A Human-Centered Multiple Instance Learning Framework for Semantic Video Retrieval

Xin Chen, *Student Member, IEEE*, Chengcui Zhang, *Member, IEEE*, Shu-Ching Chen, *Senior Member, IEEE*, and Stuart Rubin, *Senior Member, IEEE*

***Abstract*—This paper proposes a human-centered interactive framework for automatically mining and retrieving semantic events in videos. After preprocessing, the object trajectories and event models are fed into the core components of the framework for learning and retrieval. As trajectories are spatiotemporal in nature, the learning component is designed to analyze time series data. The human feedback to the retrieval results provides progressive guidance for the retrieval component in the framework. The retrieval results are in the form of video sequences instead of contained trajectories for user convenience. Thus, the trajectories are not directly labeled by the feedback as required by the training algorithm. A mapping between semantic video retrieval and multiple instance learning (MIL) is established in order to solve this problem. The effectiveness of the algorithm is demonstrated by experiments on real-life transportation surveillance videos.**

***Index Terms*—Human-centered system, multiple instance learning (MIL), neural networks, relevance feedback, video retrieval.**

I. Introduction

With the development of Web technologies and multimedia databases, there is an urgent need for mechanisms to automatically detect and retrieve semantic events in videos based on video contents. The proposed framework in this paper strives to reach this goal.

We propose a human-centered multiple instance learning (MIL) framework for semantic video retrieval. The framework first performs the object tracking and segmentation, which extracts the content features and trajectories of moving objects in the video. Then, event models are constructed to model different semantic events. In the learning and retrieval phase, human feedback is incorporated, with which the learning algorithm learns from the feedback by depressing the "irrelevant" scenes and promoting "relevant" scenes. Instead of predefined "expert" knowledge, an individual user's subjective view serves as the guidance for learning.

The use of human feedback is inspired by a well-known technique in the field of image retrieval—relevance feedback (RF) [23], [25]. The basic idea is to ask the user's opinion on the retrieval results for a user-specified query target. Based on these opinions, the learning mechanism refines the retrieval results in the next iteration. This process iterates until a satisfactory result is obtained for the user. The purposes of using feedback in the proposed framework are as follows.

1) *Reduce the semantic gap*—It is inherently hard to make the machine understand the meaning of multimedia data by reading only pixels. There exists a "semantic gap" between the low-level features and the high-level semantic meaning. It is necessary that human provide some guidance to the machine.
2) *Progressively gather training samples and customize the retrieval process*—It is different from traditional classification processes in machine learning, where prior knowledge is required to compose the "training set" for each class.

In information retrieval, especially for large multimedia databases, multiple "relevant" and "irrelevant" classes exist according to the different preferences of different users. The data in each "relevant" class may constitute only a very small portion of the entire database. It is difficult to predefine a perfect set of training sets for all "relevant" classes before the query, due to the scarcity of "relevant" samples and/or the uncertainty of users' interests. With RF, the training set is built up gradually. This mechanism provides flexibility in information retrieval as it customizes the search engine for the needs of individual users.

Videos are composed of running images (frames). A set of consecutive frames is referred to as a *video sequence* (VS) in this paper. Objects can be extracted from each frame by an object segmentation algorithm [6]. From the perspective of each such object, its moving trajectory in consecutive frames is a kind of spatiotemporal data, which is referred to as a *trajectory sequence* (TS). The goal of the proposed semantic video retrieval framework is to extract semantic scenes by analyzing the spatiotemporal relations among objects in a video. Each long video can be segmented into a set of smaller consecutive VSs. Each VS may contain one or more TSs. After the initial query, the user provides a label, i.e., "relevant" or "irrelevant," according to whether the semantic scene in the VS is of his/her interest. The user does not specify which vehicle objects, in the VS, are actually involved in the accident and which ones are driving normally. That is to say, the VS label is known while its TS labels remain unknown. Since the semantic event analysis is based on TSs, we need to find out which specific TSs in that VS contribute to the VS label. If we consider a VS as a bag and its TSs as instances, this is exactly an MIL problem, where the bag label is known and the instance labels remain unknown. MIL predicts the labels for unseen bags (i.e., VSs). A VS is "relevant" if it has at least one "relevant" TS, otherwise it is "irrelevant." In MIL, we need to learn a mapping function between bag labels and instance labels. The role of human feedback in this process is to provide labels to the retrieved bags (VSs). In this way, we map the semantic video retrieval problem to an MIL problem. Using MIL, the retrieval engine offers a more convenient and friendly query mechanism to the user, who only needs to label a whole VS but not each individual trajectory (TS) of moving vehicles.

The core learning algorithm used in this paper is a neural network for time series data. The video data are time series data as they consist of sequences of values or events changing with time. There is a large amount of literature [2], [10], [18] on applying neural networks in forecasting the behavior of real-world time series data. However, relatively few work [11], [27] have addressed the issue of event detection in time series data using neural networks. In this paper, we explore the spatiotemporal models of a neuron for semantic event mining and retrieval from videos.

The framework is designed to be of general use and can be tailored to many applications. However, we use one particular application (traffic surveillance video retrieval) throughout the paper to demonstrate the design details. The semantic events in a transportation video database are incidents captured by the surveillance cameras, such as a car crash, U-turn, and speeding. Experimental results show the effectiveness of the proposed framework for traffic accident detection.

Manuscript received July 17, 2007; revised January 7, 2008 and April 29, 2008. Current version published February 25, 2009. The work of C. Zhang was supported in part by the UAB ADVANCE Program of the Office for the Advancement of Women in Science and Engineering and NSF DBI-0649894. The work of S. Rubin was supported by the SPAWAR Systems Center (SSC) Science &Technology (S&T) Initiative. This paper was recommended by Associate Editor E. Trucco.

X. Chen and C. Zhang are with the Department of Computer and Information Sciences, University of Alabama at Birmingham, Birmingham, AL 35294 USA (e-mail: chenxin@cis.uab.edu; zhang@cis.uab.edu).

S.-C. Chen is with the School of Computing and Information Sciences, Florida International University, Miami, FL 33199 USA (e-mail: chens@cs.fiu.edu).

S. Rubin is with the SPAWAR Systems Center (SSC) San Diego, Intelligence, Surveillance, and Reconnaissance (ISR) Department, San Diego, CA 92152 USA (e-mail: stuart.rubin@navy.mil).

Digital Object Identifier 10.1109/TSMCC.2008.2007257

A literature review is provided in Section II. Section III presents an overview of the framework. Section IV briefly introduces a semantic object extraction and tracking algorithm. Section V exemplifies the semantic event modeling. Section VI illustrates the design details of the learning and retrieval process. Section VII provides experimental results. Section VIII concludes the paper.

II. Literature Review

A. MIL and Relevance Feedback

The concept of diverse density (DD) is introduced by Maron and Lozano-Perez [15]. Zucker *et al.* [28] attempt with decision trees. Ramon *et al.* [20] propose the multiple instance neural network. Andrews *et al.* use support vector machines (SVMs) to solve MIL problem. Their method is called MI-SVM [1]. Cheung and Kwok [9] propose a framework that focuses on the kernels of MIL, in which a dynamic relation between bags and instances is built up through "loss functions."

As an important technique in information retrieval, relevance feedback [23] has also been well studied in content-based image analysis. Most RF research is based on the query point movement [22] or query reweighting techniques [14], [24]. The essential idea of query point movement is to move the estimation of the "ideal query point" toward relevant example points and away from irrelevant example points specified by the user in accordance with his/her subjective judgments. The query reweighting techniques take the user's query as the fixed "ideal query point" and attempt to estimate the best similarity metrics by adjusting the weight associated with each low-level feature.

B. Spatiotemporal Event Detection for Video Data

A lot of studies in this area are based on the generic visual properties of frames, which do not utilize the spatiotemporal information by tracking each semantic object in the video. As tracking can provide more accurate and detailed information about the behavior of objects in a video, there is also some research that utilizes object trajectories as the basis for analysis. Medioni *et al.* [16] developed an event detection system by defining some scenarios based on spatial and temporal properties of object trajectories. Many other works exploit stochastic methods in learning and recognizing video events. These methods mainly include hidden Markov models (HMM), SVMs, Bayes networks, etc. Bobick *et al.* [3] proposed a coupled HMM and the associated stochastic grammars for recognizing activities. Similarly in [19], a rule-based approach is used to set up event models and HMM is adopted again for automatic learning. The authors in [21] combined HMM, Bayes networks, and belief propagation to understand human behaviors. Other learning tools being adopted include nonlinear regression model such as SVM [4]. Shot detection is achieved in [4] by utilizing the SVM-based prediction error to form a similarity measure. Belief networks are used by Huang *et al.* [13] in which a traffic scene analysis algorithm is proposed.

C. Neural Network for Time Series Data

Neural networks have shown their great potential in time series analysis—especially in forecasting. The literature cites a large body of work that has approached this problem from various directions [18]. However, there has been very little research that directly extracts spatiotemporal semantic events by modeling the dynamic relation existing in time series data. Gao *et al.* [11] explore this direction by investigating the impact of the number of inputs and hidden layers in neural network design and testing on financial data. The authors in [27] designed an event discovery pipeline for medical time series data. Naftel and Khalid [17] propose to use self-organizing maps (SOMs) for clustering and classifying object trajectories, hence detecting abnormal object behaviors.

In summary, this study mainly differs from the aforementioned algorithms for three reasons: 1) we see video event detection and retrieval from a completely new point of view—i.e., transforming it into an MIL problem in order to provide the maximum convenience and flexibility to users; 2) feedback is incorporated in retrieving semantic events, which is rarely used in video retrieval. In addition, using the feedback, the database search can be customized to meet the needs of individual users; 3) the neural network architecture for time series data prediction is adjusted to a classification architecture for event detection.

III. System Overview

The raw video is analyzed by segmenting and tracking semantic objects in it. After tracking, the object trajectories are modeled with a curve-fitting technique. In the experiment, we test its performance on retrieving traffic accidents from traffic surveillance videos. The corresponding event model is built and the feature vectors of TSs at each sampling point are extracted. When the user submits a query asking for accidents, the system performs an initial query based on some heuristics and returns the initial retrieval results to the user. The user responds to each returned VS by providing feedback. The MIL mechanism, as proposed herein, will then learn from this feedback and refine the retrieval results in the next iteration. The whole process goes through several iterations until a satisfactory result is obtained.

IV. Semantic Object Tracking

Object segmentation and tracking are the preprocessing components. In our previous work [6], an unsupervised segmentation method called the simultaneous partition and class parameter estimation (SPCPE) algorithm is used to identify vehicle objects in traffic videos. We further improve the performance of SPCPE by coupling it with a background learning and subtraction method [6]. The framework in [6] also has the capability to track moving vehicle objects (segments) within successive frames. By distinguishing the static objects from mobile objects in the frame, tracking information can be used to determine the trails of vehicle objects.

Using this framework, lots of spatiotemporal data are generated. This provides a basis for semantic video mining and retrieval. The object centroids are used in analyzing the behavioral pattern of moving vehicles.

V. Semantic Event Modeling

In this study, a spatiotemporal model is built for traffic accidents. The focus is the sudden change of behavioral pattern of each vehicle. With each vehicle trajectory, three properties of the vehicle are recorded: velocity, change of velocity, and change of motion vector. Once the sampling rate is known, the velocity at each sampling point can be directly calculated. The change of velocity Vdiff at each point can also be easily calculated by deducting the velocity sampled at the previous sampling point from the current velocity. The change of motion vector is the absolute angular difference between the current and the previous

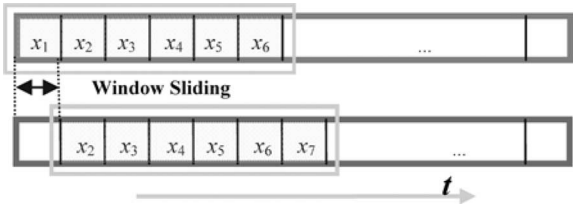

Fig. 1. Sliding window example.

motion vector. Additionally, for each vehicle, we also record its minimum distance from its nearest vehicle—mdist at each sampling point.

Some heuristics need to be established in order to process the initial query. This heuristic model is built upon the observation that a sudden change of velocity and/or driving direction may lead to an accident. Further, the closer a vehicle is to other vehicles, the greater the chance of an accident. At the ith sampling point, the property vector of a TS is $\alpha_i = [1/\text{mdist}_i, \text{vdiff}_i, \theta_i]$.

VI. Semantic Event Mining and Retrieval

A. Data Collecting and Problem Definition

In time series model of a neural network, there is a commonly used method termed a "sliding window." Fig. 1 shows an example, where a six-tuple sequence is extracted from time series data by sliding a window of size 6 one step at a time along the time axis t.

We use the sliding window technique to extract VSs. Each VS contains one or more TSs. Using user-provided feedback, some VS labels are known. In a traffic accident query, if a returned VS is labeled "relevant," then at least one vehicle's TS demonstrates abnormal behavior in that VS. If the VS label is "irrelevant," the labels of all the contained TSs are "irrelevant."

B. Learning and Retrieval Mechanisms

1) Neural Networks for Time Series Data: Suppose that m is the window size and the prediction or estimation of x_k is based on the preceding m observed data points $x_{k-m}, \ldots, x_{k-2}, x_{k-1}$. An exact value of x_k is required. Thus, prediction becomes a problem of function approximation. However, for video event mining and retrieval, only an indication of whether x_k will be an event of interest is needed. The problem now becomes how to map a TS to a class label of either "relevant" or "irrelevant:"

$$f_c : (x_{k-m}, \ldots, x_{k-2}, x_{k-1}) \rightarrow c_i \in C \tag{1}$$

where C is the set of all class labels. In this paper, our learning algorithm is based on a feedforward multilayer neural network that incorporates users' feedback. The structure of the proposed network is shown in Fig. 2.

The user's feedback is added as a node (fdk) in the input layer. The detailed design decisions are described in the following section.

2) Network Design:

1) *Window size and input nodes:* The size of the sliding window determines the number of input nodes, which can be simply decided by the typical length of an event. Take car crashes as an example. The typical length is about 15 frames. Given a sampling rate of five frames, three sampling points are needed to depict a car crash event. Thus, the window size is 3. The length of each TS is then three sampling points. Each input node, excluding the fdk node, is a feature vector at a sampling point of a TS. In a traffic accident event model, the feature vector $\alpha_i = [1/\text{mdist}_i, \text{vdiff}_i, \theta_i]$.

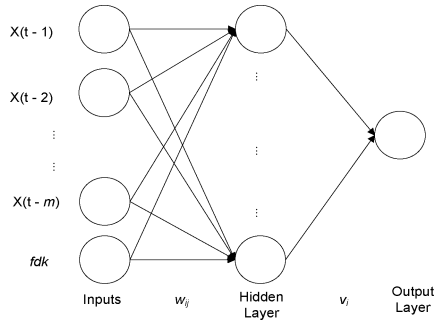

Fig. 2. Learning architecture of the proposed network.

TABLE I
Neural Network Parameters

Activation Func.	*Initial Weights*	*Search Algorithm*
$y = \tanh(\sum_i w_i x_i)$	Multiple Linear Regression [5]	Conjugate Gradient

2) *Hidden layer:* We adopt a two-layer neural network-–one hidden layer having a sigmoid transfer function and one output layer having a linear transfer function. It has been shown that this network architecture can approximate virtually any functions of interest to any degree of accuracy, provided sufficiently many hidden units are available [12].
There is one unit in the output layer indicating whether the TS is involved in the desired event or not. Suppose that the size of input is l, we tested on the hidden layer the sizes of $0.5l$, l, $1.5l$, and $2l$, and found that l generates the minimum estimated generalization error.
3) Other design issues are listed in Table I.

C. Human-Centered Event Learning and Retrieval

In the initial query, the user specifies an event of interest as the query target and there is no relevance feedback information. For each VS, we calculate its relevance (or similarity score) to the target query video event according to some event-specific heuristics. The relevance/similarity score of a VS is represented by the highest score of its containing TSs. The score of a TS is the highest score of its sampling points that is calculated as the square sum of all the three components in the feature vector $\alpha_j = [1/\text{mdist}_j, \text{vdiff}_j, \theta_j]$. The retrieval results are returned in descending order of the VSs' relevance scores.

The user identifies a returned VS as "relevant" if it lies in the province of his/her interest; otherwise, the user labels it "irrelevant." A set of training TS samples can thus be gathered. Each sample is in the form of $[\alpha_{t-2}, \alpha_{t-1}, \alpha_t, \text{fdk}, \text{opt}]$. fdk is zero if the user marks it "irrelevant;" otherwise, it is incremented by a small number ε. The number ε is set to 0.2 in our case as we assume that there are no more than five rounds of user feedback and the normalized input value is within a range of [0, 1]. opt is the desired output with the value of one for a "relevant" sample TS or zero for an "irrelevant" sample TS. The "relevant" training TS samples are collected by selecting the highest scored TSs in the "relevant" VSs and the "irrelevant" training TS samples are collected by selecting all the TSs in the "irrelevant" VSs. These training TS samples are fed into the neural-network-based learning framework. The trained neural network is then used to evaluate all the TSs. For these unknown TSs, their fdk is set to 0. The "relevant" sequences are promoted, after several iterations, by incrementing their fdk values,

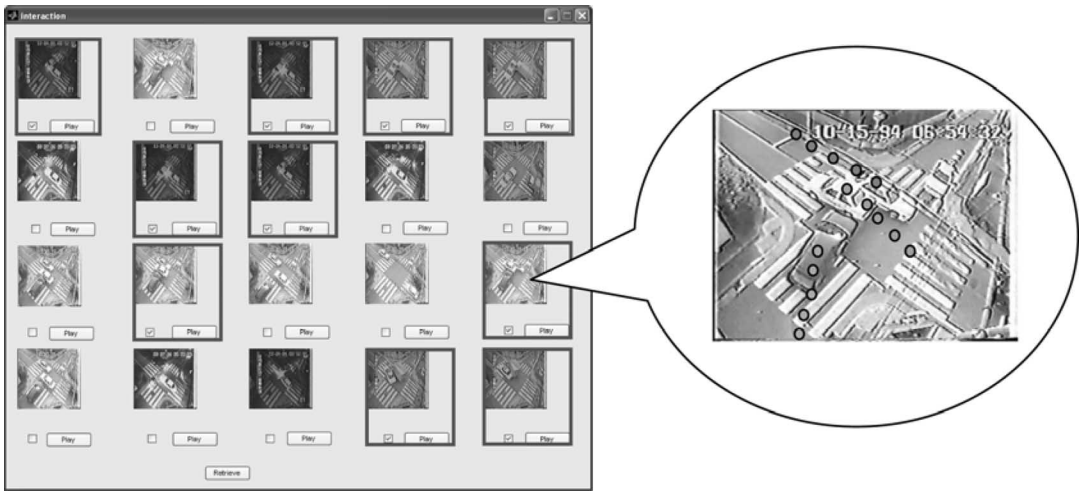

Fig. 3. User feedback interface.

while the "irrelevant" sequences are "penalized" by forcing their fdk value to stay minimal.

Like the traditional MIL, we first learn the labels of instances (TSs) in the bags (VSs) through a learning mechanism. Afterward, our approach is different from traditional MIL in that only top scored instances (TSs) in the "relevant" bags (VSs) are chosen to construct the training set, but not all relevant instances learned by the learning mechanism. This is to make sure that only "semantically relevant" instances are considered in the training process since they are all chosen by the user through RF.

VII. Experiments

A. Framework Interface

Fig. 3 shows the interface for the user to provide feedback information. The top 20 VSs are returned. The user can play the retrieved VSs. If the user believes that a VS contains an accident scene, then that VS will be selected. This is equivalent to labeling the VS "relevant." As shown in the given example, ten VSs (in blue rectangles) are labeled "relevant" given a traffic accident query. The enlargement of a sample "relevant" VS is shown on the right side of the interface with the trajectories of three vehicles marked by distinctly colored dots.

B. Performance Evaluation

The proposed framework is tested on real-world traffic video clips taken by traffic surveillance cameras at three different locations, including a road intersection in Taiwan, a road intersection in Russia, and a tunnel in Russia. There are altogether 5126 frames in these video clips. The sampling rate is five frames per sampling point and the window size is 3. Using the video segmentation method mentioned in Section VI, there are about 1000 VSs and 1343 TSs extracted from these video clips.

The proposed framework is compared with three other methods. The first one is the traditional weighted relevance feedback method. In this method, each feature in the feature vector α_i has a weight. The initial round of retrieval is the same as that of the proposed framework. The square sum of the feature components is computed as the relevance/similarity score. With RF, all relevant TSs are collected. The inverse of the standard deviation of each feature component is used as the updated weight for this feature in the next round of retrieval.

The proposed platform is also compared with the one-class SVM-based MIL algorithm [8]. The recent study on SVM active learning [26] has been shown to increase the accuracy and reduce the convergence rate on image retrieval. However, the algorithm proposed in [26] addresses the binary classification problem. In our experiment, we compare our framework with one-class SVM [8] rather than active SVM, because "relevant" class objects are "relevant" in a similar way while "irrelevant" class objects are "irrelevant" in their own different ways. Therefore, instead of considering "irrelevant" objects all in a single class, it makes more sense to treat them as outliers as does the one-class SVM. In addition, the "active learning" component in active SVM selects the images on the boundary as a "pool" and then asks the user to label these pool images since they are considered very informative. After several iterations, a classifier is learned by the active learning, which separates "relevant" images from "irrelevant" ones. However, in one-class SVM, active learning cannot be directly applied since the "boundary" is actually the "boundary" between one positive class and many negative classes, where the situation is too complicated to use active learning.

As mentioned at the outset of this paper, one of the advantages of the proposed framework is that the user is asked to identify only the relevant VSs without having to go through the trouble of further specifying the relevance of each TS within the VS. However, although the proposed framework provides such convenience to the user, the lack of information on the TS labels may degrade the performance. In order to test the robustness of the proposed framework, we compare it with the neural-network-based method without MIL [7], i.e., the retrieval results are presented in terms of tracked TSs instead of VSs. The user needs to identify the relevance of each TS. In lieu of knowing only the VS label, the learning algorithm knows the TS labels and is thus in possession of more information. The goal of the proposed framework is to render nearly equivalent performance while possessing less information than that of the non-MIL algorithm.

Five rounds of feedback are performed. The "accuracy" is the percentage of all the "relevant" VSs within the top 20 returned VSs. Fig. 4 shows the retrieval accuracies within the top 20 VSs taken at a tunnel in Russia after the initial, first, third, and fourth round of retrievals. "Weighted_RF" is the weighted RF algorithm. "SVM_MIL" is the one-class SVM-based MIL algorithm [8]. "BP" represents the neural-network-based algorithm without MIL [7]. "BP_MIL" is the proposed framework.

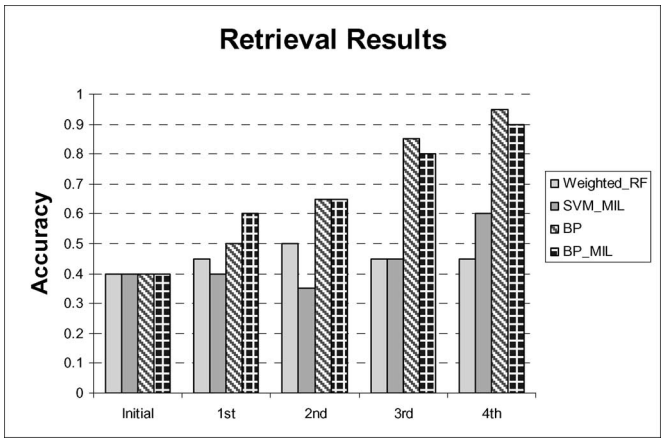

Fig. 4. Retrieval accuracies for the first video clip.

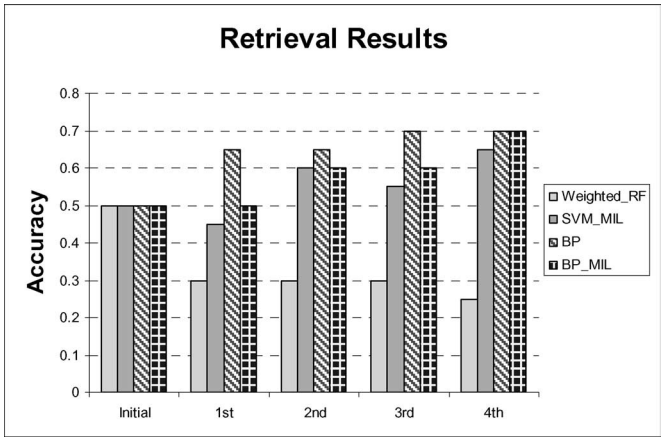

Fig. 5. Retrieval accuracies for the second video clip.

It can be gleaned from Fig. 4 that the initial accuracies of the four methods are the same since the same retrieval algorithm is used in the initial round. After that, "BP" and "BP_MIL" outperform the other two methods. By further comparing "BP" and "BP_MIL," we can see that the accuracy of "BP_MIL" is only 5% lower than that of "BP" after the fourth iteration. It only accounts for one VS among 20 VSs. The performance gain of the proposed framework over all five iterations is 0.50, while another MIL-based algorithm—"SVM_MIL" shows only 0.20 accuracy increase. The "Weighted_RF" method performs slightly better at the second and third iterations than the "SVM_MIL" method. However, its overall accuracy gain is only 0.10.

For the VSs taken at a road intersection in Taiwan (Fig. 5), while the accuracy gains with the proposed framework are not as high as that for the first clip, it is far better than that of the "Weighted_RF" method, in which performance degradation occurs right after the initial iteration. Compared with "BP", the accuracy rate of "BP_MIL," although lower in the first several iterations, is equivalent to that of "BP" at the fourth iteration. It is also worth noting that the performance of the "SVM_MIL" is much better than that in the first clip. Its accuracy rate increases from 50% to 65% after five iterations, which is only 5% lower than the proposed framework.

For the VSs taken at a road intersection in Russia (Fig. 6), since there are more TSs than in those taken at the previous two locations, the initial accuracy is much lower due to excess noise. However, the proposed algorithm achieves a performance gain of 70% in the fourth iteration;

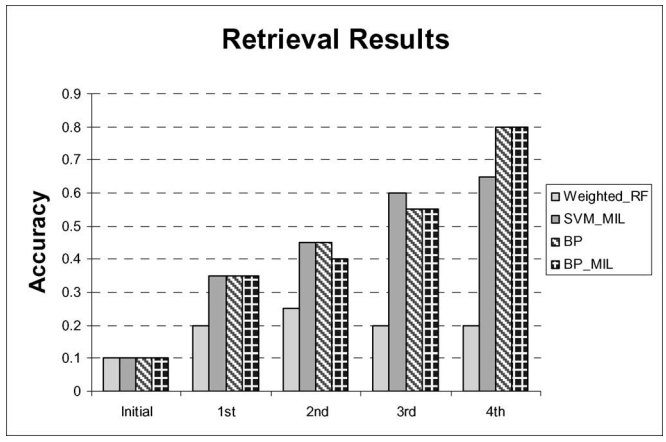

Fig. 6. Retrieval accuracies for the third video clip.

while SVM-based method has a 55% increase and the "Weighted_RF" method only increases by 15% in the second iteration.

VIII. Conclusion and Future Work

In this paper, a human-centered MIL framework for semantic video mining and retrieval is proposed. Given a set of raw videos, the semantic objects (i.e., vehicles) are tracked, and the corresponding trajectories are modeled and recorded in the database. Some spatiotemporal event models are then constructed. In the learning and retrieval phase, the user provides feedback on the relevance of each VS among the top returned VSs. The user needs to provide feedback only on the whole VS and the learning algorithm will analyze the contained TSs to find out the spatiotemporal patterns of users' interest. This can be transformed to an MIL problem. To solve this problem, the neural network for time series prediction is adapted to fit the specific needs of event identification for video data. The framework shows its effectiveness as demonstrated by our experimental results on real-life transportation surveillance videos.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," *Adv. Neural Inf. Process. Syst.*, vol. 15, pp. 561–568, 2003.

[2] S. Bengio, F. Fessant, and D. Collobert, "A connectionist system for medium-term horizon time series prediction," in *Proc. Int. Workshop Appl. Neural Netw. Telecoms*, 1995, pp. 308–315.

[3] A. F. Bobick, A. P. Pentland, and T. Poggio, "VSAM at the MIT media laboratory and CBCL: Learning and understanding action in video imagery PI report 1998," in *Proc. DARPA Image Understanding Workshop*, 1998, pp. 85–91.

[4] E. Bruno, N. Moenne-Loccoz, and S. Marchand-Maillet, "Unsupervised event discrimination based on nonlinear temporal modeling of activity," *Pattern Anal. Appl.*, vol. 7, no. 4, pp. 402–410, 2004.

[5] M.-C. Chan, C.-C. Wong, and C.-C. Lam, "Financial time series forecasting by neural network using conjugate gradient learning algorithm and multiple linear regression weight initialization," in Computing in Economics and Finance, 61, 2000, Society for Computational Economics: Barcelona, Spain. Available: http://fmwww.bc.edu/cef00/papers/paper61.pdf

[6] 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.

[7] X. Chen and C. Zhang, "An interactive semantic video mining and retrieval platform—Application in transportation surveillance videos for incident detection," in *Proc. IEEE Int. Conf. Data Mining (ICDM 2006)*, Hong Kong, China, pp. 129–138.

[8] X. Chen and C. Zhang, "A multiple instance learning framework for incident retrieval in transportation surveillance video databases," in *Proc. 2nd Workshop Multimedia Databases Data Manage., Conjunction IEEE Int. Conf. Data Eng.*, Istanbul, Turkey, 2007, pp. 75–84.

[9] P.-M. Cheung and J. T. Kwok, "A regularization framework for multiple-instance learning," in *Proc. Int. Conf. Mach. Learning*, Carnegie Mellon University, Pittsburgh, PA, 2006, pp. 193–200.

[10] R. J. Frank, N. Davey, and S. P. Hunt, "Time series prediction and neural networks," *J. Intell. Robot. Syst.*, vol. 31, pp. 91–103, 2000.

[11] D. Gao, Y. Kinouchi, K. Ito, and X. Zhao, "Neural networks for event extraction from time series: A backpropagation algorithm approach," *Future Generation Comput. Syst.*, vol. 21, pp. 1096–1105, 2005.

[12] K. M. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989.

[13] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, "Automatic symbolic traffic scene analysis using belief networks," in *Proc. Nat. Conf. Artif. Intell.*, Madison, WI, 1994, pp. 966–972.

[14] Y. Ishikawa, R. Subramanya, and C. Faloutsos, "Mindreader: Querying databases through multiple examples," in *Proc. 24th Int. Conf. Very Large Databases*, New York, 1998, pp. 218–227.

[15] O. Maron and T. Lozano-Perez, "A framework for multiple instance learning," *Adv. Natural Inf. Process. Syst.*, vol. 10, pp. 570–576, 1998.

[16] G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia, "Event detection and analysis from video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 8, pp. 873–879, Aug. 2001.

[17] A. Naftel and S. Khalid, "Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space," *Multimedia Syst.*, vol. 12, no. 1, pp. 227–238, 2006.

[18] D. W. Patterson, K. H. Chan, and C. M. Tan, "Time series forecasting with neural nets: A comparative study," in *Proc. Int. Conf. Neural Netw. Appl. Signal Process.*, Singapore, 1993, pp. 269–274.

[19] M. Petkovic and W. Jonker, "Content-based video retrieval by integrating spatio-temporal and stochastic recognition of events," in *Proc. IEEE Int. Workshop Detection Recog. Events Video*, Vancouver, BC, Canada, 2001, pp. 75–82.

[20] J. Ramon and L. D. Raedt, "Multi-instance neural networks," in *Proc. ICML 2000 Workshop Attribute-Value Relational Learning*, 2000, pp. 53–60.

[21] N. M. Robertson and I. D. Reid, "Behavior understanding in video: A combined method," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV 2005)*, Beijing, China, pp. 808–815.

[22] J. J. Rocchio, *Relevance Feedback in Information Retrieval*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[23] Y. Rui, T. S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," in *Proc. Int. Conf. Image Process.*, Washington, DC, 1997, pp. 815–818.

[24] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A power tool in interactive content-based image retrieval," *IEEE Trans. Circuits Syst. Video Technol., Spec. Issue Segmentation, Description, Retrieval Video Content*, vol. 18, no. 5, pp. 644–655, Sep. 1998.

[25] Z. Su, H. J. Zhang, S. Li, and S. P. Ma, "Relevance feedback in content-based image retrieval: Bayesian framework, feature subspaces, and progressing learning," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 924–937, Aug. 2003.

[26] S. Tong and E. Chang, "Support vector machine effective learning for image retrieval," in *Proc. ACM Multimedia*, Ottawa, ON, Canada, 2001, pp. 107–118.

[27] C. L. Tsien, "Event discovery in medical time-series data," in *Proc. AMIA Symp.*, 2000, pp. 858–862.

[28] J.-D. Zucker and Y. Chevaleyre, "Solving multiple-instance and multiple-part learning problems with decision trees and decision rules. Application to the mutagenesis problem," in *Proc. 14th Biennial Conf. Can. Soc. Comput. Stud. Intell.*, 2001, pp. 204–214.