Article Information

Title	Sunshine-Change-Tolerant Moving Object Masking for Realizing both Privacy Protection and Video Surveillance
Authors	Yoichi TOMIOKA, Hikaru MURAKAMI, and Hitoshi KITAZAWA
Citation	IEICE TRANSACTIONS on Information and Systems, Vol.E97-D, No.9, pp.2483-2492
Copyright	
IEICE Transactions Online URL	https://search.ieice.org/

PAPER

Sunshine-Change-Tolerant Moving Object Masking for Realizing both Privacy Protection and Video Surveillance

Yoichi TOMIOKA[†a)], ***Member***, **Hikaru MURAKAMI**[†b)], ***Nonmember***, ***and*** **Hitoshi KITAZAWA**[†c)], ***Fellow***

SUMMARY Recently, video surveillance systems have been widely introduced in various places, and protecting the privacy of objects in the scene has been as important as ensuring security. Masking each moving object with a background subtraction method is an effective technique to protect its privacy. However, the background subtraction method is heavily affected by sunshine change, and a redundant masking by over-extraction is inevitable. Such superfluous masking disturbs the quality of video surveillance. In this paper, we propose a moving object masking method combining background subtraction and machine learning based on Real AdaBoost. This method can reduce the superfluous masking while maintaining the reliability of privacy protection. In the experiments, we demonstrate that the proposed method achieves about 78–94% accuracy for classifying superfluous masking regions and moving objects.

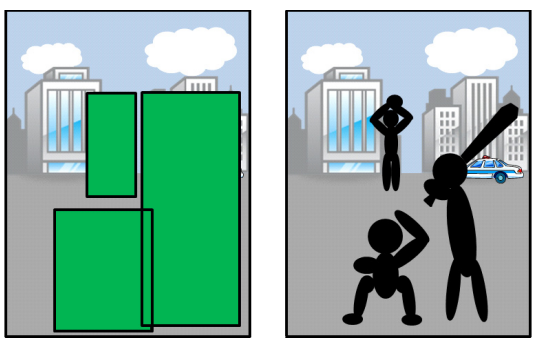
key words: *privacy protection, video surveillance, background subtraction, Real AdaBoost, sunshine change*

1. Introduction

As video surveillance systems and video streaming have come into wide use in recent decades, the following privacy protection methods for video have been receiving increasing attention: privacy-conscious video surveillance systems [1]–[4], privacy protection systems for social videos [5], a method for hiding superfluous details in the background [6], and others. In this paper, we focus on the privacy-conscious video surveillance system based on the work [4] for realizing privacy protection and real-time monitoring for video surveillance simultaneously. This system masks each object by deteriorating the pixel values of the object, and they cannot be identified by normal JPEG viewers. In order to restore the deteriorated objects by a specialized viewer, the original pixel values are encrypted and watermarked in the masked images. Thus, a specialized viewer can restore the information of a specified moving object if the password for the moving object is given.

For such a surveillance system, moving object extraction is an essential technique to mask moving objects. Although there are many moving object extraction and tracking methods, such as background subtraction, frame subtraction, and feature point tracking using mean-shift [7], background subtraction is the most suitable method for moving object masking because it can extract the detailed shape of a static object on the pixel level as well as that of a moving object. For example, if a rectangular region including a moving object is deteriorated, as shown in Fig. 1 (a), the privacy of the moving object can be protected. However, such deterioration prevents us from recognizing what happens in the surveillance video. On the other hand, if we can observe the shape of each moving object in the masked image, as shown in Fig. 1 (b), that helps us to recognize what is happening in the scene without identifying the people in it [4]. The background subtraction approach is, however, heavily affected by sunshine change; for an outdoor scene, the sun often disappears behind a cloud and then reappears. This causes a change in the RGB values in many local regions of each frame, and it results in superfluous masking shown in Fig. 2. Masking should be kept at the minimum needed for protecting the privacy of moving persons.

There have been various proposed background subtraction techniques for solving the problems arising from sunshine/illumination changes: the (a) radial reach correlation [8], (b) statistical reach feature [9], (c) statistical local difference pattern [10], and (d) intrinsic background [11] methods. Methods (a), (b), and (c) are robust to gradual sunshine change, but these methods cannot handle sudden sunshine change ranging from 4 klx to 60 klx. Although (d) addresses a sudden and uniform change in the intensity of an entire image, this method has a low tolerance for change in local regions. It is difficult to solve over-extraction problems arising from sunshine changes with the background subtrac-



(a) Over-masking using rectangles (b) Appropriate masking

Fig. 1 Examples of superfluous masking and appropriate masking. The masking in (a) ensure the privacy protection of moving objects, but it disturbs video surveillance. The black regions represent appropriately masked regions for moving objects, and the green regions represent superfluous masking.

Manuscript received December 30, 2013.
Manuscript revised May 2, 2014.

[†]The authors are with the Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology, Koganei-shi, 184–8588 Japan.
a) E-mail: ytomioka@cc.tuat.ac.jp
b) E-mail: starlit1203@gmail.com
c) E-mail: kitazawa@cc.tuat.ac.jp

DOI: 10.1587/transinf.2013EDP7465

Fig. 2 Superfluous masking due to sunshine change. The black regions represent appropriate masking, and the green regions represent superfluous masking.

tion technique alone. Such over-extraction leads to superfluous masking, as shown in Fig. 2. In order to recognize the scene correctly, superfluous masking should be reduced to the greatest extent possible, as shown in Fig. 1 (b).

In this paper, in order to decrease superfluous masking, we propose a moving object masking method combining background subtraction and machine learning based on Real AdaBoost. After extracting moving object regions including noise due to sunshine change, we classify the extracted regions into moving objects and noise. If a moving object is falsely regarded as noise by this classifier, the moving object is not masked, and its privacy cannot be protected. Thus, such an error must be kept sufficiently small. Our method can adjust the threshold to ensure that the error rate is less than a specified value experimentally. Thus, while keeping the reliability of privacy protection, the security of a privacy-conscious video surveillance system can be improved. The background subtraction technique has been used as the preprocessing of object classifications using SVM or AdaBoost in existing methods [12]–[14]. However, these methods aim to detect specific objects such as pedestrians and vehicles. On the other hand, our method aims to distinguish noise due to sunshine change from all moving objects.

In the next section, we evaluate the robustness of previous background subtraction methods with sunshine/illumination changes. In Sect. 3, we explain the proposed moving object masking method. In Sect. 4, we show the experimental results to demonstrate the performance of the proposed methods. Finally, we present our conclusions in Sect. 5.

2. Evaluations of Background Subtraction Methods Deemed to Be Robust to Illumination Changes

2.1 Brief Descriptions of Six Background Subtraction Methods

We evaluated the sunshine/illumination change robustness for the following six methods; in addition to a single Gaussian model and a Gaussian mixture model [15], which are standard and practical, we selected four methods, which are regarded as robust methods over illumination change: the radial reach correlation [8], statistical reach feature [9], statistical local difference patterns [10], and intrinsic background [11] methods.

A single-Gaussian-model-based background subtraction method (SG) represents the background intensity with a single adaptive Gaussian distribution and is tolerant to the fluctuation of the intensity of each background pixel. A Gaussian-mixture-model-based background subtraction method (GMM) is an enhancement of SG and employs an adaptive Gaussian mixture model to represent the variations arising from multiple factors, such as noise and swaying branches.

The radial-reach-correlation-based background subtraction method (RRC) uses a binary representation of the intensity difference between each pixel and its eight neighboring pixels, called reach points. The reach point for each direction is the nearest pixel in that direction such that the intensity difference is greater than a threshold T_p. Because the magnitude relation of intensities is used, it is relatively robust over a certain level of illumination change. The statistical-reach-feature-based background subtraction method (SRF) and statistical-local-difference-patterns-based background subtraction method (SLDP) are enhancements of RRC, based on statistical stability. SRF collects a variety of possible background models (ideally all possible background models), such as those under different illumination conditions. Then, for each pixel, it extracts a stable binary code that is relatively common with those background images by a similar method to RRC. SLDP represents the intensity difference between each pixel and its reach point with a Gaussian mixture model. The reach points of each pixel are concyclic points with a certain radius called a reach length.

The intrinsic-background-based background subtraction method (IB) represents each frame as being decomposed into a multiplication of a static part (intrinsic background) and a dynamic part (intrinsic foreground) based on the idea of intrinsic image estimation [16].

2.2 Results of Evaluations

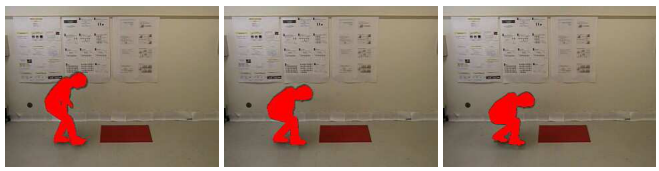
In order to evaluate the robustness under sunshine and illumination changes, we applied the above six methods to the following three scenes:

scene1 browse1 data of PETS2004 [17] in which the intensities of all pixels are artificially suppressed to 40% in the 577th frame and later,
scene2 indoor scene with exposure change,
scene3 parking lot scene with interference of clouds.

Figures 3 (a)–(c) show the ground truth of three consecutive frames of browse1 data of PETS2004 [17], in which the foreground regions are painted red. Similarly, those for the indoor scene and parking lot scene are shown in Figs. 4 and 5, respectively. We show the foreground extraction results for Fig. 3 (b), Fig. 4 (b), and Fig. 5 (b) in Fig. 6, Fig. 7, and Fig. 8, respectively. The parameters of each method

(a) (b) (c)

Fig. 3 Three consecutive ground truth images of foreground extraction for browse1 of PETS2004 [17]. In order to observe robustness to sudden illumination changes, the intensities of all pixels were uniformly suppressed to 40% in the 577th frame shown in (b).



(a) (b) (c)

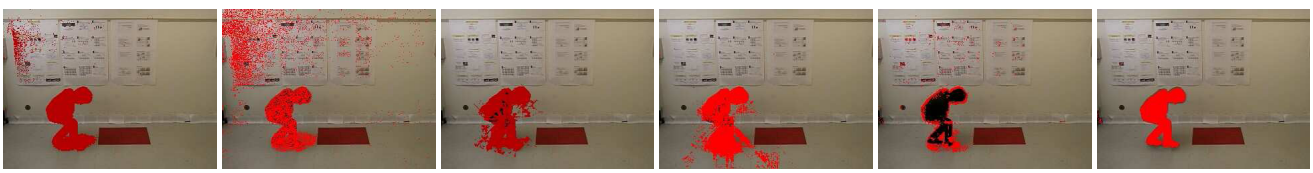
Fig. 4 Three consecutive ground truth images of foreground extraction in an indoor scene.

(a) (b) (c)

Fig. 5 Three consecutive ground truth images of foreground extraction in a parking lot scene.

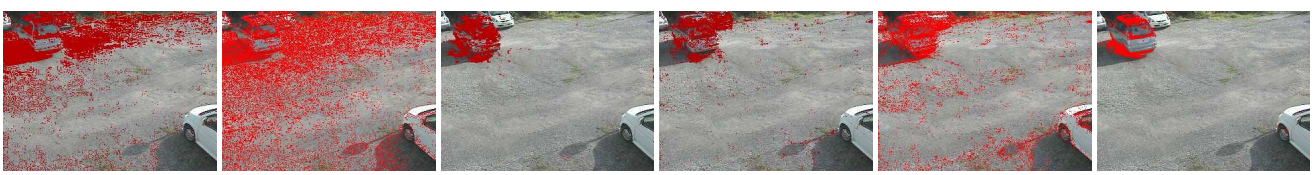
(a) SG (b) GMM (c) RRC (d) SRF (e) SLDP (f) IB

Fig. 6 Examples of foreground extraction results for Fig. 3 (b) of browse1 of PETS2004 [17]. The results from the following methods for this frame are shown: (a) single Gaussian model, (b) Gaussian mixture model, (c) radial reach correlation, (d) statistical reach feature, (e) statistical local difference patterns, and (f) intrinsic background.



(a) SG (b) GMM (c) RRC (d) SRF (e) SLDP (f) IB

Fig. 7 Examples of foreground extraction results for Fig. 4 (b). Methods (a)–(f) are the same as those in Fig. 6.

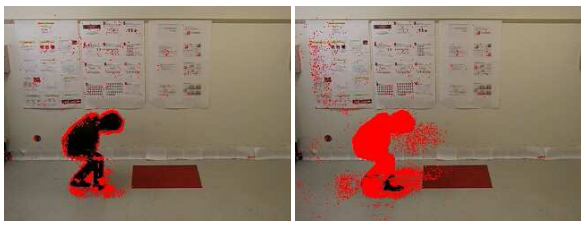


(a) SG (b) GMM (c) RRC (d) SRF (e) SLDP (f) IB

Fig. 8 Examples of foreground extraction results for Fig. 5 (b). Methods (a)–(f) are the same as those in Fig. 6.

Table 1 Parameters of each background subtraction method.

method	parameter	value
GMM	number of Gaussian distribution	4
RRC, SRF	T_p	7
SLDP	number of reach point	6
	reach length	10
	number of Gaussian distribution	4



(a) $r = 10$ (b) $r = 60$

Fig. 9 The effects of reach length r for statistical local difference patterns (SLDP).

are set as shown in Table 1. SG and GMM cannot address sudden intensity suppression. Many background pixels are falsely extracted because of exposure changes and sunshine change. However, it is relatively easy to estimate the shape of a moving object from its extracted pixels as shown in Figs. 7 (a) and (b).

RRC and SRF are relatively robust to illumination/sunshine change. However, when some reach points of a background pixel are on a moving object or its shadow, the pixel tends to be falsely extracted as an object pixel, as shown in Fig. 7 (c). This may create difficulty in recognizing the shapes of moving objects. Moreover, SRF is not effective in the case that it is difficult to sufficiently collect the background models, for example, a variety of vehicles parks in a public parking lot scene. SRF cannot address the ghost problem, i.e., the ghost of each parked vehicle remains in the scene after the vehicle leaves.

SLDP is the most robust to short-term sunshine fluctuation, such as the interference of clouds. However, SLDP is considerably affected by sudden sunshine/illumination changes and falsely extracts the edges of background objects as foreground pixels. Moreover, if the reach length is small compared to the size of a moving object, and some reach points of a pixel are located on itself, the pixel cannot be extracted, as shown in Fig. 9 (a). This under-extraction is improved by increasing the reach length, but the over-extraction due to exposure change increases in our experiment, as shown in Fig. 9 (b).

IB is the most robust method covered in this study to sudden intensity suppression, and the precise shape of a moving object can be extracted. However, it is significantly affected by short-term sunshine fluctuation.

Even these methods, which are regarded as robust methods to illumination change, are affected by sudden intensity suppression, exposure change, and/or short-term sunshine fluctuation. Although a sufficient number of pixels in each moving object can be extracted if we adjust the threshold of background subtraction, superfluous regions

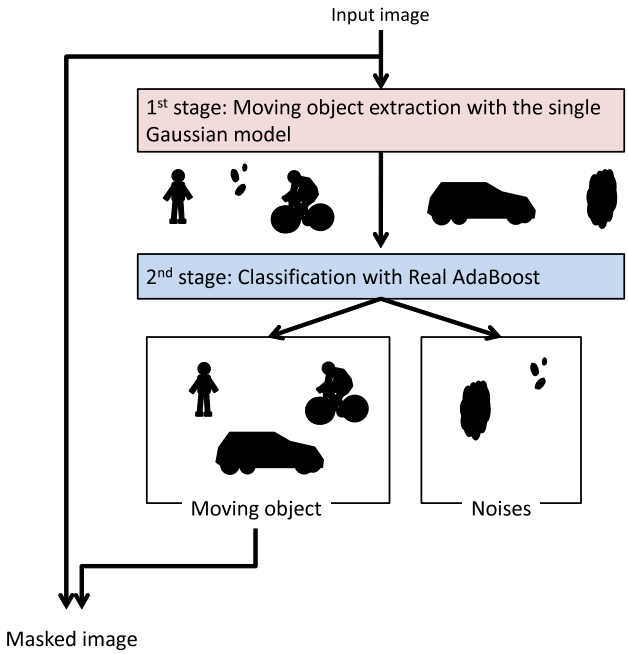


Fig. 10 Generation of masked image with two-stage moving object extraction.

arising from sunshine change are also extracted. Masking these regions may degenerate the reliability of video surveillance. In the next section, we will explain the proposed method reducing superfluous masking by using a Real-AdaBoost-based classifier.

3. Robust Moving Object Extraction over Sunshine Change

3.1 Two-Stage Moving Object Extraction

Although there are many methods for background subtraction, they all have advantages and disadvantages, as mentioned in the previous section. It is difficult to solve the superfluous masking problem mainly arising from sunshine change by a background subtraction technique alone. Thus, we employ a two-stage moving object extraction method, as shown in Fig. 10. In the first stage, we extract foreground regions by a background subtraction method, and then in the second stage, we determine if each region is a moving object. In this case, the background subtraction method should have the following functions. First, for reliable privacy protection, a sufficient ratio of pixels should be extracted for each object. Thus, it is desirable to easily adjust the threshold to determine if a pixel is an object pixel. Second, in order to recognize the action of each moving object, the shape of the moving object should be recognizable by the extracted region. Third, in order to remove ghosts, the background model should be updated appropriately in real time. In this paper, we selected the single Gaussian method, which is one of the simplest methods for practical background subtraction; this method does not require any prelearning, its processing time is short, and its memory usage

is low. Although it is known that a Gaussian mixture model can reduce the effects of repeating events, such as swaying branches, many noise are still extracted. A single Gaussian model is sufficient for the initial moving object extraction, because such noise can be reduced in the next stage with machine learning. Moreover, ghost removal for the single Gaussian model is simpler than that for the Gaussian mixture model. If we do not update the background model of the pixels regarded as moving objects, once a noise or ghost is extracted, it cannot be erased. Thus, in order to remove such noise and ghosts, we apply IIR filtering, expressed by $I(t+1) = (1-\alpha)I(t) + \alpha X(t)$, to not only background pixels but also the pixels regarded as moving objects. Here, $X(t)$ is the intensity for a pixel in the t-th frame, and $I(t)$ is the mean of the Gaussian model in the t-th frame. The duration time of such a noise or ghost can be reduced if we introduce a layered method [18] in which various background layers can be stored. However, this method is not effective in a real surveillance scene with the interference of clouds because the sunshine condition often changes until stored background layers are recalled.

We can reduce the shadowed/lighted region due to sunshine changes by using a normalized distance [19], [20]. However, a plain object region, such as plain shirts, is removed too if the texture of the corresponding region of the background is also plain because the normalized distance represents the texture for a local region.

In the second stage, we divide the extracted pixels into connected components and determine if the region of each connected component is a moving object by using a Real-AdaBoost [21]-based classifier. In the next subsection, we will explain the details of this stage.

3.2 Machine-Learning-Based Relaxation Method of the Effects of Sunshine Change

We use a Real-AdaBoost-based classifier for classifying regions into moving objects and noise mainly arising from sunshine change. We select a Real-AdaBoost-based classifier for the following two reasons (a) and (b). (a) For a privacy-conscious video surveillance, it is important to protect the privacy of moving objects, and the unmasked object rate should be guaranteed to be sufficiently low. Thus, we should employ a classifier which is suitable for adjusting the unmasked moving object rate. The unmasked moving object rate of Real AdaBoost can be adjusted by shifting its threshold. (b) The prediction speed of Real-AdaBoost-based classifier is fast and suitable for real-time applications. For example, Real-AdaBoost-based classifier with 200 weak classifiers is about three times faster than SVM-based classifier.

We introduce seven features to represent each region r: the width of r, the height of r, the area of r, the area of the sub-region of r that is regarded as the shadow by the normalized distance, and the averages of the R, G, and B values in r. Moreover, we also introduce three features for frame f in which region r is extracted: the number of extracted regions in f, the total area of the extracted regions in f, and the total area of the shadow sub-regions in f. All features that we used are listed in Table 2.

Table 2 List of basic 10 features for each region r extracted in frame f.

notation	description
w	the width of region r
h	the height of region r
a	the area of region r
s	the area of shadow sub-region of region r
R	the average of R value in region r
G	the average of G value in region r
B	the average of B value in region r
n	the number of extracted regions in frame f
A	the total area of extracted regions in frame f
S	the total area of shadow sub-regions in frame f

Table 3 List of 48 features generated by combining basic features.

operation	combinational features
addition	$w+h$, $a+s$, $R+G$, $G+B$, $B+R$, $A+S$
subtraction	$w-h$, $h-w$, $a-s$, $s-a$, $R-G$, $G-R$, $G-B$, $B-G$, $B-R$, $R-B$, $A-S$, $S-A$
multiplication	$w*h$, $a*s$, $R*G$, $G*B$, $B*R$, $A*S$, $n*A$, $n*S$, $a*A$, $s*S$
division	w/h, h/w, a/s, s/a, R/G, G/R, G/B, B/G, B/R, R/B, A/S, S/A, n/A, A/n, n/S, S/n, a/A, A/a, s/S, S/s

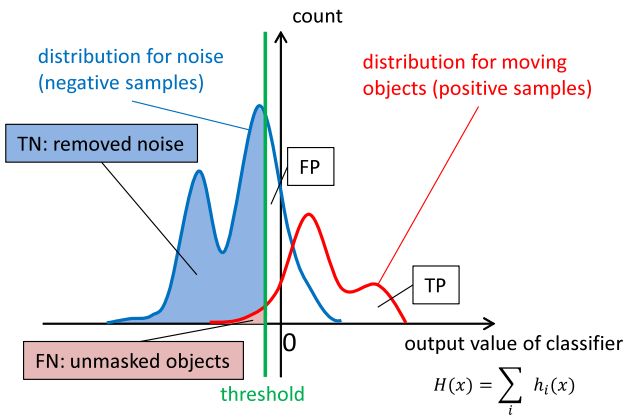


Fig. 11 The threshold determination based on the distribution of the output values of the classifier.

Real AdaBoost does not consider combinational features, i.e., those features combining two or more features, such as the multiplication of two features, in constructing a classifier. Thus, we introduced 48 features generated by combining two of ten basic features, as shown in Table 3. The combinations without physical sense are not used.

We constructed a classifier for determining if each region is a moving object based on Real AdaBoost with a training set of moving object regions (positive samples) and noise regions (negative samples). The threshold of a Real AdaBoost classifier $H(x) = \sum_i h_i(x)$ can be adjusted and should be appropriately determined to achieve a sufficiently low false negative rate and maintain the masking of moving objects. The distribution of the output value of the constructed classifier for positive samples and that for negative samples are shown in Fig. 11. The moving objects with

lower output than the threshold are regarded as noise and are unmasked. Thus, we performed n-fold cross validation using only training data sets, and we determined the threshold that will be used for prediction so that the average of the false negative rate is approximately a specified value. In the experiments, we set the value to be 5%. Although it is desirable to mask all moving objects, over-masking prevents a reliable surveillance which is one of main targets. The threshold adjustment is a technique to balance surveillance and privacy protection. Some moving objects could be identified, but the proposed method makes it hard to identify most of moving objects. This method can offer more reassurance than conventional monitoring without masking.

4. Experimental Results

For the evaluation of moving object extraction, we used three types of data sets, shown in Table 4, with sunshine changes arising from the interference from clouds. Each data set consists of training data and prediction data. The features of all samples were normalized to the range of $[-1, 1]$.

4.1 Comparison of Real AdaBoost Classifier and SVM Classifier

First, we will compare the results of a Real AdaBoost classifier with those of an SVM classifier. We used LIBSVM [22] with RBF kernel. We employ 64-fold cross-validation with Park-T shown in Table 4, and the results are compared in terms of the average accuracy. The accuracy is defined by

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

where TP, FP, TN, and FN are the number of regions of true positive, false positive, true negative, and false negative, respectively.

Figure 12 (a) shows the accuracy of Real AdaBoost with different numbers of weak classifiers for a parking lot data set. Figure 12 (b) shows the corresponding false negative rate which is the unmasked object rate and is calculated by

$$\text{false negative rate} = \frac{\text{FN}}{\text{TP} + \text{FN}}.$$

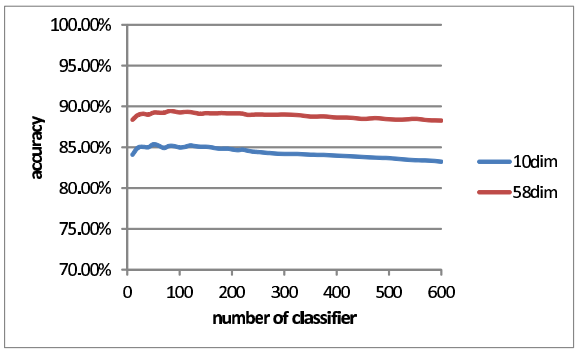
In this experiment, we set the threshold for Real AdaBoost to zero. The result with 58 features is better than that with 10 basic features, and the accuracy ranges between 88.28% and 89.45%. Although the false negative rate ranges between 11.86% and 14.27%, it can be reduced by adjusting the threshold.

On the other hand, for an SVM classifier with 10 basic features, Fig. 13 (a) shows the accuracy for SVM classifiers with different costs and γ [22]. Figure 13 (b) shows the corresponding false negative rate. Similarly, the results for an SVM classifier with 58 features are shown in Fig. 14. As same as the results of the Real AdaBoost classifier show, an SVM classifier with 58 features is better than that with 10 basic features, and the accuracy ranges between 76.80% and 93.46%.

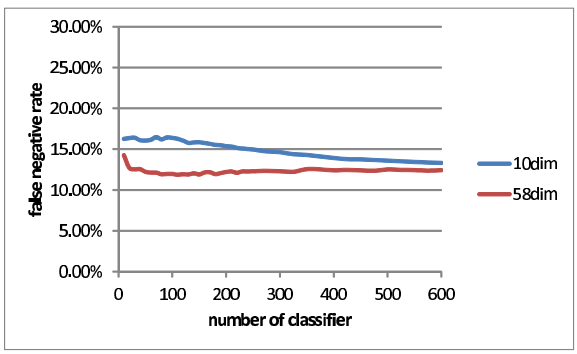
Table 4 Data set for training and prediction.

scene	category	abbr.	#regions		
			total	objects	noises
parking lot	training	Park-T	17301	4914	12387
	prediction	Park-P	3019	535	2484
outdoor1	training	Od1-T	18025	1083	16942
	prediction	Od1-P	3638	383	3255
outdoor2	training	Od2-T	6671	4978	1693
	prediction	Od2-P	1091	761	330

abbr. : abbreviation of data set name



(a) Accuracy vs. the number of selected weak classifiers



(b) False negative rate vs. the number of selected weak classifiers

Fig. 12 The results of 64-fold cross-validation for Real AdaBoost for Park-T data set.

		gamma					
		1.00E−03	1.00E−02	1.00E−01	1.00E+00	1.00E+01	1.00E+02
cost	1.00E+00	75.83%	80.54%	83.67%	88.50%	91.39%	88.01%
	1.00E+01	80.58%	82.85%	86.35%	90.50%	92.15%	87.76%
	1.00E+02	82.27%	84.06%	87.73%	92.13%	92.39%	87.20%
	1.00E+03	83.60%	86.16%	89.78%	92.34%	91.38%	87.04%
	1.00E+04	84.30%	88.06%	90.87%	93.04%	90.12%	87.08%
	1.00E+05	86.27%	89.30%	91.83%	92.83%	88.93%	87.08%

(a) Accuracy

		gamma					
		1.00E−03	1.00E−02	1.00E−01	1.00E+00	1.00E+01	1.00E+02
cost	1.00E+00	82.70%	58.65%	47.82%	30.14%	21.29%	35.19%
	1.00E+01	58.61%	51.59%	37.95%	25.09%	18.25%	33.76%
	1.00E+02	53.93%	46.70%	31.07%	17.46%	16.18%	33.86%
	1.00E+03	48.17%	38.32%	25.74%	16.71%	18.11%	33.92%
	1.00E+04	45.54%	33.03%	23.14%	14.69%	19.25%	33.84%
	1.00E+05	37.83%	27.64%	18.44%	14.20%	20.04%	33.84%

(b) False negative rate

Fig. 13 The results of 64-fold cross-validation for SVM with 10 features for Park-T data set.

		gamma					
		1.00E−03	1.00E−02	1.00E−01	1.00E+00	1.00E+01	1.00E+02
cost	1.00E+00	86.87%	87.82%	91.18%	93.46%	88.28%	76.80%
	1.00E+01	87.80%	88.58%	92.54%	93.38%	88.48%	77.28%
	1.00E+02	87.59%	91.31%	92.68%	92.23%	88.64%	77.34%
	1.00E+03	88.20%	92.24%	92.52%	91.57%	88.66%	77.35%
	1.00E+04	91.20%	92.64%	91.87%	91.12%	88.67%	77.35%
	1.00E+05	91.96%	92.72%	91.53%	91.00%	88.67%	77.35%

(a) Accuracy

		gamma					
		1.00E−03	1.00E−02	1.00E−01	1.00E+00	1.00E+01	1.00E+02
cost	1.00E+00	38.93%	36.67%	24.95%	16.06%	38.85%	81.36%
	1.00E+01	37.46%	34.37%	18.25%	13.88%	37.34%	79.57%
	1.00E+02	36.71%	23.61%	14.41%	15.55%	37.18%	79.41%
	1.00E+03	33.86%	19.50%	13.65%	16.34%	36.75%	79.39%
	1.00E+04	23.42%	15.12%	15.18%	15.26%	36.69%	79.39%
	1.00E+05	19.98%	12.92%	13.47%	15.83%	36.69%	79.39%

(b) False negative rate

Fig. 14 The results of 64-fold cross-validation for SVM with 58 features for Park-T data set.

Table 5 The accuracy of prediction when different data sets are used for training.

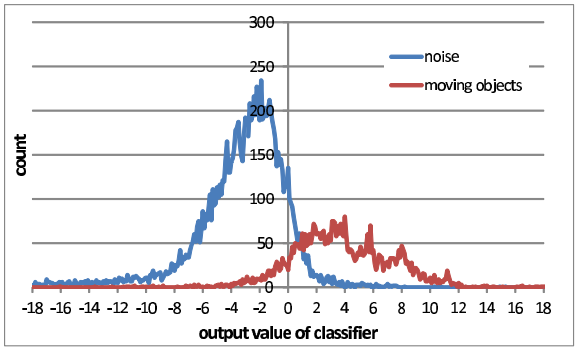
prediction data	training data	accuracy
Park-P	Park-T	78.24%
	Od1-T	42.63%
	Od2-T	18.48%
	Park-T+Od1-T	64.46%
	Park-T+Od2-T	81.45%
	Od1-T+Od2-T	40.34%
	Park-T+Od1-T+Od2-T	66.61%
Od1-P	Park-T	36.09%
	Od1-T	87.03%
	Od2-T	11.96%
	Park-T+Od1-T	76.91%
	Park-T+Od2-T	42.30%
	Od1-T+Od2-T	89.99%
	Park-T+Od1-T+Od2-T	84.44%
Od2-P	Park-T	46.10%
	Od1-T	39.05%
	Od2-T	93.68%
	Park-T+Od1-T	50.96%
	Park-T+Od2-T	86.89%
	Od1-T+Od2-T	86.62%
	Park-T+Od1-T+Od2-T	80.93%

Although we employ all 48 combined features with physical means, all features do not necessarily contribute to improve the accuracy of the classifier. Certain data also suggest that the accuracy of an SVM classifier can be improved by removing redundant features appropriately [23]. The accuracy may be improved by selecting from 48 features, and this is in our future plans.

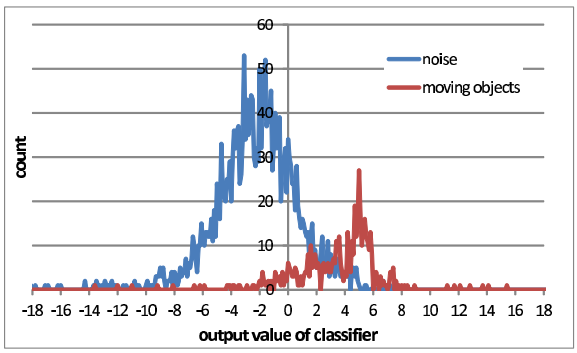
4.2 Scene Dependency of Real-AdaBoost-Based Classifier

In this subsection, we discuss the scene dependency of the Real AdaBoost classifier using the three scenes shown in Table 4. We selected a Real Adaboost classifier with 58 features, and we fixed the number of weak classifiers at 50. The threshold of the classifier was set to zero.

Table 5 shows the accuracy for each predicted data set when the classifier was trained for seven types of training data: Park-T, Od1-T, Od2-T, Park-T+Od1-T, Park-T+Od2-T, Od1-T+Od2-T, and Park-T+Od1-T+Od2-T. The high accuracy tends to be achieved when the scene of the training



(a) Sum of 64 trials of cross-validation for Park-T data set.



(b) Park-P

Fig. 15 The distribution of the output value $H(x) = \sum_i h_i(x)$ of a Real AdaBoost classifier for a parking lot scene.

data is the same as that of prediction data. Note that the accuracy occasionally increases a little when the training data of another scene is used with that of the same scene as prediction data. On the other hand, when training only the samples of different scenes, the accuracy of a classifier is significantly degenerated. Because the classifier strongly depends on the scene, the classifier should be trained with samples extracted in the same scene, and we can achieve about 78–94% accuracy for classifying superfluous masking regions and moving objects by training such samples.

The accuracy of classifier which learns using all training samples may not be the best, but it is still effective. From this characteristic, we can obtain a temporary training data for a new scene by a classifier which learns using the training data of similar scenes. Then, a training data for the new scene can be obtained easily by modifying the temporary training data with an interactive modification tool of instruction signals.

4.3 Threshold Adjustment for Reducing the Unmasked Objects

As explained in Sect. 3.2, in order to keep the false negative rate small, we determined the threshold by 64-fold cross-validation of the training data set for each scene. Then, we evaluated the accuracy for the prediction data set with the threshold.

We show the distribution of the output values of a Real AdaBoost classifier for a parking lot scene. Figure 15 (a)

Table 6 The results of superfluous mask reduction for evaluation data set.

training data	predict data	threshold	unmasked object rate	removed superfluous mask rate
Park-T	Park-P	−1.2	4.86%	58.25%
Od1-T	Od1-P	−3.8	4.96%	51.55%
Od2-T	Od2-P	0.1	4.96%	81.21%

(a) Ground truth (b) Background subtraction (c) Proposed method

Fig. 16 Example of superfluous mask removal by the proposed method for outdoor1. (a) is ground truth image in which a moving object is painted red. In (b), there are many superfluous masking based on background subtraction. The superfluous masking in (b) is reduced by the proposed method as shown in (c). The characters in each black rectangle represent the object ID, and they are used for unmasking the corresponding object [4].

Table 7 Comparison of overall performance for the scene of Od1-P.

	proposed method (SG+Real AdaBoost)		SG+SVM	SG	SLDP
	threshold = −3.8	threshold = 0			
unmasked object rate	7.37%	15.21%	14.98%	3.00%	7.60%
pose unrecognition rate	1.15%	1.15%	0.92%	1.61%	1.61%
masking area [pixels]	592925	316552	324404	1431307	774831
(ratio to proposed method)	(100.0%)	(53.4%)	(54.7%)	(241.4%)	(130.7%)

shows the distribution for sum of 64 trials of the 64-fold cross-validation using 17301 samples from the Park-T data set; Park-T is divided into 64 sets: 63 sets are used for training, and another set is used for prediction. Similarly, Figure 15 (b) shows the distribution of the output values for all samples of the Park-P data set when training on all the samples of the Park-T data set. The unmasked object rate is controllable by adjusting the threshold of the classifier although there is a trade-off relationship between unmasked objects and remained redundant masking. We can see that the distributions in Fig. 15 (b) are similar to those in Fig. 15 (a). Thus, an appropriate threshold for prediction data can be estimated by the cross validation of training data.

In this experiment, the threshold was determined so that the average of the unmasked object rate was approximately 5% for 64-fold cross-validation of the training data set. For each data set, we show the determined threshold, the unmasked object rate, and the removed superfluous mask rate in Table 6. We can see that the unmasked object rate is approximately 5%. On the other hand, the removed superfluous mask rate was reduced by this threshold adjustment. Under the thresholds, the number of superfluous masking was reduced by 58.25%–81.21%.

4.4 Results of Two-Stage Moving Object Extraction

We show an example of a superfluous mask removal using the proposed method in Fig 16. Figure 16 (a) shows a ground truth image. Figures 16 (b) and (c) show masking results in the first and second stages of our proposed method. There exist superfluous masking in Fig. 16 (b). On the other hand, by applying the proposed method, the superfluous masking was significantly removed while keeping the masking for the walking man as shown in Fig. 16 (c).

Since GMM, RRC, SRF, SLDP, and IB are designed so that they are tolerant to fluctuation, they can reduce the superfluous masking area compared to SG. However, the masking for moving objects was also reduced compared to SG. Since SLDP is the most robust to short-term sunshine fluctuation among these methods, we compared the proposed method with SLDP. In Table 7, for the scene of Od1-P, we show comparisons of overall performance between five methods: (a) the proposed method with threshold adjustment, (b) the proposed method without threshold adjustment, (c) SG which is the first stage of the proposed method, (d) a combination of SG and SVM with $\gamma = 1$ and $C = 100$, and (e) SLDP. We calculated the following three evaluation values which were visually determined. *Unmasked object rate* is the rate of moving objects whose privacy was not protected. *Pose unrecognition rate* is the rate of moving objects whose pose cannot be recognized due to over masking. *Masking area* is the total number of masked pixels.

For all methods, the poses of most moving objects could be recognized, and the pose unrecognition rate dif-

fered only slightly. The masking area of SLDP was less than that of SG, but the unmasked object rate of SLDP was 7.67% which was greater than that of SG. By applying the SVM-based classifier to the regions extracted by SG, the masking area was significantly reduced. However, the unmasked object rate was uncontrollable, and it was degenerated to 14.98%. Similarly, the proposed method without the threshold adjustment degenerated the unmasked object rate to 15.21%. By using the threshold adjustment, the proposed method kept the unmasked object rate small while the masking area was reduced to less than that of SLDP. When the threshold was set as -3.8, the second stage of the proposed method falsely removed 4.96% regions of moving objects. Note that the 4.96% regions include a part of 3.00% unmasked objects which were not sufficiently masked by SG. Therefore, the unmasked object rate increased to 7.37%. The proposed method could achieve the smallest unmasked rate and masking area of these five methods.

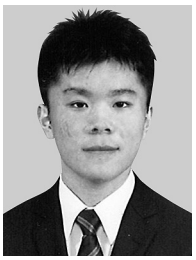
5. Conclusions

In this paper, we propose a two-stage moving object masking method to realize privacy-conscious video surveillance. By combining a background subtraction method and a Real-AdaBoost-based classifier, superfluous masking can be reduced. Moreover, the unmasked object rate can be kept small by adjusting the threshold of the classifier. This method helps us to recognize the scene in the surveillance video while protecting the privacy of moving objects. In our future work, we will search for a method to select the effective features from all combined features.

References

[1] I. Kitahara, K. Kogure, and N. Hagita, "Stealth vision for protecting privacy," Proc. 17th International Conference on Pattern Recognition (ICPR '04), vol.4, ICPR '04, pp.404–407, Washington, DC, USA, 2004.
[2] J. Wickramasuriya, M. Alhazzazi, M. Datt, S. Mehrotra, and N. Venkatasubramanian, "Privacy-protecting video surveillance," Proc. SPIE, vol.5671, pp.64–75, 2005.
[3] A. Senior, S. Pankanti, A. Hampapur, L. Brown, Y. Tian, and A. Ekin, "Blinkering surveillance: Enabling video privacy through computer vision," IBM Research Report, vol.22886, 2003.
[4] K. Yabuta, H. Kitazawa, and T. Tanaka, "Privacy protection by masking moving objects for security cameras," IEICE Trans. Fundamentals, vol.E92-A, no.3, pp.919–927, March 2009.
[5] T. Koyama, Y. Nakashima, and N. Babaguchi, "Real-time privacy protection system for social videos using intentionally-captured persons detection," 2013 IEEE International Conference on Multimedia and Expo (ICME), pp.1–6, 2013.
[6] S. Elezovikj, H. Ling, and X. Chen, "Foreground and scene structure preserved visual privacy protection using depth information," 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp.1–4, 2013.
[7] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," IEEE Trans. Pattern Anal. Mach. Intell., vol.24, no.5, pp.603–619, 2002.
[8] Y. Satoh, H. Tanahashi, S. Kaneko, Y. Niwa, and K. Yamamoto, "Robust object detection and segmentation based on radial reach correlation," MVA, pp.512–517, 2008.
[9] R. Ozaki, Y. Satoh, K. Iwata, and K. Sakaue, "Statistical reach feature method and its application to robust image registration," TENCON 2009 - 2009 IEEE Region 10 Conference, pp.1–6, 2009.
[10] S. Yoshinaga, A. Shimada, H. Nagahara, and R. Taniguchi, "Object detection using local difference patterns," Proc. 10th Asian Conference on Computer Vision, ACCV '10, pp.216–227, Berlin, Heidelberg, Springer-Verlag, 2011.
[11] F. Porikli, "Multiplicative background-foreground estimation under uncontrolled illumination using intrinsic images," IEEE Computer Society Workshop on Motion and Video Computing, Jan. 2005.
[12] L. Zhang, S. Li, X. Yuan, and S. Xiang, "Real-time object classification in video surveillance based on appearance learning," IEEE Conference on Computer Vision and Pattern Recognition, pp.1–8, 2007.
[13] M. Enzweiler and D. Gavrila, "Monocular pedestrian detection: Survey and experiments," IEEE Trans. Pattern Anal. Mach. Intell., vol.31, no.12, pp.2179–2195, 2009.
[14] D. Geronimo, A. Lopez, A. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.7, pp.1239–1258, 2010.
[15] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.246–252, 1999.
[16] Y. Weiss, "Deriving intrinsic images from image sequences," Proc. Eighth IEEE International Conference on Computer Vision, ICCV 2001, vol.2, pp.68–75, 2001.
[17] "PETS2004 benchmark data," http://www-prima.inrialpes.fr/PETS04/caviar_data.html
[18] H. Fujiyoshi and T. Kanade, "Layered detection for multiple overlapping objects (image processing and video processing)," IEICE Trans. Inf. & Syst., vol.E87-D, no.12, pp.2821–2827, Dec. 2004.
[19] S. Nagaya, T. Miyatake, T. Fujita, W. Ito, and H. Ueda, "Moving object detection by time-correlation-based background judgement method," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J79-D-II, no.4, pp.568–576, April 1996.
[20] T. Matsuyama, T. Wada, H. Habe, and K. Tanahashi, "Background subtraction under varying illumination," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J84-D-II, no.10, pp.2201–2211, Oct. 2001.
[21] R.E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," Machine Learning, vol.37, no.3, pp.297–336, 1999.
[22] C.C. Chang and C.J. Lin, "LIBSVM: A library for support vector machines," ACM Trans. Intell. Syst. Technol., vol.2, no.3, pp.27:1–27:27, May 2011.
[23] K. Aoki, S. Kuroyanagi, M. Kugler, A.S. Nugroho, and A. Iwata, "Feature selection using confident margin for SVM," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J88-D-II, no.12, pp.2291–2300, Dec. 2005.

Yoichi Tomioka received his B.E., M.E., and D.E. degrees from Tokyo Institute of Technology, Tokyo, Japan, in 2005, 2006, and 2009, respectively. He was a research associate at Tokyo Institute of Technology until 2009. Since 2009, he has been an assistant professor in the Division of Advanced Electrical and Electronics Engineering at Tokyo University of Agriculture and Technology. His research interests include image processing, security systems with mobile robots, VLSI package design automation, and combinational algorithms. He is a member of IEEE and IPSJ.



Hikaru Murakami received his B.S. degree in Electrical and Electronic Engineering from Tokyo University of Agriculture and Technology, Japan, in 2013. He is currently pursuing a master's course at the university. His research interests include image processing.

Hitoshi Kitazawa received his B.S., M.S., and Ph.D. degrees in Electronic Engineering from Tokyo Institute of Technology, Tokyo, Japan, in 1974, 1976, and 1979, respectively. He joined the Electrical Communication Laboratories, Nippon Telegraph and Telephone Corporation (NTT), in 1979. Since 2002, he has been a professor at the Tokyo University of Agriculture and Technology. His research interests include VLSI CAD algorithms, computer graphics, and image processing. He is a member of IPSJ and IEEE.