Speed Traffic-Sign Number Recognition on Low Cost FPGA for Robust Sign Distortion and Illumination Conditions

Masaharu Yamamoto[1)], Anh-Tuan Hoang[2)], Tetsushi Koide[1)2)]

1) Graduate School of advanced Sciences of Matter, Hiroshima University
2) Research Institute for Nanodevice and Bio Systems (RNBS), Hiroshima University
1-4-2 Kagamiyama, Higashi-Hiroshima-shi, Hiroshima, 739-8527 Japan
{yamamoto-masaharu, anhtuan, koide}@hiroshima-u.ac.jp

Abstract - In this paper, we propose a hardware-oriented robust speed traffic-sign recognition algorithm which can process real-time for Advanced Driving Assistant System (ADAS). In difficult conditions, such as sign distortion in various angle or at night and rain, the proposed algorithm is still be able to recognize the traffic sign with high precision. The proposed hardware oriented number recognition algorithm achieves more than 99 % in recognition rate in daytime and achieves 94.2 % including difficult conditions in rainy night.

I. Introduction

The traffic sign recognition would be very important in the Advanced Driver Assistance System (ADAS) for vehicles [1]. By 2020, almost 25 million cameras are expected to be fitted in to cars in Western Europe each year, with annual revenues surpassing €2 Billion. Much of this growth is being driven by the inclusion of Autonomous Emergency Braking (AEB) and Lane Departure Warning (LDW) into the EuroNCAP program from 2015 onwards, which is in turn pushing vehicle, manufactures to consider lower-cost ADAS sensors [2, 3]. There are three key trends related to functionality, as shown in Figure 1. The most important information is provided in the driver's visual field by the road signs, which are designed to assist the drivers in terms of destination navigation and safety. The most important of a car assistant system is to improve the driver's safety and comfort. Detecting the traffic signs can be used in warning the drivers about current traffic situation, dangerous crossing, and children path. An assistant system with speed limitation recognition ability can inform the driver about the change in speed limit as well as notify them if they drive at over speed. Hence, the driver's cognitive tasks can be reduced and safe driving is supported. However, meeting real-time (frame rate: 15 ~ 30 fps) performance for such a system is still a big challenge research, especially in compact hardware size. Our research targets to a smart and compact high performance speed traffic-signs recognition system on consumer oriented low price device: Xilinx Zynq 7020 platform [4].

Several hardware implementations or hardware accelerations for traffic sign recognition system have been introduced on FPGA. Yan Han [5] introduces a hardware/ software implementation based on Xilinx Zynq SoC. Souani [6] implemented a neuron network for shape recognition on RGB image on Virtex 4 device. Imark [7] implemented on FPGA a system with edge detection, circle detection, triangle detection and rectangle detection based on angle before matching features of templates and candidate for recognition.

In this paper, the number recognition hardware oriented algorithm of speed traffic-signs is proposed using feature quantity suitable for hardware processing. Highly precision recognition capability is possible also for the sign of a difficult distortion and illumination conditions such as rainy night and so on.

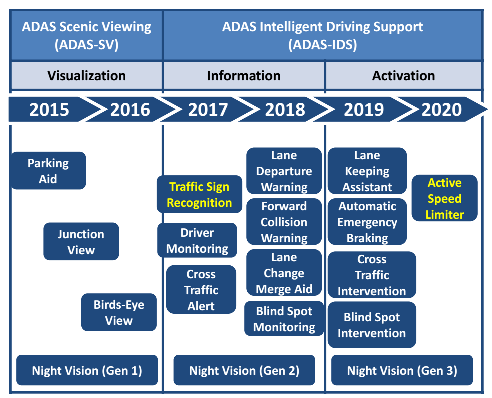


Fig. 1. Camera-Based ADAS Roadmap in EU [2, 3].

II. System Outline

Figure 2 shows the flow chart for the speed limit recognition system, which recognizes the limit speed from 640x360 pixel images (Full HD (1920x1080 pixel) images are also used together for high precision rate at night situations which are difficult illumination conditions).

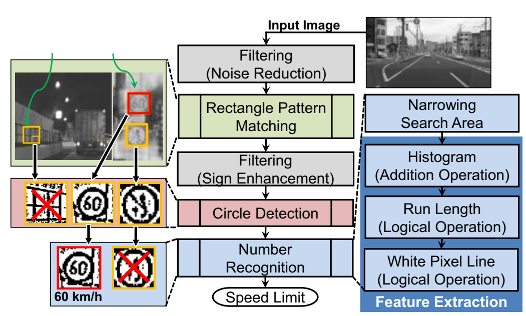


Fig. 2. Flow chart of the proposed system.

The each step is as follows:

(1) *Input Image*: Although the images taken with in-vehicle camera are Full HD images (High End Video Camera), we verify by converting from color into 8-bit grayscale images in this simulation. Moreover, in addition to Full HD images, we use 640x360 pixel images as shown in Figure 3 (a), (c). The image of 640x360 pixel (about 230,000 pixels) is carried out by dawn sampling of original image (1920x1080 image). Hence, the system supports both low cost and high end image sensors.

(2) *Noise Reduction Filter*: This conversion also applies a Mean filter to reduce noises. In addition, the image of 640x360 pixel is carried out by dawn sampling of the Full HD image (if used).

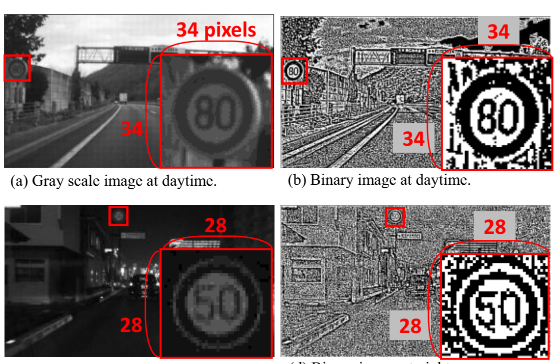
(3) *Rectangle Pattern Matching (Rough circle detection)*: Speed traffic-sign areas are searched in the gray scale image by our Rectangle Pattern Matching (RPM) [8]. Areas, which are detected as traffic-signs, are defined as Scan Window (SW).

(4) *Sign Enhancement Filter*: The 8-bit gray-scale image is converted into binary image as shown in Figure 3 (b). This conversion also applies our proposed special sign enhancement filter to increase features of the sign.

(5) *Circle Detection*: Circle is searched in the SW in order to raise accuracy using Circle Detection (CD) algorithm [8].

(6) *Number Recognition*: The Number Recognition (NR) [9] is used to define the number located inside the sign candidate areas. This module analyzes the binary image to find the limit speed. This paper emphasizes to explain about the Number Recognition. The example of "Speed-Signs" and "Not Speed-Signs" are shown in Figure 4.

(7) *Output Limit Speed*: The recognized limit speed is given to the assistant system to notify the driver if necessary. This system notifies controllers and drivers about speed limit.



(a) Gray scale image at daytime. (b) Binary image at daytime. (c) Gray scale image at night. (d) Binary image at night.

Fig. 3. Gray scale images and binary images after our proposed special sign enhancement filtering.

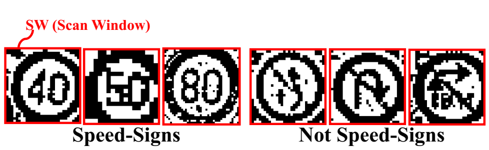


Fig. 4. The example of "Speed-Signs" and "Not Speed-Signs".

III. Histogram Feature of the Sign-Number and Recognition Algorithm

A. Feature in Run Length of Black and White Pixels

There are common features of "0" number, in which the middle of number "0" (the area is defined as R0 in Figure 5) has run length white pixels and the two vertical edges of "0" has run length of black pixels. Middle of the number is analysis for continual black and white pixels. If the features are met, the number is "0" as shown in Figure 5.

B. Feature in Existence of the Vertical White Line in Blocks

If each number is divided into four blocks as shown in Figure 5, existent and location of the *white lines* can be used as feature for the numbers recognition. However, with this feature quantity only, when there are many noises available as shown in number "7" in Figure 5, a number may be recognized incorrectly.

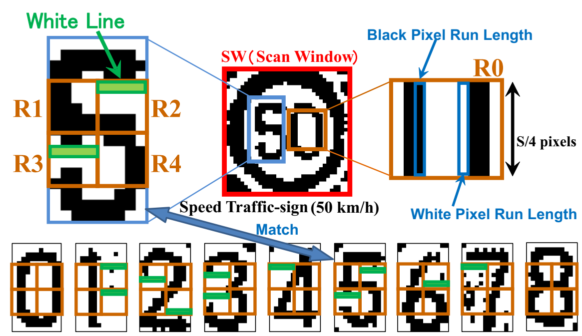


Fig. 5. Run length and white pixel line features.

C. Feature in Histogram of the Number

Histogram of the sign candidate area is calculated in vertical and horizontal axes and divided into 7 blocks (H1 ~ H7) with overlapped as shown in Figure 6 (a). Depends on the number, the maximum and minimum of the histogram in each block (H6 and H7) are different, and can be used for number recognition.

D. Feature in Rate of Black Pixels on the Search Area

The rate of black pixel of the number is different from another. It can be used in number recognition. If the rate of black pixels in the search area matches the condition in Figure 9, the search area can be considered as speed traffic-sign.

IV. Number Recognition Algorithm

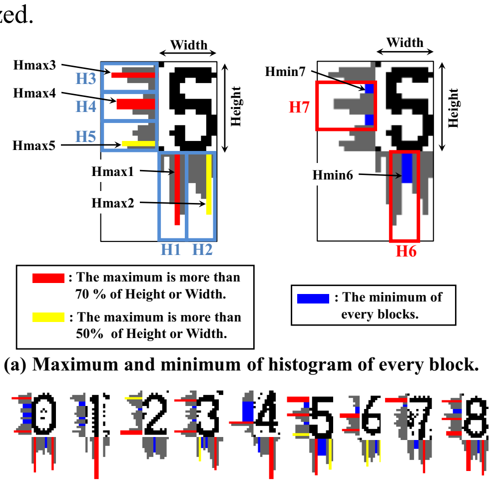
The input images in "Number Recognition" are binary images with the size in range from 20x20 to 50x50 pixels. However, the size actually used in consideration of overlap is only 14 window sizes (20, 21, 22, 23, 24, 26, 28, 30, 32, 34, 38, 42, 46 and 50 pixels).

The *Number Search Area* is defined as the area inside Scan Window (SW) that the numbers are located and is decided by narrowing down from the sign size detected at Sign Detection as shown in Figure 7 (a). The number features inside the *Search Area* are then extracted. If SW size is defined as “SxS pixels.” Size of the *Search Area* relies on the size of the SW as shown in Figure 7 (b). The maximum size of *Search Area* is 27 pixels corresponding with the biggest SW size of 50x50 pixels. In order to increase the accuracy, the surrounding area with the original *Search Area* is also processed. Hence, the original *Search Area* is extended to four other derivative areas by shifting the original to the left, right, up and down as shown in Figure 8. The number successfully recognized among those 5 *Search Areas* is defined as the speed. Since all the speed traffic-sign in Japan, ended with “0,” a sign candidate is considered as speed traffic-sign if the above features of “0” number are met at the right side of the *Search Area.*

Existence feature of vertical *white line* in 4 blocks R1 ~ R4 in Figure 5 is confirmed and compared with the pre-defined feature quantity of the sample numbers to find which number is matched. Together with the *white line* feature verification, histogram of the 5 *Search Areas* are computed in horizontal and vertical and compared in each blocks H1 ~ H7 to find the maximum and minimum.

Two flags are used for maximum checking of histogram in each block (H1 ~ H5). The first one will be set if the maximum histogram gets over 70 % of the Height or Width of the number as shown in Figure 6 (a). The other will be set if the maximum of histogram in a block is over 50 % but smaller than 70 % of “Height” or “Width”. Example about histogram pattern of a speed sign numbers is shown in the Figure 6 (b). Similarly, the minimum of the histogram of blocks H6 and H7 are also a feature for number recognition.

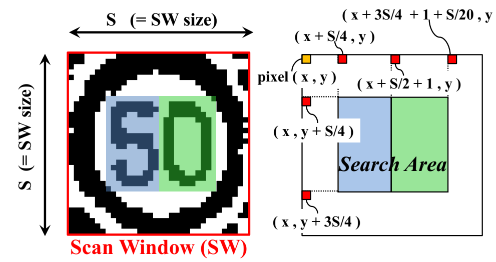
Figure 9 shows the histogram features, existent of the *white line* (white pixel run length) features and the rate of black pixels in *Search Area* features. The features extracted from the *Search Area* are compared with the standard features in Figure 9 to find the match. By that, the speed number is recognized.



(a) Maximum and minimum of histogram of every block.

(b) Sample pattern of histogram of numbers.

Fig. 6. Feature in histogram of each speed sign number.



(a) *Search Area* inside SW (b) Size of the *Search Area*

Fig. 7. Definition of *Number Search Area* in Scan Window.

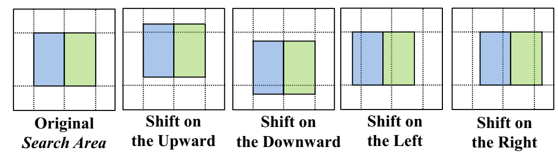


Fig. 8. Five derivation *Number Search Areas* created from the original *Search Area* by shifting.

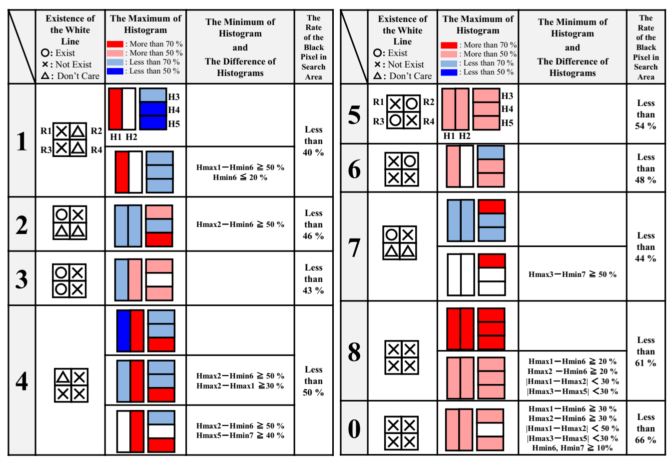


Fig. 9. Standard features for the number recognition.

V. Simulation Result

A. Benchmarking of Test Bench Images

We use two data sets in algorithm verification. The first one, the artificial data set is used to verify the robustness of the algorithm with rotation of the signs. The second one taken by the in vehicle camera in real life traffic environment is used to evaluate the recognition accuracy of a speed sign in an actual road situation.

Depends on the distance and relative location between the car and the sign, the appearance of the sign in the camera is distorted in 3 axes shown in Figure 10. The number of correctly recognized SW is calculated for precision verification. A single traffic sign can be detected by many SWs. It is considered correctly recognized if the correct speed number can be given in one among them. When a sign inclines in x axis as shown in TABLE I, it is distorted in y axis and became shorter. When it inclines in y axis, its image is distorted in x axis, and became narrower. If it rolls in z axis, the number inside is also rotated. The original signs, which have no incline in any axis, have incline degree of $(\theta x, \theta y, \theta z) = (0, 0, 0)$, in which. $\theta x, \theta y$ and θz are angle of inclination of a sign in axes x, y and z. We perform simulation with sign that inclines in various degrees in the three axes as shown in (1), (2) and (3). The number of SW with correctly recognition is filled in each cell in Table 1. The symbol "x" indicates the taller recognition.

(1) The x axis distortion angle: $\theta x = 0 \rightarrow 45$ deg.
(2) The y axis distortion angle: $\theta y = 0 \rightarrow 45$ deg.
(3) The z axis distortion angle: $\theta z = -20 \rightarrow 20$ deg.

If the signs (with all speeds from 10 to 80) incline in x and y axes less than 35 degree, they are recognizable. Table 1 also shows that sign inclined in x axis is recognized. When the inclination of a sign gets over 40 degree, it is unrecognizable. The size of the test bench signs are 30x30 pixels.

TABLE I
The Result of the Recognition for Each Axis of Distortion.

Distortion [deg]		10	20	30	40	50	60	70	80
θ_x	0 ~ 30	15	32	24	36	24	16	34	24
	35	1	2	1	1	2	2	2	2
	40	×	×	×	×	×	×	×	×
θ_y	0 ~ 20	9	14	21	24	16	11	13	7
	25	1	5	3	4	5	1	3	4
	30	1	2	3	1	×	1	×	1
	35	×	1	1	×	×	×	×	×
	40	×	×	×	×	×	×	×	×
θ_z	-20	×	×	×	×	×	×	×	×
	-18	×	×	×	×	×	×	2	×
	-16	×	×	×	×	×	1	8	2
	-14	×	5	×	×	×	5	7	2
	-12	×	3	4	×	×	5	9	8
	-10	×	2	8	×	1	5	8	6
	-8	2	3	3	×	2	3	9	7
	-6 ~ 10	29	60	48	98	53	25	63	75
	12	×	4	4	×	3	1	4	5
	14	×	×	×	×	×	×	8	3
	16	×	×	×	×	×	×	2	2
	18	×	×	×	×	×	×	×	×

High tolerance is guaranteed when the sign distorted in x and y axes. Our method is sensitive with distortion in z axis. It can correctly recognize all the speed signs if they incline in a range from -6 degree to 10 degree.

The reason is when number "40" in a speed traffic sign inclines in z axis, it is incorrectly recognized as narrow "80" number. However, the accuracy in number recognition for a sign with inclination in z axis is sufficient for practical usage as shown in the next paragraph.

B. Benchmarking of the Images Captured by an in-Vehicle Camera

(1) *False Positive Rate of Frames*

The number of frames which are incorrectly recognized speed limit during the simulation is counted. The false positive rate of frames is called "frame incorrect recognition rate (FIR_Rate)" and is calculated by equation 1.

$$\text{FIR\_Rate [\%]} = \frac{\text{\# of False Recognition Frames}}{\text{187,156 Frames}} \times 100 \quad \text{(Eq. 1)}$$

(2) *True Positive Rate in Scenes*

In the simulation, we "one scene" as extract contains 30 continuous frames, as shown in Figure 10, in which the same sign appeared in the observation field of the in-vehicle camera until it disappeared, and the rate of scenes which are successfully recognized is defined as a "scene positive recognition rate (SPR_Rate)" and is calculated by equation 2.

$$\text{SPR\_Rate [\%]} = \frac{\text{\# of True Recognition Scenes}}{\text{194 Scenes}} \times 100 \quad \text{(Eq. 2)}$$

Since a speed sign can be recognized in any frame among the 30 frames in the scene, we considered that the scene is correctly recognized if the correct speed limit can be extracted from one frame among them. The difficulty level (Level 1 ~ 4) of sign recognition is defined and the recognition result at given difficulty is shown in TABLE II and Figure 11. The yellow arc in Figure 11 indicates a sign.

- **Level 1:** The sign at a bright place of daytime.
- **Level 2:** The sign at the sign at a dark place of daytime,.
- **Level 3:** The sign at a bright place at night.
- **Level 4:** The sign which is especially hard to be seen.

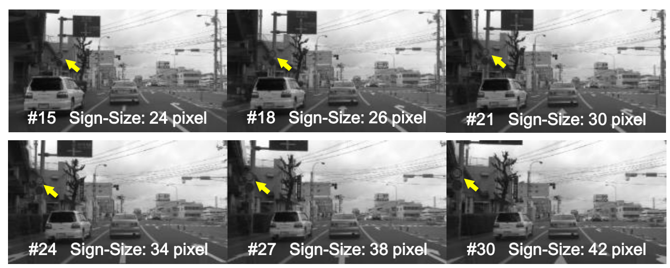


Fig. 10. Example of one Scene (1 scene = 30 frames).

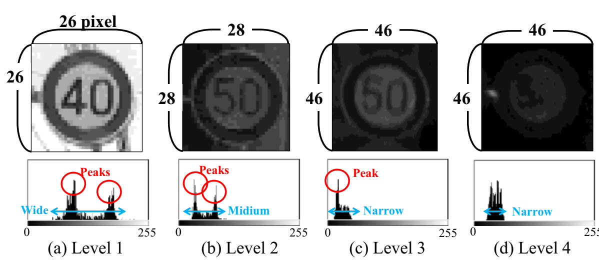


Fig. 11. Example of the difficulty level (Level 1 ~ 4).

Comparison of the recognition accuracy with related research and this research is shown in TABLE III. "Daytime 1", "Daytime 2", "Night" and "Rainy Night" are captured with a practical low price gray scale camera for production level driver assistance system. "Daytime 3" and "Daytime 4", which are captured a high-end color camera, are referred in Ref. [11] and [12], respectively. In 640x360 pixels images in daytime, SPR_Rate is 100 %. SPR_Rate reduces to 84.0 % in 640x360 pixels image at night. However, in the night scene in Full HD image, the SPR_Rate improves to 94.2 %. Concerning to sign recognition in "Rainy Night", our system is incorrectly recognizing number in 4 scenes. Moreover, as shown in Figure 13, speed numbers in these 4 scenes is recognizable with newly proposed contrast adjustment.

TABLE II
The Result of Sign Recognition Classified by Difficulty Level.

Data Set (# of Frames)	Camera	Input Image [pixel]	# of Scenes Classified by Difficulty Level 1	2	3	4	Total
Daytime 1 (41,120)	Gray Scale Cam*1	**640x360**	50/50	10/10	—	—	60/60
Daytime 2 (40,000)	Video Cam*2	**Full HD**	62/62	3/3	—	—	65/65
		640x360	62/62	3/3	—	—	65/65
Night (39,136)	Video Cam*2	**Full HD**	—	10/10	15/15	—	25/25
		640x360	—	10/10	14/15	—	24/25
Rainy Night (66,880)	Video Cam*2	**Full HD**	—	9/9	31/32	0/3	40/44
		640x360	—	9/9	25/32	0/3	34/44

*1 Gray scale camera (640x360 pixel) [10]. *2 60 fps Interlace Full HD Color Camera.

TABLE III
Comparing with Related Works.

Method	Data Set	Input Image	FIR_Rate	SPR_Rate	Speed
This Work	Daytime 1 Daytime 2	640x360 (8 bit Gray Scale)	$\frac{32}{81,140}$	100 % $\left(\frac{125}{125}\right)$	15 fps > ($\frac{60}{4}$ photos) (FPGA*5)
	Night **Rainy Night**	640x360 (8 bit Gray Scale)	$\frac{46}{106,016}$	84.0 % $\left(\frac{58}{69}\right)$	
	Night **Rainy Night**	Full HD (8bit Gray Scale)	—	94.2 % $\left(\frac{65}{69}\right)$	
SIFT*3 [11]	Daytime 3	Full HD (24 bit Color)	$\frac{16}{215,200}$	90.4 % $\left(\frac{359}{397}\right)$	0.67 fps (CPU*6)
Hough*4 [12]	Daytime 4	Full HD (24 bit Color)	—	91.4 % $\left(\frac{533}{583}\right)$	6.7 fps (CPU*7)

*3 Scale-Invariant Feature Transform. *4 Hough transform
*5 Xilinx Zynq-7020 [3]. *6 Intel Pentium D 3.2 GHz. *7 Intel Pentium 4 2.8 GHz.

C. Discussion to the Simulation Result

Experiment result on Figure 12 (a) shows that contrast adjustment is necessary for night scene to increase the recognition. In order to get the best contrast adjustment, luminosity histogram distribution is investigated to the scene at night. The result shows that, in the proposed system, it is possible to double the luminosity value of each pixel to effectively adjust the contrast of a sign possible to process in. Scene the luminosity value of each pixel needs to be double, it is one scan by only shift operation.

D. Discussion to Highway and Local Roads

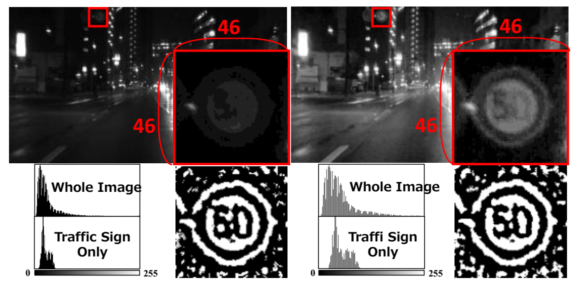
A scene consists several frames, in which the same speed traffic-sign appears on them. The Scan Window (SW) recognition rate of speed traffic-sign is defined in Equation 3, and is computed on each scene.

$$\text{SW Recognition Rate [\%]} = \frac{\text{\# of Correct SW}}{\text{\# of All SW}} \times 100 \qquad \text{(Eq. 3)}$$

"# of all SW" is number of traffic-sign recognized by the Number Recognition and "# of correct SW" is number of traffic-sign correctly recognized for limited-speed. The SW recognition rate of the scenes shown on Figure 11 is shown in Figure 13 (a) and that of a highway scene is shown in Figure 13 (b). When the camera gets closer to the sign, the size of speed traffic-sign become bigger and the SW recognition rate increase and gets 100 %. The number of frame which becomes "SW recognition rate = 100 %" is few in case of a highway as shown in Figure 13 (b), so the recognition of traffic-signs is more difficult but the proposed system can be recognized.

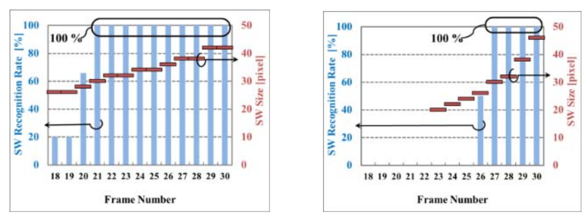
E. Discussion to the Recognition of LED Speed Signs

Figure 14 shows a LED speed sign. It shows that the color of the signs reverted in the binary image in composition with that of non LED sign. Hence, we can simply revert the color of the LED sign for recognition process. The color reverted speed LED sign can be seen in Figure 14 (c), which has same condition with other non LED sign.



(a) Before contrast adjustment. (b) After contrast adjustment .

Fig. 12. Contrast adjustment in dificult conditions (Difficulty level: 4).



(a) Local Road (Speed: 50 km/h). (b) Highway (Speed: 80 km/h).

Fig. 13. Simulation result of SW recognition rate and SW size.

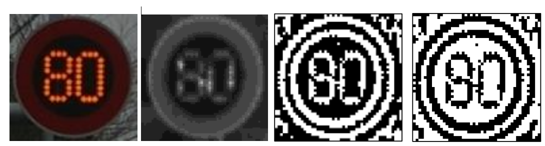

(a) LED Speed Sign in Japan (b) Sign Enhancement Filtering (c) B/W Reverse

Fig. 14. LED Speed Sign Recognition.

VI. Hardware Architecture

A. Architecture Overview

Figure 15 shows the architecture of the speed limit recognition system with number recognition inside. The system is able to scan for the traffic signs up to 50x50 pixels in size. The input image is 8-bit grayscale 640x390=249,600 pixels. It contains two main modules of Rectangle Pattern Matching (RPM) and the Number Recognition (NR). Other modules are Noise Reduction Filter, Sign Enhancement Filter, and Circle Recognition. Supported for the Noise Reduction Filter and the RPM are a number of 8-bit FIFOs. The resource amount of consumption of this system will be 6.4 % of BRAM, 39.3 % of Flip-Flops and 68.6 % of 6-input LUTs, and it can also be mounted on low price/compact device such as Xilinx Zynq-7020 sufficiently.

B. Number Recognition Architecture

The number recognition algorithm shown in section IV is one time image scanning. By scanning one line of the scan window at 2 clocks from the top to the bottom, the number recognition module can recognize the number inside *Search Area* within 100 clocks in the worst case. It extracts and verifies the feature quantity of number shown in section III (Histogram and run length of black pixels), and generates the result. Since 1 pixel in SW occupies 1 bit, max of the number in bit of one SW line is 50 bits. For this reason, the 64-bit block RAM can be used as binary image memory and one line scanning takes 1~2 clocks.

The information about location and size of traffic sign candidates are stored in Location and SW flag FIFO. Based on the information, the Number Recognition module reads the corresponding image area from Binary Image Memory, since reading 1 line from binary image memory may takes 2 clocks, the number recognition module processes 1 line by 2 clocks. As shown in Figure 16, in the number recognition module, the feature extraction processing for all SW sizes in a search area (50x50, 46x46, ... , 20x20 pixels) are performed in parallel. All the search area at a position (Top, Original, Right, Left, Bottom) are also investigated in parallel for number recognition. After matching the feature extracted from input image with the number standard features the speed limit is given out as 8 bits, one corresponds to a speed in range of 10 ~ 80 km/h (8 cases). Detected speed results of all candidates in a frame are combined and given out at the end of the detection process of that frame. As shown in Figure 17, feature extraction processing is Run length of a pixel and histogram of the number area. Since the histogram of a pixel is calculated using 1-bit addition and logical operation only, it is suitable implementation on low-cost FPGA.

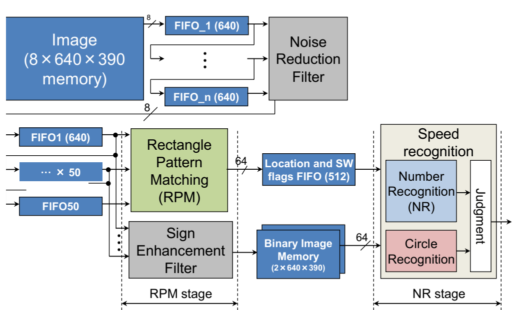


Fig. 15. Architecture overview of the speed sign recognition system.

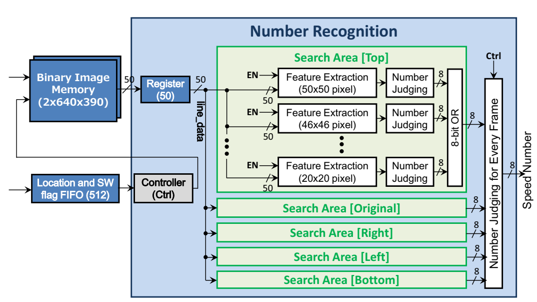


Fig. 16. Number recognition architecture.

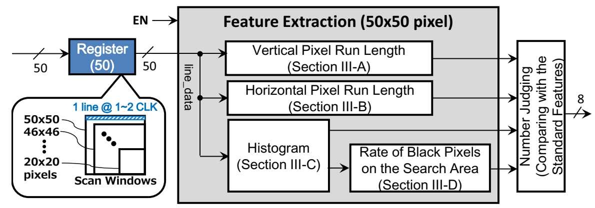


Fig. 17. Feature extraction architecture (50x50 pixels).

VII. Conclusions

The hardware oriented algorithm on number recognition for speed traffic-sign number recognition has been developed. The simulation result shows that the proposed method achieves more than 99 % in recognition rate in daytime and achieves 94.2 % include difficult conditions such as the speed sign in rainy night.

Acknowledgements

Part of this work was supported by Grant-in-Aid for Scientific Research (C) and Research (B) JSPS KAKENHI Grant Numbers 2459102 and 26280015, respectively.

References

[1] T. Hummel et al., "Advanced Driver Assistance Systems", The UDV (German Insurers Accident Research), http://udv.de/sites/default/files/tx_udvpublications/RR_FS03_AdvancedDriverAssistanceSystems_01_0.pdf, 2014-10-17 accessed.

[2] Secured by Design (SBD) Ltd, "A camera for every occasion: Technical & market trends for camera-based ADAS in EU", http://www.sbdjapan.co.jp/wp-content/jpnews/pdfs/524_Report_Sample_J.pdf, 2014-12-14 accessed.

[3] Euro NCAP, "European new car assessment program", http://www.euroncap.com/files/Euro-NCAP-2020-Roadmap---June-2014-2---0-e11c0984-af94-420e-9d63-63edc8538745.pdf, 2015-01-25 accessed.

[4] Xilinx HP, Zynq-7000 All Programmable SoC Overview, http://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf, 2014-10-14 accessed.

[5] Y. Han, "Real-time traffic sign recognition based on Zynq FPGA and ARM SoCs", *Proceeding of the 2014 IEEE International Conference on Electro/Information Technology (EIT)*, pp. 373-376, 2014.

[6] C. Souani, H. Faiedh, and K. Besbes, "Efficient algorithm for automatic road sign recognition and its hardware implementation", *Journal of Real-Time Image Processing, Vol. 9, Issue 1*, pp. 79-93, 2014.

[7] H. Irmak, "Real Time Traffic Sign Recognition system on FPGA", Master Thesis, *The Graduate School of Natural and Applied Sciences of Middle East Technical University*, 2010.

[8] A. T. Hoang, et al., "Pipeline scanning architecture with computation reduction for rectangle pattern matching in real-time traffic sign detection," *Proceeding of the IEEE International Symposium on Circuits and Systems (ISCAS2014)*, pp. 1532-1535, 2014.

[9] M. Yamamoto, et al., "Compact hardware oriented number recognition algorithm for real-time speed traffic-sign recognition," *Proceeding of the IEEE International Symposium on Circuits and Systems (ISCAS2014)*, pp. 2535-2538, 2014.

[10] G. Karanam, "Interfacing Red/Clear Sensors to ADSP-BF609 Blackfin Processors," http://www.analog.com/static/imported-files/application_notes/EE358.pdf, Rev 1–May 28 2013, 2014-10-15 accessed.

[11] M. Takagi, et al., "Road sign recognition using SIFT feature", *Proceeding of the transactions of the Institute of Electrical Engineers of Japan. C, A publication of Electronics*, 129(5), 824-831, 2009-05-01.

[12] Y. Ishizuka, et al., "Recognition system of road traffic signs using opponent-color filter", *Proceeding of the IEICE technical report. Pattern recognition and media understanding*, 103(737), 13-18, 2004-03-11. (in Japanese)