Obstacle Detection on Railway Track by Fusing Radar and Image Sensor

Masato Ukai Bogdan Tomoyuki Nassu Nozomi Nagamine
Railway Technical Research Institute
2-8-38 Hikari-cho, Kokubunji-shi, Tokyo 185-8540, Japan

Masato Watanabe Takayuki Inaba
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan

Abstract

Novel technology to recognize the situation in distant place is necessary to develop a railway safety monitoring system by which human being having fallen onto the tracks from a platform and obstacles in the level crossing can be detected. In this research, we propose a method for detecting a stationary or moving obstacle by the technology which employs the super resolution radar techniques, the image recognition techniques, and the technology to fuse these techniques. Our method is designed for detecting obstacles such as cars, bicycles, and human on the track in the range up to hundreds of meters ahead by using sensors mounted on a train.

In super resolution radar techniques, novel stepped multiple frequency radar is confirmed to provide the expected high resolution performance by experimental study using software defined radar. We designed the parameters of the stepped multiple frequency radar. The signal processing processor which employs A/Ds, D/As of high-sampling rate and the highest performance FPGAs has been designed and developed.

In image recognition techniques, the main algorithm which detects a pair of rails is based on information about rail edges and intensity gradients. As the obstacle exists in the vicinity of rails, all the detection processes can be limited to the region in the vicinity of rails. As an example, we detect an obstacle by estimating image boundaries which enclose a group of feature points exactly on the obstacle. In order to determine the group of feature points, feature points of the whole image are detected at each frames and tracked over image sequence. For robustness estimation of boundary, we use radar so as to detect the obstacle's rough position in an image region where an obstacle would exist; the motion segmentation technique is applied to the tracks that is located in the region. Note that an obstacle's position detected by radar is remarkably rough because radar has low transverse resolution. Nevertheless, the position detected by radar contributes to rapid and robust estimation of satisfactory image boundaries.

To allow for monitoring in large distances, a prototype for an onboard pan/tilt camera platform with directional control was designed and manufactured as required in a railway situation. To detect obstacles in real time, we have built a high-performance test machine with a GPU and an image processing environment.

1. Introduction

In Japan, the number of accidents resulting in injury or death on station platforms is increasing at a higher rate than ever before. Although safety messages are issued to passengers via on-board public address systems and posters, no silver-bullet solution has yet been found. Platform doors (i.e., sliding barriers on platforms) are the most effective means of preventing accidents, but their installation is progressing slowly; only 4.7% of all stations have so far been equipped with them due to the many barriers that exist, such as cost and problems with train-car and station structures.

We are currently developing obstacle monitoring system, for example, using on-board observation to detect obstacles on tracks via various sensors including mounted optical cameras. This paper mainly describes a related method of detection using optical image sensors, outlining: 1. a rail detection algorithm that focuses on the characteristics of shapes such as edges, resulting in high-level extraction performance regardless of changes in weather and line shapes; 2. an image recognition algorithm that enables comprehensive detection of obstacles using multiple sub-modules, each of which evaluates on-track objects in terms of shape, texture, movement and background contrast; and 3. the results of simulation experiments using images for verification.

1. 1 A system concept for fusing radar and image sensor

Although optical image sensors have high spatial resolution (in the cross direction) and superior object detection performance, they also have weaknesses, such as:

(1) Particular susceptibility to ambient light in backlit scenes, rainy/foggy weather, etc.

(2) Low accuracy in terms of distance to target objects

On the other hand, radars using microwaves and milliwaves have advantages over other types of sensors, such as:

(1) Stable detection regardless of rainy/snowy weather and nighttime lighting

(2) The ability to perform detection over an entire area with high data rates and significant detection distances

(3) Capability for direct speed measurement with high target movement prediction performance

Table 1 summarizes the characteristics of various sensors'. We are currently examining a method for sensor integration[1] [2] [3] because a high level of detection performance can be expected by combining a radar sensor offering long-distance detection performance and an optical image sensor in a complementary way (Figure 1).

Table 1. Summaries of various sensors' characteristics

Performance \ Sensor		Radar	Optical camera	Laser	Infrared rays	Ultrasonic
Detection distance	<2m	△	○	△	△	○
	2-30m	○	○	○	×	×
	30m<	○	△	△	×	×
Detection angle	<10deg	○	○	△	○	△
	>30deg	△	○	×	○	○
Angle resolution		△	○	○	○	×
Direct speed measurement		○	×	×	×	△
All-weather compatibility	Performance in rainy weather	○	△	△	△	△
	Performance in snowy or foggy weather	○	×	×	△	△
Performance when the sensor is contaminated		○	×	△	×	○

○:Good △:Weak X:Bad

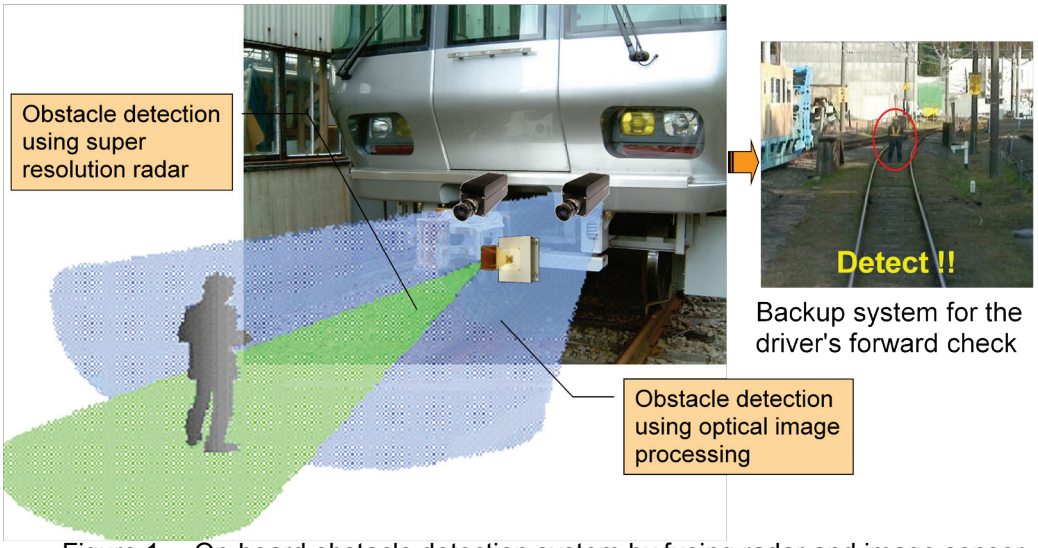


Figure 1. On-board obstacle detection system by fusing radar and image sensor

2. Optical Image Recognition Technology for Front Monitoring on Railways

2. 1 Summary of Required Performance Levels

Table 2 summarizes the performance levels required of optical image recognition technology for a front monitoring system on railways in terms of considerations such as detection distance and detection accuracy. Based on this information, we examined the specifications of individual equipment types such as cameras and image-processing devices, and fine-tuned the parameters of the image-processing program to be used.

Table 2. Required levels of performance for optical image recognition technology

Performance	Required performance level and description
Maximum detection distance	300m or more
Distance accuracy	About 1 m
Detection range	Construction gauge +/- 2 m
Angle accuracy	Object width +/- 0.5 m
Response	500 ms or less
Recognition target	People, vehicles, motorbikes, etc.
Detection output	Alarms, voices, etc.
Device specifications	Small enough to be mounted beside the driver's cab

2. 2 System Design

The image input device, which must meet a required standard of performance unique to car-mounted systems for railways, is the most important part of an image recognition system because performance largely depends on it. It is necessary to use telescopic lenses to secure relatively high resolution at a distance (e.g., a telescopic lens with a focal length of 100 mm needs to be used to ensure a view width of 10 m at a point 400 m ahead (Figure 2)). However, a related problem is that the point of regard exits the screen area on curves and slopes. To keep distant points of regard on the screen, it is necessary to develop a pan/tilt panhead with an automatic tracking function for the point of regard, and vibration isolation measures to minimize camera shake in conditions of telephotography or car vibration must also be implemented. For this purpose, we made a prototype PC-controlled pan/tilt panhead equipped with a gyrostabilizer. The unit demonstrates high-level vibration isolation performance, and Figure 3 shows a platform on which it can be installed as an image input device.

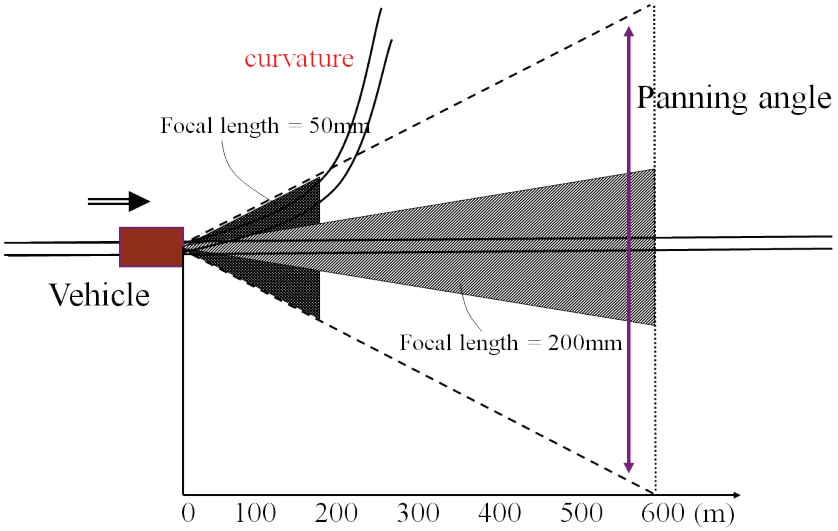


Figure 2. Required levels of performance for image input devices (The necessities of camera view control)

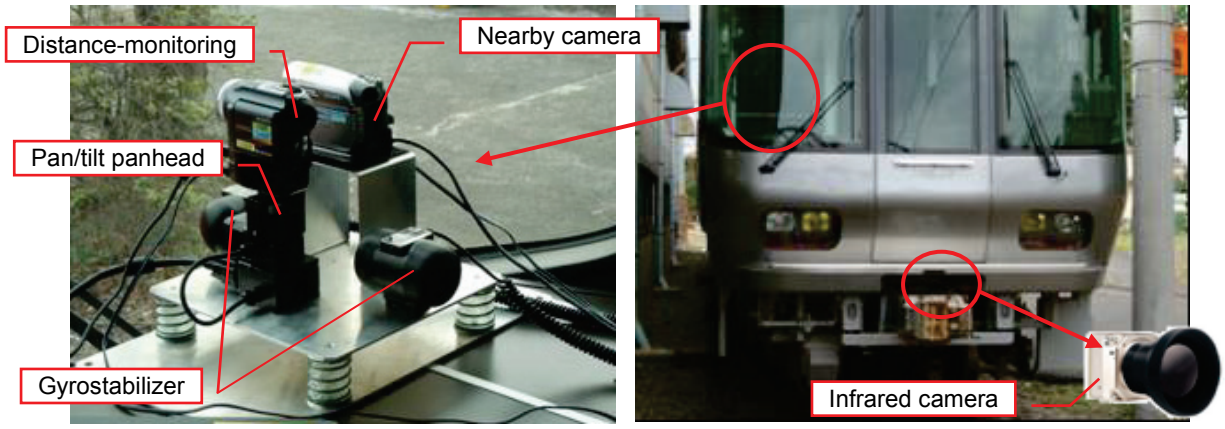


Figure 3. Optical image sensor system to be mounted on a car

3. Development of a Rail Detection Algorithm

One of the basic tasks required for identifying obstacles near the rails is determining the position and shape of the rails. We have designed an algorithm that performs rail detection by matching edge features extracted from the input images to candidate rail patterns, using the Chamfer distance [4] as a similarity metric. The algorithm is divided in two phases, one for regions near the camera and other for more distant regions.

In regions near the camera, the appearance of the rails changes only to a limited degree. Edges are extracted by pre-smoothing the input image with a 3x3 mean filter and using a symmetrical local thresholding operator [5] with hysteresis. These edges are matched to pre-computed patterns. We compute the Chamfer distance *d* from each pattern to different parts of the edge image, and obtain a matching score $s = (M-d)/M$, where M is the maximum acceptable distance. Figure 4 shows an example considering 3 patterns and 3 image positions. The best match in that example had a score of 0.89, for pattern p0 with the top-left corner positioned at x=242.

For the long distance, edges are extracted by pre-smoothing the input image with a 3x3 Gaussian filter and hysteresis thresholding the responses to a 7x7 Sobel operator. Detecting rails in the long distance is more challenging than in the short distance: the complex rail shapes prevent the use of pre-computed patterns, and the smaller resolutions make it hard to discriminate between the rails and nearby structures. We employ an iterative algorithm that considers image sub-regions progressively further away from the camera. The results obtained for a sub-region are used as a starting point for the next sub-region, starting with the results obtained for the short distance. At each iteration, the rails detected so far are projected over the new sub-region, and matching scores are computed for variations on the projection, as shown in Figure 5.

In both steps, we use weighting methods based on the knowledge that the rail shape and position do not change radically between consecutive frames, and that curvature changes are smooth: for example, if the rails are making a curve to the right, they do not suddenly turn to the left. We have tested this approach, with the detected rails being compared to ground truth data generated for over 15,000 images captured in real operation conditions, including challenging scenarios such as backlight and snow. The rail position and shape was correctly approximated in around 98% of the frames, with poor results being obtained for only 0.3% of the frames. Figure 6 illustrates some of the results.

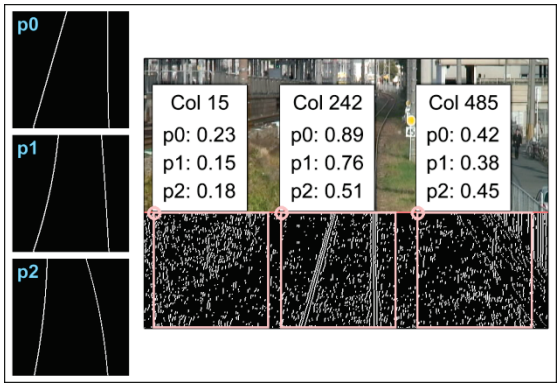


Figure 4. Example pattern matching in the short distance.

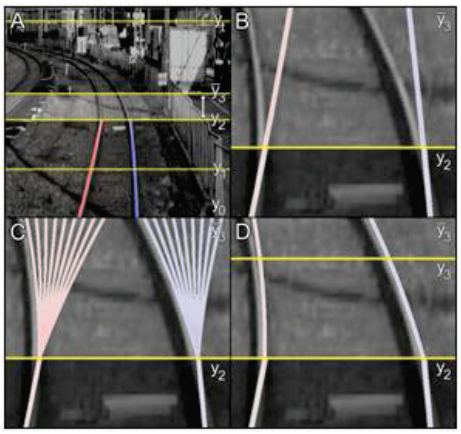


Figure 5. Detecting rails in the long distance.

Point Raindrops on car windshield backlight

Figure 6. Rail detection results

4. Development of an Image Recognition Algorithm for Obstacle Detection

As a real issue, the use of a specific algorithm alone is not sufficient to detect target obstacles perfectly. Accordingly, we applied a method using five sub-modules, each of which performs distinctive processing in parallel, and a comprehensive-assessment module that evaluates the results in a complementary manner to finally determine whether an obstacle is present or not (Figure 7).

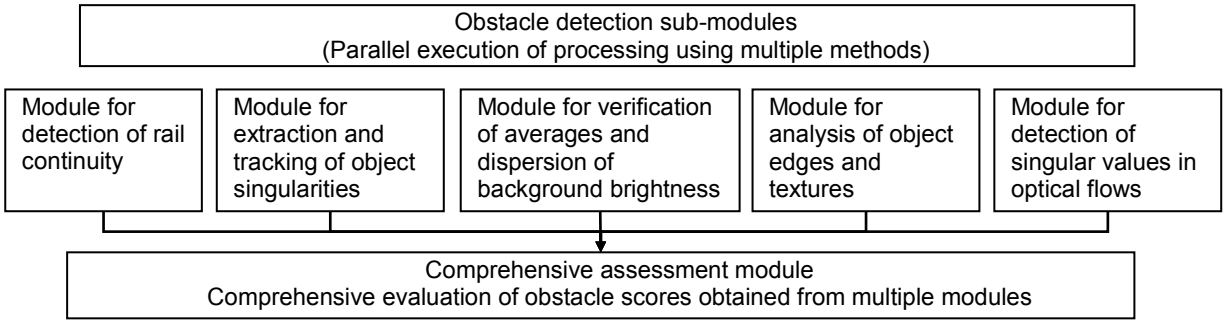


Figure 7. Image recognition method using multiple processing modules

4. 1 Normalization Method of Rail

In addition, shift operation that converts the center line of the detected rails into a straight line is performed as a type of preprocessing to absorb changes in line shape and standardize the input data so that all images can be processed as those with horizontal straight lines (Figure 8). Each of the sub-modules needs only to search a preliminarily fixed area near the tracks after this preprocessing, which is expected to improve detection performance and processing speed.

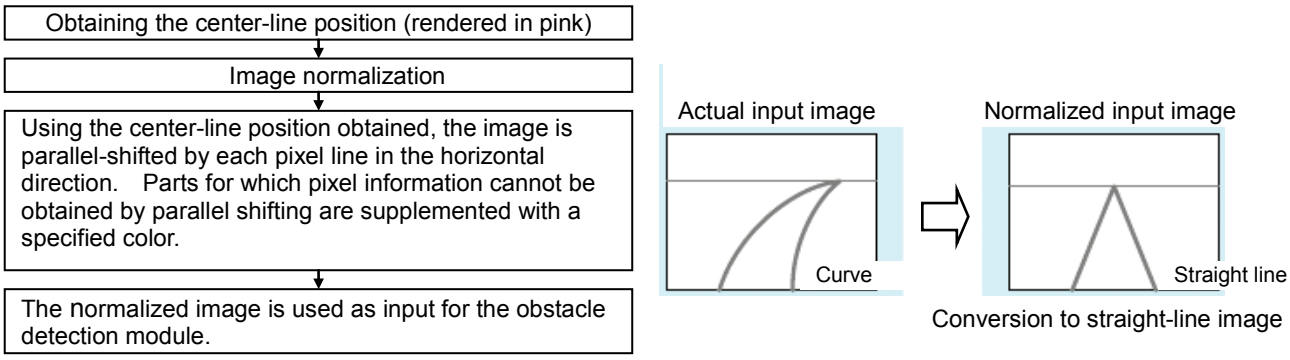


Figure 8. Overview of the rail normalization module

4. 2 Module for Verification of Rail Continuity

This module verifies whether the rails are extracted without interruption to a distant point. With a small rectangular area set on the rails, it calculates the subsequent observed values based on differences in brightness between neighboring areas (Figure 9). When the standardized data, *stdd* obtained from the observed values exceeds a certain threshold value, rail continuity is judged to have ended. It should be noted that the area size can be set arbitrarily and that standardized data are evaluated for each pair of rails.

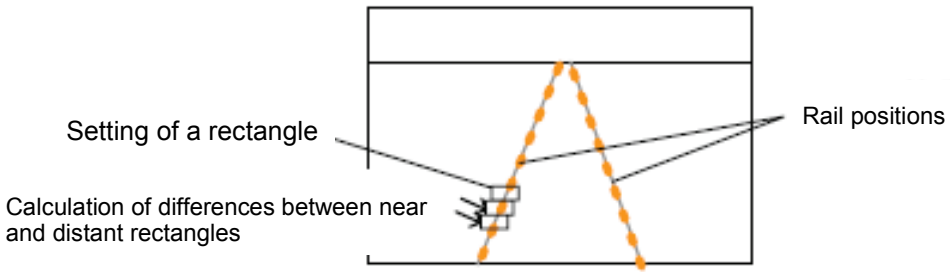


Figure 9. Calculation of the continuity index

The difference of brightness $d(i)$ in the rectangular area is calculated using the following formula:

$$d(i) = \sum_{w} \sum_{h} |I_i(w,h) - I_{i+1}(w,h)|$$

where *I* is the brightness value at certain coordinates, *i* is the type of rectangular area used, and *w* and *h* are relative positions of the rectangular area.

$$Average\ value:\ \ av = \frac{\sum_i d(i)}{Number\ of\ ranges}$$

$$Standard\ deviation:\ \ dev = \sqrt{\frac{\sum_i (d(i) - av)^2}{Number\ of\ ranges}}$$

$$Standardized\ data:\ \ stdd(i) = \left| \frac{d(i) - av}{dev} \right|$$

4. 3 Module for Extraction and Tracking of Feature Point

This module extracts feature point (SIFT) calculated from pixel values and differential values of points that do not vary in terms of scale change and rotation near the rails, and tracks these feature points to neighboring frames. If feature points are detected continually over time, a high evaluation value (obstacle score) is output.

a) DoG (difference of Gaussian) processing

Input images $I(u,v)$ are smoothed using different-scale Gaussian filters $G(u,v,k^n\sigma)$, and smoothed images $L(u,v,k^n\sigma)$ are obtained using the following formula:

$$L(u,v,k^n\sigma) = I(u,v) * G(u,v,k^n\sigma)$$

The differences in the smoothed images are derived to obtain DoG images $D(u,v,k^n\sigma)$.

$$D(u,v,k^n\sigma) = L(u,v,k^n\sigma) - L(u,v,k^{n-1}\sigma)$$

b) Extreme-value detection using DoG images

From images close to the DoG ones $D(u,v,k^n\sigma)$ in terms of scale and space, extreme values are obtained as candidate points for singularities.

c) Selection of singularities using the Laplacian

To remove feature points resulting from linear shapes such as rails, squares of the Laplacian of *u*, *v*, and *σ* are obtained using the following formula, and pixels exceeding the threshold value *th* are taken as feature points:

$$(D_{uu} + D_{vv} + D_{\sigma\sigma})^2 > th$$

Points exceeding the threshold value are selected as candidate points of obstacles.

Figure 10 shows the results of calculation using Gaussian filters with two scales. A change in the scale can be used to adjust sensitivity for obstacles in distant and close views. Accordingly, detection accuracy can be improved by performing analysis with an appropriate scale specified for each place on the screen in accordance with the expected sizes of obstacles depending on distance.

Input image DoG image Feature point extraction results (red circles)

Figure 10. Results of feature point extraction using SIFT operation

4. 4 Module for Verification of Average Values and Dispersion of Background (Track Surface) Brightness

This module measures the average value and dispersion of brightness on the track surface and retains the results while dynamically updating the brightness data for the track surface. It is designed

on the basis that there is little dispersion of brightness change on a normal track surface.

Assuming the pixel values at a certain position on the screen as $\boldsymbol{v} = (v_R, v_G, v_B)$,

$$Hue\ vector:\ \boldsymbol{v}_{hu} = \left(\frac{v_R}{|\boldsymbol{v}|}, \frac{v_G}{|\boldsymbol{v}|}, \frac{v_B}{|\boldsymbol{v}|}\right)$$
$$Brightness:\ v_{br} = |\boldsymbol{v}|$$

are feature quantities. As background data,

$$Average\ of\ the\ average\ hue\ vector\ and\ |Inner\ product\ value\ -\ 1|:\quad \boldsymbol{a}_{hu} = (a_R, a_G, a_B),\ d_{hu}$$
$$Average\ and\ standard\ deviation\ of\ brightness:\quad a_{br}, d_{br}$$

are retained while being dynamically updated. Differences from the background screen at a certain point in time can be calculated in relation to the target pixel values using the following formula:

$$dif_1 = \alpha \frac{|\boldsymbol{v}_{hu} \cdot \boldsymbol{a}_{hu} - 1|}{d_{hu}} + (1 - \alpha)\frac{v_{br} - a_{br}}{d_{br}}$$

where α is a weighting parameter for the difference between the hue vector and brightness. When this value exceeds the threshold, the presence of an obstacle is determined.

$$dif_1 > thr_1$$

The background is updated to adjust to changing the train in the movement and the weather etc..

$$\boldsymbol{a}_{hu} = \left((1 - \gamma_{hu,a})a_R + \gamma_{hu,a}\frac{v_R}{|\boldsymbol{v}|}, (1 - \gamma_{hu,a})a_G + \gamma_{hu,a}\frac{v_G}{|\boldsymbol{v}|}, (1 - \gamma_{hu,a})a_B + \gamma_{hu,a}\frac{v_B}{|\boldsymbol{v}|}\right)$$
$$d_{hu} = (1 - \gamma_{hu,d})d_{hu} + \gamma_{hu,d}|\boldsymbol{v}_{hu} \cdot \boldsymbol{a}_{hu} - 1|$$
$$a_{br} = (1 - \gamma_{br,a})a_{br} + \gamma_{br,a}\, v_{br}$$
$$d_{br} = (1 - \gamma_{br,d})d_{br} + \gamma_{br,d}(v_{br} - a_{br})$$

where $\gamma_{hu,a}, \gamma_{hu,d}, \gamma_{br,a}$ and $\gamma_{br,d}$are update speeds. These values are updated between the upper and lower limits determined for each of them.

We examined countermeasures to combat the false detection that occurs at borders between sunny and shady places. A move into the shade is seen as an overall change in a horizontal area with a narrow width. Based on this knowledge, we reviewed the background threshold conditions for a minute-width layer (with the number of pixels and pixel position moment biases used to determine the data for the foreground) and implemented stricter foreground assessment. Thus, the occurrence of false detection can be restricted by determining whether there is an overall change in a layer.

4. 5 Module for Analysis of Object Edges

This module detects obstacles by focusing on edges. First, smoothing is performed to avoid false edge detection due to noise from short-period elements, and Canny edge extraction is performed. Let E be the pixel set obtained as an edge. In addition, let E_x and E_y be edges in the X and Y directions. To extract edges by direction, a Sobel filter is used to find those in the X and Y directions. Let S_x, S_y be the set obtained. As the Sobel filter finds edges other than those of a specific direction element, the following procedure is used to extract edges of a specific direction element:

1) Obtain the complement of S_x, S_y. This is the sum of sets of edges for a specific direction element and parts that are not edges.

$$\overline{S}_x \equiv \overline{E} \bigcup E_x$$
$$\overline{S}_y \equiv \overline{E} \bigcup E_y$$

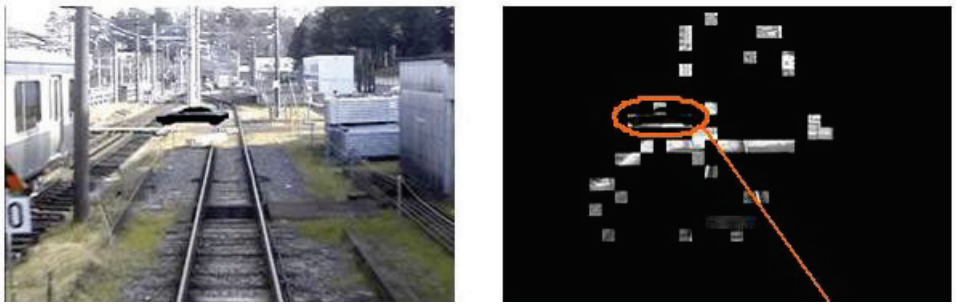
2) Obtain the intersection of this set and *E* to extract edges of a specific direction element.

$$E_x = \overline{S}_x \bigcap E$$
$$E_y = \overline{S}_y \bigcap E$$

Using the obtained edges in various directions as pixel values, this module calculates the averages and standard deviations of background pixels in the same way as for the background brightness module in order to determine whether or not the data represent a foreground area (obstacle) based on differences between the target pixel edge values and the background pixel values. To eliminate false detection, it also checks whether the data are continuously determined as a foreground area (obstacle) in multiple frames.

4. 6 Module for Analysis of Object Textures

This module divides the target image into grids of an appropriate size and extracts feature quantities by texture analysis on multiple pixels included in a grid in order to detect obstacles. For textures, feature quantities are obtained by analysis of pixel lightness histograms. This module creates background data for specific feature quantities such as contrast, secondary moment of lightness and entropy (an index of dispersion degree) in the same way as for the module for verification of averages and dispersion of brightness, and identifies edges between obstacles and the background to detect obstacles. To reduce false detection, the module also checks whether an obstacle is detected continually over time in individual grids with an eye to improving detection accuracy. Figure 11 shows example detection results.



Input image Extracted obstacle candidate

Figure 11. Results of Detection of an obstacle

4. 7 Module for Detection of Singular Values in Optical Flows

This module measures optical flows close to the track vanishing point in the center and retains them while dynamically updating the normal optical flow data associated with train movement. In comparison with optical flows in this background data, any pixels with significantly different directions and sizes (speeds) are determined as those of an obstacle. Figure 12 shows the results of obstacle detection. This module specializes in the detection of moving objects.

Normal optical flow (blue line) Optical flow when a bicycle passes (red line) Candidate obstacles

Figure 12. Results of obstacle detection using optical flows

4. 8 Comprehensive Assessment Module

The comprehensive assessment module for each of the uniformly sized grids into which the screen is divided calculates a weighted sum for the assessment results of other modules, and determines the presence of an obstacle when a certain threshold value is exceeded. To allow visual understanding of the contribution degrees of the various modules when an image is assessed as an obstacle, module scores are shown as radial line segments of a hexagon in different colors. The score values correspond to the length and width of the line segments (Figure 13 Figure 14).

In a case involving the detection of a person falling from a platform, it was discovered that the optical flow module responded to the falling movement with high accuracy. It was also discovered that the feature point extraction module and the background brightness module responded a little too sensitively. Moreover, it was learned that a grid determined by more than one module to contain an obstacle was indeed likely to contain an obstacle. Good detection results were obtained when the grid size was set between approximately 5 x 5 and 10 x 10 pixels.

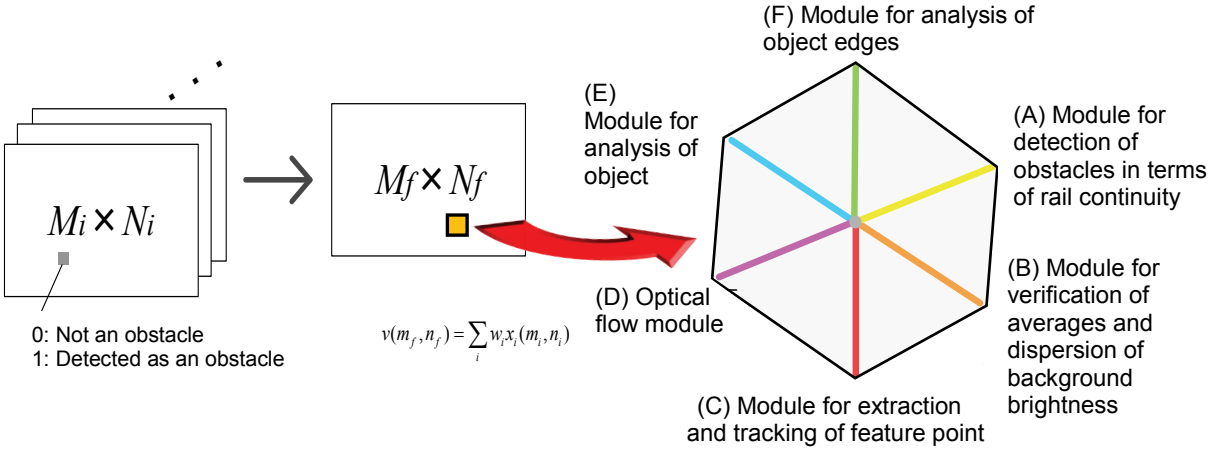
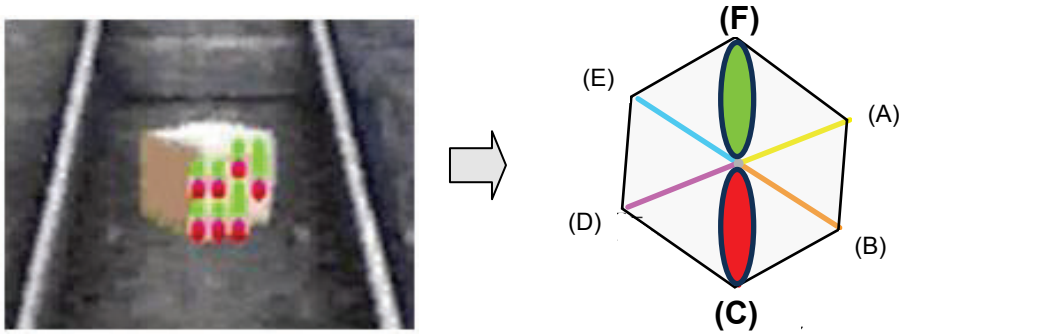


Figure 13. Visualization of module output results in relation to assessment results



(C) and (F) modules responded strongly to cardboard box

Figure 14. Results of analysis for modules that actually responded

5. Outline of Radar Sensor

A high resolution radar is under development for the purpose of the data fusion with the optical image sensor. The radar (Figure 15) is based on the stepped multiple-frequency CPC(Complementary Phase Code) method that is our proposed radar signal modulation method to achieve high resolution by the narrowband receiver in accordance with specified low-power radio station standard of the millimeter wave (transmitting power 10mW and transmitting bandwidth of 500MHz). In a complex radio wave environment with a lot of undesired reflection waves (clutter) like the railway environment, a suppression ability of the clutter decides the maximum detection distance. The high resolution in a long range is known to be an effective approach to reducing the clutter influence.

The stepped multiple-frequency CPC method is a radar signal modulation method to integrate the pulse compression method and the synthetic wideband processing. Though the synthetic wideband processing has a high resolution in a long range in the narrowband receiver, it has a problem of the range ambiguity, the increase of observation time and the range bias by error due to Doppler effect. On the other hand, the stepped multiple-frequency CPC method which employs the transmission frequency sequence shown in Figure 16 and integrates the CPC pulse compression for range gating, the synthetic wideband processing and the doppler compensation within the required observation time for train radar is the radar system to provide the high range resolution in the narrowband receiver without range ambiguity.

Figure 17 shows the schematic diagram of the stepped multiple-frequency CPC and an off-line experimental result (target range 1.2-2.8m,velocity -4km/h)with 24GHz radar equipment using general measuring instruments such as programmable vector signal generators in an anechoic chamber. In Figure 17, it is shown that an obtained target relative velocity is -4km/h. Moreover, the range sidelobes of pulse compression with comparatively short code length of 16 is reduced to -60dB on the average in the range from 0 to 200m. The pulse width of pulse compression is 15m because the receiver band of the general-purpose radar is 10MHz. Range resolution ΔR($\Delta R = c/2B$,transmission bandwidth B) is about 2.5m by the synthetic bandwidth processing. It is expected that the millimeter wave high resolution radar under development provides 30cm range resolution.

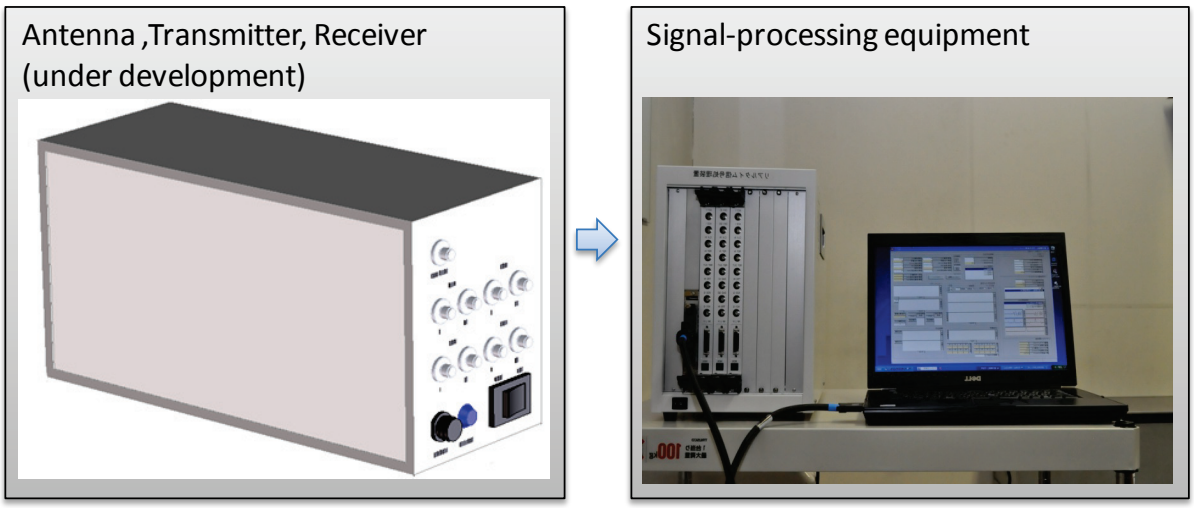


Figure 15. Millimeter wave high resolution radar (under development)

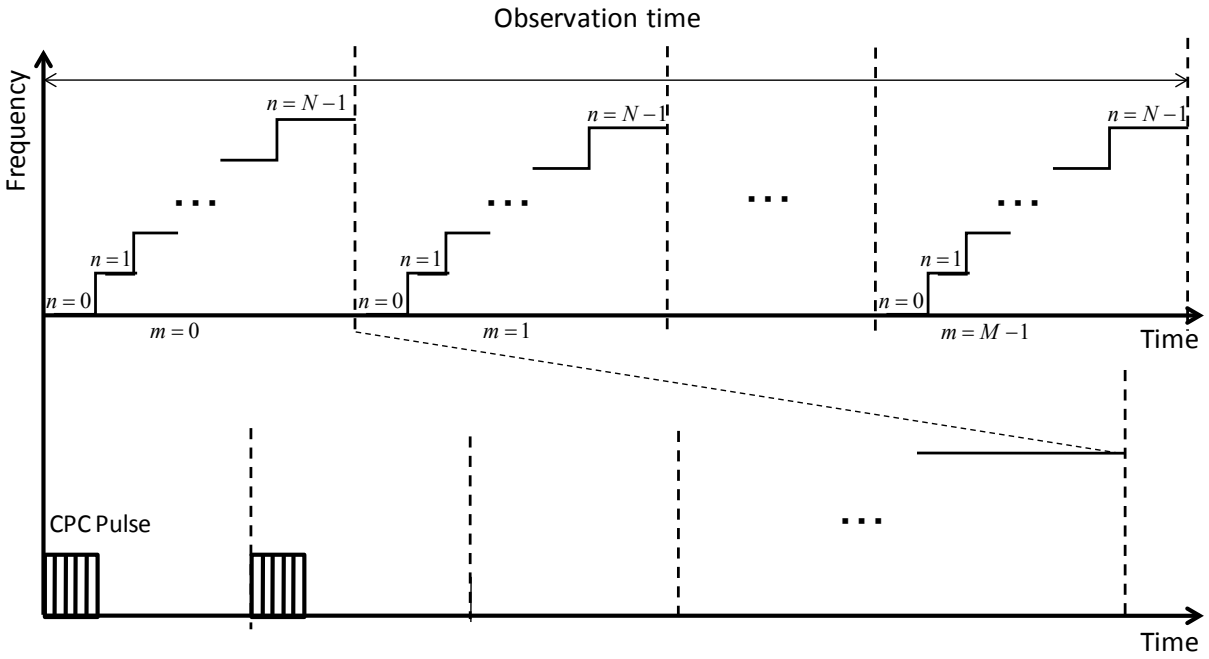


Figure 16. Transmission frequency sequence of stepped multiple-frequency CPC

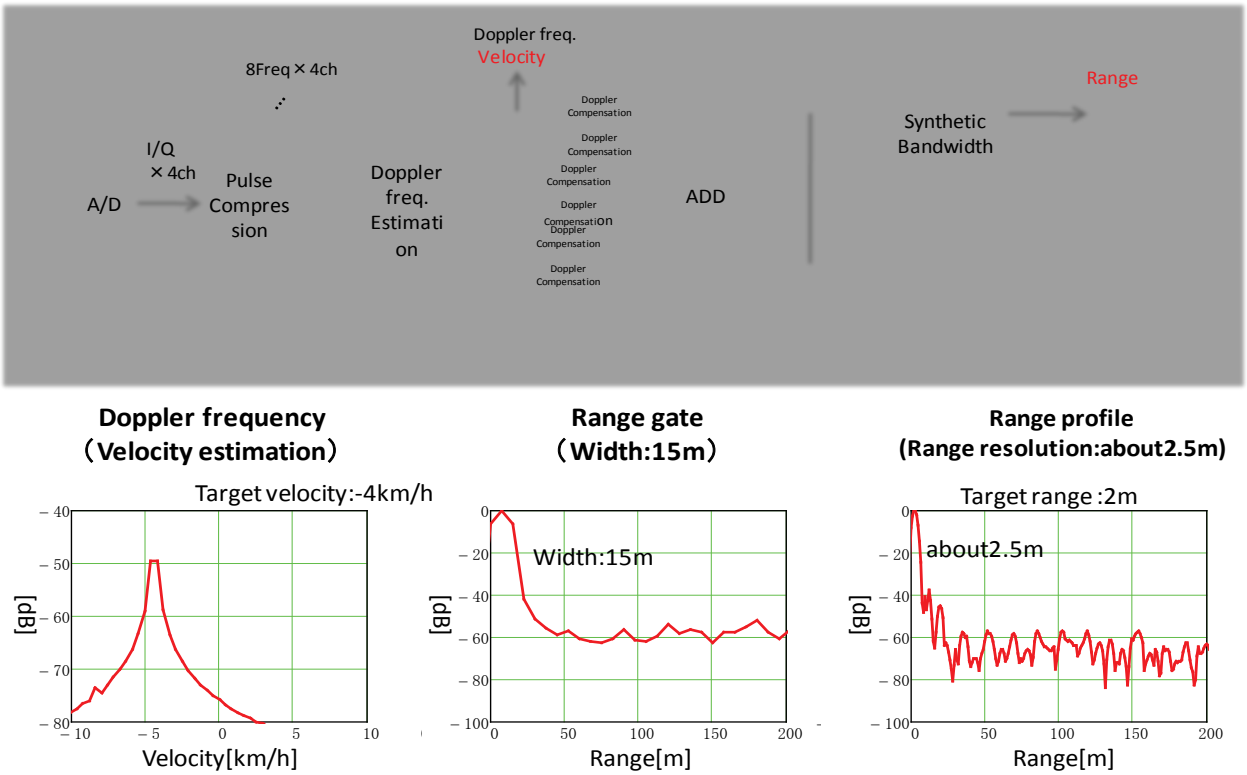


Figure 17. Schematic diagram of stepped multiple-frequency CPC and an experimental result with 24GHz radar equipment using general measuring instruments

6. Examination of a Sensor Integration Method

As a specific method for integrating sensor data, we are now examining the following method: To track an object to be detected, it is possible to introduce a target tracking filter such as a Kalman filter, which performs prediction processing in a state space in order to estimate variations in the distance, speed and angle of a target over time. Distance and speed data measured with a radar sensor can be used to estimate the arrival angle range for the next point in time, which is checked against the horizontal-direction location (angle data) measured using an optical image sensor. Then, knowledge unique to railway space such as the minimum clearance from the rails (gauge) is used in a comprehensive manner to finally realize a highly reliable obstacle detection function.

7. Conclusion

Firstly, we have introduced a new approach for rail extraction− one of the basic tasks for video-based driver support in railways. Our approach addresses the challenges created by the complex environments where trains operate. Using videos captured under real operation conditions, we have derived parameter configurations that perform well under general conditions, showing the proposed approach is feasible, running fast enough for practical use without specialized hardware, and outperforming an existing technique. Future work shall address the automatic learning of rail patterns for the region near the camera. Moreover, rail extraction is not an end on itself, so the evaluation of the proposed approach as part of a larger system is also planned.

Secondly, we have reported on the examination of an image processing method using a complementary combination of all-weather rail detection and multiple recognition modules to improve obstacle recognition performance. In the future, we will evaluate detection performance through experiments using real obstacles, and will also examine a parallel processing method for real-time processing.

One of the most important goals in obstacle detection by image processing is to improve the accuracy of recognition. There are some promising methods, such as those based on the distance information obtained by a stereo camera and color histogram information etc. However it is difficult to construct a practical detection system only by using the machine vision method. Therefore, it is important to integrate several sensors that complement different weak points.

REFERENCES

[1] Ruder, M.; Mohler, N.; Ahmed, F.; “An obstacle detection system for automated trains,” Intelligent Vehicles Symposium, 2003. Proceedings. IEEE., pp. 180-185, 9-11 June 2003.

[2] Alberto Broggi, Pietro Cerri, Pier Claudio Antonello, “Multi-Resolution Vehicle Detection using Artificial Vision,” IEEE Intelligent Vehicles Symposium, pp. 310-314,June 14-17, 2004.

[3] M. Bertozzi, L. Bombini, P. Cerri, P. Medici, P. C. Antonello, M. Miglietta, “Obstacle Detection and Classification fusing Radar and Vision,” IEEE Intelligent Vehicles Symposium, pp. 608-613,June 4-6, 2008.

[4] D. M. Gavrila, “A bayesian, exemplar-based approach to hierarchical shape matching,” IEEE Trans. Pattern Anal. Machine Intell., vol. 29, no. 8, pp. 1408–1421, 2007.

[5] T. Veit, J.-P. Tarel, P. Nicolle, and P. Charbonnier, “Evaluation of road marking feature extraction,” in 2008 IEEE Conference on Intelligent Transportation Systems, 2008, pp. 174–181

T.Inaba, ”Multiple targrt Detection for Stepped Multiple Frequency Interrupted CW Radar” John Wiley & Sons Electronics and Communications in Japan, Part 2, Vol.90, no.10, pp.49-59, 2007.