An Active Vision System for Real-Time Traffic Sign Recognition *

Jun Miura, Tsuyoshi Kanda, and Yoshiaki Shirai
Department of Computer-Controlled Mechanical Systems, Osaka University
jun@mech.eng.osaka-u.ac.jp

Abstract

This paper presents an active vision system for real-time traffic sign recognition. The system is composed of two cameras, one is equipped with a wide-angle lens and the other with a telephoto lens, and a PC with an image processing board. The system first detects candidates for traffic signs in the wide-angle image using color, intensity, and shape information. For each candidate, the telephoto-camera is directed to its predicted position to capture the candidate in a larger size in the image. The recognition algorithm is designed by intensively using built-in functions of an off-the-shelf image processing board to realize both easy implementation and fast recognition. The results of on-road real-time experiments show the feasibility of the system.

1 Introduction

Traffic signs provide the driver various information for safe and efficient navigation. Automatic recognition of traffic signs is, therefore, important for automated driving or driver assistance systems [1] [4].

Since these signs are usually painted by distinctive colors, they may be easily detected using color information [3]. However color information is sensitive to the change of weather or lighting condition and, therefore, it is sometime difficult to extract traffic signs reliably only by color. In addition, in urban cluttered scenes, other sign boards or buildings with the similar color to traffic signs may make it more difficult to extract an appropriate region for the target signs in the image. To cope with such problems, we additionally use shape information [2].

To recognize traffic signs, it is often necessary to recognize their characters and symbols. For reliable recognition of characters and symbols, it is desirable to capture a sign in a large size in the image. To do this, we use a camera with a telephoto lens and direct it to the target. The camera direction is automatically determined based on the prediction of the target motion in the image.

This paper describes an active-vision system that can recognize various traffic signs in real-time based on the above-mentioned approach. We design the recognition algorithm by intensively using built-in functions of an off-the-shelf image processing board in order to realize both easy implementation and fast recognition.

*Proc. 2000 IEEE Int. Conf. on Intelligent Transportation Systems, pp. 52-57, Dearborn, MI, Oct. 2000.

2 System Overview

Fig. 1 illustrates the system configuration. The system has two cameras, one is equipped with a wide-angle lens (called *wide-camera*) and the other with a telephoto lens (called *tele-camera*), and a PC with an image processing board. The wide-camera is directed to the moving direction of the vehicle. The tele-camera can change the viewing direction to focus the attention to the target sign.

The recognition process is composed of the following steps.

1. Detect candidates of traffic signs using color and intensity information.
2. Screen candidates using edge information.
3. Predict the motion of a screened candidate in the image and direct the tele-camera to it.
4. Extract characters and symbols (if necessary).
5. Identify the sign by pattern matching.

The following sections describe these steps. We use as examples two kinds of traffic signs: one is for indicating a speed limit (we call it *speed sign*) and the other is for guiding routes (we call it *guidance sign*).

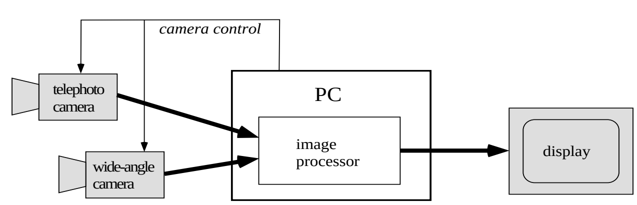


Fig. 1: System overview

3 Detecting Sign Candidate Regions

3.1 Extracting Candidates by Color and Intensity

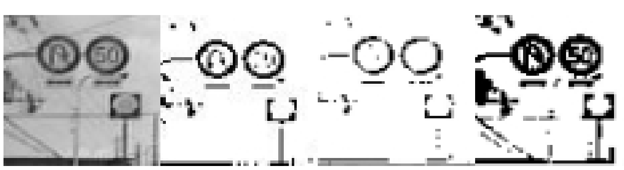
A speed sign is characterized by its circular shape and a white region with a red boundary (see Fig. 2(a)). One may think that the red boundary region is used as the cue for detection. However, since the red region has low intensity values, color information is often unreliable. Therefore, we use the white circular region as the first cue for detection. Candidate white regions are detected by binarization with *area filtering*, which only keeps white regions whose areas are within a predetermined range. Since binarization is sensitive to the threshold, as shown in Fig. 2, we perform binarization multiple times using different thresholds and search every binarized image for candidate white regions. Although this multiple-thresholds approach is for not missing any candidates, at the same time it may detect many candidate regions that do not come from the target signs. So an appropriate screening process must follow.

A guidance sign is characterized by its rectangular shape and blue color with white characters and symbols (see Fig. 11(a)). As in the case of speed signs, we perform binarization with area filtering in the color image space multiple times using different thresholds. The thresholds are determined by analyzing the actual distribution of data in the YUV color space taken in various weather and lighting conditions.

3.2 Screening Candidates by Edges

For screening candidates using shape information, we set a search area around each candidate region detected at the previous step and extract edges in the area to see if the edges form a specific shape.

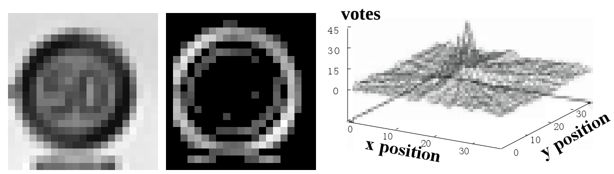
For circular shapes such as speed signs, we perform the following processing. If an edge is a part of a circle, the center of the circle should exist on the line which passes the edge and has the same direction as the gradient of the edge. Using this fact, we calculate such a line for each edge and vote the line in the search area. If there is a prominent peak in the area, we determine that a circle exists. Fig. 3 shows an example of circle detection. Fig. 4 shows a detection result of speed sign candidates screened using shape information.



(a) input (b) good (c) too low (d) too high

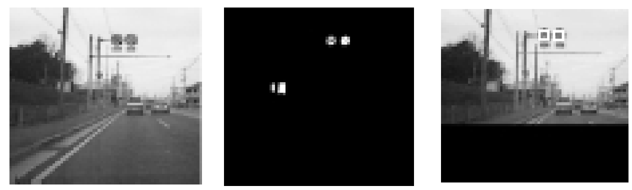
Fig. 2: Binarization results using various thresholds.

For rectangular shapes such as guidance signs, we detect four boundary line segments. As shown in Fig. 5, we extract vertical (horizontal) edges, project them to the x axis (y axis) to generate a histogram, and detect peaks in the histogram in order to determine boundary lines. Sometimes four surrounding line segments are not fully detected due to, for example, strong edges of a near object in the image. In such a case, if we have at least one strong line segment, we predict the position of other segments and try to find them with a low threshold for edge detection. Fig.



(a)search area (b)edge image (c)voting result

Fig. 3: Detecting a circle.



(a) original image (b) before screening (c) after screening

Fig. 4: Detection of speed sign candidates.

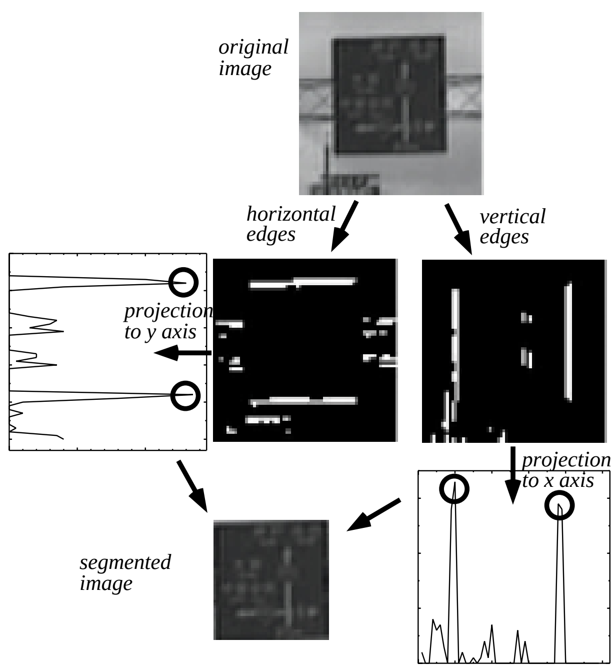


Fig. 5: Detecting four boundary line segments of guidance sign.

6 shows detection results of guidance sign candidates.

Fig. 6: Detection of guidance sign candidates.

4 Controlling Camera Direction

After detecting a candidate by the wide-camera for several times, the tele-camera is directed to it for capturing a zoomed-up image. Since it takes some time to switch the camera and direct the tele-camera, we need to predict the position *in the image* of the candidate at the time of the image input by the tele-camera. Assuming that the vehicle moves straight at a constant speed from the time of candidate detection to the end of identification, prediction is performed as follows.

Since a sign moves on a line passing the F.O.E. (focus of expansion), we only need to predict the position of the sign on the line. We here explain how to predict the horizontal position (on the x axis) of the target (see Fig. 7). In the figure, f is the focal length, m_i is the camera position at the time of image input, x_i is the x position of the target at that time, p_i is the distance between camera positions. d and w are the distance to the target from m_1 in the longitudinal and the lateral direction, respectively. We here show how to predict the position x_3 in the third image from two observations x_1 and x_2.

Let t_i be the time to move by distance p_i. t_1 is calculated by measuring the time of input of the first and the second images. t_2 depends on the time needed for camera control, and can be estimated in advance. Let v be the vehicle speed; we can calculate p_i as $p_i = vt_i$. From the similarity of triangles in the figure, we obtain:

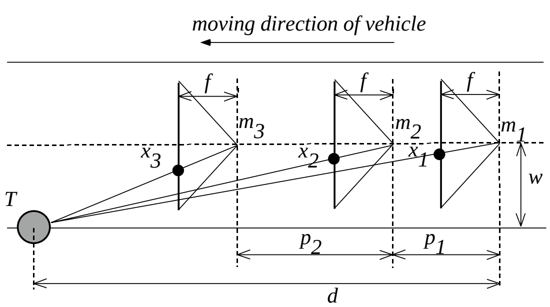


Fig. 7: Predicting target motion.

$$\begin{aligned}
\frac{f}{x_1} &= \frac{d}{w} \\
\frac{f}{x_2} &= \frac{d-p_1}{w} \\
\frac{f}{x_3} &= \frac{d-(p_1+p_2)}{w}.
\end{aligned}$$

By solving these equations, we obtain:

$$x_3 = \frac{p_1 x_1 x_2}{(p_1+p_2)x_2 - p_2 x_1} = \frac{x_1 x_2}{(1+\alpha)x_2 - \alpha x_1},$$

where $\alpha = p_2/p_1$. Since v is constant, $\alpha = t_2/t_1$. This result shows that the target position can be predicted without vehicle speed v and focal length f. From this predicted x position and the equation of the line passing the F.O.E., the direction of the camera (pan and tilt angles) is calculated.

If the tele-camera does not capture the target in a sufficiently large size in the image, the camera continuously tracks the target to obtain a larger target image using a simple tracking strategy. Fig. 8 shows an example sequence of tracking a guidance sign by the tele-camera.

Fig. 8: Tracking a guidance sign by tele-camera.

5 Identification of Signs

5.1 Identification by Pattern Matching

Identification of signs is carried out by a normalized correlation-based pattern matching using a traffic sign image database. Normalized correlation is robust to the change of lighting conditions and is suitable for pattern matching in outdoor scenes. Our image processing board has the built-in function of calculating normalized correlation between a test image and a template image.

Salgian adn Ballard [5] proposed to use responses to be obtained by convolving the input image with a set of *steerable filters* in order to localize the position of a traffic sign. This method is faster than the normalized correlation-based method, but it may be weak for the change of lighting conditions.

Fig. 9 shows the template images of speed signs in the database. We use two thresholds for the correlation value. If the correlation value of the best candidate is above the first threshold and the ratio of correlation values for the

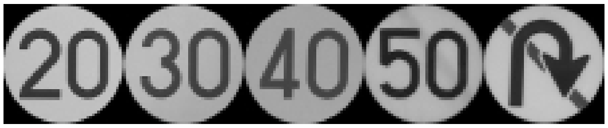
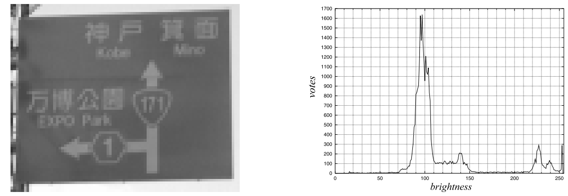


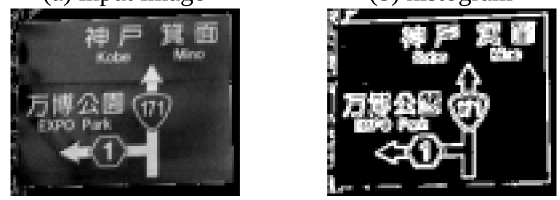
Fig. 9: Template images for speed signs.

(a) detected candidates. (b) recognition result.

Fig. 10: Recognition of speed signs.

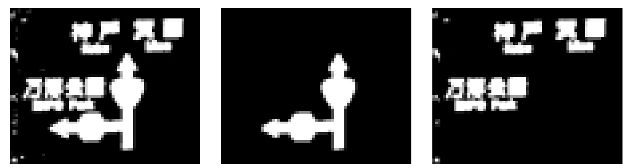


(a) input image (b) histogram



(c) transformed image (d) extracted edges

Fig. 11: Intensity transformation and edge extraction.



(a) filled (b) symbol (c) characters

Fig. 12: Extracting characters and symbols.

best and the second best candidates is above the second threshold, the best candidate is considered to be identified.

Fig. 10 shows a recognition result of speed signs. Fig. 10(a) is the result of candidate detection in the zoomed-up image. The correct templates (recognition result) are displayed at the bottom-left of the Fig. 10(b).

5.2 Extracting Characters and Symbols

Information on a guidance sign is classified into characters (Japanese Kanji and alphabets) and symbols (indicating road structure). Since information is different from sign to sign, we need to segment such characters and symbols for recognition. This segmentation is performed as follows.

If we properly locate a guidance sign region, characters and symbols may be extracted by a simple thresholding using intensity. This extraction method, however, sometimes fails due to a bad lighting condition such as a large intensity gradation over the sign. So we use edges for segmentation. However, from our experience, an appropriate threshold for extracting edges is still sensitive to the change of lighting condition; thus we first perform an intensity transformation based on the intensity histogram of the image.

Fig. 11(a) shows an input image from which characters and symbols are segmented. Fig. 11(b) is the intensity histogram. An input image usually contains a sky region, to which the high-intensity cluster in the histogram corresponds. The biggest cluster corresponds to the guidance sign. Let I_{min} and I_{max} be the minimum and the maximum intensity of the cluster. We perform an intensity transformation, from I_{org} to I_{new}, using the following equation:

$$I_{new} = \begin{cases} 255 \cdot \dfrac{I_{org} - I_{min}}{I_{max} - I_{min}} & \text{if } I_{min} \leq I_{org} \leq I_{max}, \\ 0 & \text{otherwise.} \end{cases}$$

Fig. 11(c) is the image after transformation and Fig. 11(d) is the extracted edges. Thanks to this intensity transformation, we can extract characters and symbols fairly stably using one threshold in various lighting conditions.

After eliminating edges around the sign and filling small holes, we perform a labeling on the remaining regions. Since a symbol usually forms a large region, we separate symbols from characters using size information (see Fig. 12).

Each character region is segmented as follows. Since characters are aligned almost horizontally in the image, we first project the character regions onto the vertical axis to calculate the histogram. Next we search the histogram for prominent peaks that indicate vertical position ranges where characters exist. Then for each range, we project the regions onto the horizontal axis to calculate the histogram, and to determine character positions. Fig. 13 shows the result of character extraction.

Identification of each character is performed by the normalized correlation-based matching, as in the case of speed sign recognition. Fig. 14 shows a set of template images

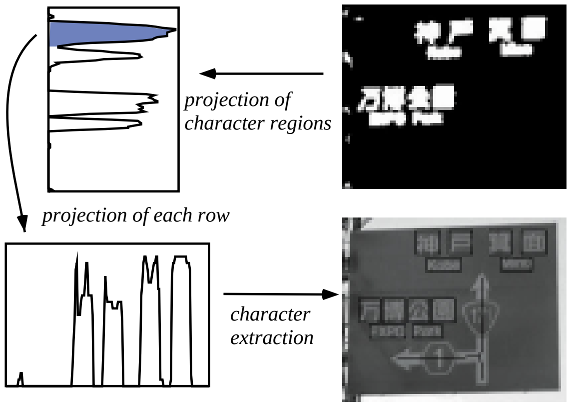


Fig. 13: Result of character extraction.

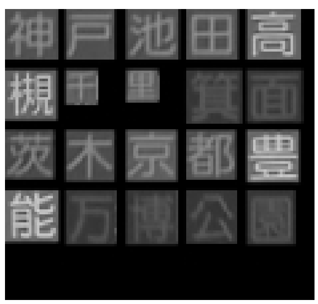


Fig. 14: Template images of characters on guidance signs. These templates are extracted from a videotaped image sequence.

of characters. In this identification, we can also use the knowledge of possible names of destination; that is, a pair (or set) of neighboring characters should make sense as the name of an actual destination. For example, “Kyoto” is represented by two kanji characters “京” (Kyo) and “都” (to); thus, if we have found “京”, its right-side character is most likely to be “都”. We have made a database of such destination names. The high-scored name is selected as the final identification result.

Fig. 15 shows two examples of on-line character identification. The identified characters are superimposed on the position of the corresponding character region. Concerning symbols indicating road structure, the skeltons of symbol regions are extracted.

Although we have not tried to recognize alphabets on guidance signs, the same method can basically be applied to alphabet recognition.

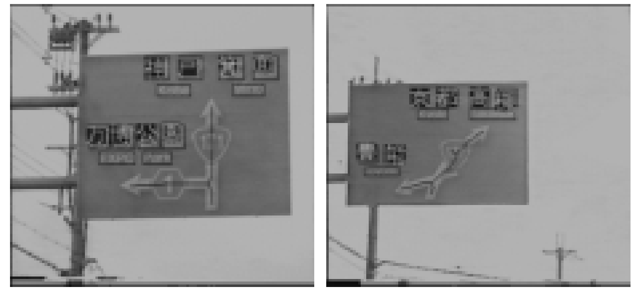


Fig. 15: Two examples of on-line character recognition.

6 Implementation and Experiments

The experimental system is composed of a PC (Pentium II (400 MHz), Linux 2.2.13), an image processing board (IP-5000 by Hitachi), and two cameras (EVI-G20 by SONY). This camera has a zoom lens; one camera uses a wide-angle position (the horizontal viewing angle is 45 degrees) and the other uses the telephoto position (the horizontal viewing angle is 15 degrees).

The recognition algorithm mentioned above has been implemented using over 50 built-in functions of IP-5000. Only circular shape detection using edges (see Fig. 3) is performed on the PC. Table 1 summarizes the processing time of each operation.

We conducted on-road experiments using the system. The results shown below were obtained during two round-cruising between two points; this means the lighting condition was different for two cruising directions.

Table 2 shows the result of recognizing guidance signs. Since the current tracking method by the tele-camera is simple and is not reliable enough, the success rate is low; but this problem will be solved soon. Once a good image was taken by the tele-camera after tracking, the system succeeded in recognizing *all* characters.

Table 3 shows the result of recognizing speed signs. Since, at the time of experiment, we used only one threshold for binarization in the tele-image, if there are several signs in an image and if the optimal threshold is different for each sign, the system detects only one sign and fails to detect the others; this is the primary cause of failure. We expect almost 100% identification rate by using multiple thresholds also for the tele-image.

7 Recognition of Other Signboards

At the roadside, there are a lot of signboards of shops, supermarkets, restaurants, and so on. These signboards can also be useful landmarks for navigation. People often use such information when they teach a route to others. Since it is possible that these shops are not on electronic maps, au-

Table 1: Processing time

contents of processing		time (msec)
speed signs	binarization & labeling	7
	screening by edges (hough transform)	8
	identification (pattern matching)	20
guidance signs	binarization & labeling	7
	screening by edges (edge segment detection)	5
	character and symbol extraction	650
	character identification (pattern matching)	30
processing time for candidate detection per frame		90-170
control of tele-camera + switch cameras		350

Table 2: Result of guidance sign recogntion.

# of signs	# of detection	# of successful tracking	# of characters	# of identified characters
17	17 (100%)	4 (23.5%)	26	26 (100.0%)

Table 3: Result of speed sign recogntion.

# of signs	# of detection	# of identification
71	69 (97.2%)	33 (46.5%)

tomatic recognition of such signboards could be very useful.

Since signboards are, as in the case of traffic signs, characterized by distinctive colors and shapes with clear characters and symbols, the above-mentioned strategy, which combines candidate detecion with the wide-camera and character recognition with the tele-camera, can be applied in a similar way. Fig. 16 shows preliminary experiments of signboard detection in the wide-camera. The signboard in Fig. 16(a) is represented by three rectangles, green, white, and blue ones, aligned vertically. The signboard in Fig. 16(b) is represented by an orange ractangle. We are now investigating character identification of such signboards for more reliable recognition.

(a) (b)

Fig. 16: Recognition of signboards.

8 Conclusions and Discussion

This paper has presented an active vision system for real-time recognition of signboards. The features of the work are:

- Use of multiple thresholds and screening by edges for robust detection of sign candidates in various lighting conditions.
- Active camera control to obtain zoomed-up images of signs for character identification.
- Real-time implementation using only off-the-shelf equipments.

References

[1] S. Azami, S. Katahara, and M. Aoki. Route guidance sign identification using 2-d structural description. In *Proceedings of the 1996 IEEE Symp. on Intelligent Vehicles*, pp. 153–158, 1996.

[2] G. Piccioli et al. Robust method for road sign detection and recognition. *Image and Vision Computing*, Vol. 14, pp. 209–223, 1996.

[3] L. Priese et al. Traffic sign recognition based on color image evaluation. In *Proceedings of the 1993 IEEE Symp. on Intelligent Vehicles*, pp. 95–100, 1993.

[4] U. Franke et al. Autonomous driving goes downtown. *IEEE Intelligent Systems & Their Applications*, Vol. 13, No. 6, pp. 40–48, 1998.

[5] G. Salgian and D.H. Ballard. Visual routines for autonomous driving. In *Proceedings of the 6th Int. Conf. on Computer Vision*, pp. 876–882, 1998.