MULTI-CAMERA SCENE ANALYSIS USING AN OBJECT-CENTRIC CONTINUOUS DISTRIBUTION HIDDEN MARKOV MODEL

Murtaza Taj and Andrea Cavallaro

Multimedia and Vision Group – Queen Mary, University of London (United Kingdom)
Email: {murtaza.taj, andrea.cavallaro}@elec.qmul.ac.uk

ABSTRACT
We propose a multi-camera event detection framework that can operate on a common ground plane as well as on the image plane. The proposed event detector is based on an object-centric state modeling that uses a Continuous Distribution Hidden Markov Model (CDHMM). Video objects are first detected using statistical change detection and then tracked using graph matching. Next, the algorithm recognizes events by estimating the most likely object state sequence using a HMM decoding strategy, based on the Viterbi algorithm. We demonstrate and evaluate the proposed framework on standard event detection datasets with single and multiple cameras, with both overlapping and non-overlapping fields of view.

Index Terms— Hidden Markov Model, Viterbi algorithm, homography, event detection, multi-camera.

1. INTRODUCTION
Manual annotation of news, sports and surveillance video is a time consuming and tedious task. For this reason, automatic algorithms capable of detecting events of interest to index videos or to trigger alarms are highly desirable. In multi-camera surveillance video, event detection can be performed on the image plane or on the ground plane. In particular, the use of the ground plane offers an extended coverage of the monitored scene as the objects can be tracked across multiple cameras.

Event detection algorithms can be classified into three main groups, namely 3-D model-based, temporal templates and trajectory-based. 3D model-based approaches treat an object as a set of connected parts and perform detections on their activities [1]. The activities can be modeled as generalized action cylinders [2]. Temporal templates use sequences of simple events as a prior to model more complex events. Examples of temporal template methods are Petri Nets ([3]) and Belief Networks ([4]). Trajectory-based techniques perform event detection by analyzing trajectories over certain time spans [5, 6]. Since events are generally composed of specific sequences of operations, HMMs are appropriate to model such events [7]. HMMs are also used to perform abnormal activity detection by modeling the behavior of crowds [8]. The main limitation of the above mentioned HMM-based techniques is the use of an evaluation strategy to obtain sequences of events, as this results in a dependence on the selected template.

When multiple cameras are available, a planar homography can be used to wrap all views on a reference view of the ground plane before object segmentation. Next, tracking is performed using connected component analysis and graph theory [10]. However, the projection of the object points that are not on the ground results in erroneous projections.

In this paper, we propose a general framework for video event detection that is applicable on the image plane as well as on the ground plane. Moreover, the proposed approach is applicable to cameras with both overlapping and non-overlapping fields of view, and operates with both calibrated and uncalibrated cameras. We segment objects in the video using a statistical color change detector and track them over time using graph matching. Next we apply on the tracked objects an object-centric state modeling based on a Continuous Distribution Hidden Markov Model (CDHMM) [11]. Unlike previous HMM-based event detectors ([12, 8]) that use the evaluation strategy to model complex events (thus requiring a sequence of simple events to be known a priori), we use a HMM decoding strategy, which allows us to deal with unknown sequences of object states, thus resulting in a more flexible event detector (i.e., not dependent on a predefined template).

This paper is organized as follows. Section 2 and Section 3 describe object detection and tracking, and the proposed HMM-based event detection algorithm, respectively. Experimental results are presented in Section 4. Finally, in Section 5 we draw the conclusions.

2. OBJECT DETECTION AND TRACKING

We decompose the event detection problem into four main steps: the extraction of objects of interest in the image plane, the projection of the mid-point object-ground intersections to the ground plane, the tracking of the projected points on the ground plane, and the detection of events on the tracked objects. These steps are detailed below.

Let an object detection module [13] generate a set of R objects \( O_t = \{O_t^1, O_t^2, \ldots, O_t^R\} \) at time \( t \) on image plane. Let \( P^r(x, y) \) be the base mid-point of the bounding box in homogeneous coordinates representing an object \( O_t^r \). \( P^r(x, y) \) represents the point of intersection between an object and the main plane of the scene.

The problem is to find the point \( P^r(u, v, 1) \), projection of the point \( P^r(x, y) \) on the world coordinates (i.e., the ground plane). We project the detections from each camera on the ground plane using homography [14]. Then \( P^r(u, v, 1) = H \times P^r(x, y, 1) \), for each \( r \), where \( H \) is a \( 3 \times 3 \) homographic matrix.

Once the points \( \hat{P}_t^r = \{\hat{P}_t^r, \hat{P}_t^r, \ldots, \hat{P}_t^r\} \) are obtained for all the \( R \) objects at time \( t \), the next problem is to associate points between consecutive frames to establish the track \( X_t^r = \{\hat{P}_t^r\} \) up to time \( t \) of each object \( O_t^r \). The trajectory \( X_t^r \) is estimated with a graph matching procedure ([15]) using the ground plane coordinates as features in each node of the graph. The gain used for each arc connecting the nodes is computed using the normalized Euclidean
3. EVENT DETECTION

As event detection can be modeled as a random process that is segmental in nature, the piecewise stationarity assumption of HMMs is well suited for event modeling. Let \( \lambda = \{ A, b_j, \omega \} \) be a continuous distribution first-order Hidden Markov Model, with \( A = \{ a_{ij} \} \) representing the state transition probabilities, \( b_j \) the emission probabilities, and \( \omega = \{ \omega_1, \ldots, \omega_n \} \) the events (states) to be detected. The track \( X_r^t \) provides the observation of the object \( O_r \), i.e., the emitting symbols of each state \( \omega_j \) at time \( t \). For each object \( r \) we compute the most likely hidden state sequence \( \omega^T_r \) up to time \( t \) as

\[
\omega^T_r = \arg \max_i [\delta_i(t-1)a_{ij}],
\]

where \( \delta_j(t) = \max_i [\delta_i(t-1)a_{ij}]b_{jt} \).

We model the emission probabilities \( b_{jt} \) as a continuous multivariate distribution \( N(\mu, \Sigma) \) with mean \( \mu \), covariance \( \Sigma \), weight \( \rho \) and range of uniform distribution \([C, D]\), therefore

\[
b_{jt} = \frac{\rho}{(2\pi)^{\frac{d}{2}} |\Sigma|^\frac{1}{2}} \exp \left( \sum_{k=1}^{K} \left[ \frac{(\theta_k - \mu \theta_k)^2}{2\sigma \theta_k} \right] \right) + \frac{1 - \rho}{\pi} \prod_{k=1}^{K} \psi_{\theta_k}, \tag{2}
\]

where \( K=2; \theta_1 = u \) and \( \theta_2 = v \) on the ground plane or \( \theta_1 = x \) and \( \theta_2 = y \) on image plane. Therefore \( \sigma_u \) and \( \sigma_v \) are the standard deviations along the \( u \) and \( v \) coordinates, respectively. The second term accounts for rapid change in probability after \( \sigma \) so that the HMM can quickly move to the next state. The functions \( \psi_{\theta_k} \) are piecewise binary and defined as

\[
\psi_u = \begin{cases} 
1 & \text{if } C_u < u < D_u \\
0 & \text{otherwise}
\end{cases}, \tag{3}
\]

and

\[
\psi_v = \begin{cases} 
1 & \text{if } \zeta(C_u) < v < \zeta(D_u) \\
0 & \text{otherwise}
\end{cases}, \tag{4}
\]

where \( \zeta = \pm \sigma_u \sqrt{1 - \left( \frac{u - (D_u - C_u)}{2\sigma_u} \right)^2} + v \), with \((u_u, v_v)\) representing the object centroid around which the model is built, and \( \pi \prod_{k=1}^{K} \psi_{\theta_k} \) is the area of an ellipse and \([C_u, D_u]\) is the range of uniform distribution along the \( u \)-axis. \( |\Sigma| \) is the determinant of the covariance matrix. Let \( \Sigma_j = diag(\sigma_u^2, \sigma_v^2) \), then \( |\Sigma| = \sigma_u \sigma_v \) in Eq (2).

The values of the elements in \( \Sigma_j \) depend on the state to be modeled, whereas the value of \( \mu \) is assigned dynamically. This is the key point of the proposed object-centric modeling (Figure 1). The value of \( \mu \) of the first state is set as the centroid of the reference object \( O_r^{ref} \) on the ground plane. The object \( O_r^{ref} \) is the object of interest around which events are to be detected (e.g., the bag in case of unattended baggage). The remaining state distributions are then placed around \( O_r^{ref} \) to estimate the possible state of \( O_r^{ref} \) with respect to the objects \( O_r \). The \( \mu \) of the other states are positioned on the line passing through the centroid of the two objects (\( O_r^{ref} \) and \( O_r \)) at a distance that is a function of the variances of the states to be detected.

The advantage of this HMM-based object-centric model is the capability of incorporating any type of distribution to best model a particular state. This makes the proposed approach flexible enough to detect different events in various scenarios. Moreover, as the behaviors of objects in real scenarios are generally characterized by fuzzy boundaries between different states, a progressive transition from one state to another is preferred to a fixed threshold-based transition [16]. If computational time is an issue, it is possible to use in the proposed framework a uniform distribution to model the states with equal state transition probabilities among the states.

The estimation of the emission probabilities \( b_{jt} \) using the proposed object-centric approach completes the computation of the HMM parameters. These parameters are now used to compute the most likely state sequence \( \omega^T_r \) for each object \( r \) by applying the Forward Viterbi algorithm every \( n \) observations. The last state \( \omega_n \) of the state sequence is used as the initial state \( \omega_0 \) for next computation. The algorithm of event detection using Forward Viterbi algorithm is sum-
marized in Algorithm 1.

To evaluate the event detection results, we estimate the accuracy, the precision and the sensitivity of the event detector. Let $FP$ be the number of false positive detections, $TP$ the number of true positive detections, and $FN$ the number of false negative detections. Moreover, let $GT$ be the frame number corresponding to an event in the ground truth and $AD$ the frame number identified by the event detector for the same event. The accuracy gives an indication of the frame-level performance of the algorithm, and is defined as

$$\gamma = \left[1 - \frac{|GT| - |AD|}{NF}\right] \times 100,$$

with $NF$ representing the minimum duration of an event. The precision is defined as $TP/(TP + FP)$ and the sensitivity is defined as $TP/(TP + FN)$.

4. EXPERIMENTAL RESULTS

We demonstrate the performance of the proposed algorithm on standard event detection sequences from the datasets PETS 2006$^1$ and ETISEO$^2$. These sequences include indoor and outdoor scenarios with pedestrians, vehicles, objects and their interactions. Examples of events to be detected are unattended / abandoned baggage, enter zone, inside zone, empty area and stopped object. The PETS dataset contains good quality sequences (duration: $94 - 136$ sec.) The ETISEO dataset contains sequences of lower quality (duration: $40 - 64$ sec.). Both datasets contain overlapping and non-overlapping regions observed by multiple cameras.

In the PETS sequences, the baggages are detected based on their size and aspect ratio (ranging between 1 and 1.8). For the attended baggage ($\omega_1$) event, $\sigma_u = \sqrt{2 \times 36}$ and $\sigma_v = \sqrt{2 \times 96}$ respectively, whereas for unattended baggage ($\omega_2$) and abandoned baggage ($\omega_3$) the values are $\sigma_u = \sqrt{36/2}$ and $\sigma_v = \sqrt{96/2}$. These values are based on the calculation that $1$m in world-coordinates corresponds in the ground plane to 36 pixels along the $u$-axis and to 96 pixels along the $v$-axis$^3$ (See Figure 1). A baggage is considered unattended when its related object (the owner) is $2m$ away and abandoned when its related object is $3m$ away for at least 30 seconds.

Table 1. Event detection precision and sensitivity for 5 test sequences of the ETISEO dataset

<table>
<thead>
<tr>
<th></th>
<th>AP-11</th>
<th>BE-19</th>
<th>RD-06</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4</td>
<td>1.00</td>
<td>0.65</td>
<td>0.87</td>
</tr>
<tr>
<td>C7</td>
<td>1.00</td>
<td>0.65</td>
<td>0.35</td>
</tr>
<tr>
<td>Precision</td>
<td>0.56</td>
<td>0.30</td>
<td>0.23</td>
</tr>
<tr>
<td>Sensitivity</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2 shows sample event detection results on the sequences S1 and S5 of the PETS 2006 dataset. The images show the detection of the object around which the model is built (first column) and the subsequent sequence of events (a warning and an alarm). To evaluate the results we computed the accuracy of the detection using $NF = 750$ (30 seconds): for the sequence S1, the accuracy for the warning event is 90.5% and for the alarm event is 92.9%; for the sequence S3, the accuracy is 100% for both events as there are no false positives; for the sequence S5, the accuracy is 88.8% and 83.02% for warning and alarm, respectively, and for the sequence S6 the accuracy is 98.5% and 95.5%. The tracks resulting from multiple object tracking (17 objects) on the ground plane are shown in Figure 3. The track highlighted in red corresponds to the object associated to the baggage events.

Figure 4 shows detection results on the ETISEO dataset for the enter zone, inside zone, stopped and empty area events. To demonstrate the flexibility of the proposed framework, in this case event detection is performed on the image plane. The green rectangle drawn on the tarmac is the zone considered for triggering the events enter zone, inside zone and empty area. The stopped event is detected anywhere in the scene. Table 1 summarizes the precision and sensitivity estimation for the results obtained on the ETISEO dataset. The lower sensitivity of the algorithm is due to the fact that the ETISEO scenarios require the classification of contextual objects, feature that is not included in our current framework. The videos with the results for object tracking and event detection are available at http://www.elec.qmul.ac.uk/staffinfo/andrea/event.html.
We presented an event detection algorithm for multiple cameras that can be applied on the image plane as well on the ground plane. The detected objects are tracked using graph matching before performing event detection using a Continuous Distribution Hidden Markov Model combined with a Viterbi decoding strategy. This approach is appropriate for the modeling of different types of events. Using a decoding strategy (instead of the common evaluation strategy) enabled us to generalize the event detection approach by eliminating the need of providing a fixed template of events to be detected. We showed using standard datasets that the proposed approach is flexible enough to be used on the image as well as on the ground plane. Current work includes the investigation of a classification step to enable the recognition of contextual objects in the scene.

5. CONCLUSIONS

6. REFERENCES


