| | |
|---|---|
| Title | Scale-adaptive spatial appearance feature density approximation for object tracking |
| Author(s) | Liu, CY; Yung, NHC |
| Citation | Ieee Transactions On Intelligent Transportation Systems, 2011, v. 12 n. 1, p. 284-290 |
| Issued Date | 2011 |
| URL | http://hdl.handle.net/10722/137288 |
| Rights | IEEE Transactions on Intelligent Transportation Systems. Copyright © IEEE. |

# Scale-Adaptive Spatial Appearance Feature Density Approximation for Object Tracking

C. Y. Liu and N. H. C. Yung, *Senior Member, IEEE*

***Abstract*—Object tracking is an essential task in visual traffic surveillance. Ideally, a tracker should be able to accurately capture an object's natural motion such as translation, rotation, and scaling. However, it is well known that object appearance varies due to changes in viewing angle, scale, and illumination. They introduce ambiguity to the image cue on which a visual tracker usually relies and which affects the tracking performance. Thus, a robust image appearance cue is required. This paper proposes scale-adaptive spatial appearance feature density approximation to represent objects and construct the image cue. It is found that the appearance representation improves the sensitivity on both the object's rotation and scale. The image cue is then constructed by both the appearance representation of the object and its surrounding background such that distinguishable parts of an object can be tracked under poor imaging conditions. Moreover, tracking dynamics is integrated with the image cue so that objects are efficiently localized in a gradient-based process. Comparative experiments show that the proposed method is effective in capturing the natural motion of objects and generating better tracking accuracy under different image conditions.**

***Index Terms*—Gaussian mixture model (GMM), image cue, object appearance representation, tracking, traffic surveillance.**

## I. Introduction

The growing demand on safety and efficiency in transportation systems requires improvements in traffic management and control. As such, it increases the reliance on traffic surveillance, of which vehicle tracking is one of its fundamental components. By tracking vehicles in a scene, detailed traffic parameters can be derived. Such information helps determine traffic abnormalities, which are a prerequisite for incident detection and management. For instance, in [1] and [2], tracking was used for event detection. In [3] and [4], tracking was applied to accident prediction. In [5] and [6], object tracks were used to learn vehicle activities. In [7] and [8], tracking was used for vehicle path prediction and driver assistance, respectively.

In a nutshell, visual-tracking methods track visual objects by their image appearance cues. These image cues could be either image features directly extracted from the image or an appearance model learned from a training image set. However, it is common knowledge that object appearance varies subject to object's translation, self-rotation, and scale (as seen by the camera) or even a change in ambient illumination or partial occlusion. Self-rotation modifies the visible parts of the object, change in object scale alters the scale of its image features, and uneven or changing illumination abruptly or gradually affects the image color/intensity as well as the saturation. Furthermore, images may be captured under poor imaging conditions (e.g., less favorable weather) or with a background of similar color. These decrease the distinctness of the object features and introduce ambiguity to the image cue. As a result, tracking under such conditions may become unreliable.

Among the tracking system in transportation surveillance, several of them rely on background subtraction, as background/foreground difference is a direct image cue of moving objects. Examples of such a tracking system can be found in [9]–[13]. However, background subtraction is susceptible to illumination variation due to shadowing or day/night changes. Shadow removal is often required in such an approach. In addition, it is scene specific, i.e., the background model must be retrained when applied to a different environment. Moreover, nearby-object merging is common; thus, in [13], foreground feature points are used as an attempt to alleviate this problem.

In addition to foreground/background difference, the object's image cue for tracking can be obtained by the appearance of the object itself. For instance, in a traffic surveillance system, the popular wire-frame methods (e.g., [14] and [15]) represent objects by a predefined 3-D edge frame and fit image edges to the model. Region-based methods (e.g., [16] and [17]) track an object by matching the candidate region to a template. Those methods assume the consistency and reliability of low-level feature detection, but the assumption is always violated by the changes in illumination, object scale, and rotation and the consequent appearance variation. To alleviate the effect of appearance variations, in [18] and [19], principle component and independent component analyses are trained to represent the image object and provide cues for detection and tracking. However, for each type of object, such methods require an image set to do offline training, and they are most applicable for texture-rich objects. This limits transferability on different types of traffic objects.

Meanwhile, in visual surveillance for a transportation system, recent developments on visual tracking improve its performance by imposing richer object information in the image cue. This could be done by either fusing different types of features or imposing prior knowledge by an object detector. For instance, in the feature-integration-based method [20], color and edge tracking are fused in a particle filter framework to improve robustness. In [21], shape, texture, and depth are integrated for tracking. However, without any notion of the object's surrounding background, these trackers have a tendency to be distracted if the object and its surrounding background contain similar color, edge, or texture features. In detection-based methods [22]–[25], either pretrained detectors or color and shape assumptions are used as image cues for tracking. Such image cues usually contain reliable knowledge of object appearance, but offline training and prior assumptions limit it in online tracking initialization and objects of different types and views.

On the other hand, feature density approximation (FDA)-based object representations have emerged as an effective method for tracking. They can be trained by densely sampled features, of which one frame can provide enough training features for online initialization. As the dense features naturally cover the range of feature variation, FDA methods can potentially accommodate object appearance variations. This is illustrated by the following FDA methods. In parametric FDA [26], object feature density is approximated by a Gaussian mixture model (GMM), which can handle the object's multimode feature distribution and appearance variations. The number of GMM components is determined by the minimum description length, and the mixture parameters are trained by expectation maximization (EM) [27]. However, their tracking dynamics is computationally complex. For nonparametric approaches [28], they approximate object feature density by the sum of kernel functions. They are efficient in learning but not efficient in prediction, as it needs to recall the learning feature set. Another example is the tracker [29] that uses a kernel weighted

Manuscript received October 18, 2009; revised April 7, 2010, July 6, 2010, August 18, 2010, and October 11, 2010; accepted October 23, 2010. Date of publication December 3, 2010; date of current version March 3, 2011. This work was supported in part by the postgraduate studentship of the University of Hong Kong. The Associate Editor for this paper was M. Á. Sotelo Vázquez.

The authors are with the Laboratory for Intelligent Transportation Systems Research, Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam, Hong Kong (e-mail: cyliu@eee.hku.hk; nyung@eee.hku.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2010.2090871

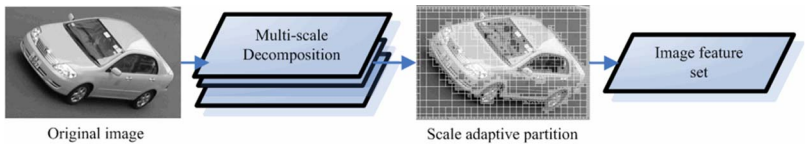


Fig. 1. Scale selection and feature extraction.

histogram to represent object appearance and efficiently locates objects by mean shift [30]. However, it fails to extract object scale and rotation. As an improvement [31], a 3-D kernel spreading on both spatial and scale domains is constructed. It tracks object and extracts its scale by performing mean shift both spatially and across scale. However, it does not work with object rotation. To extract both scale and rotation [32], an object is represented as an ellipsoidal region, whose location, scale, and rotation are determined by the mean vector and covariance matrix of the pixels in the ellipse. This histogram similarity cue is not always reliable when there is a change in illumination.

In summary, FDA methods track objects by online initialization and cope with appearance variations. There is an underlying assumption that the feature density similarity is a reliable cue. However, with poor imaging conditions, the similarity cue becomes less distinguishable. Furthermore, commonly, objects substantially change scale in wide-area surveillance, and feature scale also varies with the object's scale. The image features in fixed scale used by current FDA methods become less precise, which could be propagated to subsequent density estimation, making the appearance cue inconsistent. For the preceding reasons, this paper proposes a robust tracker that can be initialized online; work with the object's translation, rotation, and scaling; and cope with similar object/background color or poor imaging conditions.

This paper is organized as follows: Section II describes an overview of the proposed method, the advantages of the method, and a detailed description of it. Section III covers the model verification and tracking evaluation. This paper is concluded in Section IV.

## II. Proposed Method

### A. Overview

To improve the performance of FDA methods for object tracking, the proposed method is able to capture object translation, scale, and rotation under poor imaging conditions and generate a stable track. The scale-adaptive spatial appearance (SASA)-FDA uses a GMM to approximate object feature density. It represents object appearance with its spatial layout. Scale-adaptive feature extraction and consistent GMM estimation are proposed to solve the effect of changing scale. To handle the adverse imaging conditions, the cue for tracking is constructed by the GMM appearance representation of both the target object and its surrounding background. Based on this cue, this paper also presents the integrated tracking dynamics, which can lock the object in just a few iterations.

### B. Scale-Adaptive Feature Extraction

The features that represent the local structural pattern are bounded by its inner and outer scales [33], which are the scale space where they appear and the local support window where they spread, respectively. Clearly, the variation in object image scale across different frames alters the scale where the local feature pattern exists. To overcome this, we use scale-adaptive features for both inner and outer scales. In addition, as the feature's outer scale is proportional to the spatial composition of the object, we define the effective probability according its outer scale to control the contribution of the features in building the appearance model. Therefore, the appearance model is constructed the same way as the features spatially composing the object.

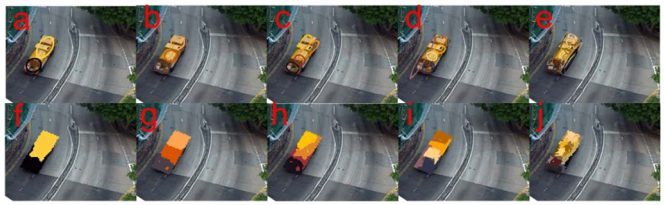


Fig. 2. Model visualization.

In scale-adaptive feature extraction, the method maps the image to its multiscale space and then selects the scale locally adapting to the image local feature. Linear scale theory [33] is chosen for multiscale image mapping, and the minimum entropy principle [34] is used for scale selection.

As shown in Fig. 1, one image is partitioned into patches with different inner and outer scales, and then, scale-adaptive features are extracted from the patches. The frame partition and scale selection procedure is given as follows.

1) Map the image into its multiscale space.
2) Initialize the partition into the largest patches and coarsest scale.
3) Split each patch, and compare the entropy of each split patch in the finer scale to its original scale.
4) Select the patch across scale by the minimum entropy principle.

After scale selection, features are extracted from these patches. They are average color, entropy of the difference of Gaussian (DOG) coefficients, average gradient orientation, and spatial coordinates of the patch center. The effective probability $P_i^{\text{creb}}$ of each feature $\mathbf{x}_i$ is defined as

$$P_i^{\text{creb}} = Size_i / Size_{\max} \tag{1}$$

where $Size_i$ is the spatial size of patch $i$, and $Size_{\max}$ is the largest patch size in one frame. Thus, the features from a larger patch play a larger role in estimating the GMM.

### C. GMM-Based Appearance Representation and Its Visualization

Given a set of scale-adaptive features $\mathbf{X} = \{\mathbf{x}_i\}$, $i = 1, \ldots, N$, GMM as defined here is used to approximate its feature density

$$P(\mathbf{x}_i) = \sum_{j=1}^{M} g(\mathbf{x}_i|\theta_j)p(\boldsymbol{\omega}_j) \tag{2}$$

where $p(\boldsymbol{\omega}_j)$ is the mixture coefficient of Gaussian component $g(\mathbf{x}_i|\theta_j)$. The spatial layout of the GMM representation is shown in Fig. 2.

As shown in Fig. 2, when the spatial coordinates are included in the features, each Gaussian component can be indicated as an ellipse (subimages a–e), which corresponds to an image region with a certain spatial layout on the object (subimages f–j).When the number of GMM

components is chosen as 2, 3, 4, 5, and 20, the object representation has a different degree of detail.

### D. Consistent GMM-Based FDA With Feature Effective Probability

To improve the FDA accuracy, we equal the contribution of the features in FDA to the proportions in which their patch composes the object. To achieve this, the log likelihood function is defined by both the feature set $\{\mathbf{x}_i\}$ and the effective probability $\{P_i^{\text{creb}}\}$ as

$$L(\boldsymbol{\theta}) = \log \prod_{i=1}^{N} P(\mathbf{x}_i, \mathbf{z}_i; \boldsymbol{\theta}) P_i^{\text{creb}} = \log \prod_{i=1}^{N} \prod_{j=1}^{M} \left[ P(\mathbf{x}_i | z_{ij} = 1; \boldsymbol{\theta}) P(z_{ij} = 1) P_i^{\text{creb}} \right]^{z_{ij}} \quad (3)$$

where $\mathbf{z_i} = [z_{i1}, \ldots, z_{iM}]^T$, and $z_{ij} = \{1, 0\}$ indicates that $\mathbf{x}_i$ is generated by which Gaussian components. The EM algorithm for GMM estimate with the effective probability $P^{\text{creb}}$ is

$$E - step: \quad h_{ij} = E[z_{ij} | x_i, \boldsymbol{\theta}^k] = p(\boldsymbol{\omega}_j) g(\mathbf{x}_i | \theta_j) / \sum_{l=1}^{M} p(\boldsymbol{\omega}_j) g(\mathbf{x}_i | \theta_l) \quad (4)$$

$$M - step: \quad p(\boldsymbol{\omega}_j) = \frac{\sum_{i=1}^{N} h_{ij} P_i^{\text{creb}}}{\sum_{i=1}^{N} P_i^{\text{creb}}} \qquad \boldsymbol{\mu}_j^{k+1} = \frac{\sum_{i=1}^{N} h_{ij} \mathbf{x}_i P_i^{\text{creb}}}{\sum_{i=1}^{N} h_{ij} P_i^{\text{creb}}}$$

$$\sum_{j}^{k+1} = \frac{\sum_{i=1}^{N} h_{ij} \left( \mathbf{x}_i - \boldsymbol{\mu}_j^{k+1} \right) \left( \mathbf{x}_i - \boldsymbol{\mu}_j^{k+1} \right)^T P_i^{\text{creb}}}{\sum_{i=1}^{N} h_{ij} P_i^{\text{creb}}} \quad (5)$$

where $g(\mathbf{x}_i | \theta_l)$ is one Gaussian component in (2).

Fig. 3 provides an illustration of the role of $P^{\text{creb}}$ in GMM estimation. Subimage (a) is the test image, and its intensity is used for GMM estimation. Subimage (b) shows the result of scale-adaptive feature extraction where the test image is partitioned into image patches of different sizes (outer scale), the patch mean intensity is used as its feature, and the effective probability is defined by its size. The GMM estimation based on the patch mean intensity should be similar to the image histogram [as shown in (c)]. In (d), the GMM estimation without the effective probability is shown, whereas (e) is the GMM estimation with effective probability. Clearly, without effective probability [in (d)], the estimate is driven to the brighter ones, as they have a larger patch number, in spite of composing a small part in the image. In (e), the features contribute according to their effective probability, and the estimate is closer to the histogram.

### E. WLR Cue for Tracking and the Tracking Dynamics

When the object and its surrounding are similar in appearance, the object image feature contains overlapping distribution with its surrounding. It makes the tracking result oscillatory. To solve the problem, a notion of the surrounding background is necessary. Based on this idea, we construct the tracking image cue as a weighted log likelihood ratio (WLR) by both the object GMM and its surrounding GMM. The object is located by searching the maximum WLR region.

To associate the object appearance model with its location, a spatial kernel is used to weigh the feature's likelihood ratio, and the object bounding window can be driven from an initial location $\mathbf{m}_s$ to the

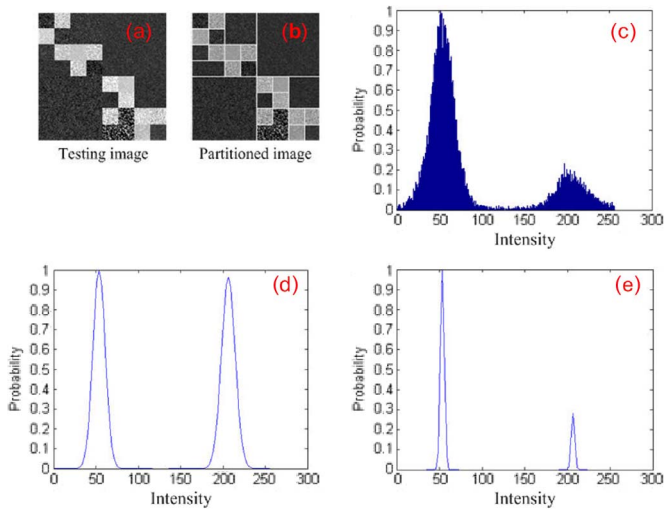


Fig. 3. Model estimation with feature effective probability.

WLR location $\boldsymbol{\mu}_s$ in a gradient-based search. As such, the tracking problem can be formulated as the optimization problem

$$\boldsymbol{\mu}_s = \arg\max_{\mathbf{m}_s} \left\{ C_{\mathbf{m}_s} = \sum_{\mathbf{x}_i \in X_{\mathbf{m}_s}} \log \left[ \frac{L_{\text{obj}}(\mathbf{x}_i | \mathbf{m}_s)}{L_{bc}(\mathbf{x}_i | \mathbf{m}_s)} \right] \cdot K\left( |\mathbf{s}_i - \mathbf{m}_s| \right) \right\} \quad (6)$$

where $C_{\mathbf{m}_s}$ is the cost function defined by the WLR, $\boldsymbol{\mu}_s$ is the optimum bounding window location to be determined, $X_{\mathbf{m}_s}$ is the feature set from the current bounding window centered in $\mathbf{m}_s$, and $\mathbf{s}_i$ is the spatial coordinates of each feature vector $\mathbf{x}_i$. $L_{\text{obj}}(\cdot)$ and $L_{bc}(\cdot)$ are the log likelihoods of a feature evaluated on the object GMM and its surrounding GMM, respectively, and $K(|\mathbf{s}_i - \mathbf{m}_s|)$ is the spatial kernel defined as

$$K\left( |\mathbf{s}_i - \mathbf{m}_s| \right) = b_s - (\mathbf{s}_i - \mathbf{m}_s)^T \boldsymbol{\Sigma}_s^{-1} (\mathbf{s}_i - \mathbf{m}_s) \quad (7)$$

where $b_s$ is the normalizing constant to make the kernel nonnegative, and $\boldsymbol{\Sigma}_s$ is a positive-definite matrix estimated from the size of the current bounding window. The derivative of the kernel is linear to object motion. The optimization problem can be solved by setting the derivative of $C_{\mathbf{m}_s}$ to zero w.r.t. to $\mathbf{m}_s$, which is

$$\nabla C_{\mathbf{m}_s} = \sum_{\mathbf{x}_i \in X_{\mathbf{m}_s}} \left\{ \log \left[ L_{\text{obj}}(\mathbf{x}_i | \mathbf{m}_s) \right] - \log \left[ L_{bc}(\mathbf{x}_i | \mathbf{m}_s) \right] \right\} \cdot \left[ \boldsymbol{\Sigma}_s^{-1} (\mathbf{s}_i - \mathbf{m}_s) \right]. \quad (8)$$

Setting $\nabla C_{\mathbf{m}_s}$ to zeros, the current optimum location $\boldsymbol{\mu}_s$ is

$$\boldsymbol{\mu}_s = \frac{\sum_{\mathbf{x}_i \in X_{\mathbf{m}_s}} \mathbf{s}_i \cdot \left\{ \log \left[ L_{\text{obj}}(\mathbf{x}_i | \mathbf{m}_s) \right] - \log \left[ L_{bc}(\mathbf{x}_i | \mathbf{m}_s) \right] \right\}}{\sum_{\mathbf{x}_i \in X_{\mathbf{m}_s}} \left\{ \log \left[ L_{\text{obj}}(\mathbf{x}_i | \mathbf{m}_s) \right] - \log \left[ L_{bc}(\mathbf{x}_i | \mathbf{m}_s) \right] \right\}}. \quad (9)$$

In (9), the log likelihood is evaluated on (3).

In summary, given a GMM density appearance model and a current starting location of the bounding window, we can estimate the next best location from the features within the current bounding window according to (9).

Fig. 4 shows the tracking dynamics and WLR in the bounding window per iteration from an initial location to the maximum WLR location. a1–a4 are the locations of the bounding window at each

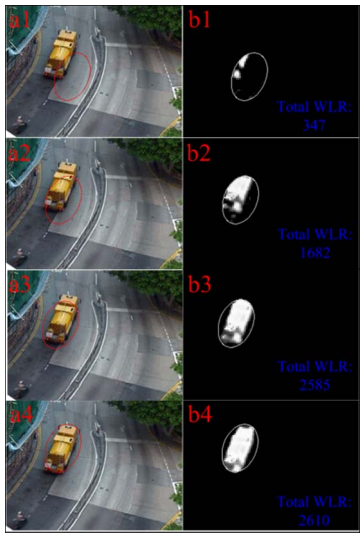


Fig. 4. Tracking dynamics.

iteration, b1–b4 show the WLR, and the total WLR in the bounding window is shown in blue. As shown in a1, the initial location of the bounding window is away from the object. In subsequent evaluations, the object features have larger WLR value than the background; thus, the bounding window is driven toward the object. After four iterations, the bounding window locked the object, and the WLR reaches the maximum.

Fig. 5 compares the effectiveness of the proposed WLR cue with the Histogram Bhattacharyya Coefficient Distance (HBCD) w.r.t. translation, rotation, and scaling. The learning and testing are conducted on frames with 20 intervals. As indicated by sub-Fig. 1, both the histogram and the object/surrounding GMM are learned in the red ellipse region. Testing is conducted on the frame in sub-Fig. 2, and the ground truth is also marked by the red ellipse. In sub-Figs. 3 and 4, the performance of the tracking cue w.r.t. translation is compared. The surface is generated by sliding the bounding window in the range $[-30:30,\ -50:50]$ along the $x$- and $y$-directions, respectively. The matching at each bounding window location to the target is measured by both the HBCD cue and the total WLR. As such, matching surfaces are generated in sub-Fig. 3 for the HBCD cue and in sub-Fig. 4 for the total WLR cue. When surfaces peak at the ground truth location(0,0), the WLR cue is unimodal with one global maximum, which indicates a stronger discrimination. In sub-Figs. 5 and 6, the performance of the tracking cue w.r.t. rotation is compared. The surface is generated by rotating the bounding window in the range $[-50°, 50°]$. As shown, the WLR cue also has a smaller rotational deviation and sharper matching curve. Moreover, it has fewer local minimum points. In sub-Figs. 7 and 8, the performance of the tracking cue w.r.t. scale is compared. The surface is generated by scaling the bounding window in the range [70%, 130%] to the scale of ground truth. They demonstrated the same property. For the histogram Bhattacharyya coefficient cue, when the scale is smaller than the ground truth, the curve is flatter, which indicates that the smaller scale with no background included tends to be selected. Table I summarizes the deviation in the preceding three comparisons.

### F. Rotation and Scale Adaptation

To capture the object heading angle (rotation) and scaling, rotation and scaling adaptation are applied to the object bounding window after it has been localized. Then, the rotation and scale factor $[\phi, \delta]$ with the largest total WLR is selected for the bounding window. For vehicles captured in the 25-frames/s video, the empirical range $\phi \in [-5°, 5°]$ and $\delta \in [0.95, 1.05]$ are large enough to cover all possible values.

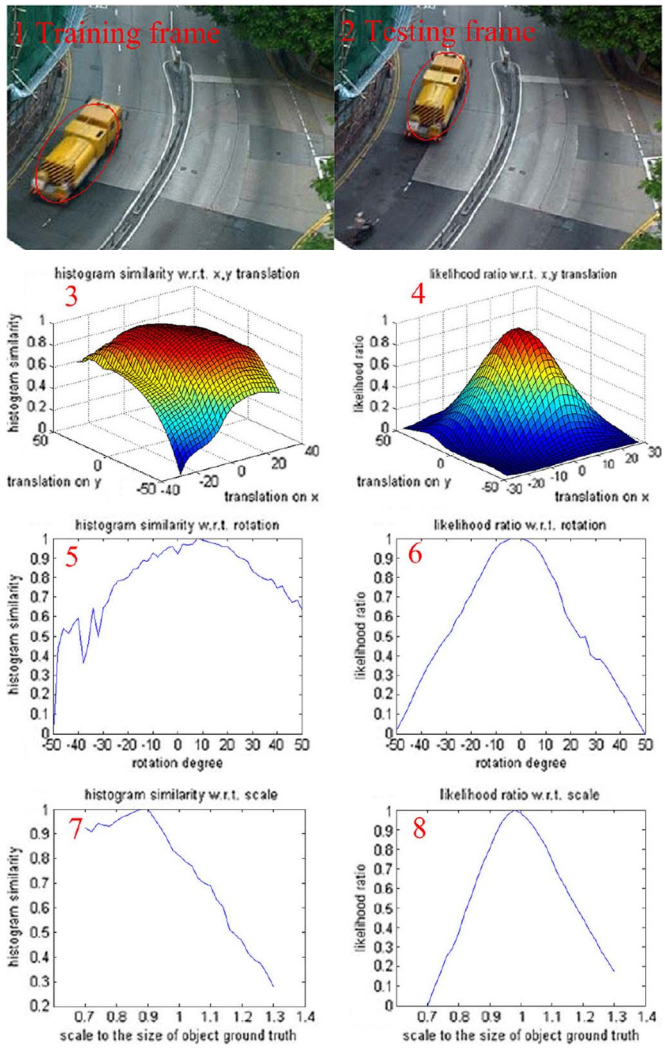


Fig. 5. Performance of tracking cue.

TABLE I
COMPARISON FOR TRACKING CUES

| Deviation | | translation | rotation | scaling |
|---|---|---|---|---|
| Image cue | WLR | 0 | -2° | 98% |
| | HBCD | 0 | 8° | 90% |

## III. EXPERIMENTS AND DISCUSSION

### A. Choice of Color Space

In the experiment, the $\mathrm{L^*a^*b}$ color space is chosen. The common choices include RGB and HSV space as well. However, RGB is prone to be affected by illumination. HSV and $\mathrm{L^*a^*b}$ both decouple illumination from chromatic information and, thus, offer higher consistency. Although HSV outperforms $\mathrm{L^*a^*b}$ in [35], it is unreliable for low-saturation colors. Moreover, HSV colors outside certain V and S ranges are normally discarded for stability. On comparison, $\mathrm{L^*a^*b}$ offers more stability across illumination and saturation ranges.


Fig. 6. Changing orientation and scale.

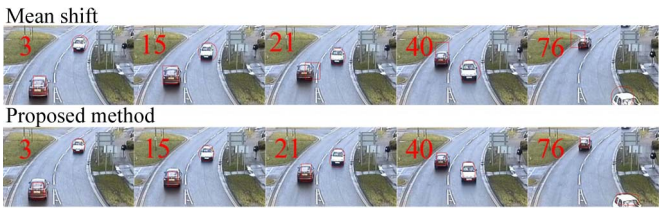


Fig. 7. Changing scale.

### B. Experiments on Tracking

In this part, we test the proposed WLR cue and the integrated tracking dynamics under different conditions and compare the results with mean shift tracking. For mean shift tracking, the scale and orientation of the bounding window are also adapted. For the tracking accuracy evaluation, the mean $M_e$ and standard deviation $M_d$ of the tracking error between the manually specified ground truth locations and the tracking results are used for quantitive evaluation.

The videos were taken outdoor at 25 frames/s and a resolution of 320 $\times$ 240. Multiple-object tracking is implemented by the apply tracker on each individual object in parallel. The proposed method individually locates the object. As such, the tracker uses the corresponding object's features for tracking, and separate data association is not necessary.

In Fig. 6, the proposed method is tested to capture objects with self-rotation and decreasing scale. In the video, the truck steers anti-clockwise and decreases in size when traveling along the road. Both methods capture the truck's motion and its scale and rotation. However, in frames 22 and 37, mean shift gives a larger scale to the object. For the proposed method, it captures the object exactly all the way.

In Fig. 7, the proposed method is tested for both increasing and decreasing scales. In the right lane, the car approaches the camera in increasing scale and is steered to the right. Both methods track motion and capture scale and rotation reasonably well. On the left lane, the car departs from the camera in a decreasing scale. Due to the shadow and the decreasing scale, its color became less saturated. In this case, the similarity cue becomes ambiguous for mean shift, and it exhibits a problem in capturing its motion (frame 21) and adapting to its scale from frame 40 onward. For the proposed method, the WLR cue captures the most distinguishable part on the object. It tracks the target and adapts to its scale very well.

In Fig. 8, the proposed method is tested using a rainy-day video. It should be noted that the water on the road and the roof of the blue car introduces reflections and causes overexposure and that the green minibus moves under the tree, which is also green. Both methods track the blue car reasonably well, although the scale is slightly larger for mean shift in frames 24, as the background has a similar color as its roof. For the proposed method, the scale is smaller. As it seeks the distinguishable part for tracking, this turns out to be most reliable. For the green minibus, mean shift tracks it before it goes under the tree. When part of it is occluded by the tree, mean shift is attracted by the rest of the minibus (as in frames 14–24). When it was mostly occluded by the tree, mean shift lost the target. However, the proposed method tracks the target all the way; even when visibility is quite low (frames 35–43), it still finds the distinguishable part for tracking.


Fig. 8. Rainy test.


Fig. 9. Changing illumination

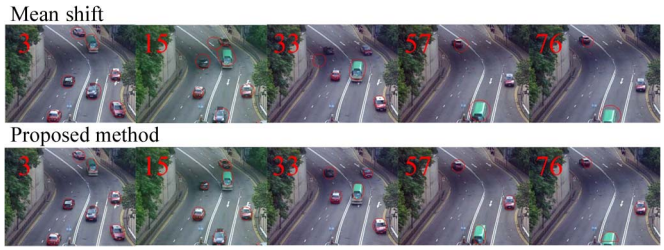


Fig. 10. Multiple vehicle.

In Fig. 9, the proposed method was tested under changing illumination. The car starts from a shadowed area in frame 3 and then travels to a bright area and reaches the peak of illumination in frame 21. It moves to a darker area afterward. As color is affected by illumination, mean shift begins to deviate at frame 15 when the scene became brighter and lost the target from frame 21. However, the proposed method tracks the car reasonably accurately throughout.

In Fig. 10, the proposed method is tested on multiple vehicles, and there is a shadowed area in the upper part of the video where illumination is poorer than the other regions of the image. As illustrated, both methods track the cars reasonably before they travel under the shadow. However, mean shift loses the sedan (frame 33) and the taxi (frame 15) and has an obvious deviation (frame 57). On the other hand, the proposed method tracks every vehicle in the sequence. The problematic case is the dark sedan on the second lane from left, whose color is unsaturated, making it difficult to track by the proposed method, as demonstrated in frame 33 with a large deviation. However, the proposed method tracks this car back after the shadow. For the green bus, the proposed method targets the roof as the most reliable part. The result is consistent tracking in every frame.

In Fig. 11, the methods are tested in a longer night video over 6000 frames in an environment mixed with pedestrians and vehicles. The vehicles are well illuminated on the right and not so well

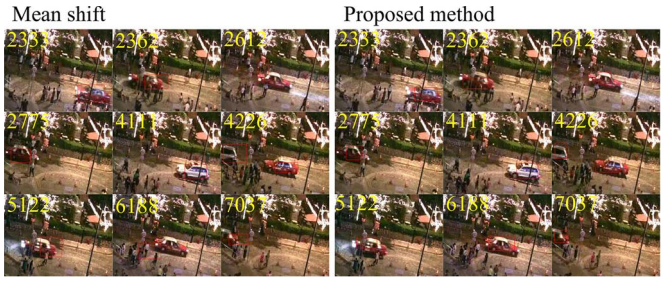


Fig. 11. Night video.

TABLE II
COMPARISON OF TRACKING ACCURACY

| Experiment in the figures | | **Proposed method** | | Mean Shift | |
|---|---|---|---|---|---|
| | | $M_e$ | $M_d$ | $M_e$ | $M_d$ |
| Fig.6 | Truck | 2.4966 | 1.7124 | 5.3678 | 2.4563 |
| Fig.7 | Red car | 8.4400 | 3.8423 | 15.6935 | 6.8983 |
| | White car | 6.4535 | 3.4883 | 8.2798 | 3.5685 |
| Fig.8 | Blue car | 4.9912 | 2.2188 | 5.1649 | 2.1136 |
| | Green bus | 10.8752 | 5.4763 | 43.4668 | 25.2133 |
| Fig.9 | White car | 2.6913 | 1.9229 | Lost after frame15 | |
| Fig.10 | Lane2 sedan | 6.5690 | 3.3914 | Lost after frame27 | |
| | Lane2 taxi | 4.6165 | 2.2061 | 4.7918 | 3.2945 |
| | Lane3 bus | 6.3466 | 2.0252 | 6.6028 | 3.7984 |
| | Lane3 jeep | 4.6585 | 3.0800 | 7.9170 | 3.1663 |
| | Lane4 taxi1 | 3.6106 | 2.2272 | Lost after frame18 | |
| | Lane4 taxi2 | 4.1916 | 2.4442 | 10.1492 | 5.2193 |
| | Lane4 taxi3 | 6.1347 | 3.2350 | 7.4494 | 3.3365 |
| Fig.11 | Taxi1 | 11.4326 | 6.4356 | 22.4672 | 9.6735 |
| | Taxi2 | 12.6531 | 5.3677 | 20.4567 | 10.5155 |
| | Taxi3 | 11.6737 | 5.2294 | Lost after frame 2613 | |
| | Taxi4 | 8.2583 | 5.4127 | Lost after frame 2673 | |
| | Taxi5 | 10.6505 | 5.6300 | 21.2948 | 7.7848 |
| | Taxi6 | 12.2731 | 4.8037 | 23.1220 | 11.0329 |
| | SUV | 10.9162 | 4.8348 | 15.7136 | 8.9360 |
| | Taxi7 | 12.5064 | 7.5509 | 26.1858 | 18.3006 |
| | Taxi8 | 16.0800 | 5.7254 | 38.4612 | 20.1123 |
| | Taxi9 | 10.6199 | 6.6717 | 24.6407 | 19.3325 |
| | Taxi10 | 11.0688 | 7.2760 | Lost after frame 5982 | |
| | Taxi11 | 19.7020 | 11.1168 | Lost after frame 6172 | |
| | Taxi12 | 8.8040 | 4.5541 | 15.0700 | 11.8097 |
| | Taxi12 | 12.7369 | 5.9010 | 18.4772 | 14.9670 |

illuminated on the left due to insufficient light and shadowing. The vehicles are also partially occluded by the pedestrians and the traffic signs on the left. Both methods accurately track vehicles in the region where illumination is consistent, i.e., either bright or dim, as illustrated in frames 2353, 2775, 4111, etc. However, when vehicles travel through regions of different illuminations and are slightly occluded by nearby pedestrians, the accuracy of mean shift abruptly decreases while the proposed method reliably tracks. For instance, in frames 2362 and 5122, the result of mean shift falls on the rear of the vehicle, but the proposed method locks the target region as before. When the vehicle has completely moved into the dimmer region (e.g., in frames 4226 and 7037), mean shift has a localization and scaling problem, but the proposed method continues to generate a stable track. It is equally challenging when the vehicle travels from the dimmer to the brighter. In this case, mean shift lost its target (as shown in frames 2612 and 6188), whereas the proposed method continues to correctly track.

The tracking accuracy referring to the manually specified ground truth position is compared in Table II. It can be seen that the proposed method has a smaller mean deviation, and standard deviation is substantially smaller in the case of the proposed method as well.

## IV. CONCLUSION

This paper proposes the SASA-FDA object appearance representation, which provides the ability to capture object scale and rotation by representing both the object appearance and its spatial layout. To solve the effect of object scaling on feature extraction, the representation is trained on scale-adaptive features in a consistent learning algorithm. The WLR measure from the appearance model is used to construct the image cue for object tracking, and tracking dynamics is provided to locate the object by the image cue. In testing and comparing with mean shift, it is verified that the proposed method works well with target objects under a variety of scale, rotation, and illumination conditions and is more accurate in tracking compared with mean shift. The proposed method can potentially be used as a generic tracker as there is no assumption imposed on the target objects, and since the appearance model is learned in one shot, an offline line object training database is not required. The proposed method can be further improved in the areas of online updating of the appearance model, exploration of other types of features, and feature fusion methods to achieve improved robustness.

## REFERENCES

[1] B. Coifman, D. Beymer, P. McLauchlan, and J. Malik, "A real-time computer vision system for vehicle tracking and traffic surveillance," *Transp. Res.: Part C*, vol. 6, no. 4, pp. 271–288, Aug. 1998.

[2] Y. K. Jung, K. W. Lee, and Y. S. Ho, "Content-based event retrieval using semantic scene interpretation for automated traffic surveillance," *IEEE Trans. Intell. Transp. Syst.*, vol. 2, no. 3, pp. 151–163, Sep. 2001.

[3] W. M. Hu, X. J. Xiao, D. Xie, T. Tan, and S. Maybank, "Traffic accident prediction using 3-D model-based vehicle tracking," *IEEE Trans. Veh. Technol.*, vol. 53, no. 3, pp. 677–694, May 2004.

[4] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Traffic monitoring and accident detection at intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 2, pp. 108–118, Jun. 2000.

[5] H. Veeraraghavan and N. P. Papanikolopoulos, "Learning to recognize video-based spatiotemporal events," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 628–638, Dec. 2009.

[6] B. T. Morris and M. M. Trivedi, "Learning, modeling, and classification of vehicle track patterns from live video," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 425–437, Sep. 2008.

[7] A. Barth and U. Franke, "Estimating the driving state of oncoming vehicles from a moving platform using stereo vision," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 560–571, Dec. 2009.

[8] J. Ge, Y. Luo, and G. Tei, "Real-time pedestrian detection and tracking at nighttime for driver-assistance systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 2, pp. 283–298, Jun. 2009.

[9] H. Veeraraghavan, O. Masoud, and N. P. Papanikolopoulos, "Computer vision algorithms for intersection monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 2, pp. 78–89, Jun. 2003.

[10] S.-C. Chen, M.-L. Shyu, S. Peeta, and C. Zhang, "Learning-based spatio-temporal vehicle tracking and indexing for transportation multimedia database systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 3, pp. 154–167, Sep. 2003.

[11] P. Kumar, S. Ranganath, H. Weimin, and K. Sengupta, "Framework for real-time behavior interpretation from traffic video," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 43–53, Mar. 2005.

[12] J.-W. Hsieh, S.-H. Yu, Y.-S. Chen, and W.-F. Hu, "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 175–187, Jun. 2006.

[13] N. K. Kanhere and S. T. Birchfield, "Real-time incremental segmentation and tracking of vehicles at low camera angles using stable features," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 148–160, Mar. 2008.

[14] C. C. C. Pang, W. W. L. Lam, and N. H. C. Yung, "A novel method for resolving vehicle occlusion in a monocular traffic-image sequence," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 3, pp. 129–141, Sep. 2004.

[15] Z. Kim and J. Malik, "Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking," in *Proc. IEEE 9th ICCV*, Nice, France, Oct. 2003, pp. 524–531.

[16] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 37–47, Mar. 2002.

[17] Y. Huang, T. S. Huang, and H. Niemann, "A region-based method for model-free object tracking," in *Proc. 9th ICPR*, Quebec City, QC, Canada, Aug. 2002, pp. 592–595.
[18] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, no. 1–3, pp. 125–141, May 2008.
[19] C.-C. R. Wang and J.-J. J. Lien, "Automatic vehicle detection using local features—A statistical approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 83–96, Mar. 2008.
[20] T. Xiong and C. Debrunner, "Stochastic car tracking with line- and color-based features," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 324–328, Dec. 2004.
[21] S. Munder, C. Schnörr, and D. M. Gavrila, "Pedestrian detection and tracking using a mixture of view-based shape–texture models," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 2, pp. 333–343, Jun. 2008.
[22] F. Xu, X. Liu, and K. Fujimura, "Pedestrian detection and tracking with night vision," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 63–71, Mar. 2005.
[23] S. Sivaraman and M. M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 267–276, Jun. 2010.
[24] R. O'Malley, E. Jones, and M. Glavin, "Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 453–462, Jun. 2010.
[25] L. Oliveira, U. Nunes, and P. Peixoto, "On exploration of classifier ensemble synergism in pedestrian detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 16–27, Mar. 2010.
[26] H. Wang, D. Suter, K. Schindler, and C. Shen, "Adaptive object tracking based on an effective appearance filter," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1661–1667, Sep. 2007.
[27] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Stat. Soc., Ser. B*, vol. 39, no. 1, pp. 1–38, 1977.
[28] B. Han, D. Comaniciu, Y. Zhu, and L. S. Davis, "Sequential kernel density approximation and its application to real-time visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1186–1197, Jul. 2008.
[29] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.
[30] 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, May 2002.
[31] R. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Conf. CVPR*, Madison, WI, 2003, pp. 234–240.
[32] Z. Zivkovic and B. Kröse, "An EM-like algorithm for color-histogram-based object tracking," in *Proc. IEEE Conf. CVPR*, Washington, DC, 2004, pp. 798–803.
[33] T. Lindeberg, *Scale-Space Theory in Computer Vision*. Dordrecht, The Netherlands: Kluwer, 1994, pp. 101–149.
[34] T. Kadir and M. Brady, "Saliency, scale and image description," *Int. J. Comput. Vis.*, vol. 45, no. 2, pp. 83–105, Nov. 2001.
[35] E. Maggio and A. Cavallaro, "Multi-part target representation for colour tracking," in *Proc. IEEE ICIP*, Genoa, Italy, 2005, pp. 729–732.