Proceedings of the Federated Conference on Computer Science and Information Systems pp. 333–340 DOI: 10.15439/2019F121 ISSN 2300-5963 ACSIS, Vol. 18

License Plate Detection with Machine Learning Without Using Number Recognition

Kazuo Ohzeki
Algorithm Lab.
Nishida Build. 5F 2-14-6 Shibuya,
Shibuya-ku 150-0002 Tokyo Japan
Email: ohzeki@shibaura-it.ac.jp

Max Geigis
University of Applied Science
Bahnhofstraße 61, 87435
Kempten (Allgäu) Germany
Email: maxgeigis@gmail.com

Stefan Alexander Schneider
University of Applied Science
Bahnhofstraße 61,
87435 Kempten (Allgäu) Germany
Email:
stefan-alexander.schneider@hs-kempten.de

***Abstract*—In autonomous driving, detecting vehicles together with their parts, such as a license plate is important. Many methods with using deep learning detect the license plate based on number recognition. However, there is an idea that the method using deep learning is difficult to use for autonomous driving because of the complexity in realizing deterministic verification. Therefore, development of a method that does not use deep learning(DL) has become important again. Although the authors have made the world's best performance in 2018 for Caltech data with using DL, this concept has now turned to another research without using DL. The CT5L method is the latest type, that includes techniques of the continuity of vertical and horizontal black-and-white pixel values inside the plate, unique Hough transform, only vertical and horizontal lines are detected, the top five in the order of the number of votes to ensure good performance. In this paper, a method to determine the threshold value for binarizing input by machine learning is proposed, and good results are obtained. The detection rate is improved by about 20 points in percent as compared to the fixed case. It achieves the best performance among the conventional fixed threshold method, Otsu's method, and the conventional method of JavaANPR.**

I. Introduction

THE issue of practical use of automatic driving is discussed in ROAD2016 [1]. With the advancement of technology, the number of test cases is said to be huge and to be 10^{12}. Because of this, Virtual Test becomes more important. With the addition of AI technology, this type of test case is expected to become even more extensive, and it is also pointed out that the deterministic test becomes more difficult.

With the technology that processes images obtained from camera sensors and recognizes them in the external world, many methods can be developed and studied by software development on a computer. Our group has been promoting the Samurai Project on ADAS since 2016, and has developed image processing technologies such as license plate detection, lane detection, vehicle detection, vehicle maker logo detection, front grill detection, etc. . Among them, LPD by deep learning has developed an improved method that surpasses the highest performance of the past [2]. Currently, many LPDs detect plate areas by number and character recognition.

Among the technologies related to autonomous driving, we are especially developing technologies for vehicle detection and license plate detection. In addition to vehicle detection, the significance of plate detection will be described. As a second step after vehicle detection, for example, plates, lights, etc. of the vehicle elements are individually detected. The verification of the element by this detection makes it possible to verify whether the event that detected the vehicle was correct. This can increase the reliability of vehicle detection. In the vehicle detection, as a detection result, both a correct detection of a vehicle (True Positive: TP) and a false detection (False Positive: FP) in which a non-vehicle is recognized as a vehicle are output. That is, detection of precision = TP / (TP + FP) is performed. In this equation, if the detection result of the vehicle element such as plate detection is evaluated as the reliability in the second stage processing and the false detection (FP) is eliminated, the FP value of the equation can be reduced and the precision rises. This is the significance of performing license plate detection in addition to vehicle detection.

Deep learning is also effective in license plate detection [2]. Convolutional Neural Network (CNN), which is used in deep learning, performs effective computations, but it is difficult to clearly describe the process of generating the correct result by including the multiplication of many coefficients and non-linearity. Therefore, there is a high possibility that the test of the autonomous driving vehicle using deep learning can not estimate the total number within the feasible range, which is a problem [3]. It has also been pointed out that there are problems with AI and safety assurance by the Cyber-Pysical System research group and the researchers of software engineering. [4] Therefore, developing a scheme that does not use CNN is effective as one method to avoid the risk arising from such an indication.

Estimated power consumption of PCs in electric vehicles is 40% of the total increase [5], and a decrease is required. Low power consumption is desired. Methods of non-deep learning etc. are also becoming more important as a study target.

IEEE Catalog Number: CFP1985N-ART ©2019, PTI

Another background is that the illegality of ANPR was pointed out in 2008 [6] [7]. The use for the crime prevention such as the police is established as a legitimate use even after that. However, there is a move that appeals to identify personal information from the number as a privacy issue, and it remains unsolved, and there is a possibility that the future controversy will continue. It is unpredictable how future laws will change, and insurance technology development and development of backup technology are necessary to cope with the uncertainty. Therefore, in consideration of the privacy problem, we have taken measures to detect the plate only from the features of the license plate without using number recognition.

In this paper, considering such background, it is characterized by examining in the range of methods that do not use number recognition and do not use deep learning such as CNN.

In our group, we are conducting research and development on practical application of automatic driving in the Samurai Project. Regarding license plate detection, Samurai Project has developed both statistical and deterministic methods, as shown in Table 1. Although Linear Regression used to determine the threshold is due to machine learning, it is a fixed factor, a finite number of realizable deterministic processes, and not deep learning. The reason that deep learning can only be performed by statistical test is considered to be due to the large number of coefficients and the fact that the circuit contains nonlinearity. Along with the development of test environment in the future, in addition to pursuing high performance using deep learning, it is also necessary to carry out technological development without deep learning and secure flexibility in technology integration .

Table 1 Two alternatives of Samurai Project for plate detection methods

item	method1	method2
process/ test	statistic non-linear dis-continuous	deterministic linear continuous
algorithm	deep Learning CNN	CT5L (threshold binarization Hough transform machine Learning Linear regression)
reference	[2]	[19], [20] [This paper]

Further structure of this paper is as follows. Section II provides the overview of the relevant works. Section III provides our number plate detection algorithm. Section IV describes machine learning tools used in this paper. Section V demonstrates the experimental results. Evaluation, Conclusions and future research directions are outlined in section VI.

II. RELATED WORKS

A. Methods by Conventional Image Processing

The license plate detection of a vehicle includes a method by conventional image processing and a method by recent deep learning, which are respectively useful techniques. First, as a conventional method of image processing, there is a method using Rectangle Detection by Rosito et al. [8]. They made it possible to detect a generalized rectangle including the case of tilting using the rectangle and symmetry of the rectangle in the Hough transform. This is used to detect License Plate and Rectangle-like semiconductor chips.

There is a scheme of Frequency of Luminance changes in the Vertical and horizontal direction, which has been studied by Martinsky [9]. The surface of the vehicle is flat, and in the area of the License Plate, there are large brightness changes in numbers, letters, and the like. If scanning horizontally, the place where the change is large corresponds to the plate area. Also, if scanning vertically, the plate part will be the place of large change as well.

B. Methods Using Deep Learning

Next, we describe a high-performance method using deep learning. Zang et al use a CNN for three datasets of RGB color channels, and integrates the results by majority decision [10]. They show the advantage of three channel integration. Montazzolli et al. detect Brazilian plates using YOLO in real-time [11]. There are three stages, cropping small region around plate, detecting plate region, and character recognition on the plate by adjusting parameters of YOLO.

Dong et al. also present two-stage detection of Chinese plates [12]. At the first stage, it detects plate region using Region Proposal Network in low resolution, then replaces by a high resolution image, and detects four vertices of the plate using Faster R-CNN, then obtains corrected plate region using affine transform. At the second stage, seven STN and CNN units work in parallel for seven characters for separation and recognition.

The highest performance using Deep learning for standard dataset of Caltech is presented by Kim et al. [13], together with the search from the second to the seventh results. They use the faster R-CNN for the vehicle region detection and candidates for license plates in each detected region with the hierarchical sampling method (CNN) are generated. Finally, non-plate candidates are filtered out by training a deep convolutional neural network. Training two different CNN's for plates and non-plates, they remove FP results using non-plate CNN. For Caltech standard dataset, precision of 98.39% and recall of 96.83% are performed, which are the best world records at the time of publication 2017. But the method by our group, which was announced in

2018, surpasses the method of Kim et al. Achieved performance [2].

By improving accuracy of character recognition, a method to detect character region at the first stage without vehicle region detection has become effective. Among them, [14] is an ambitious paper using deep learning of character recognition. At the first stage, from candidate region detected by weak character detection making saliency map, rectangle plate region is detected after removing FP by two-class CNN. At the second stage, using character separation and character recognition, together with labeling results based on connectivity, numbers and characters on the plate are confirmed. In total 37 class CNN is constructed with ten numbers and 26 uppercase letters and a single non character. Region detection at the first stage should be improved, while character recognition at the second stage improves by adding connectivity process. Using ten numbers and 26 uppercase letters for CNN, the recognition rate can be advanced, though more kinds of characters are needed depending on each country specification. For the first stage of region detection of a plate, character recognition may fail to produce FP's for logos and advertisements other than plates. In the case of serial construction, total performance is a product of each performance of each stage. Accuracy of each stage must be the highest. Even in the case of so-called Coarse-to-Fine serial construction, the first stage of Coarse should not be coarse but fine in accuracy, excluding FP and removing FN.

ANPR software vendors have published Accuracy results based on image benchmarks. In 2017, Sighthound announced an accuracy of 93.6% for original images [15]. In 2017, OpenALPR's commercial software announced 95-98% accuracy rates in public images [16]. The Brazilian team announced a scheme with an average recognition rate of 93.53% in April 2018, showing a significant improvement from the 81.8% obtained in the previous paper [17].

In the paper by Silva et al. at the 2018 ECCV, an example of detection in a moving picture is shown, but in the comparison of Accuracy, the result changes depending on the data set used [18]. As quoted in Table 2-1, the data averages 81-89% with the lowest at 57-75% and the highest at 96-98%.

Looking at these results, LPD for autonomous operation is still in development, and future performance improvement is expected. In deep learning, it is expected that the amount of computation is large and thus the power consumption will also increase. According to materials of Michigan University [5], 40% of the power consumption in the vehicle is the processing of PC, and the reduction is an issue. Development of LPD by conventional image processing technology not using deep learning adapted to such a requirement is required in the menu of the autonomous driving system. Further, a method using a technique for recognizing numbers and characters on a plate as a key is in conflict with German precedents, and a method for performing feature detection of a plate that does not perform number or character recognition is required. For these reasons, in this paper, we will develop LPD by Image Processing.

III. Proposed Method CT5L

In addition to vehicle detection, the significance of plate detection is to individually identify vehicle elements such as plates, lights, radiators, logos, tires, window glasses, shadows, tail lamps, etc., as a second step after vehicle detection. Thus, it is possible to verify whether the event that detected the vehicle was correct. This can increase the reliability of vehicle detection (see Fig. 1). Vehicle detection includes, as detection results, both correct vehicle detection (True Positive: TP) and false detection (False Positive: FP) in which a non-vehicle is recognized as a vehicle. That is, precision = TP / (TP + FP). In this equation, detection results of vehicle elements such as plate detection are evaluated as reliability in the second stage processing. Specifically, if detection of each element is confirmed, one point is added, and if the sum becomes larger than n, it is regarded as correct detection, otherwise it is regarded that vehicle detection is erroneous. Eliminating false positives (FPs) can reduce the FP value of the equation and

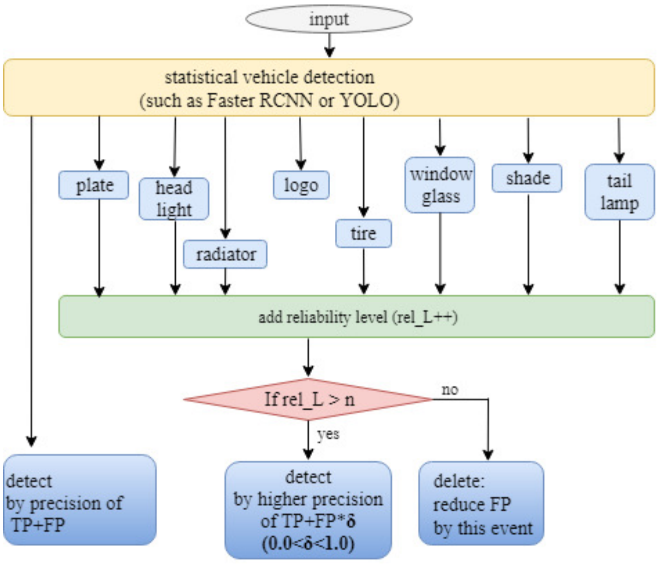

input
statistical vehicle detection
(such as Faster RCNN or YOLO)
plate
head light
logo
radiator
tire
window glass
shade
tail lamp
add reliability level (rel_L++)
If rel_L > n
no
yes
detect by precision of TP+FP
detect by higher precision of TP+FP*δ (0.0<δ<1.0)
delete: reduce FP by this event

Fig.1 Integration of vehicle and element detection

raise the precision value. This is the significance of performing license plate detection in addition to vehicle detection.

We will describe the "Combining Top Five Lines (CT5L) Method" of the proposed method. This method is an improvement of the basic method presented in [19] and [20]. In the process of development, there were several branched versions, such as the oblique detection compensation method by Okunuki and the 3D compensation method by Max Geigis. In this paper, based on the examination of variations of this, we constructed a high-performance and stable method. Fig. 2 shows a block diagram of CT5L Method.

The number plate area determined from the vehicle area of the input image is compared with the threshold value and binarized (binary image). The binarized data is evaluated as to vertical change and continuation length, horizontal change and continuation length as processing before Hough transform. Based on this evaluation result, points that are candidates for plate boundaries are registered, and points that are not so are deleted (VH continuous condition).

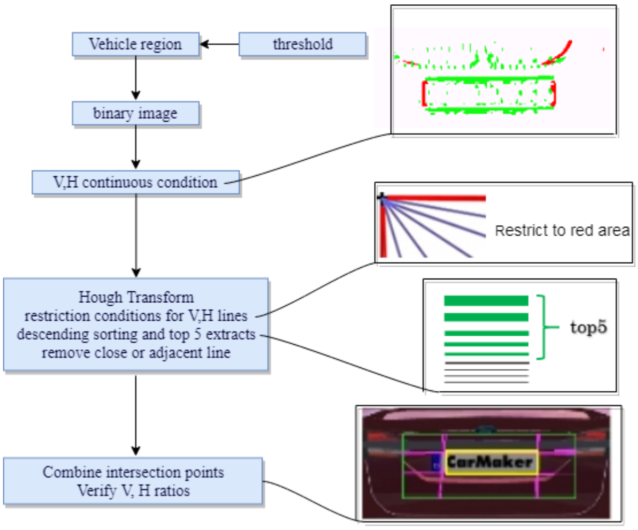

Vehicle region
threshold
binary image
V,H continuous condition
Restrict to red area
Hough Transform
restriction conditions for V,H lines
descending sorting and top 5 extracts
remove close or adjacent line
top5
Combine intersection points
Verify V, H ratios
CarMaker

Fig.2 CT5L Method Block Diagram.

Next, Hough transform is performed in the "Hough Transform" section. The Hough transform is performed only within a range in which the angles are limited to 0 ± ε degrees and 270 ± ε degrees. The transform is performed only on the red area shown in Fig. 2. As a result, unnecessary calculations are not performed, and thus speeding up can be achieved. When the search range is limited to plus and minus 3 degrees, the amount of operation is reduced to 12/360 = 1/30. Sorting is performed in descending order of the number of votes of distribution after transform, and the upper five lines are selected for each of vertical and horizontal. At this time, close lines may be dense, so if the absolute value of the difference between the line detected later and the upper-rank line up to that time is less than δ, it will be regarded as the same line and removed from the ranking process. In this way, five valid lines are extracted as candidates for the plate outline. Next, in the "Combination of intersection point", select a pair that forms a rectangle by the combination of 5 vertical lines and 5 horizontal lines, verify the aspect ratio and size, and determine the rectangle closest to the plate. If no candidate rectangle is established, no detection is made.

Fig.3 shows how plate candidate areas are extracted from the input image. The region of the vehicle is extracted as a candidate of the object region by Faster-RCNN [21] or the like. This is also demonstrated by Kido [22] and shows good results. Also, YOLO [23] can be used. The vehicle area is normalized to a size of 400 × 300 pixels as shown in Fig.4. An area 200 × 75 pixels including the plate candidate is set in this area. An example of the detected plate is shown in Fig.5.

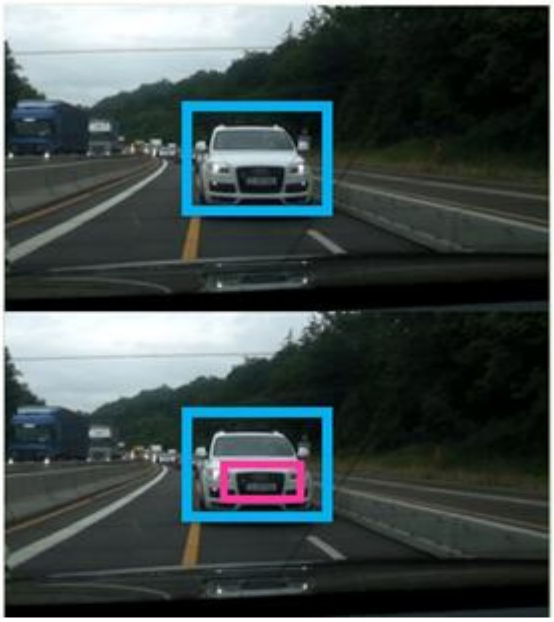


Fig.3 Vehicle area (upper) and license plate candidate area (lower).

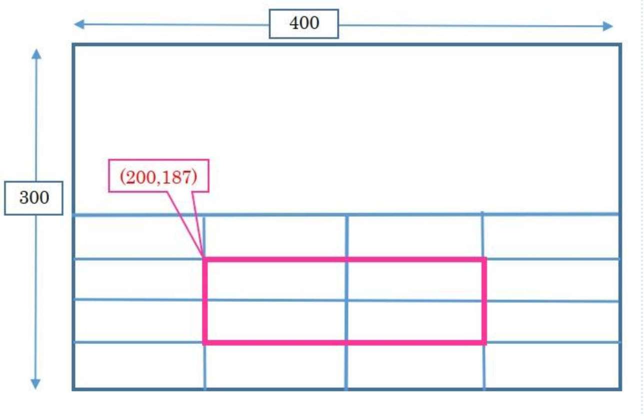


Fig. 4 Whole vehicle area(400x300) and candidate area of license plate(200x75).

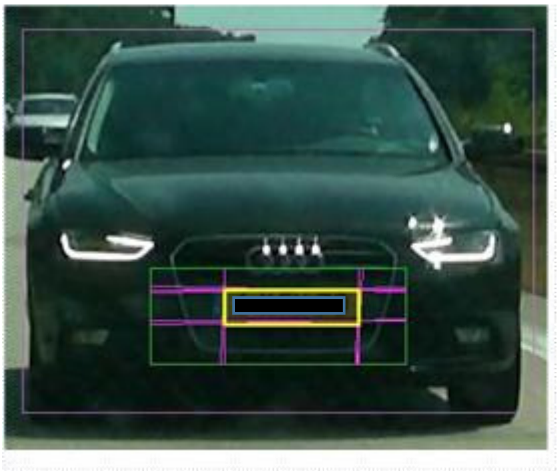
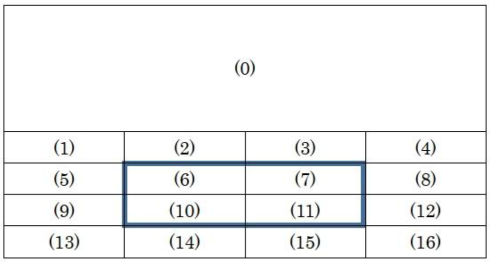


Fig.5 An example of detected plate in this paper.

IV. Machine Learning for threshold value

Machine learning for threshold value applied to input image to get binary images is introduced. The binary images are important to be used in the following process. The threshold value was at the beginning of this project, the center value of luminance which is 127 or 128. Then, taking into account of input change of luminance, average value of input luminance or shifted average value by a fixed value are tried. But through experiments, detection ratio is affected by changing threshold values. The threshold may be affected by input environment which is beyond average luminance. To cope with this problem, automatic learning by machine learning tool is effective. In this paper a machine learning method is newly introduced to derive more effective threshold from input whole image, vehicle region image, or plate candidate region.

Fig.6 shows the upper and lower two areas of the vehicle area and the lower area divided into 16 areas. As the explanatory variables used in machine learning, the average value and the variance value of the luminance of the area numbered in Fig. 6 are used. The threshold value was changed as the correct value for supervised learning, and the value at which the software was able to detect the plate normally was allocated. Therefore, a predicted value of threshold is set as an objective function. If the correct data has a width, the center value is taken as the correct value of the objective function. As a whole, machine learning system with at most 34 explanatory variables can be configured. Table 2 shows the sequence of data used. ML is a regression that predicts a threshold value, and in this case, a frequently used linear multiple regression function is used.



(0)			
(1)	(2)	(3)	(4)
(5)	(6)	(7)	(8)
(9)	(10)	(11)	(12)
(13)	(14)	(15)	(16)

Fig.6. shows the upper(0) and lower(1)-(16) two areas of the vehicle area. The lower area divided into 16 areas.

Table 2 Data structure for training and test.

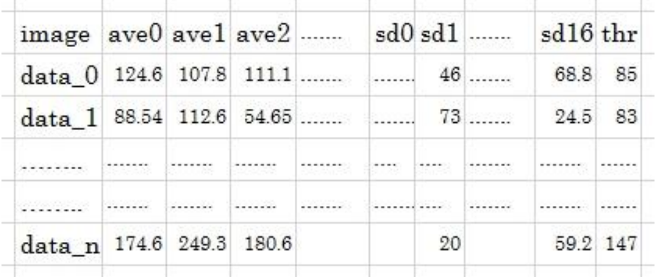


image	ave0	ave1	ave2	....	sd0	sd1	....	sd16	thr
data_0	124.6	107.8	111.1	....	....	46	....	68.8	85
data_1	88.54	112.6	54.65	....	....	73	....	24.5	83
....	......	......	......	......	....	....	......	......	......
....	......	......	......	......	......	....	......	......	......
data_n	174.6	249.3	180.6			20		59.2	147

V. Experiments

The experiment was divided into three types.

(1) Basic method: In the case of a fixed threshold as the basic configuration,

(2) ML method: When ML is introduced and threshold is automatically determined,

(3) Otsu method: The case where the famous Otsu's method is used as a method of binarization, will be described.

As the input image, a road image taken using an on-vehicle camera and a vehicle image on the road synthesized by software for performing a virtual test were used. The road images are selected from the real images [24] on the road in the front and rear of a vehicle traveling in Europe, especially in Germany (see Fig. 5). The synthesized images were generated using IPG Automotive's CarMaker® [25], which is widely used as Virtual Test Software (Fig. 6). Images are collected that are difficult to detect. Also, the size is Hi-definition (HD: 1920x1080) standard. The correct value "thr" (in Table 2) which is teacher data was obtained by changing threshold over the entire range for each image. In this preliminary experiment to obtain correct data, the median value of the threshold value when normal plate detection can be performed is determined, and up to 66 correct data can be obtained at present(n=66 in Table 2).

Table 3 is the result of performing a detection experiment by JavaANPR of the prior art [9]. In the case of HD size, in the first column, in the second column when vertical and horizontal are reduced to 1⁄2, and in the third column, the vehicle region size (400 × 300) is cut out. Because we use JavaANPR's software as is in Table 3, it is not possible to compare the same lines simply with the detection situation proposed in this paper. The images used in this paper are with increased levels of difficulty, such as those that are shot in backlight, dark images, bright images, artificial synthesized images by CarMaker®, and so on. Common road images achieved precision = 0.923 in the old version of CT5L in the experiment by Okunuki.

Table 3 results by JavaANPR [9]

	1	2	3
images	original full_size 1920x1080	quarter 960x540	cropped 400x300
TP	0.303030303	0.015151515	0.272727273

Fig.6 A car Image generated by IPG CarMaker

(1) Basic method: CT5L method was executed with threshold fixed. In the experiments up to now, as the threshold value, for example, the intermediate value (127 or 128) of luminance, a value of 95 or the like have been used in the preliminary experiment with a fixed value. In this Basic experiment, changes in the detection rate of the entire search were examined in which the threshold was changed from 0 to 255 on the obtained input image. As a result, as shown in Fig.7, using 89 as the fixed threshold value was the best result. As for the other values, it can be seen that the detection rate randomly fluctuates and gradually decreases as it gets away from 89. Also, there is a second peak at 104, 105.

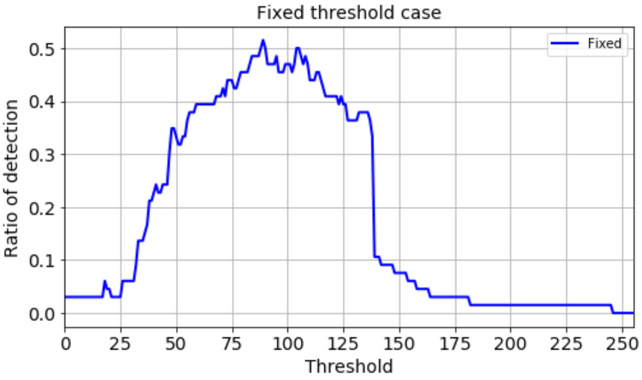


Fig.7 The case of fixed threshold

(2) ML method: The implementation of ML used the Linear Regression function of Scikit-learn included in the Library of Python. The explanatory variable is extracted from the numerical data described in Section 4, and learning is performed by Linear_Regression. When the predicted value of threshold output as a result is included in the range of the correct answer, it is regarded as the correct answer. That is to minimize the sum of squares of errors with the correct answer. The learning used Leave-One-Out Cross-Validation method (LOOCV) [26] which is a type of cross-validation [27]. In LOOCV, one test data is separated from learning data, and all remaining data are used for learning. Using the regression coefficient after learning, test data that has been isolated without being used for learning is input, a predicted value is determined, and the accuracy rate is measured. First of all, the performance was evaluated when the explanatory variable was limited to one. This is useful because it evaluates the effectiveness of each explanatory variable and is also a source of feature variable selection in subsequent multiple regression experiments.

Fig.8 shows the result when there is one explanatory variable. Using the luminance average value as an explanatory variable is better than using the standard deviation value, and the range is about 0.46 to 0.6. COD is also better for average than that for Sd. Also, as the position of the variable, 2, 6, 10, etc. are bad, and 4, 5, 8. 9, 11, 12, 13, etc. are good in results.

he combination which takes out several explanatory variables from 34 explanatory variables becomes huge. In fact, the total number of combinations that extract n from 34 is 2^{34}-1, if n = 34 from the formula $2^{n} = \sum_{k=0}^{n}$ nCk. Taking out multiple explanatory variables is called “Feature Selection”, and methods to reduce the number of searches have been studied for a long time. The method of selecting the one with good performance in the case of one explanatory variable and sequentially increasing the combination is called “Forward Method” [28]. In addition, when increasing sequentially, it is considered as a bad effect that it can not be removed after registration once, and a flexible method of adding p pieces and excluding r pieces

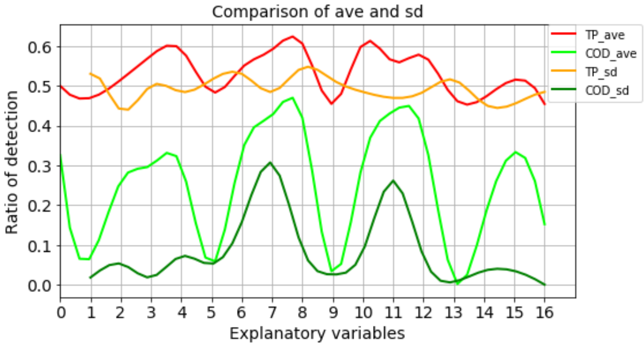


Fig.8 The case of a single explanatory variable.
For each explanatory variable, detection ratio is evaluated by average or variance. TP_ave=True positive rate, ave means average of detection ratio, sd=Standard deviation, and COD=Coefficient Of Determination. Other than grid points are interpolated by the spline function. The same is true for the graphs below.

has been proposed as "+p-r method" [29l. Here, based on the Forward method, while increasing sequentially from the high precision explanatory variables, we will try to replace as appropriate when increasing the number. Similarly, Fig. 9.,Fig.10, Fig. 11, Table 4, Table 5 and Table 6 show the results when 3, 4, 5, 6 and 7 explanatory variables are used.

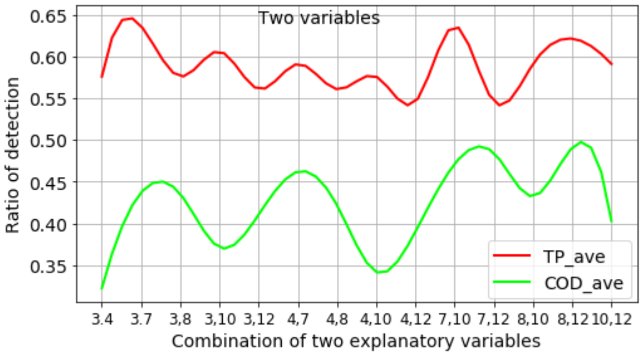


Fig.9 Results when using two explanatory variables.

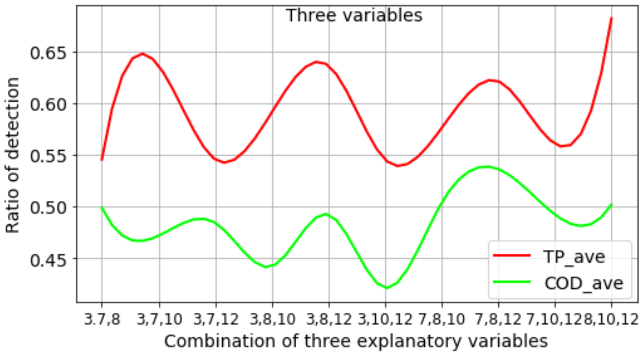


Fig.10 Results when using three explanatory variables.

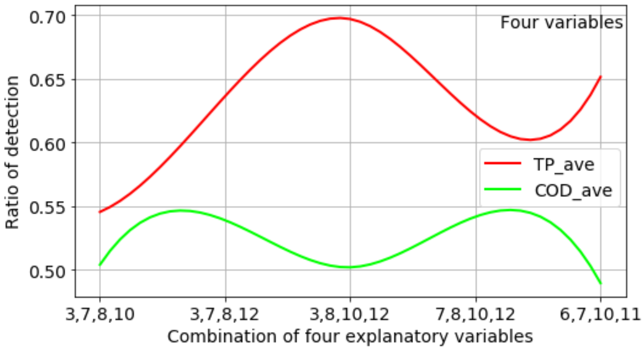


Fig.11 Results when using four explanatory variables.

Table 4 the case of Five Variables.

rate	ast6,ast7,ast10,ast11,ast12
TP	0.621212121
COD	0.514388478

Table 5 The case of six variables.

rate	ave4,ave6,ave7,ave10,ave11,sd11
TP	0.560606061
COD	0.543577619

Table6 Two cases of seven and eight variables.

	The case of seven variables
rate	ave5,ave6,ave7,ave8,ave9,ave11,sd11
TP	0.742424242
	The case of eight variables
rate	ave5,ave6,ave7,ave8,ave9,ave10,ave11,sd11
TP	0.712121212
COD	0.695194388

(3) Otsu method: There is a famous Otsu's method as a scheme of binarization. Here, image data is given, and experiments are performed in the case of plate detection using a threshold value obtained by the Otsu binarization method [29_Otsu] as a predicted value. Otsu's method is to automatically find the optimum threshold iteratively so as to maximize the ratio of the interclass variance divided by the intraclass variance. The implementation of Otsu's binarization used the threshold function in OpenCV2. The image was given in three ways: an original image (1920x1080), a vehicle area (400x300) and a plate candidate

The fixed threshold used in the past (95, etc.) has been improved in some cases, but has not reached a good example by ML.

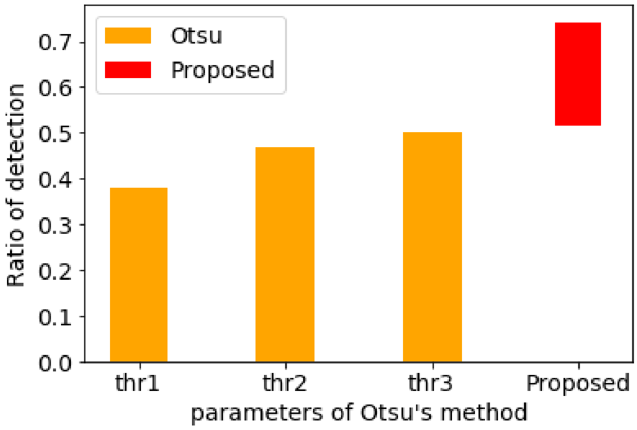


Fig. 12 ratio(TP) vs. area of threshold calculation. thr1=1920x1080, thr2=960x540, thr3=400x300.

VI Evaluation and Conclusion

Table 7 shows the best results of for fixed threshold values case, for Otsu's automatic optimization method case and for proposed Machine Learning method case. In Otsu's method, in addition to the automatically determined threshold value, the value when improved manually is added. In ML, the proposed method with seven variables at present is the best, and it is improved by 0.227 points in rate representation compared to the conventional fixed case.

In the field of deterministic verification, that is, in the field of the method that does not use deep learning at present, optimization of threshold is performed by fixed ML in the method of detecting features of the plate region. It is possible to construct the method which surpasses the conventional automatic optimization method Otsu Method and JavaANPR.

In the future, we will investigate the possibility of further performance improvement, such as mitigation for overdetection from the data distribution obtained during optimization. We will also incorporate and implement Virtual Test's LPD detection function as reference software.

Table 7 Comparison of the best value of three methods

	Fixed threshold	Otsu Method	ML(proposed)
param.	thr=89	thr3(200x75)	ave5,6,7,8,9,11, sd11
rate (TP)	0.515152	0.5 (automatic) 0.68(plus manual adjustment)	0.742424242

REFERENCES

[1] Documentation_1st_RoundTableAutonomousDriving.pdf https://www.hs-kempten.de/fileadmin/fh-kempten/HK/news/2016/

[2] Kazuo Ohzeki, Yoshikazu Kido, Yutaka Hirakawa, Stefan Schneider, "Multi-Module Deep Learning Using Training Data from the first-Stage Error -- Effective For License Plate Detection --", IEICE Technical report vol. 117, no. 514, PRMU2017-176, pp. 25-30, March 2018,(in English)

[3] Xiaowei Huang, Marta Kwiatkowska, Sen Wang and Min Wu, “Safety Verification of Deep Neural Networks”, Keynote of CAV2017 July 2017 Heidelberg

[4] Alexander, Robert David , Ashmore, Rob and Banks, ”The State of Solutions for Autonomous Systems Safety.”, White Rose Research Online, February 2018

[5] James H. Gawron, Gregory A. Keoleian, Robert D. De Kleine, Timothy J. Wallington, and Hyung Chul Kim, "Life Cycle Assessment of Connected and Automated Vehicles: Sensing and Computing Subsystem and Vehicle Level Effects", Environ. Sci. Technol., 2018, 52 (5), pp 3249–3256, American Chemical Society Feb. 2018.

[6] "Das Bundesverfassungsgericht" (in German). Bverfg.de. 3 November 2008. Retrieved 16 February2009.

[7] https://en.wikipedia.org/wiki/Automatic_number-plate_recognition#Germany

[8] Cláudio Rosito Jung and Rodrigo Schramm,”Rectangle Detection based on aWindowed Hough Transform”, Computer Graphics and Image Processing, 2004. Proceedings. 17th Brazilian Symposium, pp. 113 – 120, 17-20 Oct. 2004,

[9] Ondrej Martinsky, "Algorithmic and mathematical principles of automatic number plate recognition systems", B.SC. THESIS, BRNO University of Technology, 2007.

[10] Di Zang, Zhenliang Chai, Junqi Zhang, Dongdong Zhang, and Jiujun Cheng, "Vehicle license plate recognition using visual attention model and deep learning" Journal of Electronic Imaging SPIE 24(3) 033001-pp.1-10. May/Jun 2015.

[11] S´ergio Montazzolli and Claudio Jung, "Real-Time Brazilian License Plate Detection and Recognition Using Deep Convolutional Neural Networks", 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), pp.55-62, Oct. 2017.

[12] M. Dong, D. He, C. Luo, D. Liu, W. Zeng, "A CNN-Based Approach for Automatic License Plate Recognition in the Wild", 28th British Machine Vision Conference BMVC pp.1-12, Sept 2017.

[13] S.G. Kim, H.G. Jeon and H.I. Koo, "Deep-learning- based license plate detection method using vehicle region extraction", Electronics Letters Vol. 53, Issue: 15, 7 20 Institution of Engineering and Technology, pp.1034-1036, 2017.

[14] Hui Li, and Chunhua Shen, "Reading Car License Plates Using Deep Convolutional Neural Networks and LSTMs", arXiv:1601.05610 Jan, 2016.

[15] Dehghan, Afshin; Zain Masood, Syed; Shu, Guang; Ortiz, Enrique G. (19 February 2017). "View Independent Vehicle Make, Model and Color Recognition Using Convolutional Neural Network". Archived from the original on 30 May 2018. Retrieved 30 May 2018 – via ResearchGate.

[16] "OpenALPR Benchmarks". openalpr.com. 31 October 2017. http://www.openalpr.com/benchmarks.html

[17] Laroca, Rayson; Severo, Evair; Zanlorensi, Luiz A.; Oliveira, Luiz S.; Resende Goncalves, Gabriel; Robson Schwartz, William; Menotti, David, “A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector “International Joint Conference on Neural Networks (IJCNN). pp. 1–10. arXiv:1802.09567

[18] Sérgio Montazzolli SilvaEmail, and Cláudio Rosito Jung, “License Plate Detection and Recognition in Unconstrained Scenarios”, European Conference on Computer Vision, ECCV 2018: pp. 593-609, 2018

[19] Kazuo Ohzeki, Takuya Okunuki, Stefan Schneider, “Number Plate Region detection using Luminance and Sobel filter data”, The 18th Meeting on Image Recognition and Understanding (MIRU) DS1-9 pp.1-2 July 2015.

[20] Kazuo Ohzeki, Takuya Okunuki, and Stefan Schneider, "AN ADVANCED NUMBER PLATE DETECTION METHOD FOR REAR-END COLLISION AVOIDANCE SYSTEM” Proc. Irish Transport Research Network (ITRN) Session 5b 27th-28th Aug. 2015.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", Arxiv:1506.01497 Part of: Advances in Neural Information Processing Systems 28 (NIPS 2015)

[22] Yoshikazu kido, Yutaka Hirakawa, Kazuo Ohzeki, “Recognition of Driving Environment by Deep Learning Algorithm using adaptive Environment Model”, Proc. PCSJ/IMPS P-5-15 Nov. 2017.

[23] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Las Vegas June 2016 https://arxiv.org/pdf/1506.02640.pdf

[24] https://www.youtube.com/user/alg0z/videos

[25] https://ipg-automotive.com/products-services/simulation-software/carmaker/

[26] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, "An Introduction to Statistical Learning", Springer Texts in Statistics 2013.

[27] Ron Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection", Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (II) IJCAI-95 Contents Vol 2, Pages 1137-1143 Aug. 1995.

[28] T. Marill and D. M. Green. “On the effectiveness of receptors in recognition systems”, IEEE Transactions on Information Theory, 9:11–17, 1963.

[29] P. Pudil , J. NovoviEova , J. Kittler, “Floating search methods in feature selection”, Pattern Recognition Letters 15 (1994) 1 1 19-1 125.

[30] Nobuyuki Otsu, ”An Automatic Threshold Selection Method Based on Discriminant and Least Squares Criteria”、ieice transaction D Vol.J63-D no.4 pp.349-356 April. 1980.