Moving Object Tracking in Driving Environment

Jae-Yeong Lee and Wopil Yu
Electronics and Telecommunications Research Institute, Daejeon, 305-700, Korea
(E-mail: jylee@etri.re.kr, ywp@etri.re.kr)

Abstract - This paper presents a visual tracking system developed for the tracking of pedestrians and vehicles in a driving environment. The presented system applies a partition-based mean-shift for the tracking of pedestrians and a HSV template matcher for vehicles. The experiment results on real video sequences confirm the validity of our method.

Keywords - Tracking, Mean-Shift, Vehicle, Pedestrian.

1. Introduction

Object tracking is an important and challenging problem in computer vision. For many high-level applications such as surveillance, human-robot interaction, action recognition and navigation of intelligent vehicles, reliable tracking is essentially required. In natural scenes, however, tracking is still a challenging problem with respect to robustness.

Ryoo et al. [1] recently proposed a personal driving diary system, which is an automated system that recognizes important interactions between the driving vehicle and the others (e.g. accident, overtaking, ...) by analyzing videos obtained from a vehicle-mounted camera, and labels them together with its contextual knowledge on the vehicle. The driving diary system is composed of four components: geometry component, detection component, tracking component, and event analysis component. These components obtain visual inputs (i.e. videos) from the camera and interact each other to analyze events involving the driving vehicle itself, other vehicles, and pedestrians.

In this paper, we describe a visual tracking system for the tracking of pedestrians and vehicles in a driving environment, which was developed as a tracking component of the personal driving diary system [1]. The tracking system uses a histogram-based approach for the tracking of pedestrians and a template matching method for vehicles. The detailed methodologies and experimental results are given in the following sections.

2. Pedestrian Tracking

Histogram-based object tracking methods like mean-shift have been used widely because of its simplicity and real-time performance [2]. Mean-shift is well suited for the tracking of non-rigid objects and works well for the objects having discriminant colors from the background. However, it is well known that the mean-shift is very sensitive to the background colors and shows low localization accuracy in complex environment.

Fig. 1. A sample image and corresponding backprojection images: (a) input image, (b) backprojection image using the upper part histogram of the object, (c) backprojection image using the lower part histogram of the object, (d) combined backprojection image of (b) and (c), (e) conventional backprojection image using single histogram.

One of the main sources of the problem is the loss of spatial information in the use of histograms. Motivated from this observation, we have developed a new partition-based mean-shift tracking method that divides an object region into fixed partitions and then applies the mean-shift for each partition conjunctively. While original mean-shift uses a single histogram, the proposed partition-based mean-shift builds and maintains a histogram for each partition.

Let \( P = \{ P_i \} \) be a set of partitions of an initial object region. Histograms \( H_i \) are then built from each partition, forming an object model. Each histogram \( H_i \) captures color distribution of partition \( P_i \). For a given input image \( I \), backprojection images are then computed by projecting each histogram on the current search region as follows:

\[
\psi_j(x) = \frac{H_j(I(x))}{H_x(I(x))},
\]

where \( H_x \) is a histogram of the current search region and \( H(\cdot) \) is the normalized frequency of a pixel color in the histogram. Note that we have \( n \) backprojection images which have the same size with the search region, where \( n \) is the number of partitions.

Finally, the mean-shift vector is computed as follows:

\[
\Delta x = \frac{\sum_{j \in \hat{R}} K(x - \hat{x}_{old}) \psi_j(x)}{\sum_{j \in \hat{R}} K(x - \hat{x}_{old}) \psi_j(x)},
\]

where \( K \) is a kernel function defining an influence zone.
Figure 1 illustrates this procedure when using $2 \times 1$ partitions. Figure 1(b) and Fig. 1(c) show backprojection images using the upper part histogram and the lower part histogram of the partition-based object model, respectively. Figure 1(e) shows conventional backprojection image using single histogram model. We can see that the object is locally better discriminated from the background in the backprojection images when using partition-based model, compared to the case of single histogram model. It is because each partition will have the smaller color variance than the case of the whole object region, and thus each partition will have the higher chance of discrimination from the background locally. Finally, the combined backprojection image (Fig. 1(d)) is used for the computation of mean-shift vector in Eq. (2).

The scale of the target region (i.e. bounding box) is then adjusted by using Bhattacharyya coefficient as like [2]. Let $P_{H} = \{P_{k}^{c}\}$ be a set of partitions of the target candidate region located by iterative application of Eq. (2). Candidate histograms $H_{k}^{c}$ are then built from each partition $P_{k}^{c}$. We compute Bhattacharyya coefficient between the object model and a target candidate by the sum of every Bhattacharyya coefficient between corresponding partition histograms as the following:

$$\rho = \sum_{i=1}^{m} \sum_{k=1}^{n} b_{i}^{c} b_{i}^{c} \cdot$$  \hspace{1cm} (3)

where $m$ is the number of histogram bins, $b_{i}^{c}$ and $b_{i}^{c}$ are the normalized $k$-th bin densities of the histogram $H_{k}$ and $H_{k}^{c}$, respectively. This similarity measure is computed for the previous scale (scale in the previous image frame), enlarged scale, and reduced scale. The scale with highest score is selected as the current scale of the target object.

The presented partition-based mean-shift contributes to the performance in two aspects. Firstly it enables more reliable tracking as the object region is better discriminated from the background. Secondly we can locate a target object more accurately since the spatial information of each partition is preserved in Eq. (2). The second aspect is very important in our system because the performance of event analysis highly depends on the exact trajectories.

### 3. Vehicle Tracking

For the tracking of vehicles, we adopt a simple template matching technique. For each initial detection of a vehicle, a HSV template is built from the corresponding image region, forming an object model. A target object is then located by minimizing the average color difference between the object model and candidate region.

The color difference is measured by Euclidean distance between two color points in HSV color space (conic model). The 3D coordinate of a color point in conic HSV color space is given by the following:

$$x = s \cos(h)v / 255 \cdot y = s \sin(h)v / 255 \cdot z = v$$  \hspace{1cm} (3)

where $h$, $s$, $v$ denotes hue, saturation and intensity, respectively.

After the target object is located in the current frame, the scale is then adjusted such that the average color difference between the object model and the located candidate region is minimized.

Different from histogram-based approaches, template matching utilizes raw colors of an object with their spatial information in pixel level, and therefore it is less sensitive to background colors and able to locate object region more accurately if the objects have rigid body and their orientation is fixed.

### 4. Experiments

In this subsection, we evaluate the performance of the visual tracker in our tracking component. For the evaluation, we randomly selected 10 pedestrian video scenes and 10 vehicle video scenes from the personal driving diary dataset [1]. Using this test set, we performed experiments to evaluate how accurately and how long the tracker is able to track a target object, given an initial position. The tracking accuracy was evaluated based on the overlap ratio between a searching result and the corresponding ground truth as the following:

$$\lambda = \frac{|R \cap G|}{|R \cup G|} \times 100 \cdot$$  \hspace{1cm} (4)

where $R$ denotes a searched region and $G$ a ground truth. This accuracy measurement involves both translation and scale error, which is commonly used in the vision community for object detection problems [3-5]. Figure 2 shows several examples of tracking results.

<table>
<thead>
<tr>
<th>Table 1. Comparison of tracking accuracies.</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Table" /></td>
</tr>
</tbody>
</table>

- 140 -
Table 1 shows the average tracking accuracies of Comaniciu’s mean-shift (MS) [2], our partition-based mean-shift (PMS), and our template tracker (TM) for the test dataset. We are able to observe that the partition-based mean-shift shows the best performance for pedestrian dataset and the template tracker shows the best for the vehicle dataset. In fact, the average performance of 55% accuracy for the pedestrian videos is not low when we consider the fact that a searching result having 50% overlap with the ground truth has 33% accuracy by Eq. (4).

Table 2. Continuous tracking performance of the system.

<table>
<thead>
<tr>
<th>Pedestrian</th>
<th>Video length (frames)</th>
<th>Successful tracking (%)</th>
<th>Pedestrian</th>
<th>Video length (frames)</th>
<th>Successful tracking (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ped02</td>
<td>156</td>
<td>100</td>
<td>veh22</td>
<td>230</td>
<td>100</td>
</tr>
<tr>
<td>ped03</td>
<td>152</td>
<td>100</td>
<td>veh23</td>
<td>227</td>
<td>100</td>
</tr>
<tr>
<td>ped04</td>
<td>83</td>
<td>100</td>
<td>veh28</td>
<td>237</td>
<td>100</td>
</tr>
<tr>
<td>ped05</td>
<td>78</td>
<td>100</td>
<td>veh29</td>
<td>212</td>
<td>100</td>
</tr>
<tr>
<td>ped06</td>
<td>500</td>
<td>100</td>
<td>veh33</td>
<td>235</td>
<td>100</td>
</tr>
<tr>
<td>ped07</td>
<td>78</td>
<td>71</td>
<td>veh35</td>
<td>238</td>
<td>100</td>
</tr>
<tr>
<td>ped12</td>
<td>94</td>
<td>12</td>
<td>veh40</td>
<td>370</td>
<td>100</td>
</tr>
<tr>
<td>ped13</td>
<td>171</td>
<td>100</td>
<td>veh42</td>
<td>208</td>
<td>100</td>
</tr>
<tr>
<td>ped53</td>
<td>153</td>
<td>100</td>
<td>veh45</td>
<td>234</td>
<td>100</td>
</tr>
<tr>
<td>ped54</td>
<td>166</td>
<td>100</td>
<td>veh47</td>
<td>510</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 2 shows the continuous tracking performance of the system. 'Video length' shows total number of frames in each video clip. 'Successful tracking' shows the percentage of an interval of a video clip that the tracker succeeds to track an initial target object continuously without failure. The results were obtained by applying the partition-based meanshift tracker for pedestrian videos and the template tracker for vehicle videos. The results show that our visual tracker is able to track a target object correctly without failure for most cases. An exceptional case of ‘ped12’ video clip where the tracker fails at very early stage of tracking is because the target pedestrian disappears temporarily due to occlusion and our tracker does not handle occlusion currently.

5. Conclusion

In this work, we presented a visual tracking system that consists of a partition-based mean-shift tracker and a HSV template tracker. The presented system uses different tracking methodology according to object type (rigid object and non-rigid object), giving better overall performance. The experimental results on real video sequences show that the presented system is able to track pedestrians and vehicles effectively in driving environment.

Acknowledgement

This work was supported by the R&D program of the Korea Evaluation Institute of Industrial Technology (KEIT). (The Development of Low-cost Autonomous Navigation Systems for a Robot Vehicle in Urban Environment, 10035354)

References