ONLINE SELECTION OF DISCRIMINATIVE PIXEL-PAIR FEATURE FOR TRACKING

Kenji Nishida
Neuroscience Research Institute
National Institute of Advanced Industrial
Science and Technology (AIST)
Central 2, 1-1-1 Umezono
Tsukuba, Ibaraki, 305-8568, Japan
Email: kenji.nishida@aist.go.jp

Takio Kurita
Neuroscience Research Institute
National Institute of Advanced Industrial
Science and Technology (AIST)
Central 2, 1-1-1 Umezono
Tsukuba, Ibaraki, 305-8568, Japan
Email: takio-kurita@aist.go.jp

Masakatsu Higashikubo
Information & Communication Laboratories
Sumitomo Electric Industries Ltd.
1-1-3, Shimaya, Konohana-ku
Osaka, 554-0024 Japan
Email: higashikubo@sei.co.jp

ABSTRACT

A novel visual tracking algorithm is proposed in this paper. The algorithm plays an important role in a cooperative driving support system (DSSS) that is aimed at reducing traffic fatalities and injuries. The input to the algorithm is a gray-scale image for every video frame from a roadside camera, and the algorithm can be used to detect the existence of vehicles on the road and then track their trajectories. In this algorithm, discriminative pixel-pair feature selection is adopted to discriminate between an image patch with an object in the correct position and image patches with objects in an incorrect position. The proposed algorithm showed stable and precise tracking performance when implemented in various illumination conditions and traffic conditions; the performance was especially good for the low-contrast vehicles running against a high-contrast background.

KEY WORDS

Visual Tracking, Discriminative Tracking, Online Feature Selection, Intelligent Traffic System

1 Introduction

According to government statistics[1], in Japan, 5,155 people were killed in 766,177 traffic accidents in 2008. Although the accidents have been decreasing over the past eight years, the prevention of traffic accidents is still one of the most critical tasks in our society. For this purpose, we are developing a cooperative driving safety support system (DSSS) comprising roadside visual sensors to recognize traffic conditions for providing driving assistance. The traffic conditions are determined on the basis of the existence (number) of vehicles on the road and their speed (moving trajectories). While we have developed vehicle detectors to estimate these conditions, more precise estimation would be possible if detection and tracking can be combined to estimate the vehicle movement. Therefore, in this study, we focus on the vehicle tracking system.

We use gray-scale images from a roadside (above the road) video camera, and therefore, color-based tracking algorithms such as the mean-shift algorithm are difficult to implement. Because of the traffic conditions on the road, the input images may contain scenarios in which it is difficult to track vehicles, such as the presence of many vehicles with similar appearances, partially occluded or low-contrast vehicles, illumination changes, and high-contrast background texture. Therefore, a robust shape-based tracking algorithm is required.

In this paper, we propose a tracking algorithm that can be used to select discriminative pixel-pair features in every video frame. The pixel-pair feature is determined by a relative difference in the intensities of two pixels[2, 3, 4]; therefore, it is considered to help realize robustness to illumination changes. Our algorithm showed good tracking performance for low-contrast vehicles; the tracking was performed by selecting pixel pairs on the basis of the discriminant criterion.

Related studies such as those on previous tracking algorithms are described in the next section, and the tracking algorithm involving discriminative pixel-pair feature selection is described in section 3. In section 4, we present our experimental results. Some variations of the discriminative pixel-pair selection are also considered.

2 Previous Visual Tracking Algorithms

More than two decades of vision research has resulted in the development of some well-known approaches for object tracking. The first is based on the background subtraction algorithm [14]. In this approach, the background is dynamically estimated from incoming images, and the difference between the current and the background images is estimated to detect the presence of vehicles. While this approach enables reliable vehicle detection in favorable illumination conditions, the performance of the background estimation is degraded in heavy traffic conditions because the movement of vehicles is small and a significant part of the background is not observable.

The second approach involves the use of a feature-based tracking algorithm [7, 6]. In this approach, salient features such as corner features are individually extracted and tracked and are grouped on the basis of the proximity of their positions to each other and the similarities in movements. This approach is robust to changes in the illumination condition. However, the difficulties in feature grouping (e.g., features of nearby vehicles not being correctly separated or features of large vehicles not being decomposed) affects the precision of the vehicle location and dimension. Although Kim[8] employed the intensity profile as the feature, Kim's approach can be classified as an approach of this type.

The third approach is called the mean-shift[9, 10] approach, in which local features (such as color histograms) of pixels corresponding to the object are followed. The mean-shift approach enables robust and high-speed object tracking, if a local feature that can be used to discriminate between the object and the background exists. However, it is difficult to discriminate between nearby objects with similar colors and to adopt this method for gray-scale images.

The fourth (and final) approach can be classified as a discriminative tracking approach. Avidan[11] redefined the tracking problem as the problem of classifying (or discriminating between) objects and the background. In this approach, features are extracted from both the objects and the background; then, a classifier is trained to classify (discriminate between) the object and the background. Hidaka[12] employed rectangle feature for their classifier and tracked objects by maximizing the classification score. Grabner[13] trained a classifier to discriminate an image patch with an object in the correct position and image patches with objects in the in-correct position, thereby, the position of the object could be estimated in higher precision. While this approach enables stable and robust object tracking, a large number of computations are necessary. The approach of Collins[5] and Mahadevan[15] is classified as an approach of this type, but they selected discriminative features instead of training classifiers.

Object trackers must cope with the appearance change (illumination change, deformation etc.), thus most of the trackers update feature set while tracking a particular object. Although updating feature set improves the tracking performance, it may affects the precision (or stability) of tracking such as drifting[16]. Therefore, Grabner[16] introduced on-line boosting to update feature weights to attain a compatibility of adaptation and stability of tracking classfiers. Woodley[17] employed discriminative feature selection using a local generative model to cope with appearance change while maintaining the proximity to a static appearance model.

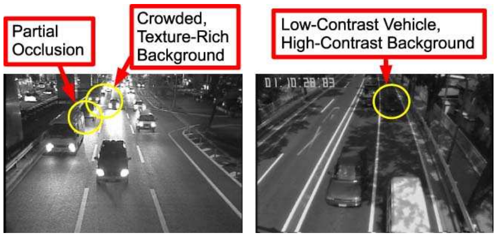

Partial Occlusion
Crowded, Texture-Rich Background
Low-Contrast Vehicle, High-Contrast Background

Figure 1. Example of input images

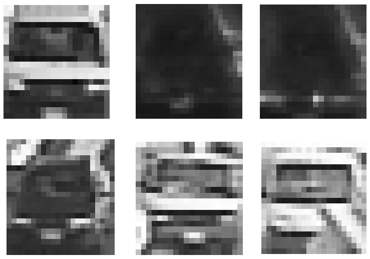


Figure 2. Example of initial image patch for objects

3 Discriminative Pixel-Pair Selection for Tracking

In this section, we first describe our system environment and then our tracking algorithm, including the definition of pixel-pair features.

3.1 System Environment

Our purpose is to develop a robust and stable vehicle tracking system for images input from a roadside video camera; the input images are gray-scale images and are obtained under various illumination conditions, such as daytime or night-time illumination conditions, varying illumination conditions during tracking, conditions with a high-contrast shadow on the road plane, and those with low-contrast vehicles (fig. 1). Because of these characteristics, the background subtraction and mean-shift approaches are difficult to adopt, and the illumination conditions necessitate robust feature extraction. In addition, the initial position of the vehicle is given by the vehicle detector as a square image patch; therefore, the initial patch for a vehicle should contain some background. Figure 2 shows examples of an initial patch; the background in the patch may contain other vehicles and high-contrast road texture (such as road-markings, shadows, and sunlight), and this makes it difficult to discriminate between the object/background.

Therefore, we employed pixel-pair feature selection, which is considered to be robust to illumination changes and redefined the tracking problem as that of discriminating between the image patch containing the object in the correct position and the image patches containing the object in an incorrect position, rather than a problem of discriminating between object/background[13].

3.2 Problem Definition

We define a tracking problem as a classification problem of obtaining an image patch that contains the object in the correct position from a new image frame. A tracking procedure is briefly illustrated in figure 3. For the t_{th} frame, the image frame $\boldsymbol{V}_t$ and a vehicle position (and scale) $\boldsymbol{L}_t$ are obtained from the $(t-1)_{th}$ frame (the initial position is given by the vehicle detector). Our tracking system can be used to crop a positive (correct) image patch $\boldsymbol{I}_t$ using $\boldsymbol{V}_t$ and $\boldsymbol{L}_t$; then, F false (incorrect) image patches $\boldsymbol{J}_t^1, \dots, \boldsymbol{J}_t^F$ surrounding $\boldsymbol{L}_t$ are cropped (figure 4 shows examples). Next, the features for discriminating between $\boldsymbol{I}_t$ and $\boldsymbol{J}_t$s are extracted, instead of training a classifier. Finally, a search for an image patch $\boldsymbol{I}_{t+1}$, which is the most similar to the positive (correct) image patch $\boldsymbol{I}_t$, is carried out from the next frame $\boldsymbol{V}_{t+1}$.

We adopt a discriminative pixel-pair feature selection for our tracking system.

3.3 Pixel-Pair Feature

The pixel-pair feature is an extension of the statistical reach feature (SRF)[2] in which the restriction on the distance between pixel pairs is removed. The definition of the pixel-pair feature and the similarity index $c(I, J)$ of a given pair of images I and J of the same size are described as follows (figure 5). Suppose the size of the input images is $W \times H$. Let grid Γ represent a set of pixel coordinates in the images I and J. To be specific,

$$\Gamma := \{(i, j) | i = 1, \dots, W, j = 1, \dots, H\}. \qquad (1)$$

We regard the image of size $W \times H$ as an intensity function defined on Γ. For an arbitrary pair (p, q) of grid points in Γ, we define the value $ppf(p \succ q; T_p)$ as follows:

$$ppf(p \succ q; T_p) := \begin{cases} 1 & \boldsymbol{I}(p) - \boldsymbol{I}(q) \geq T_p \\ -1 & \boldsymbol{I}(p) - \boldsymbol{I}(q) \leq -T_p \\ \phi & otherwise \end{cases} \qquad (2)$$

Here, $T_p (> 0)$ is the threshold of the intensity difference. We adopt the grid-point pair (p, q) as a feature when $ppf(p \succ q; T_p) \neq \Phi$. Hereafter, we write $ppf(p \succ q)$ rather than $ppf(p \succ q; T_p)$, unless there is any ambiguity.

Since we do not restrict the selection of p and q from the image I, it is possible to extract a huge number of pixel-pair features. Therefore, we limit the number of pixel-pair features to N. By selecting a set of pairs $(\boldsymbol{p}, \boldsymbol{q})$ with selection policy s, we denote a random pixel-pair-feature set

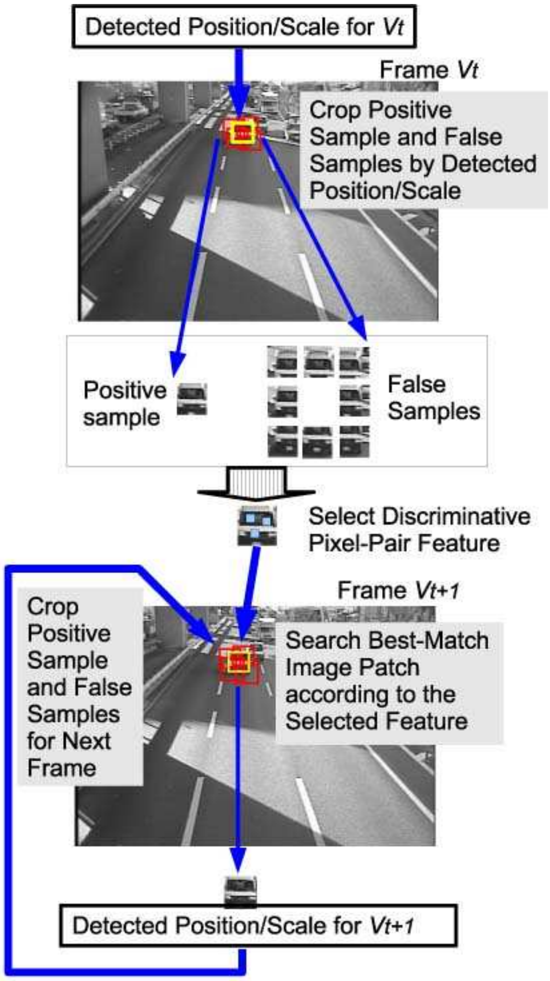

Detected Position/Scale for Vt
Frame Vt
Crop Positive Sample and False Samples by Detected Position/Scale
Positive sample
False Samples
Select Discriminative Pixel-Pair Feature
Frame Vt+1
Crop Positive Sample and False Samples for Next Frame
Search Best-Match Image Patch according to the Selected Feature
Detected Position/Scale for Vt+1

Figure 3. Tracking procedure

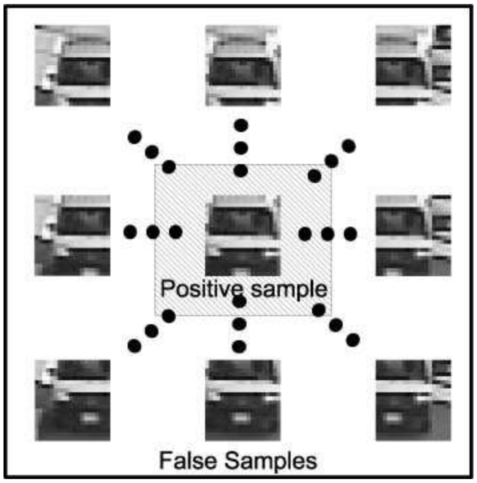


Figure 4. Examples of positive and false samples

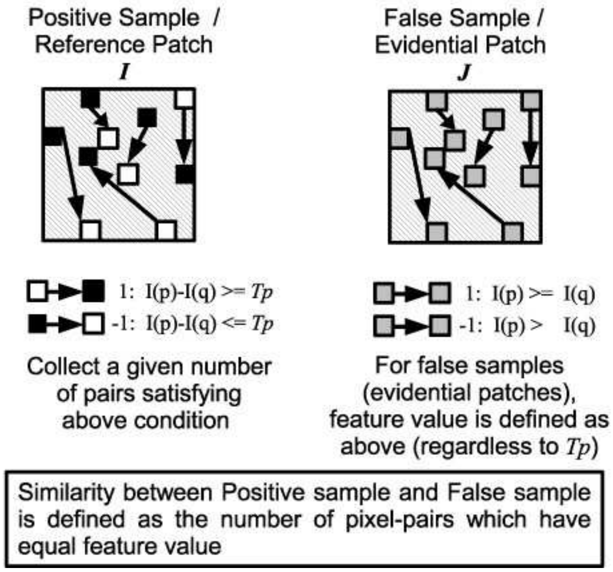

Positive Sample / Reference Patch
I
False Sample / Evidential Patch
J
1: I(p)-I(q) >= *Tp*
-1: I(p)-I(q) <= *Tp*
1: I(p) >= I(q)
-1: I(p) > I(q)
Collect a given number of pairs satisfying above condition
For false samples (evidential patches), feature value is defined as above (regardless to *Tp*)
Similarity between Positive sample and False sample is defined as the number of pixel-pairs which have equal feature value

Figure 5. Pixel-pair Features

RP_s as follows:

$$RP_s(\boldsymbol{p},\boldsymbol{q},\boldsymbol{I},T_p,N) := (\{ppf(\boldsymbol{p}\succ\boldsymbol{q}) \neq \Phi), \qquad (3)$$

where $\{\boldsymbol{p},\boldsymbol{q} \in \Gamma\times\Gamma\}$, $\boldsymbol{p} = \{p_1,\ldots,p_N\}$, and $\boldsymbol{q} = \{q_1,\ldots,q_N\}$. We define the incremental sign $b(\boldsymbol{p} > \boldsymbol{q})$ for the image $\boldsymbol{J}$ for computing the similarity between images $\boldsymbol{I}$ and $\boldsymbol{J}$ as follows:

$$b(\boldsymbol{p}\succ\boldsymbol{q}) := \begin{cases} 1 & \boldsymbol{J}(p) \geq \boldsymbol{J}(q) \\ -1 & otherwise \end{cases} \qquad (4)$$

For a pixel pair $(p,q) \in RP_s$, a single-pair similarity $r(p,q,\boldsymbol{J})$ is defined as follows:

$$r(p,q,\boldsymbol{J}) = \{ppf(p\succ q) = b(p\succ q)\}. \qquad (5)$$

The similarity index $c_s(\boldsymbol{I},\boldsymbol{J},RP_s)$ measured by using a pixel-pair feature set RP_s is defined as follows:

$$c_s(\boldsymbol{I},\boldsymbol{J},RP_s) = \frac{\sum_{(p,q)\in RP_s} r(p,q,\boldsymbol{J})}{|RP_s|} \qquad (6)$$

3.4 Discriminative Pixel-Pair Selection

Pixel-pair features are selected to maximize the discriminant criterion used for discriminating between a correct image patch $\boldsymbol{I}$ and incorrect image patches $\boldsymbol{J}$s.

According to our problem definition, the discriminant criterion is defined for following condition:

- the feature takes binary values $+v, -v$
- only ONE positive sample exists
- a large number F of false samples exist

The feature values for the positive sample p and false samples n^i are defined as

$$p = v, \quad \{n^i\}_1^F = v, -v. \qquad (7)$$

Assuming $F \gg 1$, the total mean $\bar\mu_T$ is nearly equal to the mean for false samples $\bar\mu_n$. Defining m as a number of false samples of which values are $n^i = -v$,

$$\begin{aligned} \bar\mu_T \approx \bar\mu_n &= \frac{1}{F}\sum_1^F n^i \\ &= \frac{1}{F}(\sum_1^m(-v) + \sum_1^{F-m} v) \\ &= \frac{1}{F}(F-2m)v. \qquad (8) \end{aligned}$$

The total variance and inter-class variance are defined as follows:

$$\begin{aligned} \sigma_T^2 &= \frac{1}{F+1}\{(v-\bar\mu_T)^2 + (\sum_1^F(n^i-\bar\mu_T)^2\} \\ &\approx \frac{1}{F+1}\{(v-\bar\mu_n)^2 + (\sum_1^F(n^i-\bar\mu_n)^2\} \qquad (9) \end{aligned}$$

$$\begin{aligned} \sigma_B^2 &= \frac{1}{F+1}(v-\bar\mu_T)^2 + \frac{F}{F+1}(\bar\mu_n-\bar\mu_T)^2 \\ &\approx \frac{1}{F+1}(v-\bar\mu_n)^2 + \frac{F}{F+1}(\bar\mu_n-\bar\mu_n)^2 \\ &= \frac{1}{F+1}(v-\bar\mu_n)^2. \qquad (10) \end{aligned}$$

Therefore, the discriminant criterion (ratio of inter-class variance to the total variance) is defined as

$$\frac{\bar\mu_B^2}{\bar\mu_T^2} = \frac{(v-\bar\mu_n)^2}{(v-\bar\mu_n)^2 + (\sum_1^F(n_i^F-\bar\mu_n)^2}. \qquad (11)$$

This equation indicates that minimizing the variance of false samples is equivalent to maximizing the discriminant criterion.

The variances of false samples are redefined by substituting $\bar\mu_n = \frac{1}{N}(F-2m)v$ as

$$\begin{aligned} \sum_1^F(n^i-\bar\mu_n)^2 &= \sum_1^m(-v-\bar\mu_n)^2 + \sum_1^{F-m}(v-\bar\mu_n)^2 \\ &= \frac{4v^2}{F}m(F-m). \qquad (12) \end{aligned}$$

The variance of false samples attains the minimum value zero at $m = 0$ and $m = F$. Since $m = 0$ implies that all the false samples have the same value as the positive sample, discrimination is impossible. At $m = F$, the discriminant criterion is maximized to one, where the similarity between a positive sample and a false sample is minimized to 0. Therefore, minimizing the single-pair similarity index for a pixel-pair feature $c_s(\boldsymbol{I}, \boldsymbol{J}, RP_s)$ is equivalent to maximizing the discriminant criterion. It is also equivalent to minimizing the sum of similarity indices for a feature set C_{min} as follows:

$$C_{min} = \sum_{i=1}^{F} \{c_{min}(\boldsymbol{I}, \boldsymbol{J}^i, RP_{min})\}, \tag{13}$$

where C_{min}, c_{min}, and RP_{min} represent the selection policy for minimizing the single-pair similarity for each pixel pair in the feature set.

Three implementations of discriminative pixel-pair selection were considered for our experiment:

1. Randomly generate pixel-pair features and collect the features whose similarity indices c_s are smaller than a certain threshold.

2. Randomly generate more pixel-pair features than needed; then, select the number of required features from among those with similarity indices less than c_s.

3. Randomly generate a certain number of pixel-pair feature sets; then, select the feature set with the lowest similarity index as the set C_s.

We adopted the third implementation in our experiment.

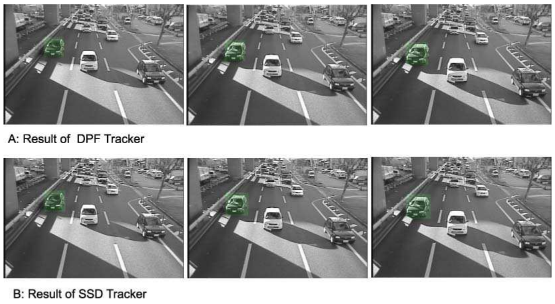
4 Experiment

In this section, we present the performance results of our system for challenging conditions, such as illumination changes for a vehicle, presence of a low-contrast vehicle, a vehicle with partial occlusion, and night-time crowded traffic. To validate our algorithm based on the discriminative pixel-pair feature (DPF tracker), the results are compared with the results of tracking algorithm based on the least sum of the squared difference (SSD tracker).

The parameters for our DPF tracker are as follows: threshold of intensity difference T_p, is 20 (range: 0–255); the number of false samples, F, is 120 (implying that the false samples are extracted from the region within ± 5 pixels of the positive sample region in the horizontal or vertical directions); the number of pixel-pair features, N, is 200, regardless to the patch size of the object region. 100 feature sets are generated, and the one with the lowest similarity index C_s is adopted.

4.1 Illumination Change for a Vehicle

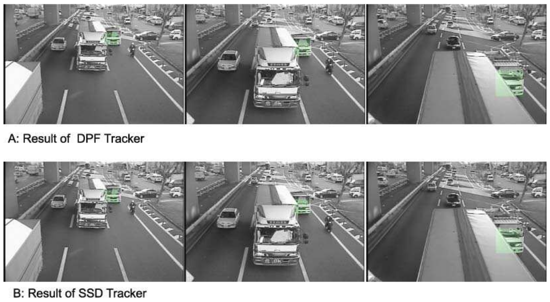
Figure 6 shows the result in the case of illumination changes for the tracking vehicle. The green-shaded area indicates the tracked vehicle position. The black car in the left lane is tracked, and the patch size for the object is 100×100 in the leftmost video frame. In the SSD tracker, obviously, all the pixels in the image patch are used, and hence, 10,000 pixel are compared as features. The results for the DPF tracker and the SSD tracker are almost the same, even though a small number of features are used in the former (200).



A: Result of DPF Tracker

B: Result of SSD Tracker

Figure 6. Tracking result: illumination change



A: Result of DPF Tracker

B: Result of SSD Tracker

Figure 7. Tracking result: partial occlusion

4.2 Partial Occlusion

Figure 7 shows the result for a partially occluded vehicle. The tracking is very poor when the SSD tracker is used and the precision (especially for the scale) is not sufficient, while the object was successfully tracked when the DPF was used.

4.3 Crowded Night-Time

Figure 8 shows the result for the night-time conditions, yellow area indicates manually defined ground-truth, while green area indicates tracker estimation. Since the initial patch includes some parts of the other vehicles as background, the SSD estimations are affected by the background (parts of other vehicles), and thus, the scale of estimations are larger than ground-truth. While the scale of estimations by the DPF tracker are larger than ground-truth, the estimations are more sufficient than those by the SSD tracker.

Figure 9 shows the relative position error: (position error)/(scale of ground-truth) and figure 10 shows the scale error: (estimation size)/(ground-truth size). These results indicates that the DPF tracker attains higher precision with both of the position and scale estimation.

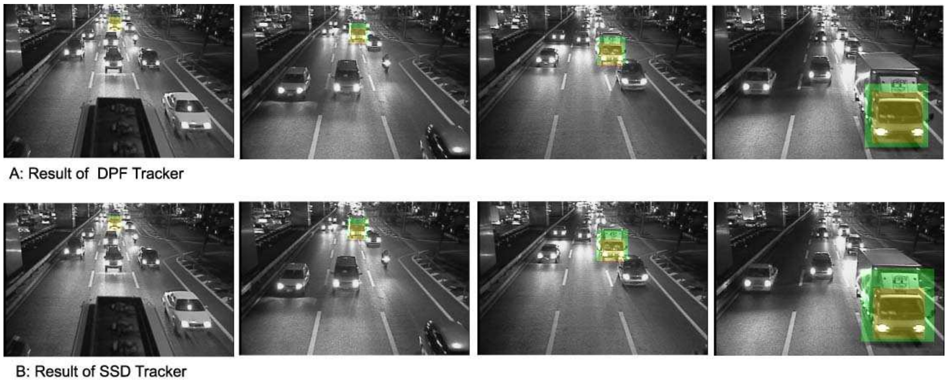


Figure 8. Tracking result: crowded traffic at night

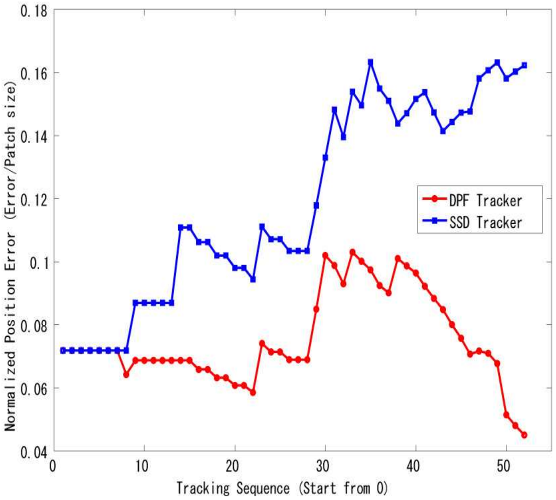


Figure 9. Relative position error in tracking

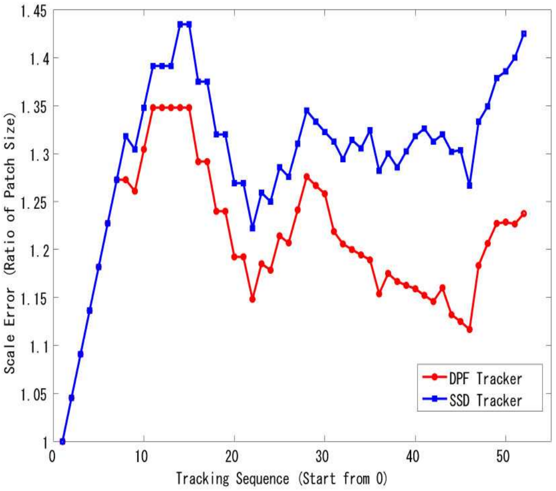


Figure 10. Scale error in tracking

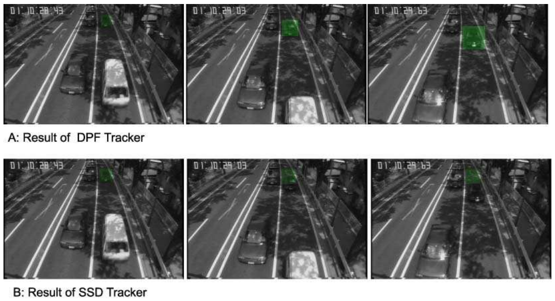


Figure 11. Tracking result: low contrast vehicle

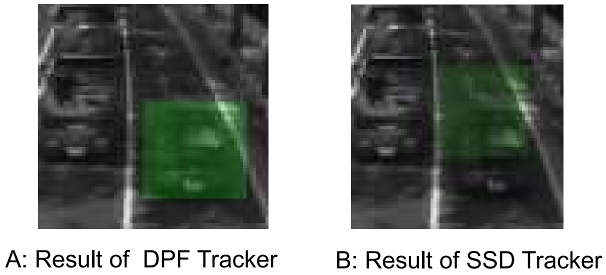


Figure 12. Tracking miss by SSD tracker

4.4 Low-Contrast Vehicle

Figure 11 shows the results for a low-contrast vehicle. The black car in the right lane is tracked. The DPF tracker was successfully used to track the vehicle, while when the SSD tracker was used, the vehicle was missed in an early frame. Figure 12 shows a magnified version of the second video frame of figure 11. The result indicates that the SSD tracker was affected by the high-contrast shadow in the background (road plane), while the DPF tracker showed robustness to the background texture.

Figure 13 and figure 14 shows the confidence map for the first frame of the sequence in figure 11 by using DPF tracker and SSD tracker. The confidence map by DPF

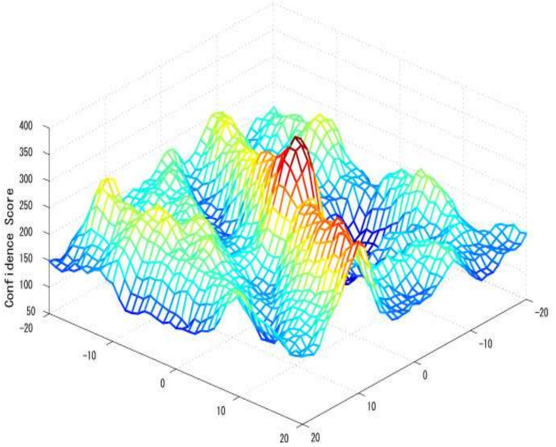


Figure 13. Confidence map for DPF tracker

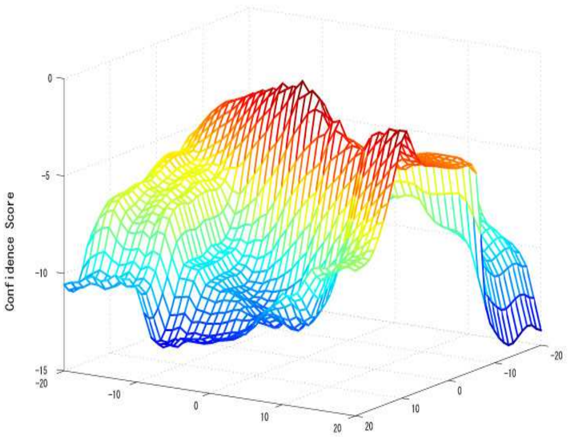


Figure 14. Confidence map for SSD tracker

tracker shows a significant peak, while SSD tracker could not show such peak in confidence. This result indicates that selection of discriminative pixel-pair feature improves the precision of tracking.

4.5 The Effect of T_p

Figure 15 shows the effect of the threshold T_p. We used the same sequence as that in subsection 4.4. For $T_p = 50$, the tracking failed in the early video frame, and the tracking is considered to be affected by the difference between the background with high-contrast texture and the low-contrast vehicle. For $T_p = 20$, the optimal tracking performance was obtained (this is why we set $T_p = 20$ for other experiments), and for $T_p = 10$, the object was tracked almost completely, except for at the end. The extraction of pixel-pair features with low T_p seems to be too sensitive to insignificant difference in intensities, such as the fluctuations caused by noise.

5 Conclusion

We proposed a novel algorithm for object tracking. In the proposed method, pixel-pair features that are considered to be robust to changes in illumination conditions are adopted along with discriminative feature selection. The pixel-pair features are selected to discriminate between the image patch containing the object in the correct position and image patches containing the object in an incorrect position. The proposed algorithm showed a stable and precise tracking performance with high robustness to changes in illumination and traffic conditions, especially for low-contrast vehicles. We also evaluated the effect of a threshold parameter for pixel-pair features; this is important for tracking of low-contrast vehicles. We are now developing a cooperative detection and tracking system to help recognize traffic conditions.

References

[1] `http://www.npa.go.jp/toukei/koutuu48/H20.All.pdf`

[2] R.Ozaki, Y.Satoh, K.Iwata, K.Sakane, “Statistical Reach Feature Method and Its Application to Template Matching”, in *Proc MVA 2009*, pp.174-177, 2009.

[3] S. Kaneko, et al., gRobust image registration by increment sign correlationh,*Pattern Recognition*, vol.35, no.10, pp.2223-2234, 2002.

[4] M.Özuysal, P.FUa, V.Lepetit, “Fast Keypoint Recognition in TenLines of Code”, in *Proc. CVPR 2007*, 2007.

[5] R.T.Collins, Y.Liu, M.Leordeanu, “Online Selection of Discriminative Tracking Features”, in *IEEE PAMI*, Vol.27, No.10, pp.1631-1643, 2005.

[6] B.Coifman, D.Beymer, P.Maclauchlan, J.Malik, “A real-time computer vision system for vehiclr tracking and traffic surveillance”, *Transportation Research Part C*, No.6, pp.271-288, 1998.

[7] D.Beymer, B.Coifman, J.Malik, “A Real-Time Computer Vision System for Measuring Traffic Parameters”, in *proc. IEEE CVPR*, pp.495-501, 1997.

[8] Z.Kim, J.Malik, “Fast Vehicle Detection with Probabilistic Feature Grouping and its Application ot Vehicle Tracking”, in *Proc. ICCV*, 2003.

[9] D.Comeniciu, P.Meer, “MeamShift: A Robust Approach Toward Feature Space Analysis”, *IEEE PAMI*, Vol.24, No.5, pp.603-619, May, 2002.

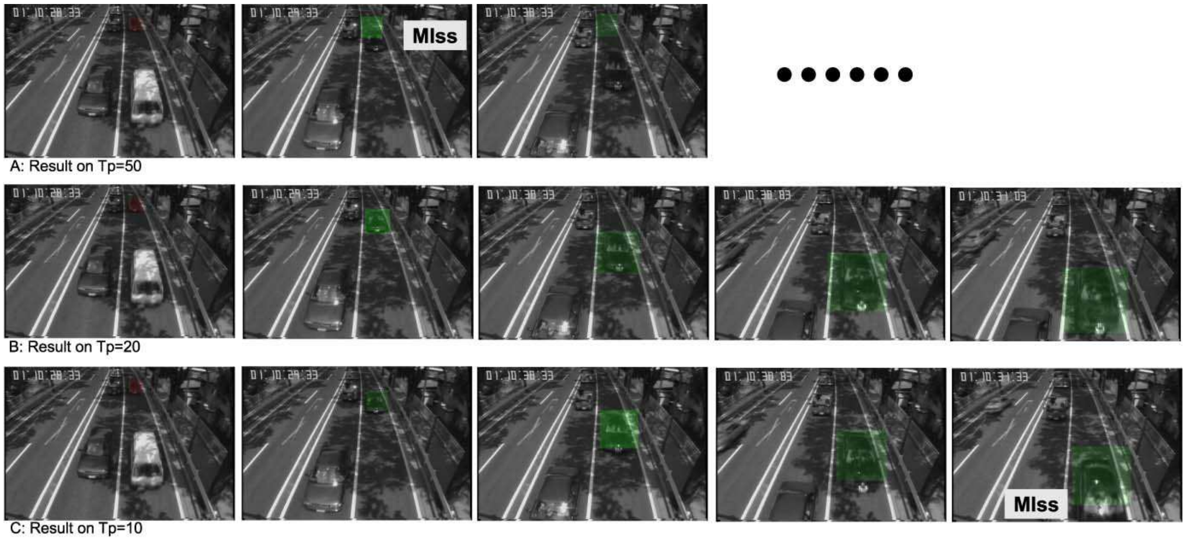


Figure 15. Effect of T_p for low-contrast vehicle

[10] D.Comaniciu, V.Ramesh, P.Meer, “Real-time tracking of non-rigid objects using mean shift”, in *proc. CVPR 2000*, Vol.2, pp.142-149, 2000.

[11] S.Avidan, “Ensemble Tracking”, *IEEE PAMI*, Vol.29, No.2, pp.261,271, 2007.

[12] Akinori Hidaka, Kenji Nishida, and Takio Kurita, “Object Tracking by Maximizing Classification Score of Detector Based on Rectangle Features,” *IEICE Trans. on Information and Systems*, Vol.E91-D, No.8, pp.2163-2170, 2008.8.

[13] H.Grabner, M.Grabner, H.Bischof, “Real-Time Tracking via On-line Boosting”, in *Proc. BMVC*, pp.47-56, 2006.

[14] D.Koller, J.Weber, J.Malik, “Robust multiple Car Tracking with Occlusion Reasoning”, in *proc. ECCV*, Vol.A, pp.189-196, 1994.

[15] V.Mahadevan, N.Vasconcelos, “Salliency-based Discriminant Tracking”, in *proc.of CVPR 2009*, pp.1007-1013, 2009.

[16] H.Grabner, C.Leistner, H.Bischof, “Semi-Supervised On-Line Boosting for Robust Tracking”, in *Proc. ECCV 2008*, 2008.

[17] T.Woodley, B.Stenger, R.Chipolla, “Tracking using Online Feature Selection and a Local Generative Model”, in *Proc. BMVC 2007*, 2007.