Multi-cue Road Boundary Detection using Stereo Vision

Li Wang, Tao Wu*, Zhipeng Xiao, Liang Xiao, Dawei Zhao, Jiarong Han
College of Mechatronic Engineering and Automation
National University of Defense Technology
Changsha, Hunan, P.R.China
Email: wangli2014@nudt.edu.cn; wt.cs@163.com

***Abstract*—In this paper, we propose a multi-cue fusion approach to detect the road boundary using stereo vision, which fits road boundary with a few edge points. Firstly boundary areas are determined in accordance with the normal vector information. Based on the cues of normal vector, height and color in the boundary area, three Bayes models are established respectively. Then the Naive Bayes framework could provide the confidence level of each point in the boundary area, which fuses three kinds of cues. In each boundary area, the point with the highest confidence level would be output. Finally, the support vector regression (SVR) method fits curb curves according to the correct edge points. Extensive experiments tested on KITTI-Road benchmark demonstrate that our method reaches the state-of-the-art.**

***Keyworks*—Road boundary detection; Normal vector; Naive Bayes; SVR;**

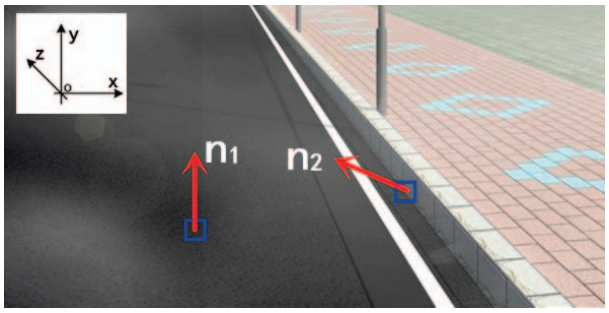


Fig. 1. Normal vectors on road 3D model. In this model, the characteristics are obviously defferent between the road boundary area and the road area, including the normal vector, height and color.

I. Introduction

Recently, with Chang'an driverless cars completing long-distance road test [1], intelligent vehicle technology has become a hot spot, which was widely discussed in China. Road boundary detection is one of the most essential task for intelligent vehicle, which plays a fundamental role in autonomous cars navigation and driver assistance system.

Based on vision sensors, road detection approaches can be divided into different types, such as monocular-image-based method, LIDAR-based method and stereo-based method. Monocular image based detection is mostly used, which is inexpensive in prices and rich in scene information. Using pixel gradient of gray image to detect curb is the earliest method with the typical operator, such as Sobel operator, etc. Later, the cues of color and texture are widely used in road detection. In the past few years, most approaches are based on machine learning. Structured random forests and multi-channel characteristics are used for curb detection with good results, such as [2]. [3] detected road and understood urban scene through deep convolutional neural networks, which learns to classify different image patches by pixel-wise labeling. However, road detection based on monocular image is much more challenging due to the drastic change of road scenes, illumination, and the clutter of background.

High definition LIDARs are widely used in self-driving car, owing to their accurate 3D point cloud, which could construct 3D structure of surrounding scene. In contrast to monocular image, LIDAR shows adaptable and robust well on condition that road scenes have weak texture or drastic illumination change. Using height difference, gradient and gradient direction of point cloud, Zhao [4] got smooth road edge results. In [5] and [6], they proposed a mean to detect curb from the top-view LIDAR data. Besides, using least trimmed squares (LTS) method, the literature [7] fitted the boundary points that were detected by LIDAR data. However, this kind of approaches only uses 3D information to classify the road area and non-road area while the color and texture information are lacked. And it is not robust under the condition of a little or no differences between these two areas in height.

Compared to monocular-image-based and LIDAR-based methods, stereo may be a better choice for this task, which can use not only color, texture and feature points of images but also 3D information. With digital elevation map (DEM), both [8] and [9] detected curb by the height gradient of each gird and fitted road. In [10], Jan Siegemund presented a method that classified 3D point cloud into different parts of the model (i.e. road and sidewalk) using temporally integrated Conditional Random Field (CRF).

Based on detection ways, road detection methods can be mainly divided into two types. One method detects road by the classifications of each pixel. This kind of method clusters the road class pixels and outputs the road area, such as [3] and [10]. However, the detection results would be irregular, and many obvious noise cant be filtered out. The other kind of method detects road through fitting boundary pixels, which could get regular road results, such as [2]. However, the fitting

978-1-5090-2933-4/16/$31.00 ©2016 IEEE

algorithm is essential for the detection result. Because the fitting algorithm determines whether the pixels of result are accurately classified.

In this paper, we present a novel approach for road boundary detection based on stereo vision that is fused with multiple cues. Based on disparity maps, the normal vector of each sampling point is calculated, which obviously responds to road boundary, as Fig.1 shows. And then boundary areas are determined in accordance with the normal vector. According to the information of normal vector, height and color in boundary areas, three Bayes models are established respectively. A Naive Bayes model outputs correct points, while the error sampling points that belong to road class with small probability are deleted. Our algorithm is tested on the authoritative KITTI-Road dataset [11] and the results show that the proposed approach reaches the state-of-the-art.

The rest of this paper is organized as follows. The framework of proposed algorithm is interpreted in section II. The method of normal vector calculation is described in details in section III, and the detail of road boundary is introduced in the Section IV. Then experimental results and comparisons are shown in section V. At last, the conclusion and future work are given in section VI.

II. Algorithm Framework

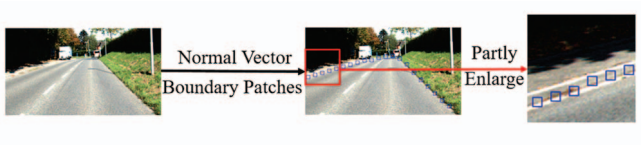
In this paper, the proposed method just uses a few road boundary points of an image to fit the road boundary, which can be divided into three steps: searching road boundary areas, determining road boundary pixels and fitting road boundary.

Searching road boundary areas is the preliminary work. The disparity maps are calculated firstly through left and right images. And then the normal vectors are computed. According to variation of the angle of normal vectors, the road boundary areas that are also called boundary-patches can be searched, as Fig.2 (a) shows.

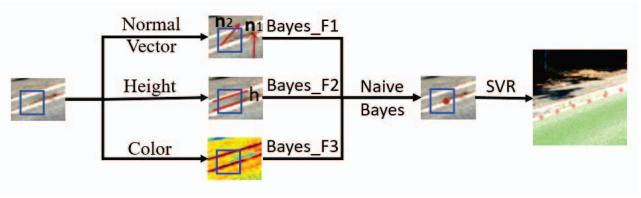
Determining road boundary pixels takes each pixel in the boundary-patch as processing unit. Based on the characteristics of normal vector, height and color, three Bayes models are established respectively. Fused these three cues, the Naive Bayes model selects an optimal pixel as the road boundary point in each boundary-patch. Finally, SVR method fits road boundary by the sparsely located boundary points, as Fig.2 (b) shows.

III. Normal Vector

The normal vector is usually applied in the field of space analytic geometry, which change markedly at the road curb than height information (e.g. DEM). As a point on the surface, the normal vector is a special vector whose direction is perpendicular to the local surface. Using the sharp directional characteristic, the method can distinguish whether the points are on a same plane to detect road boundary, as Fig.1 shows. Then, two means of solving the normal vector are introduced in the rest of this section.



(a) Process of searching boundary-patches



(b) Process of determining boundary pixels

Fig. 2. Process of the proposed method.

A. Plane Equation Based Method

A method of normal vector calculation was used in [12], which is based on plane equation. General equations of plane, three-variable linear equation, can be expressed as

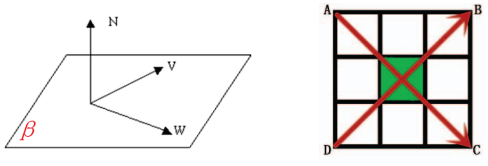
$$Ax + By + Cz + D = 0 \quad (A^2 + B^2 + C^2 \neq 0) \qquad (1)$$

The vector $\mathbf{n} = (A, B, C)$ is the normal vector of this plane. So Eq.1 can be solved by the least square method, and then through generalized inverse matrix, the normal vector can be gotten when the origin point is not on the plane.

B. Vector Cross Product Method

In addition to the plane equation based method, vector cross product method can be used to compute the normal vector. As Fig.3 (a) shows, given a plane β in the right-handed coordinate system, we can derive the normal vector by the cross product of two arbitrary vectors in the plane which are not parallel, such as $\mathbf{w}$ and $\mathbf{v}$. The normal vector $\mathbf{n}$ of plane β is equal to the cross product of the two vectors, $\mathbf{n} = \mathbf{w} \times \mathbf{v}$, and the direction upward perpendicular to β. Define the coordinate of $\mathbf{w} = (x_a, y_a, z_a)$ and $\mathbf{v} = (x_b, y_b, z_b)$, and $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are the unit vector of X, Y, Z axis direction. Though, the normal vector $\mathbf{n}$ can be expressed as:

$$\mathbf{n} = (y_a z_b - z_a y_b)\mathbf{i} + (z_a x_b - x_a z_b)\mathbf{j} + (x_a y_b - y_a x_b)\mathbf{k} \quad (2)$$



(a) Space model (b) Patch model

Fig. 3. Vector cross product model.

As one of the fundamental information, the stereo disparity map is obtained by SPSS method that is proposed in [13]. Then the normal vector can be calculated. As Fig.3 (b) shows, a 3×3 pixel patch, the coordinates of four corners can be

obtained by the disparity map. $\overrightarrow{AC}$ and $\overrightarrow{DB}$ are expressed by the coordinates. Then the normal vector can be written as $\boldsymbol{n_i} = \overrightarrow{AC} \times \overrightarrow{DB}$, which means the normal vector of the middle pixel of patch.

These two methods are compared with the same condition and parameters. The cross product method outperforms the plane equation based method, especially at the changed plane area. Because the plane equation based method has smooth filter characteristic, and this characteristic would weaken the response of plane changing. What's more, the cross product method just use 8 ms to traverse a 1242×375 pixels image while the first method uses 1363 ms. Therefore, the normal vectors are calculated by vector cross product method in this paper. All pixels of the image are traversed and the normal vectors of each pixel are computed.

IV. Road Boundary Detection

The coarse candidate patches of road boundaries on the sparsely sampled image rows are located by the normal vector cues. Then pixel-level localization of the road boundary points are realized by fusing the normal vector, height and color cues with naive Bayes model. Finally, the curb is fitted by support vector regression (SVR) method, based on correct sampling points.

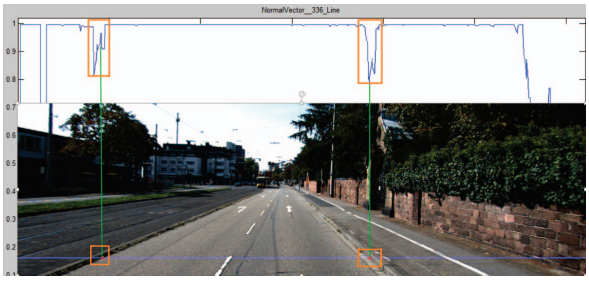
A. Searching Boundary-patches

The normal vector just describes a vertical direction of a plane. To be more specific and scalar, dihedral angle is chosen as the conditional variables, which is the included angle of standard horizontal plane and pixel neighborhood plane. The cosine value of dihedral angle can be calculated by the following formula :

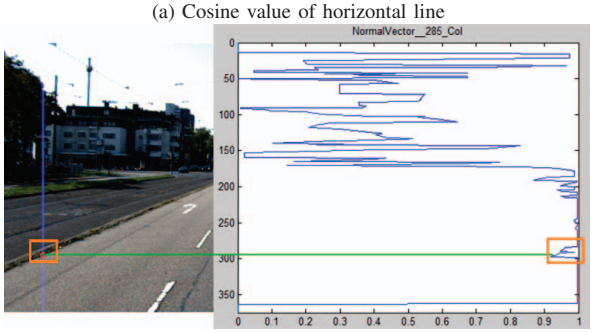
$$\cos\theta = \frac{\mathbf{n} \cdot \mathbf{n}_i}{|\mathbf{n}| \cdot |\mathbf{n}_i|} \tag{3}$$

in which, $\mathbf{n}$ is the standard normal vector in the coordinate system as Fig.1 shows, $\mathbf{n} = (0, 1, 0)$, and $\mathbf{n}_i$ is the pixel normal vector. It is obvious that $\theta \in [-\pi/2, \pi/2]$ in 3D space, so $\cos\theta \in [0, 1]$. For an image, the cosine values of one row and one column are drawn, as Fig.4 shows. It is clear that road boundary corresponds to a strong response of cosine value.

Though, based on this characteristic, the road boundary area can be found. However, in the Fig.4, there are many responses including noise interference, such as sidewalks and complex environment. In order to find the correct area, we take the search strategy that combines horizontal and vertical search. At the bottom of the image, several lines are searched horizontally from the middle to both sides and different lines are given different weight. According to the weighted average values of the searching results, the left and right limit points are calculated. And then, the sampling lines are searched vertically from bottom to top between the left and right limits. So the correct change-points are discovered, which are the red circle shown in Fig.4. The 10×10 boundary areas called boundary-patches can be generated, which take the change-points as the center. Then finding the highest confidence level road boundary points in the boundary-patches is the next work.



(a) Cosine value of horizontal line



(b) Cosine value of vertical line

Fig. 4. Correspondence relation between curb and cosine value.

B. Normal Vector for Road Boundary Detection

Since the boundary-patches are gotten, the probability of every pixel in each boundary-patch is calculated based on the normal vector, which describes the confidence level of the pixel belongs to road boundary class.

For a boundary-patch, the average difference values of 4-connected pixels regions are used to calculate the probability of pixels belonging to the road boundary class. $e_1(i)$ is defined as the average difference value of the normal vectors among the ith pixel and the 4-connected pixels in the boundary-patch. The maximum $e_1(i)$ in this boundary-patch is written as W. A function about $e_1(i)$ and W is constructed, as:

$$\varphi(e_1(i), W) = e_1(i)/W \tag{4}$$

in which, i represents the number of pixels in boundary-patches. It is clear that $\varphi(e_1(i), W) \in [0, 1]$.

Because road boundaries are similar in an image, the values of W are concentrated in a value range. Nevertheless, the W value of the false road boundary-patch is not in the value range. According to this characteristic, the false boundary-patch would be deleted. All W values of boundary-patches in the whole image are counted and voted in corresponding region of histogram. It is obvious that the histogram obeys normal distribution and the W values of correct boundary-patches are in the middle region. Based on the histogram statistics, the proportion of different regions quantity is the probability that indicates the confidence level of a boundary-patch belongs to real road boundary, written as $\chi(W_n)$.

So the results of 4-connected pixels difference values can be regard as a likelihood probability, written as $p(e_1(i)|R_n)$. And the voting results can be regard as a prior probability, expressed as $p(R_n)$. Obviously, $p(e_1(i)|R_n) \sim \varphi(e_1(i), W_n)$

and $p(R_n) \sim \chi(W_n)$. A Bayes model of these patch pixels can be expressed as

$$p(R_n|e_1(i)) = p(e_1(i)|R_n) \cdot p(R_n) \tag{5}$$

in which, i represents the number of pixels in boundary-patches, and n represents the number of boundary-patch in the whole image. For example, when the maximum difference value of a boundary-patch is very small or large, $p(e_1(i)|R_n)$ may be large, but $p(R_n)$ is small. So the false boundary-patch would be deleted.

C. Height and Color for Road Boundary Detection

Since the disparity map is gotten, height information of each pixel can be computed in boundary-patches. In the changed area, the height value will also change. It is similar to the normal vector, and $e_2(i)$ is defined as the average difference value of the height information among the ith pixel and the 4-connected pixels in the boundary-patch. The probability of pixels belonging to the road boundary class can be written as

$$p(R_n|e_2(i)) = p(e_2(i)|R_n) \cdot p(R_n) \tag{6}$$

where i represents the number of pixels in boundary-patches, and n represents the number of boundary-patch in the whole image.

The use of color information is also similar to the normal vector cue. In the changed area, the color information is also different. RGB and HIS channel information is used to calculate the probability of pixels belonging to the road boundary class. The probability of these boundary-patch pixels can be expressed as

$$p(R_n|e_3(i)) = p(e_3(i)|R_n) \cdot p(R_n) \tag{7}$$

in which, i represents the number of pixels in boundary-patches, and n represents the number of boundary-patch in the whole image. $e_3(i)$ is defined as the average difference value of the color channel information among the ith pixel and the 4 adjacent pixels in the boundary-patch.

D. Naive Bayes Model and SVR Fitting

Because the normal vector, height and color information are different and independent, the probabilities, $p(R_n|e_1(i))$, $p(R_n|e_2(i))$, $p(R_n|e_3(i))$, are also conditional independent, as Fig.5 shows. In a boundary-patch, the algorithm just searches only one pixel which is the optimal solution. A Naive Bayes can be written as

$$\mathbf{max} \quad p(R_n|e_m(i)) = \prod_{m=1}^{3} p(R_n|e_m(i)) \tag{8}$$

in which, i represents the number of pixels in boundary-patches, m represents the three characteristics, and n represents the number of boundary-patch. The maximum probability of one pixel means that the pixel is regard as road boundary with the highest confidence.

And then, SVR method fits road boundary. Because road boundary is non-linear and irregular, the Radial Basis Function (RBF) is chosen as kernel function to deal with the task, which is also called Gaussian Kernel Function.

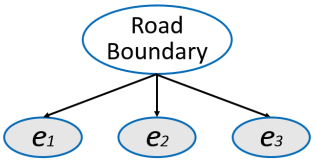


Fig. 5. Naive Bayes Model. In this model each observed variables e_i is independent of the other observed variables given the value of the probability.

V. Experiments

A. Datasets and Running Environment

In order to test the effectiveness of the presented approach, the proposed method is tested on the KITTI Benchmark [11], which is one of the most influential database. The KITTI Road dataset contains 289 training images and 290 testing images, which consist of three different classes of road scenes: UM (urban marked), UMM (urban multiple marked lanes) and UU (urban unmarked). Here, we just deal with road detection and don't care of lanes. For evaluation, the dataset offered pixel-based evaluation and evaluating indices include maximum F1-measure (**MaxF**), average precision (**AP**), precision (**PRE**), recall (**REC**), false positive rate (**FPR**) and false negative rate (**FNR**).

B. Evaluation

There is a key parameter in our method : the radius of RBF kernel function, which is essential for the output results of road detection. The radius of the RBF kernel function is determined by the gamma value, and the gamma value is positively correlated with the radius of the RBF kernel function. We show the qualitative and quantitative evaluation of the gamma values. If the gamma value is large, the experience risk would be low, and it could fit complex curves which can describe more road boundary details. On the contrary, if the gamma value is small, the experience risk would be high, and it could fit simple curves which can filter noise and error sampling curb pixel. Fig.6 shows the above properties. But neither of the two cases is the best result. In order to be smooth in near and flexible in far, let gamma value is 0.00002 within the range of 15 meters in 3D space, and let gamma is 0.0002 far from the range of 15 meters. The experiments about gamma values are tested on the training dataset. Table I, Table II and Table III show the quantitative evaluation (%) of detection results with different gamma values. From the tables, it is obvious that the multi-gamma method is better than others. So the multi-gamma method is the best choice for the fitting problem.

Choosing the best parameters, the proposed approach can output good results of road detection. Fig.7 shows the process of detection. The road results are uploaded onto the website and get a state-of-the-art result. Fig.8 shows the result of road detection in bird eye view.

To show the superiority of the proposed method, we compared the results with other stereo methods, including NNP [14], ProbBoost [15], SPlane+BL [16], GRES3D+SELAS [17], HistonBoost [18] and BM [19], which

(a) gamma= 0.00002

(b) gamma= 0.0002

Fig. 6. Effect of gamma on fitting results.

TABLE I
COMPARISON ON UM (PERSPECTIVE VIEW)

Gamma	MaxF	*AP*	PRE	REC	FPR	FNR
0.00002	92.07	86.73	93.63	**90.56**	1.33	**9.44**
0.0002	91.96	**86.92**	93.84	90.15	1.27	9.85
Multi-gamma	**92.45**	82.58	**96.99**	88.32	**0.59**	11.68

TABLE II
COMPARISON ON UMM (PERSPECTIVE VIEW)

Gamma	MaxF	*AP*	PRE	REC	FPR	FNR
0.00002	90.05	**82.91**	**95.52**	85.17	1.41	14.83
0.0002	89.33	82.22	94.69	84.54	1.68	15.46
Multi-gamma	**90.72**	80.59	95.17	**86.67**	**0.78**	**13.33**

TABLE III
COMPARISON ON UU (PERSPECTIVE VIEW)

Gamma	MaxF	*AP*	PRE	REC	FPR	FNR
0.00002	85.43	71.21	83.44	87.51	3.35	12.49
0.0002	84.45	72.12	84.58	84.32	2.93	15.68
Multi-gamma	**86.71**	**72.23**	**84.95**	**88.55**	**2.77**	**11.45**

Fig. 7. Illustration of the proposed algorithm processes, the first row shows the boundary patches, the middle row shows the results of Navie Bayes model and the third row shows the results of SVR fitting, in which the blue and green areas use different gamma parameters.

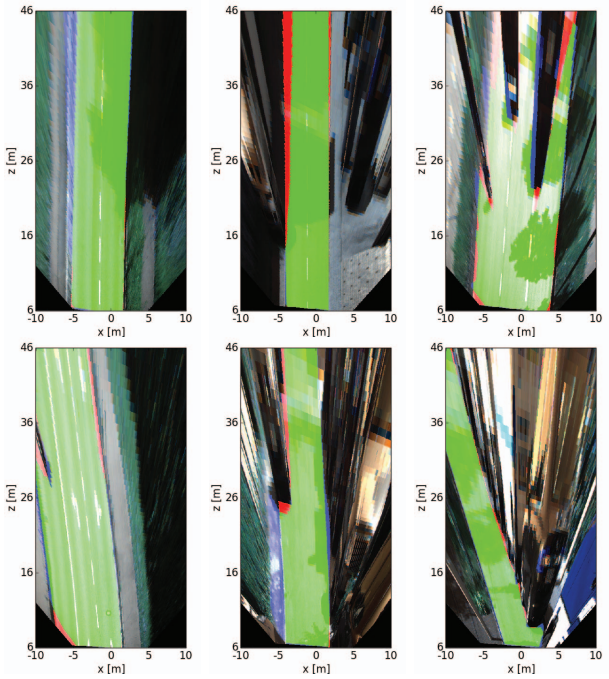


Fig. 8. Road detected in bird eye view. Here, red denotes false negatives, blue areas correspond to false positives and green represents true positives.

are the latest and the top ranked algorithms. Table IV, Table V, Table VI and Table VII shows the compared quantitative evaluation (%) of detection results on UM, UMM, UU and Urban Road. From the table, it is clear that the proposed algorithm gets the best MaxF, PRE and FPR in UMM. In these stereo algorithm, the proposed algorithm ranks the first place in UMM and urban road scene, and it ranks the second place in UM and UU scene. The ranking can be query on the website [20].

VI. CONCLUSION

In this paper, a multi-cue fusion road detection approach is proposed. The method is divided into two parts: road boundary detection and fitting road boundary. The normal vector, height and color information are used to calculate the road boundary probability of each pixel in the boundary-patches. The Naive Bayes framework fuse these three cues to get the highest probability road boundary points in the boundary-patches. Then these points are fitted into the boundary curve by SVR method. Our algorithm is tested on the KITTI road datasets and get the first place in UMM and unban road scene [20].

Yet, the proposed approach uses more normal vector information and rarely uses color and feature information of the images. In the future, we will focus on the color and feature information and make full use of information contained in images for road detection. At the same time, optimizing algorithm and parallel processing with GPU, the method would run faster.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (No. 61375050). The authors acknowledge the postgraduate laboratory and all partners within a laboratory.

TABLE IV
RESULTS OF ONLINE EVALUTION ON UM (BEV)

Algorithm	MaxF	*AP*	PRE	REC	FPR	FNR
NNP	**90.50**	87.95	**91.43**	89.59	**3.83**	10.41
ProbBoost	87.48	80.13	85.02	90.09	7.23	9.91
SPlaneBL	85.23	**88.66**	83.43	87.12	7.89	12.88
GRES3D+SELAS	83.69	84.61	78.31	89.88	11.35	10.12
HistonBoost	83.68	72.79	82.01	85.42	8.54	14.58
BM	78.90	66.06	69.53	**91.19**	18.21	**8.81**
Ours	89.42	83.13	88.31	90.55	5.46	9.45

TABLE V
RESULTS OF ONLINE EVALUTION ON UMM (BEV)

Algorithm	MaxF	*AP*	PRE	REC	FPR	FNR
NNP	91.34	**88.66**	91.07	91.60	9.87	8.40
ProbBoost	91.36	84.92	88.18	94.78	13.97	5.22
SPlaneBL	82.04	85.56	75.11	90.39	32.93	9.61
GRES3D+SELAS	88.19	88.65	83.98	92.85	19.48	7.15
HistonBoost	88.73	81.57	84.49	93.42	18.85	6.58
BM	89.41	80.61	83.43	**96.30**	21.02	**3.70**
Ours	**92.21**	87.99	**91.55**	92.89	**9.43**	7.11

TABLE VI
RESULTS OF ONLINE EVALUTION ON UU (BEV)

Algorithm	MaxF	*AP*	PRE	REC	FPR	FNR
NNP	**85.55**	76.90	**85.36**	85.57	4.79	14.25
ProbBoost	80.76	68.70	85.25	76.72	**4.33**	23.28
SPlaneBL	74.02	79.61	65.15	85.68	14.93	14.32
GRES3D+SELAS	82.70	**83.95**	78.54	87.32	7.77	12.68
HistonBoost	74.19	63.01	77.43	71.22	6.77	28.78
BM	78.43	62.46	70.87	**87.80**	11.76	**12.20**
Ours	85.46	74.07	85.06	85.86	4.91	14.14

TABLE VII
RESULTS OF ONLINE EVALUTION ON URBAN ROAD (BEV)

Algorithm	MaxF	*AP*	PRE	REC	FPR	FNR
NNP	89.68	86.50	**89.67**	89.68	**5.69**	10.32
ProbBoost	87.78	77.30	86.59	89.01	7.60	10.99
SPlaneBL	79.63	83.90	72.59	88.17	18.34	11.83
GRES3D+SELAS	85.09	**86.86**	82.27	88.10	10.46	11.90
HistonBoost	83.92	73.75	82.24	85.66	10.19	14.34
BM	83.47	72.23	75.90	**92.72**	16.22	**7.28**
Ours	**89.75**	84.15	89.02	90.49	6.15	9.51

REFERENCES

[1] $http://en.people.cn/n3/2016/0417/c90000-9045643.html$

[2] Dollár P., Zitnick C.L.: Structured Forests for Fast Edge Detection[C]. International Conference on Computer Vision (ICCV) 2013: 18411848.

[3] Brust CA, Sickert S, Simon M, Rodner E, Denzler J. Convolutional Patch Networks with Spatial Prior for Road Detection and Urban Scene Understanding. arXiv preprint arXiv:1502.06344v1 (2015).

[4] Zhao G, Yuan J. Curb detection and tracking using 3D-LIDAR scanner [C]. In Image Processing (ICIP), 2012 19th IEEE International Conference on. 2012:437440.

[5] Clode S, Kootsookos P J, Rottensteiner F. The automatic extraction of roads from LIDAR data [C]. In ISPRS 2004.

[6] Hu X, Tao C V, Hu Y. Automatic road extraction from dense urban area by integrated processing of high resolution imagery and lidar data [J]. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences. Istanbul, Turkey. 2004, 35: B3.

[7] AY Hata, FS Osorio, DF Wolf. Robust curb detection and vehicle localization in urban environments [C]. In Intelligent Vehicles Symposium Proceedings, 2014 IEEE. 2014: 12571262.

[8] Kellner M, Hofmann U, Bouzouraa ME, Stephan N. Multi-cue, Model-Based Detection and Mapping of Road Curb Features Using Stereo Vision [C]. International Conference on Intelligent Transportation Systems, 2015 IEEE. 2015: 1221 - 1228.

[9] Oniga F, Nedevschi S. Polynomial curb detection based on dense stereovision for driving assistance[C]. Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference. 2010:1110 - 1115.

[10] Siegemund J, Franke U, FoRstner W. A temporal filter approach for detection and reconstruction of curbs and road surfaces based on Conditional Random Fields [C]. IEEE Intelligent Vehicles Symposium. IEEE, 2011:637-642.

[11] Geiger A, Lenz P, Stiller C, and Urtasun R, Vision meets robotics: The KITTI dataset. The International Journal of Robotics Research, vol. 32, pp. 12311237, 2013.

[12] Einecke N and Eggert J. Block-Matching Stereo with Relaxed Fronto-Parallel Assumption. Intelligent Vehicles Symposium Proceedings, 2014 IEEE, June 2014, pp. 700705.

[13] Yamaguchi, Koichiro, David McAllester, and Raquel Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. Computer Vision−ECCV 2014. Springer International Publishing, 2014. 756-771.

[14] Chen X, Kundu K, Zhu Y, Berneshawi AG, Ma H, Fidler S and Urtasun R. 3D object proposals for accurate object class detection. In Advances in Neural Information Processing Systems, 2015 (pp. 424-432).

[15] Vitor GB, Victorino AC, Ferreira JV. Victorino, and Janito V. Ferreira. A probabilistic distribution approach for the classification of urban roads in complex environments. Workshop on IEEE International Conference on Robotics and Automation (ICRA). 2014.

[16] Wang B, Fremont V, and Rodriguez S. Color-based road detection and its evaluation on the kitti road benchmark. Intelligent Vehicles Symposium Proceedings, 2014 IEEE, June 2014, pp. 3136.

[17] Shinzato, Patrick Y., Diego Gomes, and Denis F. Wolf. Road estimation with sparse 3D points from stereo data. Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on. IEEE, 2014.

[18] Vitor G B, Victorino A C, Ferreira J V. Comprehensive performance analysis of road detection algorithms using the common urban Kitti-road benchmark[C]. Intelligent Vehicles Symposium Proceedings. IEEE, 2014:19-24.

[19] Pradeep V, Medioni G, Weiland J. Piecewise Planar Modeling for Step Detection using Stereo Vision[J]. Workshop on Computer Vision Applications for the Visually Impaired, 2012.

[20] $http://www.cvlibs.net/datasets/kitti/eval\_road.php$