Image Recognition of Vehicle Applying Fusion of Structured Heuristic Knowledge and Machine Learning

Tokihiko Akita ¹) Yuya Yamada ²)

1)-2) AISIN SEIKI Co., Ltd.
2-1 Asahi-machi, Kariya, Aichi, 448-8650, Japan (E-mail: akita@rd.aisin.co.jp)

ABSTRACT: Image recognition of vehicles is still difficult for practical use under various actual environments. Recently machine learning algorithm utilizing general feature amount have been often adopted. However, they utilize only a part of information obtained from images. Also, it’s difficult for human to understand the classifier, so care for individual recognition errors is hard. And so, we propose a method to design feature amounts hierarchically structuralized human’s empirical knowledge and adjust them by machine learning. Applying this method to images in actual environment, they were evaluated. As result of the experiments, 90% or more of recognition rate was achieved.

KEY WORDS: electronics and control, image recognition system, rear end collision, vehicle detection, machine learning, Adaboost [E1]

1. Introduction

It is well-known that rear-end collision accidents are the most common cause of traffic accidents and occupies about a third of all accidents. Driver assistance systems are effective to reduce them. Thus front collision warning systems (FCW) or pre-crash brake systems (PCS) are becoming common, which utilize a milliwave radar, laser radar or stereo camera. Among these sensors, a monocular camera system is expected to become widespread due to the low cost and multi-functionality. However, performance of the image recognition of vehicles under various actual environments is still a concern for practical use.

To improve performance of the image recognition, huge quantities of image in many varieties of environments must be recorded and pattern recognition algorithm corresponding to them must be developed. Recently various design methods utilizing general feature amounts based on edge information and machine learning algorithms have been proposed (1), (2). Histograms of Oriented Gradients (HOG) (3) are often utilized as the feature amount in the researches. Various algorithms that improved the original one have been proposed. Co-occurrence of 2 HOG features is utilized in the paper (4). In another paper (5), each part model is extracted by applying HOG features and vehicles are recognized using their relationships obtained by the learning. However, they use only edge information in the image and don’t utilize essential useful information for human to recognize objects such as luminance or color. The Bag-of-Features algorithm has been proposed, which classifies objects as set of various feature amounts (6). However, it doesn’t utilize correlation among each feature amount, thus false positive errors tend to occur. These methods design the classifier by machine learning utilizing image samples. When a detection error is occurred, the classifiers cannot be directly and manually corrected. Furthermore, they are difficult for human to understand the classifiers. Only way to improve is to add error samples. Thus, many error samples are necessary and it is difficult to correct individual errors.

In this paper, combination method of human identification knowledge and machine learning is proposed for those concerns. This method combines features which human seems to utilize for the recognition with the parameters designed by machine learning. The method can be considered as the algorithm which substitutes the edge features in the previously-cited paper (2) with appearance features extracted by human. This method needs trial and error in design of the structure, thus it may take time. However, this can efficiently utilize image information corresponding to the detection objects. And, it is easy for human to understand structure of the classifiers, thus individual errors can be easily corrected. This method was applied to recognition of preceding vehicles in the actual environment, and the evaluation results are reported in this paper.

2. Recognition algorithm

2.1. Outline

Overview of the proposed recognition algorithm to detect a preceding vehicle is explained. Processing flow is shown in Fig. 1. After capturing image of the front camera, lane boundaries are recognized. Processing area for vehicle recognition is set based on the lane boundaries. Vehicle image patterns are searched in the area. If any vehicle pattern cannot be continuously found, optical flow is searched. This is a backup procedure in case of missing vehicle pattern. Approaching vehicle in the near-field area must be detected for collision avoidance. In this case, optical flow is...
more robust than pattern recognition. After detecting optical flow, they are grouped and judged whether the obstacle is approaching. If so, time to collision (TTC) is estimated by the optical flow. Warning is issued, when the TTC becomes below threshold time set from vehicle motion and driver’s response time. If vehicle patterns are continuously found and confidence of the recognition is high, TTC is calculated by change rate of the vehicle size after judging approaching vehicle. Warning is issued depending on the collision risk.

![Flowchart](image)

**Fig. 1** General flowchart of preceding vehicle detection

### 2.2. Pattern recognition algorithm

Proposed pattern recognition algorithm is explained. First, edge histograms in the horizontal and vertical direction are calculated. Then, candidate rectangles for vehicles obtained by combination of the peak positions are extracted as shown in Fig.2. Number of candidate rectangles is reduced by restriction of vehicle size and the aspect ratio, assuming that vehicles exist only on the road with given gradient and radius or less. Unnecessary candidates are eliminated by the constraint, thus calculation cost and probability of false detection can be reduced compared with the conventional raster scan method.

![Histograms](image)

**Fig. 2** Candidate rectangles obtained by edge histograms

The candidate rectangles are checked whether matching to vehicle image patterns. Templates of vehicle image patterns are configured in 2 layers so that they can flexibly adapt to fluctuation of illumination and vehicle appearance in various actual environments. The lower layer expresses each part of vehicle appearance such as outline of a body, tires, shadow under a body and tail lamps. The upper layer expresses combination of these parts so that they can be totally recognized as a vehicle. Appearance combination of the parts is changed depending on illumination condition, vehicle class, weather and so on. Thus, the various combination patterns with statistical appearance probability above certain level are prepared.

Pattern recognition method for each part in the lower layer is explained. We haven’t utilized general feature amounts such as HOG or Haar-like feature, but we have designed specific appearance features of vehicle elements utilized by human to discriminate. These samples are shown in Fig. 3. Vehicle outline, tire shape, shadow under a body and tail lamps were adopted in this research. Only tail lamps can be sometimes seen in the night (right figure in Fig. 3). Recognition must be executed by only this appearance information. A license plate, bumper or rear window can be utilized as appearance features in some cases. However, a license plate could not be discriminated in far area due to the camera image resolution. Other features were not adopted in this research since the appearance patterns were widely changed depending on vehicle class. For discrimination of each part, feature amounts that human seems to utilize as hints of the recognition were extracted and quantified, and then classifiers are designed by machine learning adopting the Real Adaboost. Furthermore, result of the classification can be obtained as the confidence in real number. It is generally considered that the Real Adaboost is more robust with fewer classifiers than the Adaboost, thus this was adopted in this research.

The feature amounts can be treated in real number by adopting the Real Adaboost. Furthermore, result of the discrimination can be obtained as the confidence in real number. This information is integrated to recognize a vehicle. Relative position between the outline and parts are also utilized. The co-occurrence information is contained in them so that expressive power can be improved.

![Appearance patterns](image)

**Fig. 3** Appearance pattern of vehicle elements

As the example, design method for the tire classifier is explained. In a candidate rectangle for a vehicle, candidate tires are extracted as paired rectangles by combination of edge line segments and the luminance in the lower right and left area as shown in Fig. 4.

---

Copyright © 2014 Society of Automotive Engineers of Japan, Inc. All rights reserved
Then, the following various feature amounts were calculated. The luminance feature amount of a tire \( l \) is quantified by average luminance in the candidate tire area bounded by red rectangles in Fig. 4 as equation (1).

\[
l = \frac{1}{M_i \cdot N + M_i \cdot N} \left[ \sum_{j \in y, x} l(i, j) \right] (1)
\]

where, \( M_i \) and \( M_i \) denote width of left and right candidate tire rectangles respectively, \( N \) denote height of candidate tire rectangles, \( l(i, j) \) denotes luminance at image coordinates \( i, j \). \( jctb \) denotes vertical coordinates at bottom edge of candidate tire rectangle, \( ictl \) and \( ictr \) denote horizontal coordinates at left and right edges of candidate tire rectangles respectively. Height of the tires \( N \) and bottom edge of candidate tire rectangle \( jctb \) were extracted as the same value in this case.

And other feature amounts of particular tires are aspect ratio of the left and right rectangle \( artl, arrt \) respectively, relative width to the vehicle outline candidate (yellow dashed rectangle in Fig. 4) of the left and right area \( rvll, rvtr \) respectively, average edge density of the left and right tires (line intensity on red rectangles in Fig. 4) \( edal, edar \) respectively.

\[
artl = \frac{M_i}{N}, \quad arrt = \frac{M_i}{N} \quad (2, 3)
\]

\[
rvll = \frac{M_i}{ictr - ictl}, \quad rvtr = \frac{M_i}{ictr - ictl} \quad (4, 5)
\]

\[
edal = \frac{1}{M_i + 2N} \left[ \sum_{i=1}^{M_i} \sum_{j=1}^{N} lbh(i, jctb) + \sum_{i=1}^{M_i} \sum_{j=1}^{N} lbh(icvl + 1 - j, jctb) \right] \quad (6)
\]

\[
edar = \frac{1}{M_i + 2N} \left[ \sum_{i=1}^{M_i} \sum_{j=1}^{N} lbh(i, jctb) + \sum_{i=1}^{M_i} \sum_{j=1}^{N} lbh(icvr + 1 - j, jctb) \right] \quad (7)
\]

where, \( lbh \) and \( dlh \) denote horizontal and vertical edge density respectively. These edge intensity values were calculated by Sobel filter in vertical and horizontal direction.

The symmetry feature amount of the left and right tires was quantified by difference of the tire rectangle areas \( dta, \) aspect ratios \( dart, \) luminance \( dlt \).

\[
dta = \left| M_i - M_i \right| \quad (8)
\]

\[
dart = artl - arrt \quad (9)
\]

\[
dlt = \frac{1}{M_i \cdot N} \left[ \sum_{j=1}^{N} l(i, j) - \frac{1}{M_i \cdot N} \sum_{j=1}^{N} l(i, j) \right] \quad (10)
\]

Difference of the relative positions to the candidate vehicle rectangle \( dpr \) was also quantified and utilized as a feature amount, which represents symmetric property of position of the candidate tires (red rectangles in Fig. 4).

\[
dpr = \frac{1}{icvr - icvl} \left[ (ictl + M_i) - (icvl - ictr - M_i) \right] \quad (11)
\]

where, \( icvl \) and \( icvr \) denote horizontal coordinates at left and right edges of candidate vehicle rectangle respectively.

Furthermore, edge density of a vehicle body outline between top of candidate tires (line intensity in the green dashed rectangles in Fig. 4) \( edvb \) was also added as an indirect feature.

\[
edvb = \frac{1}{(ictr - M_i) - (icvl + M_i)} \sum_{i=1}^{M_i} lbh(i, jctb + N) \quad (12)
\]

Utilizing the extracted human knowledge, classifiers are statistically designed by the Real Adaboost. Learning algorithm of the Real Adaboost is shown as follows;

Step 1. Preprocessing
Prepare \( N \) of leaning sample \( x_1, x_2, \ldots, x_N \) and the label \( y \{ +1, -1 \} \) corresponding to the class

Step 2. Initialize weight of learning sample \( D_i \)

Step 3. For \( t = 1, 2, \ldots, T \), \( T \) times of learning number
For \( m = 1, 2, \ldots, M \), \( M \) of candidate weak classifier
Step 3.1. Design probability density function \( W_m, W \)
Step 3.2. Calculate evaluation value \( z \)
Step 4. Select a weak classifier with the maximum \( z \) in the weak classifier candidates
Step 5. Update weight of learning sample \( D_i \) and normalization

Step 6. Construct a strong classifier

In Step 1, labeled learning samples are prepared. In this case, \( \{x_i\} \) is a sample group of the candidate rectangle images for each elemental part such as a tire. For each image, vehicle images are labeled as positive (+1), and other noise images are labeled as negative (-1). In Step 2, weight of learning sample \( D_i \) is equalized as equation (13).

\[
D_i(\hat{t}) = \frac{1}{N} \quad (13)
\]

Step 3 to 5 are repeated for number of times for learning \( T \) and number of candidate weak classifier \( M \). In Step 3.1, each feature amount is converted to BIN and the sample weight is accumulated in each BIN so that probability density functions (PDF) \( W_m, W \) are built as equation (14), (15).

\[
W_i^j = \sum_{i, j, j, j \in y, y} D_i(\hat{t}) \quad (14)
\]

\[
W_i^j = \sum_{i, j, j, j \in y, y} D_i(\hat{t}) \quad (15)
\]

where, \( j \) denotes BIN number, \( W_i^j \) denotes a value of BIN \( j \) of PDF \( W_i \) for positive samples, \( W_i^j \) denotes a value of BIN \( j \) of PDF \( W_i \) for negative samples and \( J \) denotes a total number of the BIN.

In Step 3.2, their similarity is evaluated by the Bhattacharyya distance (7), which is adopted as the evaluation value \( z \) in equation (16).

\[
z = 1 - \sqrt{\sum_j W_i^j W_i^j} \quad (16)
\]
The samples of the histograms for feature amounts of a tire at beginning of the learning are shown in Fig. 5-7. Fig. 5 shows PDF of average luminance in areas of the candidate tire. Quality of this feature amount is not so high since the histograms are overlapped according to the Bhattacharyya distance of 0.11. Thus this classifier is weak. Fig. 6 shows PDF of difference between aspects in the area. Their histograms are almost overlapped, so the Bhattacharyya distance was 0.04. As this classifier is very weak, it’s infrequently-used. Fig. 7 shows PDF of edge density of a vehicle body between top ends of tire candidate rectangles. The histograms are separated very well, thus a good classifier can be constructed by this feature amount. The Bhattacharyya distance was 0.72. Feature amounts to construct the classifiers are selected in descending order of the evaluation value as the following steps.

And, output of the classifier \( h(x) \) is calculated in equation (18).

\[
h(x) = \frac{1}{2} \ln \frac{W_t' + \epsilon}{W_s' + \epsilon}
\]

where, \( \epsilon \) is a small number to avoid divide-by-0 (e.g. 0.0000001).

In Step 5, weight of learning sample \( D_t \) is updated by result of the classification in equation (19). And then, they are normalized in equation (20).

\[
D_t(i)(t) = D_t(i)\exp[-y_t h(x_i)]
\]

\[
D_t(i) = D_t(i)/\sum_{n=0}^{N} D_t(n)
\]

\( T \) weak classifiers are designed by the learning, and middle classifier for a tire is calculated by the summation in equation (21). They are not classified in this stage, and the result \( H(x) \) is transferred to the upper classifiers as confidence of the lower level classification.

\[
H(x) = \sum_{i=1}^{N} h_t(x)
\]

Vehicles are classified by utilizing confidence of recognizing each part. Strong classifiers for vehicles are constructed by various combinations of middle classifiers for each part as shown in Fig. 8. Vehicles are classified for each combination since some parts cannot be seen and features of some combinations cannot be calculated depending on illumination condition. Then, relative position to the road boundaries was added to feature amounts since obtained feature information is too few to recognize in some situations. 23 patterns of feature combination were statistically selected in this research. The classifiers utilizing confidence of the middle classifiers and the Real Adaboost are constructed for each pattern in equation (22), and then the pattern with the largest output of classifier is selected in equation (23). If it exceeds given threshold value, it is recognized as a vehicle.

\[
H_p(x) = \sum_{i=1}^{N} h_i(H_p(x))
\]

\[
H_{vehicle}(x) = \max_{p \in P} [H_p(x)]
\]

where, \( h_p(x) \) denotes output of classifier for pattern \( p \), \( h_i(x) \) denotes the \( i \)th classifier, \( S \) denotes a number of the classifiers, \( H_i(x) \) denotes the \( i \)th middle classifier, \( H_{vehicle}(x) \) denotes final confidence for vehicle classification and \( P \) denotes a total number of the patterns.

Fig. 8 Samples of vehicle appearance pattern
Images in actual environment were prepared, and part classifiers and vehicle classifiers were separately learned in this proposed method. The images were intentionally selected to have wide variety of part appearance so that learning image samples can be efficiently collected. If the image samples are randomly selected and learned as conventional methods, the feature space becomes multiple dimension and huge image samples are necessary to cover entire feature space. From restriction of cost to collect image samples, 50 to 100 images of positive samples were prepared and negative samples of 10 to 20 times the positive samples were prepared and learned. About 300 of images for positive and negative sample each were used for vehicle discrimination.

### 2.3. Optical flow

The pattern recognition is reliable for learned appearance pattern. However it is weak in unlearned patterns caused by illumination condition or special vehicle shape. In order to maintain the recognition performance even in the cases, optical flow is utilized when any vehicle pattern cannot be detected. To avoid a collision to a preceding vehicle, it is sufficient to detect an approaching vehicle. Optical flow can robustly detect approaching objects without depending on the appearance pattern.

The detection method utilizing optical flow while a vehicle is approaching to a subject vehicle is explained. Fig. 9 shows the optical flows on a preceding vehicle and the vanishing point (VP). Assuming that a preceding vehicle to be detected exists in the traveling direction of a subject vehicle, it is only necessary to detect optical flows approaching radially from around a focus of expansion (FOE) of the camera.

Optical flows are calculated by the template matching method adopting the normalized cross-correlation (NCC) for the adjacent area of local feature points adopting the KLT method \(^{(9)}\), considering the calculation cost and the detection performance as results of the evaluation for various methods \(^{(9)}\). This tracking method for local feature points is robust for partial shadow or fluctuation in illumination under the actual environment \(^{(10)-(11)}\).

Assuming that the start and end points of these optical flows exist on the planes parallel to the camera image plane respectively, these optical flows can be grouped by using the following simple invariant \(C\) in equation (24), which is derived from projection geometry. Approaching vehicles can be detected in the two conditions of grouping only optical flows that converge to the VP and satisfying this equation (24).

\[
\begin{align*}
A &= x_l - x_{2i} \\
B &= x_l - x_{vp}
\end{align*}
\]

where, \(x_l\) and \(x_{2i}\) are \(x\) coordinates of the end and start point of each optical flow, \(x_{vp}\) is \(x\) coordinate of a vanishing point of the optical flows.

Grouping processing is actually difficult only from this equation. Because actual optical flows do not converge to one point of the VP due to the noise. The equation is modified as equation (25), and each optical flow is plotted on the 2-dimensional graph as shown in Fig.10. Points on a line with some tolerance in this graph can be grouped as an oncoming obstacle by the RANSAC algorithm. Fig. 11 shows the grouped points corresponding to the graph in the actual scene, though the vehicle is approaching from behind in this picture.

\[
dx_i = C \cdot x_l - C_1
\]

where, \(dx_i = x_l - x_{2i}\), \(C_1 = C \cdot x_{vp}\)

2.4. Collision warning

Collision risk quantified by TTC can be estimated by change rate of size of the detected feature patterns or the grouping result of optical flows. If the feature patterns with high confidence are continuously detected, the TTC can be calculated by change rate of the pattern width as shown in equation (26). In this case, calculation amount can be reduced because it does not need to search optical flows.

\[
TTC_c = \frac{\Delta t}{w_1/w_2 - 1}
\]

where \(TTC_c\) is prediction time reaching to the camera, \(\Delta t\) is time interval of image capturing for optical flow calculation, \(w_1\) and \(w_2\) are width of vehicle feature pattern at the latest detection and \(\Delta t\) ago.

If enough recognition confidence of the feature patterns cannot be obtained, optical flows are detected and the TTC can be calculated by equation (27). As a value of \(C\) in this equation has already calculated in the equation (24) in the grouping process, calculation amount can be saved. In addition, compared with the calculation method using derivation of the distance to a vehicle,
higher accuracy can be obtained due to no influence from road surface condition since the distance must be calculated by the triangulation to the road boundary.

\[
TTC = \Delta t \left( \frac{1}{C} - 1 \right)
\]  

(27)

When this TTC falls below the set threshold, a warning is issued. The minimum threshold of TTC is specified in CA NCAP, so they were set based on these values (12).

3. Experimental result

3.1. Experimental condition

Recognition performance was evaluated off-line for the above algorithm. 269 scenes for the evaluation were extracted according to ratio of day-and-night, weather and type of vehicle in the real world, referring to Japanese statistics of traffic flow and weather. Breakdown of the statistics is shown in Fig. 12. The evaluation samples contain preceding vehicles at about 80m of constant inter-vehicular distance or approaching ones to several meters.

![Fig. 12 Ratio of environment in Japan for evaluation](image)

3.2. Experimental result

As result of the evaluation for all scenes, ratio of correct recognition was 91.2% and false alarm occurred twice. The rate rose to 94.0% for the vehicle within 30 m. Sample scenes of the correct recognition are shown in Fig. 13-15. These samples are difficult scenes to recognize vehicles in various poor illumination conditions.

![Fig. 13 Correct recognition scene in rain](image)  ![Fig. 14 Correct recognition scene in tunnel](image)  ![Fig. 15 Correct recognition scene in the evening](image)

Factors of detection error are shown in Table 1. In rainy weather, vehicle features such as outline and tail lamps were blurred by water splash from a preceding vehicle, thus they could not be sometimes detected. The sample scene is shown in Fig. 16. In the night, only tail lamp feature could be seen in some scenes. However color or shape of tail lamps in the camera image were sometimes changed, the errors were caused since parameters for recognition in the image were deviated from the learned parameters. The sample scene is shown in Fig. 17. Features except the outline were lost due to backlight when driving toward sunset, and then detection errors were caused. The sample scene is shown in Fig. 18.

| Factor                  | Error rate (%) |
|-------------------------|----------------|
| Rain                    | 3.4            |
| Back light              | 1.6            |
| Night                   | 1.3            |
| Specially-shaped truck  | 0.6            |
| Other                   | 1.9            |

Table 1. Factors of detection error (Total error rate: 8.8%)

![Fig. 16 Undetected vehicle in rain](image)  ![Fig. 17 Undetected vehicle in the night](image)  ![Fig. 18 Undetected truck in back light](image)

Warning function was evaluated for the scenes including an approaching vehicle. When TTC of warning threshold was set to 2.4sec, warning was issued in 83 scenes which were evaluated. As result of the evaluation, rate of correct detection was 89.2% and no false alarm was issued. It was judged as correct warning in the case that warning was issued at 2.4±0.5sec. Breakdown of the warning errors was 7.2% of late alarm (6 scenes), 2.4% of miss alarm (2 scenes) and 1.2% of early alarm (1 scene). Sample scenes of the warning error are shown in Fig. 19-21. Vehicle outline and tail lamps are blurred by a windshield wiper in Fig. 19. Contrast of vehicle becomes low and tail lamps lose the color in twilight gloom in Fig. 20. Luminance of tail lamps is saturated and the shape is deformed in the night due to their overlap in Fig. 21, though the alarm was issued with some delay in this case.
Cause of detection errors in this evaluation is discussed. One of the factors is that fundamental feature amount such as the outline cannot be calculated due to lack of the visibility depending on the environment. For the countermeasure, they can be considered that normalization process or the camera with high resolution and sensitivity is adopted in order to detect objects even with low contrast. However, they are challenges in the future due to the calculation amount or cost of the camera. Another factor is limitation of the parameter adjustment for the classifiers. The classifiers are statistically designed to separate them in the feature amount space, thus some vehicles that cannot be discriminated from background may remain due to the trade off. We will improve the recognition performance by adding further weaker feature amounts in the future.

3.3. Comparison with conventional method

In our previous paper (12), the recognition algorithm was manually developed by trial and error. This was evaluated for 378 scenes in the actual environment. As the result, 87.0% of true positive detection rate with 5.3% of false positive one was achieved. This data shows that the recognition performance was improved by this machine learning algorithm. However, this is not a result from fair comparison but only reference data because there is difference in the conditions such as evaluation scenes and adjustment methods of the algorithms. As another effect, development efficiency is discussed. Though it took about 6 months to adjust the previous algorithm, it took about 2 months to adjust this proposed method. This result is also only reference data because they were not compared in the same conditions. However, we actually feel that the development efficiency could be improved.

With respect to recognition performance of the weak classifiers, the proposed method was compared to the previous one. In this paper, we will explain about feature amounts of tires, tail lamps, and brake lamps as the typical cases. For the comparison, the rates of true and false positive detection for same image samples of vehicle and background were plotted in the graphs. This graph denotes discrimination performance is higher in the lower right area, which means accuracy rate is higher and error detection rate is lower. Classification results are obtained as degree of the confidence, thus recognition rates for various thresholds (0 - 1 at internals of 0.1) were calculated and plotted. Fig. 22 shows the recognition results for the tire feature amount. In this case, 336 images of vehicle and 322 images of background were evaluated. The classifier applying this proposed machine learning method provided better performance. This is assumed because it is difficult to manually adjust the classifier due to complexity of the feature amount structure.

Fig. 22 Performance comparison of tire classifiers

Fig. 23 shows the recognition results for the tail lamp feature amount. In this case, 210 images of vehicle and 214 images of background were evaluated. There was almost no difference in recognition performance between the classifier applying this proposed machine learning method and the previous one. This is assumed because there is little room to improve due to many images which is absolutely difficult to recognize only the tail lamps. These sample images are shown in Fig. 24. The tail lamp areas are small and color feature is almost lost in these images, thus it is found the recognition is essentially difficult.

Fig. 23 Performance comparison of tail lamp classifiers

Fig. 24 Sample images which are difficult to see tail lamps

Fig. 25 shows the recognition results for the brake lamp feature amount. In this case, 127 images of vehicle and 140 images of background were evaluated. The classifier applying this proposed machine learning method provided slightly better performance. This is assumed because they can be easily recognized due to high brightness and strong red color of the brake lamp images, thus it is hard to appear the difference.
As shown above, it could be shown that recognition performance can be improved approximately though there is some difference in effect depending on the feature amounts. However, it must be noted that these are not results from strict evaluation because conditions of the adjustment were different.

4. Conclusion

Feature amounts were designed by structuring human knowledge to discriminate vehicles, and the classifiers were statistically designed by machine learning adopting the Real Adaboost. The method was adjusted and evaluated for images in the actual environment, and the recognition performance was quantified. As result of the evaluation, 91.2% of correct recognition rate was achieved though there is some room to improve furthermore by adding another feature amounts. Learning image samples can be intentionally selected by structuring feature amounts obtained from human knowledge, thus the classifiers can be designed from relatively fewer image samples. And then effectiveness of this proposed method was quantified comparing with our previous method, though this comparison is not strict because it was difficult to evaluate in the same condition. In the future, the feature amounts will be improved so that the recognition performance will be increased. Furthermore, this method can be applied to various objects to be recognized, thus we will try to apply to other systems in the future.

References

(1) T. Mitsui, Y. Yamauchi and H. Fujiyoshi : Human Detection by Two Stages Adaboost with Joint HOG, The 14th Symposium on Sensing via Image Information, IN1-06 (2008).
(2) H. T. Niknejad, K. Takahashi, S. Mita and D. McAllester : Vehicle Detection and Nighttime for Urban Autonomous Driving, IEEE/RSJ International Conference on Intelligent Robots and Systems, p. 4442-4447 (2011).
(3) N. Dalal and B. Triggs : Histograms of Oriented Gradients for Human Detection, IEEE CVPR, p. 886-893 (2005).
(4) G. Csurka, C. Bray, C. Dance and L. Fan : Visual categorization with bags of keypoints, in Proceedings of ECCV Workshop on Statistical Learning in Computer Vision, p. 59-74 (2004).
(5) R. E. Schapire and Y. Singer : Improved Boosting Algorithms Using Confidencercated Predictions, Machine Learning, p. 297-336 (1999).
(6) Y. Freund and R. E. Schapire : A decisiontheoretic generalization of on-line learning and an application to boosting, Journal of Computer and System Sciences, No. 1, Vol. 55, p. 119-139 (1997).
(7) A. Bhattacharyya : On a measure of divergence between two statistical populations defined by their probability distributions. Bulletin of the Calcutta Mathematical Society 35, p. 99-109 (1943).
(8) J. Shi and C. Tomasi : Good Features to Track, 1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), p. 593-600 (1994).
(9) http://opencv.jp/opencv-1.0.0/document/opencvref_cv_opticalflow.html
(10) T. Akita and K. Hayashi : Rear Obstacle Warning System Utilizing a Rear Monitor Camera, Proceedings of the 13th ITS World Congress, No.1526 (2006).
(11) T. Akita : Image recognition application systems utilizing surround view cameras, Proceedings of the 19th World Congress on ITS, No. AP-00055 (2012).
(12) T. Akita and Y. Yamada : Image Recognition Algorithm of Front Collision Warning System Using Monocular Camera for CA NCAP, FAST-Zero'11 Proceedings, No.TS1-6-1-2 (2011).