INTERPRETATION OF STATE SEQUENCES IN HMM FOR ACTIVITY REPRESENTATION

*Naresh P. Cuntoor**

Center for Automation Research*
University of Maryland College Park
College Park, MD 20742 USA
cuntoor,rama@cfar.umd.edu

*B. Yegnanarayana and Rama Chellappa**

Department of Computer Science and Engineering
Indian Institute of Technology Madras
Chennai-600036 India
yegna@cs.iitm.ernet.in

ABSTRACT

We propose a method for activity representation based on semantic events, using the HMM framework. For every time instant, the probability of event occurrence is computed by exploring a subset of state sequences. The idea is that while activity trajectories may have large variations at the data or the state levels, they may exhibit similarities at the event level. Our experiments show the application of these events to activity recognition in an office environment and to anomalous trajectory detection using surveillance video data.

1. INTRODUCTION

Activity modeling has several applications includiing activity recognition, object classification, segmentation, video indexing etc. Approaches using statistical models, temporal templates among others have been proposed to represent repetitive activities such as walking, running, gestures etc. [1], [2]. Not all activities can be represented using direct statistical models due to large data variability and semantic ambiguity [3], i.e., multiple samples corresponding to the same activity may have drastically varying appearances. In such cases, activities may be regarded as a sequence of certain events that are semantically significant. We propose an automatic event detection method using the HMM for activity recognition and anomalous trajectory detection.

We illustrate the use of events in activity representation through the following example. Consider the activity of *picking up an object*. Picking up a pen lying on the desk and picking up a book from the cabinet may create dissimilar trajectories though the act of *picking up* closely resemble each other in the two cases. Only the few frames when the object is picked up is relevant to the recognition task. Similarly, in a surveillance scenario in an airport where people deplane and walk toward the gate, one may be interested in knowing whether the people follow a normal path. The exact trajectories of the people may not be important, and certain key frames in the video sequence may be sufficient to characterize the activity.

The HMM has become a popular tool in many recognition tasks including speech, gesture, action etc. owing to its powerful representation and tractability. However, a direct application of HMM is not feasible as we have already mentioned. Further, the optimal state sequence of the HMM is used to evaluate the likelihood without requiring a physical interpretation. We explore a subset of state sequences that identifies events by analyzing certain transitions in the hidden state. An event is defined as the maximum over all possible transitions between two distinct hidden states such that the past and future states each have a support of p frames of observation vectors. The event probability sequence is used to compare two activities.

This work was sponsored by the Advanced Research and Development Activity, a U.S. Government entity which sponsors and promotes research of import to the intelligence community.

2. EVENT DETECTION USING HMM

Before describing the event probability sequence, we review the basic HMM notation. Detailed explanation of the HMM may be found in several sources including [4].

- Let $O = \{o_1, \ldots, o_T\}$ represent the observation symbol sequence of length T.
- Let $Q = \{q_1, \ldots, q_T\}$ be the (hidden) state sequence with $q_t \in \{1, \ldots, N\}$, where N is the number of states.
- $A = [a_{ij}]_{N \times N}$ is the state transition matrix whose elements $a_{ij} = P(q_t = j / q_{t-1} = i)$ are transition probabilities.
- $b_j(o_t) = P(o_t / q_t = j)$ is the observation symbol probability.
- Initial probability of states is given by $\Pi = [\pi_1, \ldots, \pi_N]$.

The model parameters are estimated such that $P(O/\lambda)$ is maximized using either the max of the propabilities over all possible state sequences Q, or by summing over all possible Q.

- The Viterbi algorithm computes the likelihood for the optimal state sequence using $P(O/\lambda) = \max_Q P(Q, O/\lambda)$.
- The likelihood by summing over all possible state sequences is given by $P(O/\lambda) = \sum_Q P(Q, O/\lambda)$

The variables used in estimating the model parameters are

$$\begin{aligned} \text{Forward variable } \alpha_t(i) &= P(q_t = i, \mathbf{o}_1^t/\lambda) \\ \text{Backward variable } \beta_t(j) &= P(\mathbf{o}_{t+1}^T/q_t = j, \lambda) \end{aligned}$$

Probability of passing through state i at time t and state j at $t+1$ conditioned on the data is given by $\xi_t(i,j) = P(q_t = i, q_{t+1} = j/O, \lambda)$.

The optimal state sequence is not driven by the events that occur during the activity, but by the likelihood of the observed data. In this study, we explore a subset of these state sequences, which are likely to highlight some events. The hypothesis is that events

can be extracted more easily by looking for certain changes in the state sequence. Though the states themselves may look different across samples of the same activity, certain transitions in the hidden states may be preserved. Such transitions represent events. An activity may be described as a sequence of events.

The large variability in the trajectories associated with an activity makes it difficult to describe the events directly from the observed data. Further, rapid fluctuations due to events tend to be reflected in the state sequence as well, unless the HMM is broadly tuned to mask the events. We propose a method to capture the events from the sequences of hidden states. Define a variable $\eta_t^p(i,j)$, $p = 1, 2, \ldots,$, which is similar to $\xi_t(i,j)$, as follows.

$$\begin{aligned}
\eta_t^1(i,j) &= P(q_{t-1} = i, q_t = i, q_{t+1} = j, q_{t+2} = j/O, \lambda) \\
\eta_t^p(i,j) &= P(q_{t-p} = i, q_{t-p+1} = i, \ldots, q_t = i, \\
&\quad q_{t+1} = j, q_{t+2} = j, \ldots, q_{t+p+1} = j/O, \lambda) \quad (1) \\
e_t^p(k,l) &= \max_{i \neq j} \eta_t^p(i,j) \quad (2)
\end{aligned}$$

where $(k,l) = \arg\max_{i \neq j} \eta_t^p(i,j)$. The quantity $e_t^p(k,l)$ represents the event probability at time t. A large value of $e_t^p(k,l)$ indicates the presence of an event. The nature of the event is specified by (k,l) and the strength of the event by the probability value.

$$\begin{aligned}
\eta_t^1(i,j) &= \frac{P(q_{t-1} = i, q_t = i, q_{t+1} = j, q_{t+2} = j, O/\lambda)}{P(O/\lambda)} \\
&= \frac{\alpha_{t-1}(i) a_{ii} b_i(o_t) a_{ij} b_j(o_{t+1}) a_{jj} b_j(o_{t+2}) \beta_{t+2}(j)}{P(O/\lambda)}
\end{aligned}$$

Similarly, for $p \geq 1$,

$$\begin{aligned}
\eta_t^p(i,j) &= \alpha_{t-p}(i) a_{ii}^p b_i(o_{t-p+1}) b_i(o_{t-p+2}) \ldots b_i(o_t) a_{ij} \times \\
&\quad b_j(o_{t+1}) b_j(o_{t+2}) \ldots b_j(o_{t+p+1}) a_{jj}^p \times \\
&\quad \beta_{t+p+1}(j)/P(O/\lambda)
\end{aligned}$$

The choice of p determines the relative strength of the dominant and spurious peaks (events) in e_t.

3. RECOGNIZING ACTIVITIES USING EVENT PROBABILITY SEQUENCE

In this section, we will outline our approach to activity recognition based on the event probability sequence e_t. From the video data, the motion trajectory of the object is extracted and smoothed out. Using the Baum-Welch algorithm, the parameters of the HMM are computed from trajectories of an activity[4]. Two activities that are the same in the semantic sense but differ in context are used to train two separate models. Similarly, trajectories of an activity that are captured from different viewpoints are used to train separate models. Thus, the trajectories corresponding to picking up a pen from the desk and picking up a book from the cabinet will have two different models, while two instances of picking up a pen lying on the table, without changing the viewpoint will be used as multiple observations to train an HMM.

We compute η_t in (1) and the event probability sequence e_t in (2), using the trained HMM for all the training data. The event probability sequence forms the signature for the trajectory. Activities that are similar but appear differently will resemble at the event probability level, even though the state descriptions may be different. Events are thus abstractions of the hidden states, which are themselves abstractions of the observed data.

Given a test trajectory belonging to an activity, we compute the event probability sequences for each of the distinct HMMs available in the gallery. The event probability sequence for each HMM is compared with each of the event probability sequences of the trajectories of the training data corresponding to this HMM. If there are a total of N trajectories in the training data set for all the activities (HMMs), then there will be N matching scores for each test trajectory. The matching score for comparing two event probability sequences is obtained using dynamic time warping algorithm [5].

For a better understanding of the role of the event probabilities in activity representation, suppose that the segmental k-means algorithm is used in training. Roughly speaking, the motion trajectory is segmented into different states. Consider a fictitious trajectory obtained by drawing a horizontal line in air. This can be done by moving the hand from left to right, or from right to left. Since the appearance of the two trajectories differs significantly, we cannot use an HMM to represent the activity directly. On the other hand, using event probabilities in both the cases, we ge, 3 maxima along the event probability sequences as the trajectory transits from one segment to the next.

Ivanov and Bobick [3] split the recognition task into two steps - they segment the activity using HMMs that are tuned to some parts of the activity, and then use stochastic context-free grammar to parse the activity. Our method computes the events within the HMM framework. Also, a vocabulary of activities is not required in our method. Rao et al. [6] represent actions using dynamic instants, which are points of maximum curvature along the trajectory. We will discuss the relation of the dynamic instants to our method in Section 5.

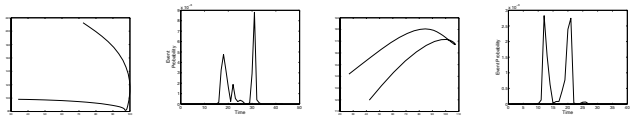
4. DETECTING ANOMALOUS TRAJECTORY

In this section, we will outline our approach to anomaly detection based on the event probability sequence e_t. In a surveillance scenario, one of the tasks is to automatically check whether an activity is occuring along expected lines. It is common to have several instances of the normal activity, and very few samples of an unexpected case, which makes it hard to model the unusual activity. Some approaches like [7] use a model to describe the usual activity, while [8] proposes an unsupervised scheme to compare two activities, and detect an unusual case based on the the extent of similarity. We use HMMs along with the event probability sequences to represent the normal activity. We compare trajectories at the event probability level to check for anomalous behavior.

For a particular activity, we use multiple observations of the normal activity to train an HMM. For each of the normal trajectories in the database, we compute the variable $\eta_t(i,j)$ as defined in (1) and the event probabilitiy sequence e_t defined in (2). Given a new trajectory O^{new}, we compute its associated event probability sequence $e_t(k,l) = \max_{i \neq j} P(q_{t-p} = i, q_{t-p+1} = i, \ldots, q_t = i, q_{t+1} = j, q_{t+2} = j, \ldots, q_{t+p+1} = j/O^{\text{new}}, \lambda_g)$, where λ_g is the HMM associated with the normal activity, and (k,l) is the maximizing argument (i,j). We compare this event probability sequence with the set of events in the database. An anomaly is said to be present in that part of the trajectory where either an event did not occur as expected or if an unexpected event occured.

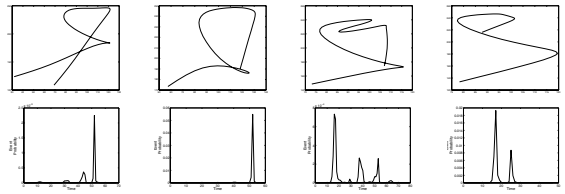
5. EXPERIMENTS

We demonstrate our activity represention method using two datasets - the UCF dataset and the TSA dataset.



(a) (b) (c) (d)

Fig. 1. (a) $x-$, y coordinates of hand trajectories for "pick up object from desk", (b) corresponding event probability. (c) shows the trajectory of "pick up umbrella from cabinet" and (d), its event probability sequence. Both (b) and (d) have two dominant events.



(a) (b) (c) (d)

Fig. 2. Trajectories are shown in the top row and the corresponding event probability sequences in the bottom row. (a) and (b): Two instances of "open door"; (c) and (d):"Close door"

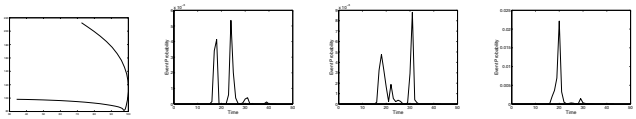
5.1. Activity Recognition

The UCF dataset consists of 60 trajectories of common activities, and are described in [6]. We divide the list of activities into 7 main categories: open door, pick up, put down, close door,erase board, pour water into cup and pick up object and put down elsewhere. The hand is tracked as it performs the different activities and the resulting trajectory is smoothed out using anisotropic diffusion. Figures 1 and 2 show some examples of the hand trajectory executing different activities. Typically, most of the activities in the dataset last for a few seconds i.e., of the order of 100 frames. The number of events in an activity ranges from 1 to 6. At the matching stage, only these events are compared.

Finding events in the training data: The Baum Welch algorithm is used to estimate the parameters of the HMM. In our experiments, we use a 4 state, single Gaussian, left-to-right model. Using the parameters of the trained HMM, we compute the η_t^p and e_t for different values of p as defined in (1) and (2). In our recognition experiments, p was set to 5. The HMM model along with the event probability sequence is stored as the signature of the activity.

The trajectories may appear differently, but resemble at the event level. In other words, though the state descriptions of the associated HMMs are different, the activities have similarities in the dominant transitions of the hidden states. For example, referring to the "pick up" action in figure 1(a) and (c), the trajectories for picking up an object from the desk looks different from that of the umbrella, but the two activities resemble each other at the event level (Figures 1(b) and (d)). In the close door activities in Figures 2(a) and (c), we see that the two leading events correspond to the closing action, and the third event occurs because of some random motion of the hand. Our method is robust to such false alarms in the detected events.

Matching the test sequence: Given an unknown (test) trajectory, we compute the event probability sequence for every model



(a) (b) (c) (d)

Fig. 3. (a) shows the hand trajectory as the action "pick up object" is executed. (b)-(d) show a plot of event probability, as a function of time for the region of support, $p = 3, 5, 7$ respectively. Increasing p in $\eta_t^{\mathrm{p}}(i,j)$ causes two events to merge, at the time when the actual pick up action occurs.

in the database i.e., compute $e_t^{p,g}(k,l)$ using HMM λ_g for $g = 1, \ldots, G$, where G is the number of trajectories in the database. The similarity score between the two event probability sequences is computed using the dynamic time warping algorithm.

Table 1 summarizes the recognition results using cumulative match scores (CMS) as the performance measure. CMS is computed by accumulating recognition rates from rank 1 onwards. For instance, a CMS of 90% at rank 5 means that on an average, within the top 5 matches, the action is correctly recognized 90% of the time. The first two columns are the CMS scores at rank 1 and 5 respectively, obtained using the proposed method. We compare our recognition rates with those in [6], whenever the experiments are comparable.

Comparison with the UCF method[6]: The *dynamic instants* defined in [6] can be considered to be a subset of the event probabilities e_t. The events in e_t do not require sharp curvatures in the trajectory unlike the dynamic instants. Even a single missed instant can cause an incorrect match in the dynamic instants method. For instance, the action "pick up umbrella while twisting hand" is not recognized as a pick up action in [6] because of excessive peaks in curvature caused by the twisting action. To make quantitative statements about the tolerance level to the number of such missed events in our method, we need a more detailed analysis. Further, collinearity in the trajectory is an issue in [6] and causes incorrect matches. In our case, the changes in the direction of motion of the hand is sufficient to produce the event probabilities. For similar reasons, recognition using dynamic instants cannot deal with composite activities that have subactivities.

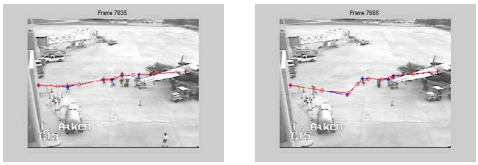
Choice of p: The effect of increasing the region of support p depends on the extent of variation of the trajectory. If the variation in the trajectory itself is less, then we can expect strengthening of those events that are less dominant at lower values of p. This is because of the regularity at the observation level that is preserved at the state level (see Figure 3). On the other hand, if the motion trajectory has significant fluctuations, then this tends to get reflected to a large extent in the state sequence as well. Increasing the value of p will therefore cause some of the events that were less significant initially, to gradually disappear. At the same time, the dominant event may be reinforced. In other words, the number of events detected is tied to the choice of p.

5.2. Anomalous trajectory detection

The TSA dataset consists of surveillance video captured in an airport. A part of it contains images of passengers deplaning and walking across the tarmac to the terminal gate. The scene is 320×240 pixels wide, and the people are about $10 - 15$ pixels tall. (see Fig-

Table 1. Cumulative match score (CMS) percentages for activity recognition at rank 1 (recognition rate) and rank 5. Recognition rates in column 2 for proposed method is compared with those reported in [6] in column 4)

Expt. set	CMS rank 1	CMS rank 5	UCF results CMS rank 1
Open door	72	94	50
Pick up	70	95	-
Put down	72	94	-
Close door	80	100	53
Erase board	50	100	75
Pour water	33	100	33
Pick up and put down	67	100	56
Overall	65	95	61



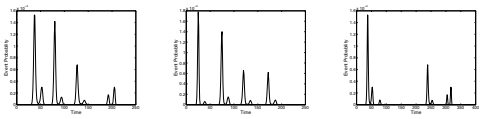
(a) (b)

Fig. 4. (a) Snapshot of TSA dataset (b) Simulated anomaly - person walks away from the virtual path and rejoins path

ure 4). Tracking is difficult and error-prone. The 5 best trajectories of people walking (out of 11) are chosen as the normal trajectories. Since we do not have any instances of anomalous trajectories, we simulate one by considering deviation from the path as an anomaly (Figure 4(b)).

Using the set of normal trajectories, a $5-$state HMM is trained, assuming a left-to-right model. From our experiments, we observe that the choice of the number of states is not critical. For each of the normal trajectories, the event probability sequence is computed. Figures 5 (a) and (b) show two such sequences. Given a new (anomalous) trajectory, we use the HMM in the database to compute the event probabilities. If the anomaly is present in a part of the trajectory and resumes the normal activity after sometime, then this is reflected in the events detected. The method does not accumulate errors at all time instants, but only based on the times when events occur or when an event was expected to occur as seen in the training data.

To declare the presence of an anomaly, we use both the number of events detected (spurious and missing events are both anomalies) as well the location of the detected events. Figure 5 (c) shows the event probability sequence for a person who deviates from the normal path and later rejoins the virtual path. We observe that the latter two dominant peaks in (c) resemble the latter half of the normal event sequence, whereas a missing event in the first half indicates an anomaly.



(a) (b) (c)

Fig. 5. (a) and (b) show the event probability sequence for two normal trajectories. (c) shows the events for an anomalous trajectory.

6. CONCLUSION

We have presented a technique to represent activity trajectories using 'event probabilities', using an HMM framework. The motivation for this approach was the need for a semantically viable representation rather than a purely statistical one. These events capture the essence of the activity, and provide a way of focussing our attention on the salient portions of the trajectory, rather than using the entire trajectory for recognition and for other tasks. We have demonstrated the application of such events to activity recognition and anomaly detection. A more complete evaluation of the proposed method requires a larger dataset. We also need a way of quantifying the discriminating capacity of the event probability sequence, perhaps through the use of the number of bits required to represent the events associated with the different activities.

One of the drawbacks of using the motion trajectories as the feature is that appearance and other information is lost. Instead of using the centroid of the moving object, a more descriptive feature vector is the time variation of a bounding box around the moving object. Presently, an event probability is based on a simple step edge with a certain support region. For example, state sequence $Q = \{2, 2, 2, 1, 1, 1\}$ is an event with support $p = 2$. We categorize events based on the two states involved in the transition, the region of support and the probability of the transition. This can be modified to include more complex patterns so that the events of various types can be detected; for example, an event of the type $Q = \{2, 1, 2, 1, 2, 1\}$. A more interesting problem is discovering such event patterns given multiple observations of an activity.

7. REFERENCES

[1] J.W. Davis and A.F. Bobick, "The representation and recognition of action using temporal templates," *Proc. CVPR*, 1997.

[2] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," *Proc. Intl. Symposium on Computer Vision*, pp. 265–270, 1995.

[3] Y.A. Ivanov and A.F. Bobick, "Recognition of visual activities and interactions by stochastic parsing," *IEEE Trans. PAMI*, vol. 23, pp. 852–872, August 2000.

[4] L.R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. of the IEEE*, vol. 77, no. 2, pp. 257–285, February 1989.

[5] B. H. Juang, "On the hidden markov model and dynamic time warping for speech recognition - a unified view," *Technical Journal*, vol. 63, pp. 1213–1243, 1984.

[6] C. Rao, A. Yilmaz, and M. Shah, "View-invariant representation and recognition of actions," *IJCV*, vol. 50, no. 2, 2003.

[7] N. Vaswani, A. Roy Chowdhury, and R. Chellappa, "Activity recognition using the dynamics of the configuration of interacting objects," *Proc. CVPR*, June 2003.

[8] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," *Proc. CVPR*, pp. 819–826, 2004.