Development of Image Processing Sensor for Cooperative Driving Safety Support Systems

Masakatsu HIGASHIKUBO*, Epifanio BAGARINAO and Takio KURITA

To realize Cooperative Driving Safety Support Systems (DSSS) aiming at the reduction of traffic fatalities and serious injuries, an image processing sensor needs to be installed on the road. With such a sensor, automobiles, motorbikes and pedestrians in a driver's blind corners are spotted, and the obtained traffic data is provided to the driver. For this purpose, the sensor should be able to detect the position and travel speed of objects with a higher precision than that of traffic counters, regardless of weather conditions and time zones. We have developed an image processing algorithm which responds to the demand and enables the reconfiguration of a highly precise detector, even in unknown situations, by adding small amount of training data and computation. In this paper, we report the outline and experimental results of the developed algorithm.

Keywords: DSSS, image sensor, HOG, SVM, incremental SVM, re-training

1. Introduction

To realize Cooperative Driving Safety Support Systems (DSSS) aiming at the reduction of traffic fatalities and serious injuries, an image processing sensor needs to be installed on the road. With such a sensor, automobiles, motorbikes and pedestrians in a driver's blind corners (**Fig. 1**) are spotted and the obtained traffic data is provided to the driver. For this purpose, the sensor should be able to detect the position and travel speed of vehicles and motorbikes with a higher precision than that of traffic counters which simply aim to reduce travel time, regardless of weather conditions and time zones. We have developed an image processing algorithm which responds to the demand and enables the reconfiguration of a highly precise detector, even in unknown situations, by adding small amount of training data and computation. In this paper, we report the outline and experimental results of the developed algorithm.

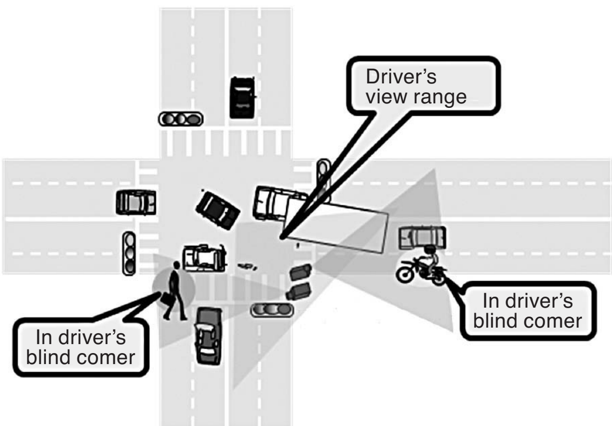


Fig. 1. Example of measuring range of the image processing sensor for DSSS

2. Background

2-1 Traffic accidents situation in Japan and the necessity of cooperative Driving Safety Support Systems

The National Police Agency has announced the traffic accidents situation in Japan [(1)] (**Fig. 2**). Traffic fatalities have been decreasing for 8 years, and became to 5,155 in 2008. This largely depends on improvements in medical technology and the spread of various kinds of in-vehicle safety equipment. Given this situation, Prime Minister Aso's policy speech in January 2009 announced that in the next 10 years we will reduce the traffic fatalities to half and become the most safety country in the world.

Analyzing the accident situation in detail, the following things became clear. **Table 1** shows the type of the accident and it turns out that rear-end collision, head-on collision and turning right collision occupy more than 60 percent of the total number of accidents. Moreover, **Table 2** shows the type of law violation and we can find that violation of safe driving practices (such as failure to make safety check, not keeping eyes on the road or failure to confirm traffic movement) occupy about three quarters of the total number of accidents.

In such accidents situation, to attain the aim of the policy speech "to reduce the traffic fatalities to half in the next ten years," we need to realize the Cooperative Driving Safety Support Systems [(2)] in which infrastructure equipment and in-vehicle equipment cooperate and prevent an accident that cannot be prevented only with each equipment.

To realize this system, it is necessary to develop various technologies such as sensors, communication and traffic signal control. Especially in sensor technology, we think image processing is leading other sensors by the total points of the measuring range, a product life, cost, and performance.

2-2 Goal of the image processing sensor for Cooperative Driving Safety Support Systems

To realize DSSS, the image sensor should spot automobiles, motorbikes and pedestrians in a driver's blind corners and provide the obtained traffic data and images

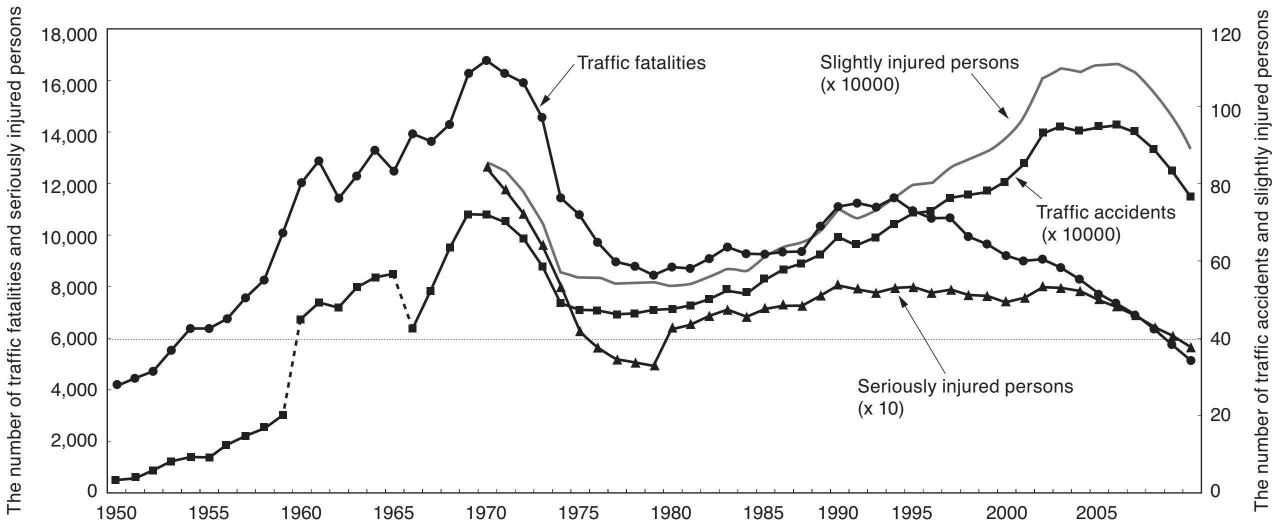


Fig. 2. Transition of traffic accidents situation

Table 1. Number of traffic accidents according to accident type

Type of accidents	Number of accidents
Rear-end collision	239,236
Head-on collision	208,290
Person to vehicle	70,704
Turning right collision	68,147
Collision with structures	38,671
Frontal collision	19,247
Others	121,852
Total	766,147

Table 2. Number of traffic accidents according to law violation type

Type of law violation	Number of accidents	
Disregarding traffic signal	22,512	
Speeding	8,613	
Failure to stop	33,670	
Improper steering and/or braking	49,724	Violation of safe driving practices
Careless driving	47,929	Violation of safe driving practices
Not keeping eyes on the road	115,139	Violation of safe driving practices
Failure to confirm traffic movement	78,233	Violation of safe driving practices
Failure to make safety check	227,553	Violation of safe driving practices
Others	140,147	
Total	723,520	

to the driver. Moreover it can measure the object's position, speed and direction of movement, and can offer drivers the information they need to prevent accidents.

The following systems are considered as examples of application of the image sensor; which tells the driver existence of motorbikes in the dead angle of the large-sized vehicles in the case of right-turn, and tells the driver of a pedestrian's existence not noticed when the driver's mind is taken by other vehicles in the case of right-turn or left-turn.

Also, in terms of performance, to prevent accidents, the sensor should be able to detect the position and travel speed of each vehicle with a higher precision than that of traffic counters, regardless of various environmental conditions (time zone, weather, a camera installation position, a camera angle, etc.).

A common specification in Japan is under consideration, so we set original one shown in **Table 3**, which is based on the experience of our previous projects.

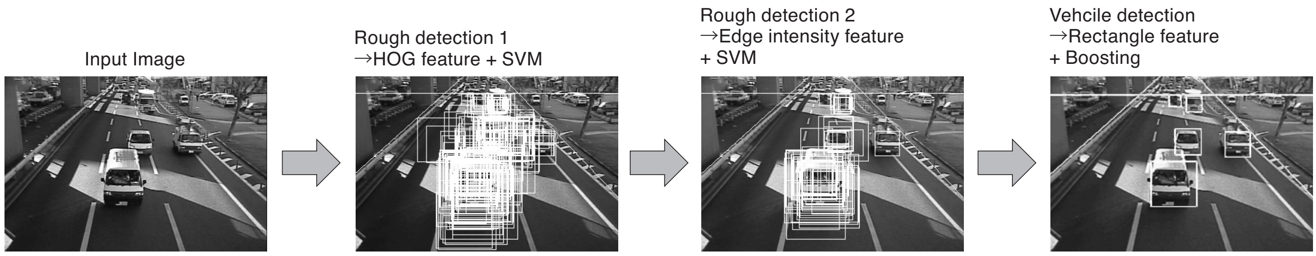
Table 3. Target specification of image processing sensor for DSSS

Articles	Specification	Regulation
Measuring range	up to 4 lanes and 150 m length	
Processing time	100 msec / frame	
Vehicles detection accuracy	less than 3% of false negatives under one false positive per frame	Rate of false negatives = 100 – number of detected vehicles / number of ground truth * 100
Vehicle type classification accuracy	5% or less of incorrect classification	number of incorrect classification / number of detected vehicles
Speed accuracy	10% or less of the difference	(measured speed – true speed) / (true speed)

3. Vehicles Detection Algorithm

3-1 Flow of the vehicles detection algorithm

Figure 3 shows the flow of the developed vehicles de-



Detection method	Strong point	Weak point	Usage
HOG feature + SVM	Almost no negative positives and position gap is detectable	Able to detect a little break into the detection area Undetectable of an exact position	Detection of existence
Edge intensity feature + SVM	Almost no negative positives and more precise detection than HOG + SVM	Able to detect a little break into the detection area More false positives than HOG + SVM	Detection of existence
Rectangle feature + Boosting	Detection of an exact position	False positives of a similar object	Detection of an exact position

Fig. 3. Flow of vehicle detection algorithm and characteristics of each method

tection algorithm. Each image processing method has a strong point and so we combined them.

3-2 Outline of each image processing method

Here, we will explain the outline of the image processing method shown in **Fig. 3**.

(1) Histograms of oriented gradients (HOG) feature

HOG feature is the feature vector of the histogram for each direction of the intensity gradient image in the local area, which was proposed as the feature to detect human by Dalal [3].

HOG feature creates a histogram of the intensity gradients for each local area, so it has an advantage of robustness against illumination changes and geometric change (parallel translation, rotation).

We developed a parameter set which is optimum for vehicles detection. Parameters are such as the size of the local area, the number of gradient directions and so on. And then we can calculate the feature vector that narrows the vehicle's existing range from the whole image without false negatives.

(2) Support vector machine (SVM) [4]

As shown in **Fig. 4**, SVM is the method to calculate the hyper-plane to maximize the margin (distance between the boundary of each class sample), and it is known as the one of the methods that can classify with smallest error.

Training complexity of SVM is large, and for a huge problem with more than 10000 samples it is impossible to solve directly, while the identification process is fast, especially linear SVM can be performed with product sum operation.

Also in the SVM training, we discovered how to set the parameters for the vehicles and motorbike. Furthermore, even if it was a large-scale problem, we developed the way to extract optimal sample data and complete the training in feasible time.

(3) Rectangle feature and boosting

As shown in **Fig. 5**, we set the two adjacent rectangles in the window, and define the rectangle feature as the difference of the average luminosity of each rectangular. Although the discrimination ability of each rectangle feature is weak and it is called the weak classifier, it is known that combining with the boosting method will make one a strong classifier.

As an example, Viola [5] has proposed the face detection method in which some ideas were added to the rectangle feature and boosting method.

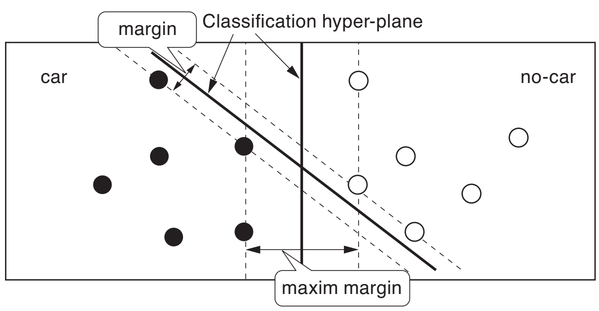


Fig. 4. Concept of the margin in SVM

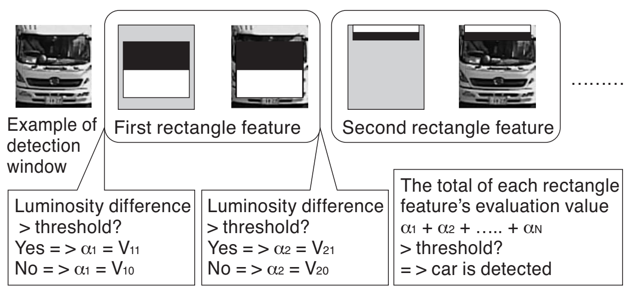


Fig. 5. Outline of rectangle feature + boosting

Table 4. Contents of experimental data

Scene ID	Type of scene	Number of frames	Number of scan positions	Training data: Total number	Training data: Number of negative data (no-car)	Training data: Number of positive data (car)	Total number of vehicles
No.379	shadow of vehicle	12948	4816	58453	37037	21416	19749
No.383	shadow of leaves	26204	3999	50058	34920	15138	13373
No.85	vehicle going away at daytime	9999	4369	88689	61066	27623	22781
No.86	vehicle going away at night	9999	4369	46236	27872	18364	15231
No.122	night	3001	3427	12762	11730	1032	869
No.384	night	11119	4816	61214	45584	15630	13839

Table 5. Result of training and detection by HOG+SVM

	Subset of training data			Classification accuracy of training data			Classification accuracy of scanned dat			
Scene ID	Total number	Number of negative subset data	Number of positive subset data	Total accuracy	Accuracy of negative data	Accuracy of positive data	Total number of detection	Number of true positives	Number of false negatives	Rate of false negatives
No.379	584	311	273	100.00%	100.00%	100.00%	5141359	19579	170	0.86%
No.383	500	279	221	99.97%	100.00%	99.89%	6495247	13371	2	0.01%
No.85	886	528	358	99.60%	99.79%	99.19%	8755370	22653	128	0.56%
No.86	462	269	193	99.92%	99.98%	99.83%	5481497	15097	134	0.88%
No.122	127	117	10	100.00%	100.00%	100.00%	400799	869	0	0.00%
No.384	612	447	165	100.00%	100.00%	100.00%	10378419	13836	3	0.02%

We also have added some original ideas for the detection of vehicles and motorbikes to the rectangle feature and boosting method, and have developed a method to minimize false negatives.

4. Experimental Result of Vehicles Detection

4-1 Rough vehicles detection

(1) The scanning range and the HOG feature

As shown in the “rough detection 1” of **Fig. 3**, we set the measuring range inside of lines, and change the range of detection window size according to the y coordinate (vertical direction).

Then, within the measuring range, while scanning the position of the detection window, we resize the window to 16 x 16 pixels size, calculate HOG feature, and classify the existence of vehicles by SVM.

In the calculation of HOG feature, we extract 8 x 8 pixels cells and overlap them by half size, so 9 cells can be set in a 16 x 16 pixels image. And we set the bin number of the intensity gradient direction to 8, and then the number of dimension of feature vector would be 72.

(2) Introduction of EDGE+SVM

HOG+SVM (HOG feature and SVM) is poor at the position shift, so it detects too many false positives. So, we introduced the vertical edge intensity image and the horizontal edge intensity image which were different from HOG feature. In calculating edge intensity image, we used sobel filter except for the outer pixels. So the dimension of this feature vector is $14 \times 14 \times 2 = 392$. By using EDEG+SVM, we reduce the false positives of HOG+SVM.

(3) Improvement in the training time of SVM

The computational complexity of SVM training becomes huge when the size of training data exceeds 10000. So we trained SVM by having extracted random part of the training data, and verified the classification accuracy to all the training data. And by repeating this we got the best SVM. Moreover, when the best SVM is updated, for the next training, we leave some of the training data, and add some data by extracting randomly from all the training data. Furthermore, by performing some improvement, we succeeded in calculating a highly precise SVM in feasible computing time.

(4) The experimental result of rough vehicles detection

The contents of experimental data are shown in **Table 4**, and a detection result is shown in **Table 5** and **Table 6**. Positive data are manually extracted at the position where vehicle exists in the center. Moreover, we choose 3 or 4 frames in which vehicles do not exist, and extract negative data from the frames while varying size or position. Moreover, the extraction size is limited to more than 16 x 16 pixels.

(5) Consideration

Applying EDGE+SVM to the detection result by

Table 6. Result of using EDEG+SVM to the result of HOG + SVM

Scene ID	Subset of training data			Classification accuracy of training data			Result of using EDEG+SVM to the result of HOG ＋ SVM			
	Total number	Number of negative subset data	Number of positive subset data	Total accuracy	Accuracy of negative data	Accuracy of positive data	Total number of detection	Number of true positives	Number of false negatives	Rate of false negatives
No.379	584	338	246	99.92%	99.98%	99.81%	574528	18945	804	4.07%
No.383	564	632	496	99.88%	99.97%	99.73%	1998130	13303	70	0.52%
No.85	886	526	360	99.74%	99.91%	99.37%	3107169	22288	493	2.16%
No.86	462	252	210	99.78%	99.93%	99.56%	1964435	14867	364	2.39%
No.122	127	103	24	100.00%	100.00%	100.00%	48002	856	13	1.50%
No.384	612	429	183	100.00%	100.00%	100.00%	697575	13768	71	0.51%

HOG+SVM, false positives are largely reduced, and false negatives were also suppressed to 4% or less. In addition, most false negatives had been detected in the neighborhood of the correct position.

4-2 Position detection by rectangle feature and boosting

(1) Training the detector

As a position detection method, we introduced the rectangle feature and boosting. Furthermore, we developed the training method to train a highly precise detector with small computation. Since the computational complexity of large-scale data became huge, we extracted random part of the training data and verified the classification accuracy to all the training data. Then, by repeating this, we got the best detector.

(2) Vehicles position detection

Position detection processing was performed to the rough vehicles detection result using the trained detector. Furthermore, when a detection window overlapped, the center-of-gravity position of the evaluation value of each detection window was set as the vehicles position.

(3) Experimental result

Table 7 shows the detection accuracy by rectangle feature and boosting to the result of rough detection.

(4) Consideration

By rectangle feature and boosting, we can pinpoint the vehicles position with high precision, and false positives were suppressed to one or less per frame. Moreover, by center-of-gravity position calculation, some vehicles which were failed to detect in the rough detection result were detected.

Table 7. Result of rectangle feature + boosting to the result of rough detection

Scene ID	Number of frames	Total number of vehicles	Number of true positives	Rate of true positives	Number of false positives	Rate of false positives per frame
No.379	12948	19749	19320	97.8%	8857	0.68
No.383	26204	13373	13256	99.1%	8032	0.31
No.85	9999	22781	22271	97.8%	10853	1.1
No.86	9999	15231	14708	96.6%	7987	0.80
No.122	3001	869	861	99.1%	326	0.11
No.384	11119	13839	13704	99.0%	8505	0.76

rate of rue positives = number of rue positives / total number of vehicles x 100

4-3 Summary of experiments

Detection accuracy of more than about 97% was obtained by performing rectangle feature and boosting method to the detection result of HOG+SVM & Edge+SVM.

In particular, on the night scene, detection performance is very high and it is more than 99% and there is almost no false positive by surface reflection.

However, on the scene (No.379, 383) of daytime, the false positives by the shadow of trees or buildings are not completely removed. Moreover the cause of false negatives is due to mostly low contrast.

We will improve them by combining the tracking result in the future.

5. Respond to Unknown Situations

"Rough detection" in **Fig. 3** limits the measuring range where vehicles exist, and needs to discriminate from non-vehicles such as shadows or signs on the road, regardless of various environmental conditions (time zone, weather, a camera installation position, a camera angle, etc.).

Although SVM is applied as the classifier, SVM trained at the condition of A may be unable to detect with high precision at condition of B. And training from scratch needs huge costs of collecting training data for each condition.

5-1 Proposed method

We have developed the method to re-train SVM which was suitable for the condition of B by adding small amount of computation and data of the condition B to SVM trained on the condition of A. The re-training method we developed is shown in **Fig. 6**.

Here, an initial training set is a data set extracted from the condition of A, and we prepared many data labeled as "car" or "no car". On the other hand, a target set is the labeled data of the condition of B and we prepared necessary minimum amount of data.

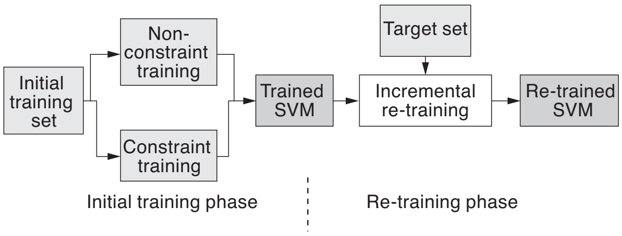


Fig. 6. Re-training method

First, at the initial training phase, using an initial training set, SVM is trained with the incremental SVM method by non-constraint or constraint. In this phase, you may train with the usual SVM. Next, at the re-training phase, data is added from the target data set and SVM is re-trained with the incremental SVM method.

5-2 Re-training method

Here, we will explain the outline of the re-training method shown in **Fig. 6**.

(1) Non-constraint training

Non-constraint training is the method of training SVM using an initial training set, without checking the improvement of the classification accuracy of the target set.

The flow of non-constraint training is shown in **Fig. 7**, and the outline is as follows.

(1) Get a new training data from the training set.
(2) Update SVM using the incremental SVM method.
(3) Do the classification test of the target set using updated SVM.
(4) Repeat from (1) until all training data are processed.

(2) Constraint training

Constraint training method holds the support vectors of SVM and support vectors are used for the next training only when the classification accuracy of the target set has been improved using updated SVM. It is expectable to be more suitable for the target set than non-constraint training.

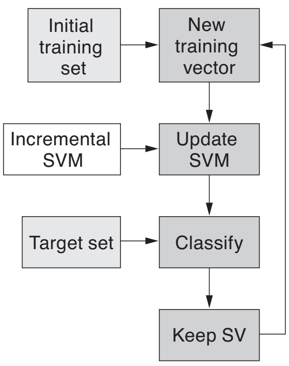


Fig. 7. Non-constraint training

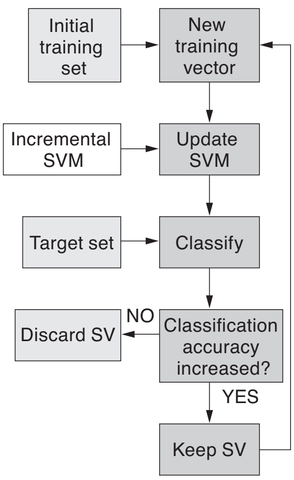


Fig. 8. Constraint training

The flow of constraint training is shown in **Fig. 8**, and the outline is as follows.

(1) Get a new training data from the training set.
(2) Update SVM using the incremental SVM method.
(3) Do the classification test of the target set using updated SVM.
(4) If the classification accuracy doesn't increase, discard the updated support vector (SV).
(5) If classification accuracy increases, keep the updated SV.
(6) Repeat from (1) until all training data are processed.

(3) Incremental SVM

While the usual SVM uses all training data at the same time, incremental SVM adds only one training data. An exact solution [(6)] to incremental SVM was proposed by Cauwenberghs and Poggio. In this solution, the Kuhn-Tacker (KT) conditions are stored in all the above-mentioned training vectors while a new vector is added to the solution set.

The practical advantage of incremental SVM is in online re-training. That is, new training data can be included in the existing solution, without re-training from scratch. For especially large-scale training data, usual SVM training time becomes very large, but incremental SVM can solve it in less computation and will be possible online. Furthermore, the contribution to the classification accuracy of each support vector can also be evaluated easily. It also becomes possible to choose only the suitable support vector which optimizes classification of the target set.

6. Experimental Result

We prepared data set from four types of image scenes (DAY1, DAY2, NIGHT1, NIGHT2) shown in **Fig. 9**. This dataset contains "car" data and "no-car" data, and the numbers of each data are almost same. In the experiment, the data of each scene was divided into ten groups, and was used for training. Each group is called DAY1_n (n is a group number). Moreover, HOG feature is calculated from those data and used as the feature vector.

(1) Non- constraint training result

Non-constraint training was performed for every group of NIGHT1, and the classification ("car" or "no-car") experiment was performed to the data of all the 4

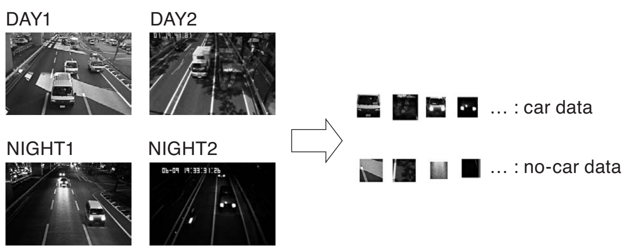


Fig. 9. Type of scenes and training dataset

scenes. As the result shown in **Table 8**, to the same scene data, the classification was highly precise and its accuracy was more than 99%. Moreover, the same high precision classification result was also obtained to NIGHT2 which is a similar scene to NIGHT1. On the other hand, to other scenes (DAY1, DAY2), there was a case where the classification accuracy was under 70%.

(2) Constraint training result

Non-constraint training and constraint training were performed for every DAY1 group, and the classification experiment to DAY2 was conducted. The result is shown in **Table 9**, and two effects by constraint training are found. That is, the classification accuracy of DAY2 is improving about 10%, and on the other hand the classification accuracy of DAY1 is down about same points. However, since it is the purpose to raise the classification accuracy of DAY2, the accuracy decrease of DAY1 is not a problem.

(3) Re-training adding target data

To each group of DAY1, non-constraint training or constraint training was performed in the initial training phase, and re-training was performed by adding data of NIGHT1. The classification result by the re-training result is shown in **Table 10** and re-training was performed by both of constraint training and non-constraint training. The classification to NIGHT1 was highly precise and its accuracy was more than 99%. Moreover, by non-constraint training, its accuracy of DAY1 became more than 99%.

Table 8. Accuracy of true positives by non-constraint training (%)

NIGHT1_n	NIGHT1	DAY1	DAY2	NIGHT2
0	99.9739	84.6013	75.1565	98.5896
1	99.9804	90.4915	67.9231	99.8433
2	99.9869	91.2442	72.4156	99.8668
3	99.9902	88.5310	69.6002	99.8198
4	99.9755	91.1023	73.3942	99.7649
5	99.9739	90.5240	73.3907	99.7649
6	99.9771	91.4256	72.7755	99.7963
7	99.9820	91.2152	80.9450	99.8041
8	99.9902	90.1887	70.3555	99.8590
9	99.9788	91.6240	77.9559	99.8746

Table 9. Accuracy of true positives by constraint training (%)

	Non-constraint training		Constraint training	
DAY1_n	DAY1	DAY2	DAY1	DAY2
0	99.7331	81.6683	91.2186	91.0664
1	99.7160	82.7728	86.4113	94.7753
2	99.6801	82.5388	86.3121	93.6920
3	99.7143	80.9981	86.3172	93.9863
4	99.7844	83.1912	79.6931	93.3215
5	99.7998	85.2336	78.2252	92.8127
6	99.7571	77.4063	87.0819	95.2256
7	99.8341	81.8793	83.0770	93.3481
8	99.7759	82.4342	86.0674	88.6038
9	99.7109	82.9519	88.8885	95.1299

Table 10. Accuracy of true positives after incremental re-training (%)

	Non-constraint training		Constraint training	
DAY1_n	DAY1	NIGHT1	DAY1	NIGHT1
0	99.6750	99.9935	89.0613	99.9951
1	99.7109	99.9935	90.9996	99.9886
2	99.6613	99.9902	93.0765	99.9935
3	99.7006	99.9951	96.0669	99.9706
4	99.7793	99.9967	90.9329	99.9967
5	99.8084	99.9918	90.6660	99.9967
6	99.7536	99.9886	82.0351	98.6686
7	98.8777	99.4527	92.5308	99.9984
8	99.7827	99.9918	90.9466	99.9951
9	99.6921	99.9820	97.9693	99.9918

(4) The number of additional data at re-training

The change of the number of additional target data and the classification accuracy of a target set in re-training was summarized in **Fig. 10**. The left graph of **Fig. 10** is the result when DAY1_0 is as the initial data set and DAY2 is as the target data. And the right graph of **Fig. 10** is the result when DAY1_0 is as the initial dataset and NIGHT1 is as the target data. Even when non-constraint training was performed in the initial training phase, it turned out that re-training for highly precise SVM is possible by adding only hundreds of data.

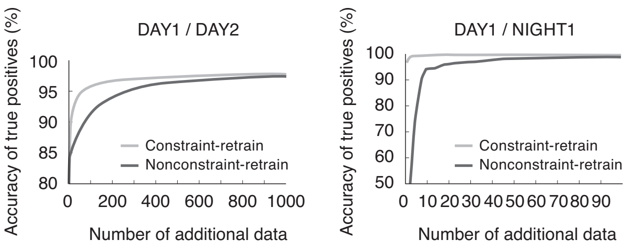


Fig. 10. Relation between accuracy of true positives and the number of additional data at re-training

(5) Experimental result conclusion

We found that SVM could be trained to raise the classification accuracy of the target set using constraint training by the initial training set. Furthermore, by re-training adding hundreds of target data, highly precise SVM to the target data was calculated. By applying the incremental SVM method to re-training, drastic computation time reduction is expected compared with the case where all the training data are trained from scratch. Moreover it turned out that highly precise SVM could be trained for the initial training set and the target set by performing re-training with additional target data after performing non-constraint training in the initial training phase.

From now on, we will perform detailed verification of training time, and detection accuracy verification on further many scenes, and try to develop the method which automatically collects the labeled target data to add.

7. Conclusion

Each image processing method has a strong point and so we combined them. In order to achieve the detection function of vehicles and motorbikes, we developed the algorithm by combining the strong points of each image processing method and we made some original improvement for them. And the detection accuracy has reached less than 3% of false negatives and under one false positive per frame in the measuring range of up to 4 lanes and 100 m length. Moreover, we succeeded in developing the method of solving large-scale training data efficiently. Although not reached the target specification of less than 1% of false negatives, we have started the improvement of the detection algorithm and combining the tracking result.

As the second result, we have developed the reconfiguration method for highly precise detection under various environmental conditions (time zone, weather, a camera installation position, a camera angle, etc.), by adding small amount of training data and computation.

By using this method, it will be able to retrain online and by adding the automatic collection function of additional data, even if the initial training data doesn't know the scene, a highly precise detection classifier will be reconfigured automatically.

And providing drivers with the detection results by the developed algorithm will contribute to realization of traffic accident prevention.

References

(1) National Police Agency, "The traffic accidents situation in 2008," http://www.npa.go.jp/toukei/koutuu48/H20.All.pdf, 2009.

(2) Universal Traffic Management Society of Japan, "a driving safety support system (DSSS)," http://www.utms.or.jp/japanese/system/dsss.html.

(3) N.Dalal and B. Triggs and "Histograms of. Oriented Gradients for Human Detection," IEEE Conference on Computer Vision and Pattern Recognition, pp.886-893, Jun, 2005.

(4) Takio Kurita, "Introduction to support vector machine," http://www.neurosci.aist.go.jp/~kurita/lecture/svm/svm.html.

(5) Viola and Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in Proc.of. CVPR, vol.1, pp.511-518, December, 2001.

(6) Cauwenberghs, G.and Poggio, T., "Incremental and Decremental Support Vector Machine Learning," Leen, T.K., Dietterich, T.G., and Tresp, V. (eds) Advances in Neural Information Processing Systems, vol.13, pp. 409-415. MIT Press, 2001.

Contributors (The lead author is indicated by an asterisk (*)).

M. HIGASHIKUBO*

• Manager,
Applied ICT Systems Department , Information & Communication Laboratories
He is engaged in the research and development of the image processing sensor for road traffic measurements.

E. BAGARINAO

• Ph.D
National Institute of Advanced Industrial Science and Technology

T. KURITA

• Dr. Engineering
National Institute of Advanced Industrial Science and Technology