Clustering Human Behaviors with Dynamic Time Warping and Hidden Markov Models for a Video Surveillance System

Kan Ouivirach and Matthew N. Dailey
Computer Science and Information Management
Asian Institute of Technology
kan@ieee.org, mdailey@ait.ac.th

***Abstract*—We propose and experimentally evaluate a new method for clustering human behaviors that is suitable for bootstrapping an anomaly detection module for intelligent video surveillance systems. The method uses dynamic time warping, agglomerative hierarchical clustering, and hidden Markov models to provide an initial partitioning of a set of observation sequences then automatically identifies where to cut off the hierarchical clustering dendrogram. We show that the method is extremely effective, providing 100% accuracy in separating anomalous from typical behaviors on real-world testbed video surveillance data.**

I. Introduction

Human behavior understanding is an important component of a wide variety of desirable intelligent systems. However, the problem is very difficult, due to the wide range of activities possible in any given context and the large amount of variability within any particular activity. Many researchers have attempted to build systems able to interpret and understand human behaviors. The most classic work is from Yamato et al. [1], who model tennis actions using hidden Markov models (HMMs). Du et al. [2] present an approach to recognize interaction activities using dynamic Bayesian networks (DBNs) that outperforms conventional HMMs. Gao et al. [3] use a single mixture-of-Gaussian HMM, in which the states represent the stages of dining activities, to monitor the eating behavior of elderly people in a nursing home.

As video monitoring is becoming more ubiquitous in our lives, research on advanced video surveillance analysis is increasingly important. To help security personnel work reliably and efficiently, we would like to filter out typical events, and in cases of anomalous events, automatically raise an alarm or present the event to a human operator for consideration as a security threat. One limitation of most of the work is that the number of "normal" behavior patterns need to be known beforehand. One example is that of Nair and Clark [4], who built an automated video surveillance system using HMMs, each modeling a common, predefined activity in a scene.

More recently, research has started to focus on unsupervised analysis and clustering of behaviors in a particular scene for a variety of purposes including anomaly detection, surveillance, and classification. Zhong et al. [5] treat video segments as documents and cluster the documents based on the co-occurrence information. Li et al. [6] cluster human gestures by constructing an affinity matrix using dynamic time warping (DTW) [7] then apply the normalized-cut approach to cluster the gestures. Hautamäki et al. [8] apply DTW and use the pairwise DTW distances as input to a hierarchical clustering process in which k-means is used to fine-tune the output.

In this paper, we use a combination of clustering and HMMs to group human behaviors in a scene. There is some recent related work using graphical models such as HMMs to cluster behavior patterns. Li and Biswas [9] use the Bayesian information criterion (BIC) for HMM model selection and construct a binary hierarchical clustering dendrogram to initialize data partitions based on a sequence-to-model likelihood distance measure. They then compare each pair of clusters using a partition mutual information (PMI) criterion [10] to find the optimal number of clusters.

Xiang and Gong [11] model the distribution of activity data in a scene using a Gaussian mixture model (GMM) and also employ the BIC to select the optimal number of behavior classes prior to HMM training.

Swears et al. [12] propose hierarchical HMM-based clustering to find and cluster motion trajectories and velocities in a highway interchange scene. They build up a set of HMMs incrementally. For each new trajectory, they first test the likelihood of the trajectory according to each existing HMM model. If the new observation is not fit by any existing model, it is considered deviant and is grouped with other deviant observations to form a new HMM.

Alon et al. [13] propose a method to discover groupings of similar object motions. They apply a finite mixture of HMMs where the number of mixture components is assumed to be known. They estimate the number of clusters using the minimum description length (MDL) criterion [14], a penalized likelihood measure.

We propose a new method for clustering human behaviors in the context of video surveillance. After extracting sequences of features representing individual human behaviors in a given scene, we use DTW to measure the pairwise similarity between sequences. Then we construct an agglomerative hierarchical clustering dendrogram based on the DTW similarity measure. To find the optimal set of behavior clusters, we start at the root of the tree, train a HMM on the patterns in that cluster, and determine how well the HMM models the set

of patterns in the cluster. When we find that a HMM is an insufficient representation of the patterns in a given cluster, we throw away that HMM and recursively consider each of the child clusters according to the pre-calculated DTW-based dendrogram. Our method is able to automatically find the common human behaviors occurring in a given scene. In an experiment with a testbed video surveillance data set, we find that the method separates typical behaviors and abnormal behaviors into separate sets of clusters with 100% accuracy.

The most similar related work is that of Oates et al. [15], who first proposed the idea of using the DTW with HMMs to cluster time series. They use the DTW dendrogram cut off at an arbitrary depth as an initial partition of the training sequences, then they train HMMs on each partition iteratively until they have a set of HMMs that models all of the training sequences. They apply their method to simulated time series with good results but report obtaining poor clustering results in an experiment with real robot sensor data.

Our method has the potential to improve upon the state of the art in intelligent video surveillance applications by bootstrapping human behavior classification and anomaly detection modules in a given installation. Once a set of initial clusters and corresponding HMMs representing typical behavior is determined, we can easily find which cluster a new sequence should fall into by performing statistical tests on the sequence's likelihood according to each HMM model. When the likelihood is low according to all of the pre-existing clusters, we can consider the sequence to be anomalous and alert a human operator. When the likelihood is sufficiently high for one of the pre-existing models, we can simply incrementally update the sufficient statistics for that model. This approach would allow raising alerts for behaviors inconsistent with the automatically-derived typical behavior profile for the scene while providing adaptation to gradual changes in typical behavior patterns over time. We do not focus on these incremental learning and anomaly detection issues in this paper, but we plan to in future work.

II. Human Behavior Pattern Clustering

A. Overview

Fig. 1 provides an overview of the architecture of our proposed method. The blob extraction phase (Fig. 1a) generates observation sequences from videos as follows:

1) Grab a few initial frames from an the input video to model the background scene.
2) Perform foreground extraction to get a list of blobs.
3) Find the single largest blob in the scene, remove any pixels likely to be shadow pixels, and extract feature vector $\vec{f_t}$ for the blob at time t.
4) Apply vector quantization using the k-means algorithm to convert features into symbols.
5) Aggregate symbol sequences and store for batch cluster analysis.

In the behavior clustering phase (Fig. 1b), for the set of all discrete symbol sequences, we perform the following steps:

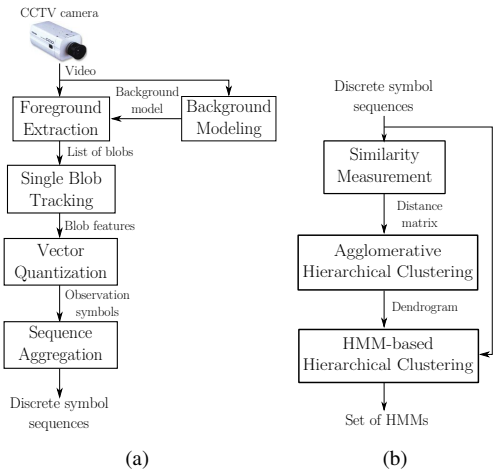

Fig. 1. Block diagrams for the proposed method. (a) Blob extraction flow. (b) Behavior clustering flow.

Fig. 2. Sample foreground extraction and shadow removal results. (a) Original image. (b) Foreground pixels according to background model. (c) Foreground pixels after shadow removal.

1) Apply dynamic time warping to the set of sequences to construct a similarity-based distance matrix.
2) Run agglomerative hierarchical clustering on the distance matrix to get a dendrogram.
3) Perform HMM-based hierarchical clustering using the set of sequences and the dendrogram to get a set of HMMs modeling the typical behaviors in the scene.

B. Blob Extraction

After discarding the no-motion video segments, we use Poppe et al.'s [16] background modeling technique to separate the moving foreground pixels from the background. Poppe et al. extend the standard mixture of Gaussians background model [17] to handle gradual illumination changes.

To remove shadows cast by moving objects, we eliminate any foreground pixels whose gray level normalized cross correlation (NCC) with the background model is above some threshold. This method works well in outdoor scenes with visible background texture inside shadows. Sample results from the foreground extraction and shadow removal procedures are shown in Fig. 2.

We next apply morphological opening and closing operations to remove noise and connect foreground regions, then we obtain the connected foreground components and filter out any components whose size is below threshold. For simplicity, we focus only on videos containing a single moving blob.

We then represent a blob (connected foreground component)

at time t by the feature vector

$$\vec{f_t} = \begin{bmatrix} x_t & y_t & s_t & r_t & dx_t & dy_t & v_t \end{bmatrix},$$

where (x_t, y_t) is the centroid of the blob, s_t is the size of the blob in pixels, r_t is the aspect ratio of the blob's bounding box, (dx_t, dy_t) is the unit-normalized motion vector for the blob compared to the previous frame, and v_t is the blob's speed compared to the previous frame.

After extracting the feature vectors $\vec{f_t}$ over a training set, we quantize them into discrete symbols using k-means. To prevent differing numeric scales of the features from affecting the distance metric, we normalize each feature independently by z-scaling to a mean of 0 and standard deviation of 1 over the training set. For the vectors (x_t, y_t) and (dx_t, dy_t), rather than normalize the x and y components independently, we use a common isotropic scale factor for the two dimensions to avoid overemphasizing small deviations from typical trajectories in directions without much deviation in the training data.

Currently, we empirically tune the free parameters (frame buffer length, thresholds, and number of k-means clusters) to the training data. We hope to automate the blob extraction parameter selection process in future work.

C. Behavior Clustering

Here we model the common behaviors in a scene by clustering a set of observation sequences acquired over some period of time. First, we apply DTW to estimate the similarity between every pair of training sequences despite variations in length and speed, to obtain a similarity matrix. Second, we use the similarity matrix for hierarchical agglomerative clustering by first combining the most similar two sequences into a single cluster then repeatedly merging clusters until just one cluster is left at the root of the tree or dendrogram. To determine the similarity of two clusters during this step, we use the similarity of the most similar pair of sequences between the two clusters.

The resulting DTW dendrogram provides a convenient representation of the similarity structure within a set of time sequences, but hierarchical clustering always comes with the practical issue of determining the optimal cutoff or number of clusters to use in a particular application. We solve this problem using HMMs as described below.

The flow of the algorithm is summarized in Fig. 3. We begin at the root of the hierarchical clustering dendrogram and attempt to model the sequences in that cluster (all training sequences, for the root) using a HMM. When there are more than N sequences in parent cluster c whose per-observation log-likelihood (calculated using the forward algorithm [18]) is less than a threshold p_c, we consider the HMM to be inadequate, throw it away, and then recursively attempt to model each of c's children in the DTW dendrogram. We use $N = 10$ in our experiments.

To determine the optimal rejection threshold p_c for cluster c, we use an approach similar to that of Oates et al. [15]. We generate random sequences from the HMM and then calculate the mean μ_c and standard deviation σ_c of the per-observation log likelihood over the set of generated sequences. For the lengths of the generated sequences, we simply use the average length of the training patterns in cluster c. After obtaining the statistics of the per-observation log likelihood, we let p_c be $\mu_c - z\sigma_c$, where z is an experimentally tuned parameter that gives us convenient control over the probability of making Type I errors in classifying a particular sequence as having been generated by a particular HMM model.

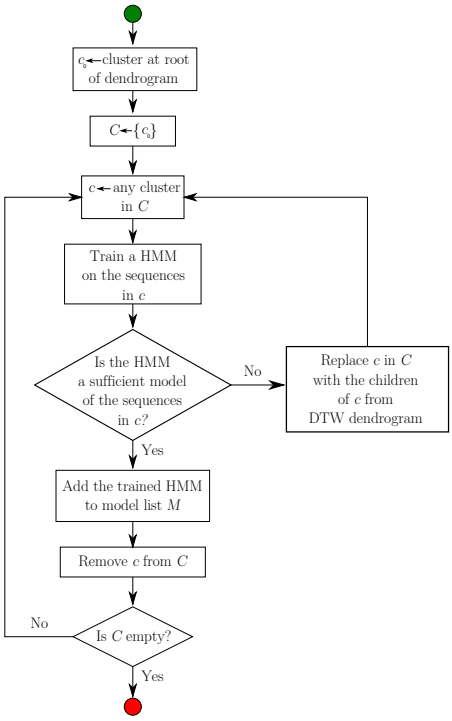

Fig. 3. Processing flow of the use of HMM clustering method

III. Experimental Results

To create a testbed data set, we mounted a CCTV camera to view the scene in front of an academic building, as seen in Fig. 2a. We recorded videos at a resolution of 320×240 and 25 frames per second over one week during working hours (9:00–17:00). To save disk space, we used a motion detection technique that automatically segments a raw video stream into separate videos containing motion. We obtained videos corresponding to over 500 motion events then manually selected the 298 videos containing only a single motion.

We found that there are at least four common behaviors in this scene: people walking into the building, walking out of the building, parking a bicycle, and riding a bicycle out. Fig. 4 shows examples of each of these behaviors. Other less common activities include people walking into the scene then walking out or people walking while telephoning and leaving the scene. For purposes of evaluating the results of our algorithm, we hand-labeled each of the videos with the categories Walk-in, Walk-out, Cycle-in, Cycle-out, or Other.

We performed three experiments to evaluate our method. In Experiment I, we applied our proposed method, as previously

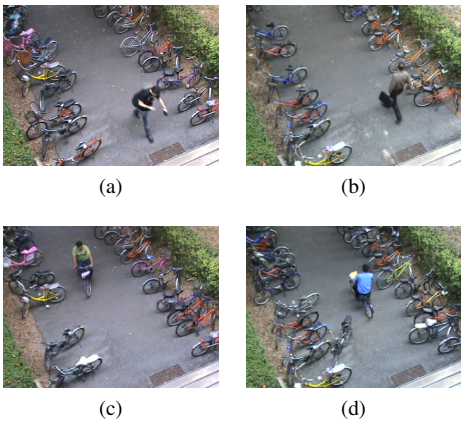

Fig. 4. Examples of common human activities in our testbed scene. (a) Walking in. (b) Walking out. (c) Cycling in. (d) Cycling out.

described, to cluster the 298 single-motion videos in our testbed data set. In Experiment II, we used an alternative clustering method that only uses recursive modeling by HMMs, without DTW. In Experiment III, we used an alternative method combining HMMs with supervised learning. In every experiment, we evaluated the clustering results (or classification results in the case of the supervised system of Experiment III) according to how well the induced categories separate the anomalous sequences (hand-labeled with the category "Other") from the typical sequences (Walk-in, Walk-out, Cycle-in, Cycle-out). Our main hypothesis was that *using DTW as a pre-process prior to HMM-based clustering should improve the quality of the clusters* in terms of separating anomalous from typical behaviors. One might also have hypothesized that supervised learning (Experiment III) would be better than either of the unsupervised methods (Experiments I and II), but as we shall see, we obtained a somewhat surprising result to the contrary.

In all three experiments, we used linear HMMs with four states and bypass transitions. That is, each HMM had transitions from state 1 to states 1, 2, and 3, from state 2 to states 2, 3, and 4, from state 3 to states 3 and 4, and from state 4 to itself. We chose this model structure based on our previous empirical experience [19].

To find the distribution (parameters μ_c and σ_c) of the per-observation log likelihood for a particular HMM, we always generated 1000 sequences of 120 observations then used a z-threshold of 2.0, corresponding to a Type I error (probability of misclassifying a sequence generated by the HMM as not generated by the HMM) of 0.0228. We fixed the parameter N (the number of deviant patterns allowed in a cluster) to 10.

A. Experiment I (DTW+HMMs)

The clustering results are shown in Table I. The method obtained 97 clusters. For the 17 clusters containing more than one sequence, we show the distribution of the activities represented by each sequence. For the 80 clusters containing only a single sequence, we summarize their distribution across the activity categories in the last row of the table.

TABLE I
CLUSTERING RESULTS FOR EXPERIMENT I (DTW+HMMS).

Cluster #	Walk-in	Walk-out	Cycle-in	Cycle-out	Other
1	96	0	18	0	0
2	0	54	0	5	0
3	0	3	0	8	0
4	0	2	0	0	0
5	0	1	0	2	0
6	0	0	0	2	0
7	0	0	0	2	0
8	0	0	0	2	0
9	0	0	0	2	0
10	0	0	2	0	0
11	0	0	2	0	0
12	0	0	3	0	0
13	0	0	1	1	0
14	0	0	0	0	4
15	0	0	0	0	4
16	0	0	0	0	2
17	0	0	0	0	2
One-seq clusters	4	17	34	21	4

It is clear from the results that the separation of anomalous and typical behaviors is excellent (100% accuracy), with the caveat that 80 sequences (26.8% of the data set) fall into single-sequence clusters that would have to be manually examined by a human operator if the method was used to bootstrap a real-world surveillance system.

B. Experiment II (HMMs only)

To determine the extent to which our system benefits from preprocessing using DTW, in this experiment, we used the concept of recursive modeling of the data using HMMs without using DTW as a pre-process. The method is similar to that of Swears et al. [12]. We begin by training a single HMM on all sequences and computing the distribution of the per-observation log-likelihood for that HMM as previously described. We then assign every sequence with a per-observation log-likelihood above threshold p_c to a cluster then repeat the process by training a new HMM on the remaining sequences. Similarly to the method of Experiment I, we stop splitting whenever the number of deviant sequences in the cluster is less than 10.

The results are shown in Table II. The method obtains only three clusters, and the clusters are incapable of separating anomalous behaviors from typical behaviors. With manual assignment of all three clusters to the "typical" category, we would achieve a recall of 0, a precision of 100%, and an accuracy of 94.6%. By manually assigning cluster 1 to the anomalous category, we would achieve a recall of 100%, a precision of 8%, and an accuracy of 38.3%. These strikingly poor results confirm our main hypothesis, and we conclude that recursive HMM modeling without DTW-based preprocessing is useless for video surveillance.

C. Experiment III (Supervised classification with HMMs)

As an alternative approach to separating anomalous from typical behaviors, in this experiment, we trained four HMMs

TABLE II
CLUSTERING RESULTS FOR EXPERIMENT II (HMMS ONLY).

Cluster #	Walk-in	Walk-out	Cycle-in	Cycle-out	Other
1	15	77	49	43	16
2	80	0	11	2	0
3	5	0	0	0	0

on 80% of each of the four typical behaviors observed in our testbed data set. We retained 20% of each typical sequences and the 16 anomalous sequences as a test set. After HMM training, we manually determined the best per-observation log-likelihood threshold for each HMM by maximizing the F1 value (a measure combining both precision and recall) for the separation between the positive and negative test patterns. The anomalous pattern detection rate of the the combined classifier was 50% (8 patterns) with a false alarm rate of 24.6% (16 out of 65 normal testing patterns).

A priori, one might have hypothesized that supervised sequence classification should outperform the unsupervised method we have proposed in this paper. To the contrary, we find that the simple-minded approach of training a single HMM on each typical behavior is far inferior. Clearly, there is more variation within the behavior categories than can be handled precisely by single simple HMMs. These supervised results could presumably be improved by applying our clustering method within each behavior category, thus obtaining a collection of HMMs for each behavior category. However, we have already demonstrated that our method achieves perfect separation of anomalous and and typical behaviors *without any information about the labels*, so this approach would be unnecessarily laborious.

IV. CONCLUSION

In this paper, we have proposed and evaluated a new method for clustering human behaviors. The method could be used to bootstrap an anomaly detection module for intelligent video surveillance systems. The combination of DTW partitioning with linear HMM training turns out to be quite powerful; manual examination of the clusters obtained from our method shows a perfect separation between typical and anomalous behaviors on a real-world testbed video surveillance data set.

The combination of DTW with the type of linear HMMs we use in this work is surprisingly effective. It is likely that the patterns DTW groups together are perfectly suited for modeling by this type of HMM. We plan to further explore this idea in future work.

There are two key limitations to our current method: our blob extraction process is currently only robust for single-motion events, and, although the method achieves 100% accuracy in separating typical from anomalous events, it does so at the cost of creating a fairly large number of single-sequence clusters that would have to be manually identified as typical or anomalous by a human operator in a real surveillance setting.

In future work, we plan to address these limitations, combine the method with incremental learning for behavior understanding and anomaly detection, and integrate the module with the ZoneMinder open source video surveillance system [20].

ACKNOWLEDGMENTS

This work was partly supported by a grant from the Royal Thai Government to MND and by graduate fellowships from the Royal Thai Government to KO. Special thanks are due to Shashi Gharti, who assisted with the implementation of human tracking and feature extraction.

REFERENCES

[1] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov model," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1992, pp. 379–385.

[2] Y. Du, F. Chen, W. Xu, and Y. Li, "Recognizing interaction activities using dynamic Bayesian network," in *International Conference on Pattern Recognition (ICPR)*, 2006, pp. 618–621.

[3] J. Gao, A. G. Hauptmann, A. Bharucha, and H. D. Wactlar, "Dining activity analysis using a hidden Markov model," in *International Conference on Pattern Recognition (ICPR)*, 2004, pp. 915–918.

[4] V. Nair and J. Clark, "Automated visual surveillance using hidden Markov models," in *Vision Interface Conference*, 2002, pp. 88–92.

[5] H. Zhong, M. Visontai, and J. Shi, "Detecting unusual activity in video," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004, pp. 819–826.

[6] H. Li, Z. Hu, Y. Wu, and F. Wu, "Behavior modeling and recognition based on space-time image features," in *International Conference on Pattern Recognition (ICPR)*, 2006, pp. 243–246.

[7] H. Sakoe, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, pp. 43–49, 1978.

[8] V. Hautamäki, P. Nykänen, and P. Fränti, "Time-series clustering by approximate prototypes," in *International Conference on Pattern Recognition (ICPR)*, 2008, pp. 1–4.

[9] C. Li and G. Biswas, "Temporal pattern generation using hidden Markov model based unsupervised classification," in *Advances Intelligent Data Analysis (IDA)*, 1999, pp. 245–256.

[10] L. Bahl, P. Brown, P. de Souza, and R. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1986, pp. 49–52.

[11] T. Xiang and S. Gong, "Video behaviour profiling and abnormality detection without manual labelling," in *IEEE International Conference on Computer Vision (ICCV)*, 2005, pp. 1238–1245 Vol. 2.

[12] E. Swears, A. Hoogs, and A. Perera, "Learning motion patterns in surveillance video using HMM clustering," in *IEEE Workshop on Motion and video Computing (WMVC)*, 2008, pp. 1–8.

[13] J. Alon, S. Sclaroff, G. Kollios, and V. Pavlovic, "Discovering clusters in motion time-series data," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 375–381 vol.1.

[14] J. Rissanen, "Hypothesis selection and testing by the MDL principle," *The Computer Journal*, pp. 260–269, 1998.

[15] T. Oates, L. Firoiu, and P. R. Cohen, "Using dynamic time warping to bootstrap HMM-based clustering of time series," in *Sequence Learning: Paradigms, Algorithms, and Applications*, 2001, pp. 35–52.

[16] C. Poppe, G. Martens, P. Lambert, and R. V. de Walle, "Improved background mixture models for video surveillance applications," in *Asian Conference on Computer Vision (ACCV)*, 2007, pp. 251–260.

[17] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999, p. 252.

[18] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, pp. 257–286, 1989.

[19] K. Ouivirach, "Human behavior profiling for a video surveillance system," Master's thesis, Computer Science and Information Management, Asian Institute of Technology, 2006.

[20] P. Coombes, "ZoneMinder: A Linux-based camera monitoring and analysis tool," 2007, open source software available at http://www.zoneminder.com.