fig. 9: Parallelogram shaped mask

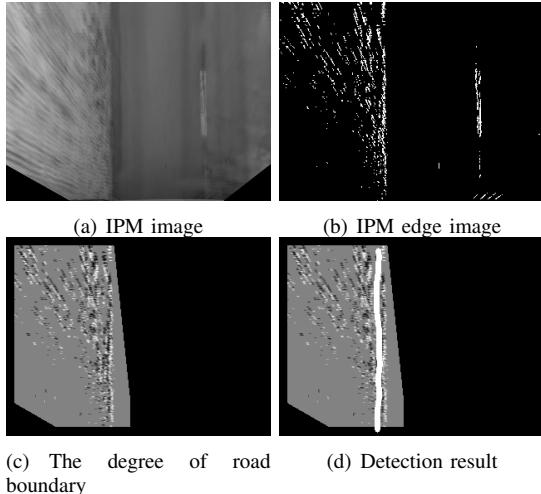


Fig. 10: Road boundary detection

than other road side objects, we prepare four different IPM transformation parameters for class 21, class 22, class 23 and other classes. The appearance of slanting edges in the IPM edge image is different between artificial objects and natural objects. Therefore, two types of parallelogram shaped masks whose size are different are used for artificial objects like a curb, a sidewall, a parked vehicle and a traffic rail and natural objects like grass and a snow side wall. The mask size for the artificial object is set smaller than the mask size for the natural object. Each mask size was determined by experiments conducted in advance. In experiments, W and H are set to 20 and 3 for artificial objects and 20 and 7 for natural objects respectively.

VI. EXPERIMENTS

We conducted experiments to recognize the situation of the road boundary and to detect the left boundary of the

driving lane based on the recognition result by an in-vehicle monocular camera. The image size is 640×480 pixels. In experiments, it's assumed that a vehicle drives in the lane closest to the shoulder. The proposed method was evaluated under various situations of the road boundary. The tests were conducted on a computer equipped with Intel Core i7-8700K, 32GB RAM, and an NVIDIA GeForce GTX1080 Ti.

A. Evaluation of road boundary situation recognition

Images taken at fine daytime are used for evaluation. The image whose size is 256×256 pixels was input to the CNN was trained to recognize the situation of the road boundary. First, the recognition performance of the road boundary situation was evaluated using the KITTI dataset which includes six kinds of classes. Figure 11a shows the recognition rate. The vertical axis shows the recognition rate and the horizontal axis shows the class number. The recognition rate of all six classes exceeds 90%. The lowest recognition rate was 93.7% of class 10. Since the KITTI dataset does not contain other 19 classes, especially snowy road scenes (class 21 - 24), the recognition rate including these classes was evaluated by our own dataset. The recognition rate of all classes also exceeds 90% as shown in Fig. 11b. The lowest recognition rate was 92.8% of class 21. Slight error has occurred in some scenes. For examples, since Figure 12a contains a lid of a side gutter and white line, it should be identified as Class 0 (white line). However, this image was misrecognized as class 4 (curb). It is the cause of this misrecognition that the projection images of a lid of a side gutter and a white line are similar to the projection image of a curb because the distance between them is close. Fig. 12b is the example of false recognition in the snowy road. Although the correct class of this scene is class 23 (snow side wall - smooth snow surface), it is misrecognized as class 21 (snow side wall - rough snow surface). To avoid this false recognition, it's necessary to acquire more snow road image and to re-learn the network since appearance of the snow surface is similar.

B. Driving lane detection based on road boundary situation

We conducted experiments to confirm that the proposed driving lane detection method is effective for change in road boundary. Figure 13 shows in the experimental result in the scene where the situation of the road boundary changes in the order of a white line, a blurred white line and the grass. The correct boundary of the driving lane is detected since the proposed method switches detection algorithms and parameters according to the situation of the road boundary. TABLE II shows the recognition result of the road boundary situation in this scene. The class of the situation of the road boundary is identified correctly in each frame. Since the type of road boundary does not change from frame to frame, the situation of the road boundary was recognized every 15 frames. Figure 14 demonstrates that the proposed method is effective for detecting the blurred white line. In this scene, it's difficult to detect a white line since it's too blurred. Figure 14a shows the result without recognizing the road boundary situation. Instead of the white line, the boundary

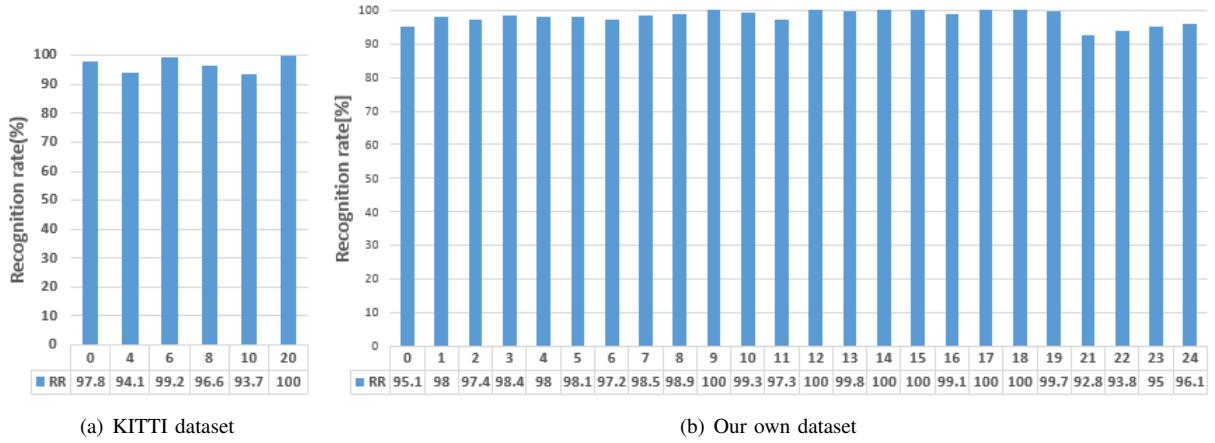
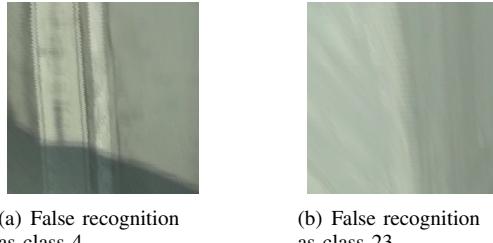


Fig. 11: Recognition rate for each class



(a) False recognition as class 4
(b) False recognition as class 23

Fig. 12: Examples of false recognition



Fig. 13: Detection results of driving lane

with the curb is falsely detected as the left boundary of the driving lane. On the other hand, the proposed method can detect the blurred white line correctly since it switches detection parameter according to the situation of the road boundary as shown in Fig. 14b. Figure 15 shows the other result. Although grass on the curb is erroneously detected as a driving lane instead of the curb (Fig. 15a). On the other hand, the proposed method can detect the curb correctly as a driving lane since the size of the parallelogram shaped mask for the artificial roadside object is applied according to the situation of the road boundary, as shown in Fig. 15b. Figure 19 shows some detection results of a curb, grass, parked vehicles and a snow side wall. Recognition of road boundary situation is performed on the GPU. On the other hand, lane mark and road boundary detection are performed on the CPU. Although the processing time of recognition of road boundary situation is 60ms for VGG 16 and 80ms for GoogLeNet presently, real time processing will be possible since the performance of GPU is improving day by day. Lane mark detection can be processed in real time but detection of road boundary has not reached real time. However, real time processing of road

TABLE II: The recognition result of road boundary situation is Fig. 13

Frame number	1	16	31	46	61	76	91	106
Identification class	6	6	6	7	7	7	8	8
Truth class	6	6	6	7	7	7	8	8



(a) Result without recognition (b) Result of the proposed of road boundary situation method

Fig. 14: Driving lane detection for blurred white line



(a) Result without recognition (b) Result of the proposed of road boundary situation method

Fig. 15: Driving lane detection for curb

boundary detection will be possible by parallelization and software optimization. We are currently working on it.

C. Quantitative evaluation

The performance of the proposed method was evaluated quantitatively in our dataset and KITTI dataset. In the KITTI dataset, images containing road side objects such as curbs, grass, vehicles and so on were used for evaluation. 127 frames of the curb and 56 frames of the other road side object including parked vehicles were evaluated. Only the left side of the driving lane in the mirror image was compared with the ground truth tracing the left boundary of the true road area shown in the KITTI dataset. Since there is no image covered

TABLE III: Detection accuracy (KITTI dataset)

	# of frames	DA
Curb	127	0.095
Other	56	0.071
Total	183	0.088

TABLE IV: Detection accuracy (our dataset)

	# of frames	DA
Curb	100	0.046
Grass	161	0.118
Snow side wall (rough surface)	90	0.088
Snow side wall (rutted surface)	164	0.096
Snow side wall (smooth surface)	90	0.073
Total	605	0.084

TABLE V: Average pixel error in the horizontal direction

Test images	Our method	ZongNet	UNV	VGGFCN-6D
um_45	11.17 [pixel]	11.94 [pixel]	19.04 [pixel]	11.70 [pixel]
um_77	1.52 [pixel]	7.59 [pixel]	9.04 [pixel]	5.31 [pixel]
umm_25	23.67 [pixel]	17.15 [pixel]	28.15 [pixel]	31.28 [pixel]
umm_40	2.86 [pixel]	5.99 [pixel]	4.50 [pixel]	3.97 [pixel]
umm_66	6.27 [pixel]	7.40 [pixel]	4.86 [pixel]	6.72 [pixel]
uu_27	22.67 [pixel]	5.84 [pixel]	18.73 [pixel]	11.97 [pixel]
uu_82	12.66 [pixel]	12.54 [pixel]	21.43 [pixel]	10.84 [pixel]

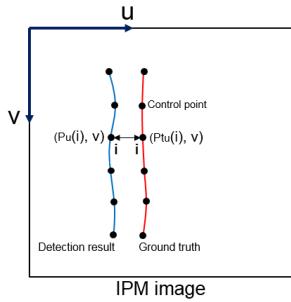


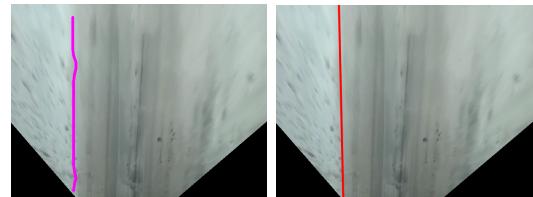
Fig. 16: DA measurement method

with snow in the KITTI dataset, we evaluated road boundary detection in the snowy road using our data set. Our dataset contains 100 frames of the curb, 161 frames of the road side grass, 90 frames of the snow side wall (rough surface), 164 frames of the snow side wall (rutted surface) and 90 frames of the snow side wall (smooth surface). Figure 17 shows detection results and ground truth in our dataset. The ground truth was obtained by tracing the road boundary manually in the IPM image. The detection accuracy DA given by the equation (3) is estimated in the IPM image (Fig. 16). Therefore, ground truth of the KITTI dataset is projected onto the IPM image.

$$DA = \frac{\sum_{i=0}^{n-1} \frac{|p_u(i) - pt_{u,x}(i)|}{LaneWidth}}{n}, \quad (3)$$

where $p_u(i)$ is the u coordinate value of the control point calculated by the snakes, $pt_u(i)$ is the u coordinate value of the ground truth whose v coordinate value is same as $p_u(i)$, n is the number of the control points in the Snakes and $LaneWidth$ is the width of the driving lane in the IPM image.

Table III indicates the detection accuracy DA in the KITTI dataset. The average DA of all scenes is 0.088. This result shows that the error of $0.27m$ occurs when the width of the driving lane is about $3m$. However, this result shows that a vehicle has some space to run in the driving lane if the vehicle width is less than $2.5m$. Therefore, the proposed method can

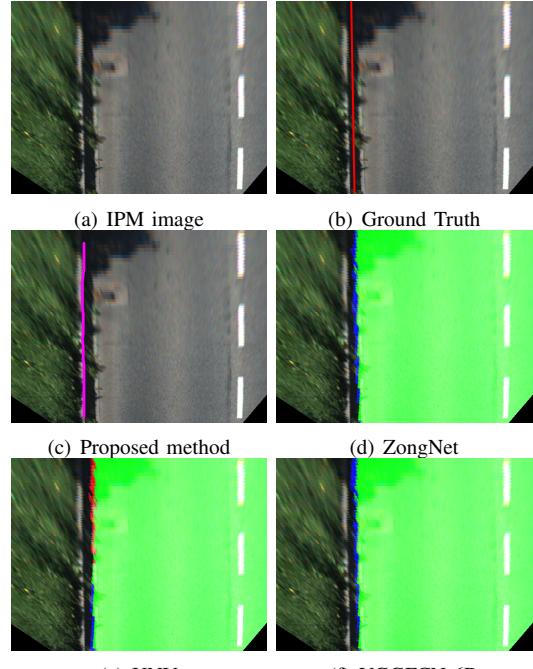


(a) Sherbet-like snow surface (b) Ground Truth of (a)

(c) Snow side wall (wet surface) (d) Ground Truth of (c)

(e) Smooth snow surface (f) Ground Truth of (e)

Fig. 17: Result of road boundary detection in the snowy road



(a) IPM image (b) Ground Truth

(c) Proposed method (d) ZongNet

(e) UNV (f) VGGFCN-6D

Fig. 18: Comparison of detection results

also be applied to ordinary vehicles. Table IV shows the detection accuracy DA in the our dataset. The average DA of all scenes is 0.084. This result shows that the proposed method is effective for road boundary detection in the snowy roads. We compared the detection accuracy between the proposed method and some semantic segmentation methods published in the KITTI Vision Benchmark Suite Road/Lane Detection Evaluation 2013. Figure 18 shows results of ZoneNet, UNV,

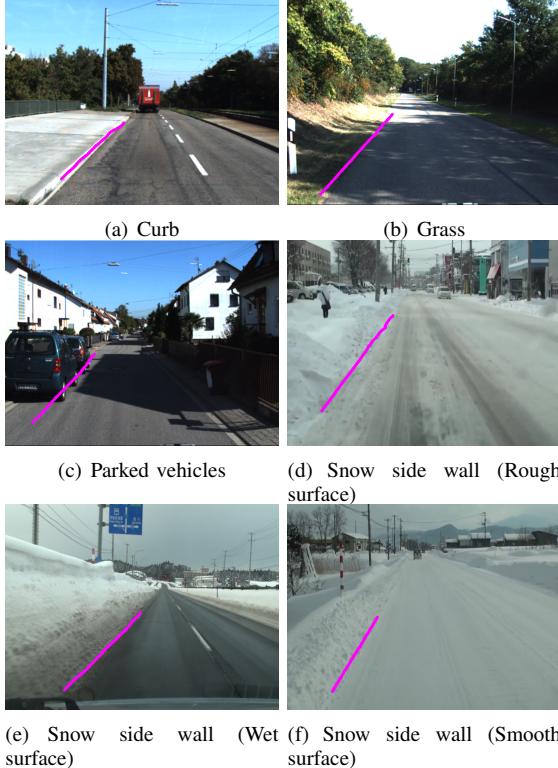


Fig. 19: Detection result of various roadside objects

VGGFCN-6D and the proposed method. The ground truth was obtained by tracing the boundary of the road area manually in the IPM image. Table V displays the average pixel error in the horizontal direction between the detection result and the ground truth in the IPM image. It can be observed that the performance of our approach is equal to or better than other approaches using semantic segmentation. To our best knowledge, these methods using semantic segmentation have not been evaluated in snow road scenes. The proposed method shows good performance for snowy road as shown in Table IV.

VII. CONCLUSION

This paper proposed the method to recognize the situation of the road boundary from an image and to detect a driving lane based on the recognition result. The situation of road boundary is classified into 25 classes. The recognition rate of the proposed method was over 90.0% in all classes. Since the proposed method can select suitable lane detection algorithms and parameters according to the situation of the road boundary, the driving lane divided by various kinds of roadside objects can be detected stably. In the future, we will improve the performance of the recognition model by adding various road images taken under various weather conditions. In addition, our model created at VGG16 is a huge network of 15.5GB. Therefore, we plan to apply SqueezeNet to VGG16 [26] [27] to reduce the parameters. We think that it would be possible to greatly reduce the network size. We aim to create a lightweight network while maintaining a high recognition rate.

REFERENCES

- [1] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, System, and Evaluation", *IEEE Trans. on Intelligent Transportation Systems*, Vol.7, No.1, pp.20-37, 2006.
- [2] B. Hillel et al., "Recent progress in road and lane detection: A survey", *Machine Vision and Applications*, Vol.25, Issue 3, pp.727- 745, 2014.
- [3] J.Douret et al., "A Reliable and Robust Lane Detection System based on the Parallel Use of Three Algorithms for Driving Safety Assistance", *Proceedings of IAPR Conference on Machine Vision Applications*, pp.398-401, 2005.
- [4] M. Meuter et al., "A Novel Approach to Lane Detection and Tracking", *Proceedings of International IEEE Conference on Intelligent Transportation Systems*, pp.582-587, 2009.
- [5] S. Hold et al., "ELA - an Exit Lane Assistant for Adaptive Cruise Control and Navigation Systems", *Proceedings of International IEEE Conference on Intelligent Transportation Systems*, pp.629-634, 2010. 10
- [6] Z. W. Kim, "Robust Lane Detection and Tracking in Challenging Scenarios", *IEEE Trans. on Intelligent Transportation Systems*, Vol.9, No.1, pp.16-26, 2008.
- [7] J. M. Alvarez et al., 2012, "Semantic road segmentation via multi-scale ensembles of learned features", In *Proceedings of ECCV2012*, pages 586-595.
- [8] J. M. Alvarez et al., 2012, "Road scene segmentation from a single image", In *Proceedings of ECCV2012*, pages 376-389.
- [9] D. Levi et al., 2015, "Stixelnet: A deep convolutional network for obstacle detection and road segmentation", In *Proceedings of BMVC2015*.
- [10] C. Brust et al., 2015, "Convolutional patch networks with spatial prior for road detection and urban scene understanding", In *Proceedings of VISAPP2015*.
- [11] R. Mohan, 2014. "Deep deconvolutional networks for scene parsing", In *ArXiv.org*.
- [12] G. L. Oliveira et al., 2016. "Efficient deep models for monocular road segmentation", In *Proceedings of IROS2016*, pages 586-595.
- [13] A. Laddha et al., 2016. "Map-supervised road detection", In *Proceedings of IV2016*.
- [14] D. Costea and S. Nedevschi (2017). "Traffic scene segmentation based on boosting over multimodal low", intermediate and high order multi-range channel features, In *Proceedings of IV2017*.
- [15] Y. Lyu and X. Huang (2018), "Road Segmentation Using CNN with GRU", In *ArXiv.org*.
- [16] X. Liu and Z. Deng, "Segmentation of Drivable Road Using Deep Fully Convolutional Residual Network with Pyramid Pooling", *Cognitive Computation*, Issue 2/2018.
- [17] Q. Wang et al., "Embedding Structured Contour and Location Prior in Siamesed Fully Convolutional Networks for Road Detection", 2018 *IEEE Transactions on Intelligent Transportation Systems*, pp.230-241, 2018.
- [18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", In *ICLR*, 2015.
- [19] C. Szegedy et al., "Going Deeper with Convolutions", *Computer Vision and Pattern Recognition (CVPR)*, 2015 *IEEE Conference on*, pages 1-9, 2015.
- [20] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", In *International Conference on Artificial Intelligence and Statistics*, 2010
- [21] J. Cristy, *ImageMagick*, *ImageMagick Studio LLC*.
- [22] T. Yasuda and K. Onoguchi, "Lane Estimation based on Lane Marking Recognition," *Proceedings of 19th ITS World Congress Vienna*, 2012.
- [23] A. Geiger et al., "Are we ready for autonomous driving? the kitti vision benchmark suite". In *CVPR*, 2012.
- [24] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of ACM*, vol.15, pp.11-15, 1972.
- [25] K. Goro and K. Onoguchi, "Road Boundary Detection Using In-vehicle Monocular Camera", *7th International Conference on Pattern Recognition Applications and Methods, ICPRAM 2018*
- [26] Forrest N. Iandola et al., "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5MB model size.", *arXiv:1602.07360*, 2016.
- [27] H. Qassim et al., 2017, "Residual Squeeze VGG16", In *ArXiv.org*.