A Multiple Instance Learning Framework for Incident Retrieval in Transportation Surveillance Video Databases

Xin Chen, Chengcui Zhang, and Wei-Bang Chen
Department of Computer and Information Sciences,
University of Alabama at Birmingham
Birmingham, Alabama, 35294 USA
Tel: (205)-934-8606, Email: {chenxin, zhang, wbc0522}@cis.uab.edu

Abstract

Traffic incidents are frequent query targets in a transportation surveillance video database. Therefore, understanding and retrieving transportation videos based on their semantic contents becomes an urgent task. For this purpose, this paper proposes an interactive Multiple Instance Learning (MIL) framework for semantic video retrieval. It incorporates techniques in multimedia processing, data mining, and information retrieval. By tracking vehicles' trajectories in a video and modeling semantic events, the framework initiates a progressive learning process guided by the user's Relevance Feedback (RF). The choice of RF is for reducing the "semantic gap" between the machine-readable features and the high level human concepts, which is a popular technique in the area of Content-based Image Retrieval (CBIR). With the information provided by RF, a mapping between semantic video retrieval and MIL is established. Due to its robustness to high-dimensional data, One-class SVM is selected to be the core learning algorithm for MIL in this framework. Although the proposed work is intended for transportation surveillance videos, it is designed as a general framework and can be tailored to other applications as well. The effectiveness of the algorithm is demonstrated by our experiments on real-life transportation surveillance videos.

1. Introduction

With the development of intelligent transportation system, a large amount of transportation surveillance videos are collected and stored in the database. Usually, these videos are organized with the corresponding meta-data such as the time and place a video is taken. However, a typical database query would be more interested in the semantic events in the video i.e. traffic incidents in this case. To manually add such information in the meta-data for query is simply not feasible in the view that such a database is usually gigantic in size. There is a need for mechanisms to detect and retrieve semantic events in videos based on the video contents. The proposed framework in this paper strives to reach this goal.

Relevance Feedback (RF) is a well-known technique in the field of image retrieval. It is used to incorporate the user's subjective perceptions with the learning process [2, 4] for Content-Based Image Retrieval (CBIR). The basic idea of Relevance Feedback is to ask the user's opinion on the retrieval result for a user-specified query target. Based on these opinions, the learning mechanism tries to refine the retrieval result in the next iteration. The process iterates until a satisfactory result is obtained for the user. As a supervised learning technique, Relevance Feedback has been shown to significantly increase the retrieval accuracy. In this paper, we borrow the idea of Relevance Feedback from CBIR and apply it to the retrieval of video data. This is one of the major contributions of the framework.

The purposes of using RF in the proposed framework are to: **1) Reduce the semantic gap** - It is inherently hard to make the machine understand the meaning of multimedia data by only reading pixels, frames or signals. There exists a "semantic gap" between the low level features and the high level semantic meaning. It is necessary that human provides some guidance to the machine. **2) Progressively gather training samples and customize the retrieval process -** The purpose of the proposed framework is to automatically learn and retrieve semantic scenes from videos according to the user's query. RF is used as a bridge between multimedia processing and information retrieval. It is different from traditional classification process in machine learning, where prior knowledge is required to compose the "training set" for each class. In the scenario of information retrieval, especially for large multimedia databases, multiple "relevant" and "irrelevant" classes exist according to the different preferences of different users [19]. The data in each "relevant' class may only constitute a very small portion of the entire database. Thus, in a large-scale multimedia database, it is difficult to pre-define a perfect set of training sets for all "relevant" classes before query, due to the scarcity of "relevant" samples and the uncertainty of users' interest. With RF, the initial query results are returned based on some heuristics i.e. the models of some generally categorized events. The training set for the user's specific query is built up gradually with the help of the user's

1-4244-0832-6/07/$20.00 ©2007 IEEE.

feedback. Therefore, RF provides more flexibility in information retrieval as it customizes the search engine for the need of individual users.

Video is composed of running images (frames). A set of consecutive frames is referred to as a **Video Sequence (VS)** in this paper. Objects i.e. vehicles can be extracted from each frame by a vehicle segmentation algorithm [20]. With image analysis, the content features of each vehicle object (e.g. its spatial location, texture features, shape, and color) in a frame can be extracted. From the perspective of each such object, its moving trajectory in consecutive frames is a kind of spatio-temporal data, which is referred to as a **Trajectory Sequence (TS)** in this framework. The goal of the proposed semantic video retrieval framework is to extract semantic scenes by analyzing the spatio-temporal relations among moving and still objects in the video.

Each long video can be segmented into a set of smaller consecutive VSs. Each such VS may contain one or more moving vehicle objects. In other words, one 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. For example, if the query target is traffic accidents, the user will label a VS with an accident scene "relevant" and vice versa. 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, the VS label is known wile its contained TS labels are 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 contained TSs as instances, this is exactly a Multiple Instance Learning (MIL) problem, where the bag label is known and the instance labels remain unknown. The goal of MIL is to predict the labels for unseen bags. A VS is "relevant" if it has at least one "relevant" TS, otherwise it is "irrelevant". Therefore, in MIL, we need to learn a mapping function between bag labels and instance labels. The role of RF in this process is to provide labels to retrieved bags (VSs) at each retrieval iteration. In this way, we map the semantic video retrieval problem to a MIL problem. This is another major contribution of the proposed work. To our best knowledge, this is the first paper to apply MIL in semantic video retrieval. With MIL, the retrieval engine offers a more convenient and friendly query mechanism to the user, who only needs to label a whole video segment (VS) but not each individual trajectory (TS) of moving vehicles in that video segment.

The core learning algorithm used in this paper is One-class Support Vector Machine (SVM). In our previous work [3], we applied One-class SVM [18] to solve MIL problems for region-based image retrieval and showed its effectiveness in manipulating high dimensional data. Same as images, video data are also high-dimensional and require a robust learning algorithm like One-class SVM. Some works such as [16] apply SVMs to solve MIL problems, however, most of which are in the area of CBIR. The proposed framework uses it in a novel and non-traditional way for semantic video retrieval.

In summary, the proposed interactive video retrieval framework first performs the object tracking and segmentation. Then event models are constructed to model different semantic events. In the learning and retrieval phase, Relevance Feedback is incorporated, with which the user provides feedback and the learning algorithm learns from it by depressing the "irrelevant" scenes and promoting "relevant" scenes. Instead of pre-defined "expert" knowledge, individual user's subjective view serves as the guideline for learning. In this framework, One-class SVM serves as the key learning mechanism to solve a MIL problem. It learns the spatio-temporal characteristics of user-interested video events, which is dynamic rather than static. By using users' feedbacks, human knowledge is incorporated into such a database. Although the framework illustrated in the paper targets traffic surveillance application, it is designed to be of general use and can be tailored to many other fields. In this study, the semantic events in a transportation video database are incidents captured by the surveillance cameras on the road, such as car crash, bumping, U-turn and speeding. Experimental results show the effectiveness of the proposed framework for traffic accident detection.

In the rest of the paper, a literature review is provided in Section 2. Section 3 briefly introduces a semantic object extraction and tracking algorithm for traffic surveillance videos. Section 4 exemplifies the semantic event modeling. Section 5 presents the design details of the learning and retrieval process. Section 6 provides the experimental results. Section 7 concludes the paper.

2. Literature review

2.1. Multiple Instance Learning (MIL)

A great amount of research has been done to solve Multiple Instance Learning problems. A representative approach by learning the axis-parallel rectangles is first developed by Dietterich et al. [5]. The concept of Diverse Density (DD) is introduced by Maron and Lozano-Perez [6] and a two-step gradient descent with multiple starting points is applied to find the maximum Diverse Density. The EM-DD algorithm is proposed by Zhang and Goldman [7] based on Diverse Density. Its main difference from Maron's method is that it searches maximum DD points by Expectation Maximization. It is shown that EM-DD is more robust in dealing with high-dimension data. Wang et al. [10] explore the lazy learning approaches in Multiple Instance Learning. Zucker et al. [11] attempt to solve the Multiple Instance Learning

problem with decision trees and decision rules. Ramon et al. [15] propose the Multiple Instance Neural Network. Andrews et al. use Support Vector Machines (SVMs) to solve MIL problem. Their method is called MI-SVM [16].

Some of the above mentioned algorithms have been applied to image classification or image retrieval. Our proposed learning framework applies One-class SVM to solve a MIL problem in semantic video retrieval.

2.2. Relevance feedback

In order to overcome the obstacle posed by the gap between high-level concepts and low-level features, the concept of relevance feedback (RF) associated with CBIR was first proposed in [2]. In the past few years, the RF approach to image retrieval has been an active research field. This powerful technique has proven successful in many application areas. In addition, various ad hoc parameter estimation techniques have been proposed for the RF approaches. Most RF techniques in CBIR are based on the most popular vector model [8, 21, 22, 24] used in information retrieval [26]. The RF technique estimates the user's ideal query by using relevant and irrelevant examples (training samples) provided by the user. The fundamental goal of these techniques is to estimate the ideal query parameters accurately and robustly.

Most previous RF research has been based on query point movement or query re-weighting techniques [26]. The essential idea of query point movement is quite straightforward. It represents an attempt to move the estimation of the "ideal query point" towards relevant example points and away from irrelevant example points specified by the user in accordance with his/her subjective judgments. Rocchio's formula [23] is frequently used to iteratively update the estimation of the "ideal query point". The re-weighting techniques, however, 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 [21, 25, 27]. The basic idea is to give larger weights to more important dimensions and smaller weights to less important ones.

As the Relevance Feedback techniques in the above mentioned works are applied to content based image analysis, we adjust it to fit the needs of semantic video retrieval in this paper.

2.3. Spatio-temporal event detection for transportation surveillance videos

In the field of transportation surveillance videos, most of the research is focusing on vehicle extraction and tracking, which is only the first phase towards studying the semantic meaning of videos captured by the surveillance camera. With the growing popularity of Intelligent Transportation Systems, automatic traffic incident detection is drawing the attention of more researchers. Various machine learning algorithms are explored for this purpose: 1) Hidden Markov Model is used in Porikli et al. [28] to estimate traffic congestion without vehicle tracking, and used in Kamijo et al. [32] for traffic monitoring and accident detection at intersections; 2) Belief Networks are used in Huang et al. [29] in which a traffic scene analysis algorithm is proposed based on that; Buxton and Gong [30] use Bayesian belief networks to model dynamic dependencies between parameters involved in visual interpretation. 3) Self-Organizing Map (SOM) is another popular choice as the hierarchical SOM in [31] and the fuzzy SOM in [33].

Dance and Caelli [34] present a traffic scene interpretation system, which is based on a cognition model in AI. This model is originally suggested by Marvin Minsky [9] and is implemented with the object oriented approach in [34]. Some statistical methods are also applied in this area. Fernyhough et al. [1] construct a set of qualitative event models in their work. In [35], a searching scheme is proposed that provides the functions of query by example, by sketch, and parameter weighting. The search algorithm in this scheme is based on parameters of moving trajectories. A nonparametric regression algorithm is examined in [12] for forecasting traffic flows. In [17], a threshold is set on the distance between two vehicles, which is used as the measurement for possible collision. Similarly, in [14], the overlap of two vehicles is regarded as the occurrence of collision.

The main difference of this paper from the above mentioned algorithms is that we see video event detection and retrieval from a completely new point of view i.e. transform it into a MIL problem in order to provide the maximum convenience and flexibility to users. No pre-defined event-specific models are needed prior to the retrieval, and the database search can be customized to meet the needs of individual users. Consequently, One-class SVM is chosen to solve this problem.

3. Semantic object tracking

3.1. Automatic vehicle tracking and segmentation

As traffic surveillance videos are the target of our study in this paper, in this section, we provide some background information on the processing of transportation surveillance videos. In our previous work [20], an unsupervised segmentation method called the Simultaneous Partition and Class Parameter Estimation (SPCPE) algorithm, coupled with a background learning and subtraction method, is used to identify the vehicle objects in a traffic video sequence. The technique of background learning and subtraction is used to enhance

the basic SPCPE algorithm and to better identify vehicle objects in traffic surveillance videos.

The framework in [20] also has the ability to track moving vehicle objects (segments) within successive video frames. By distinguishing the static objects from mobile objects in the frame, tracking information can be used to determine the trails of vehicle objects. Figure 1 shows an example of the tracking result of three vehicles. The yellow rectangular area is the Minimal Bounding Rectangle (MBR) of the vehicle. ($x_{centroid}$, $y_{centroid}$) are the coordinates of a vehicle segment's centroid represented by a red dot in the figure. It is used for tracking the positions of vehicles across video frames. The last phase of the framework is to classify vehicle objects into different classes such as SUVs, pick-up trucks, and cars, etc. The classification algorithm is based on Principal Component Analysis [13].

With this framework, lots of spatio-temporal data is generated. This provides a basis for semantic video mining and retrieval.

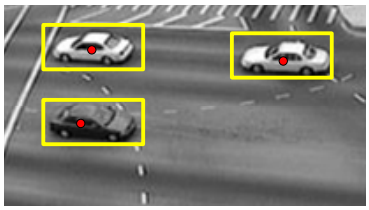

Figure 1. Tracked vehicle segments and their centroids

3.2. Trajectory modeling

By tracking each moving vehicle in the video, a series of object centroids on successive frames are recorded. We can approximate the trajectory of a vehicle by using the least-square curve fitting. A k^{th} degree polynomial for the curve is:

$$y = a_0 + a_1 x + \ldots + a_k x^k \quad (1)$$

Given n centroids on a trajectory, the k+1 unknowns [a_0, a_1, ..., a_k] can be resolved by n equations through minimizing the squared sum of the deviations of the data from the model. The n equations can be represented as:

$$\begin{bmatrix} 1 & x_1 & \ldots & x_1^k \\ 1 & x_2 & \ldots & x_2^k \\ \ldots & \ldots & \ldots & \ldots \\ 1 & x_n & \ldots & x_n^k \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ \ldots \\ a_k \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \ldots \\ y_n \end{bmatrix} \quad (2)$$

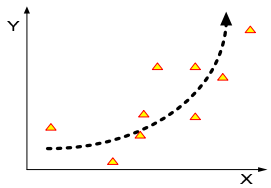

Figure 2. An example of polynomial curve fitting

The fitted curve represents a rough shape of the moving trajectory. It can be described by only a few polynomial coefficients. The first derivative of a polynomial curve is a tangent vector, which represents the velocities of that vehicle at different time. Figure 2 demonstrates the fitted curve of a group of centroids by a 4^{th} degree polynomial. The small triangles around the curve represent the original centroids.

4. Semantic event modeling

With different event types, different properties of semantic objects can be extracted to build the models for specific event types. In this study, a spatio-temporal model is built for traffic accidents.

Under some circumstances, a car accident may involve only one vehicle. Examples are sudden stops, crashes onto side walls in the tunnel or cashes into crowd in car races. If a vehicle crashes into another vehicle or several vehicles bump into each other, the accident will involve more than one vehicle. In all cases, the focus shall be the sudden change of behavior 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 checking point from the current velocity. A motion vector is a vector with its starting point being the centroid of some vehicle at the previous sampling point and the ending point being the centroid of the same vehicle at the current sampling point. As illustrated in the figure below, the change of motion vector is denoted as the angle between the current motion vector and the previous motion vector. $\overrightarrow{M_1}$ and $\overrightarrow{M_2}$ are two consecutive motion vectors. θ is the difference angle between them. Since we only record the absolute angle difference, there is no need to normalize these vectors along the axis.

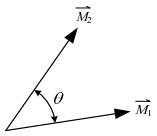

Figure 3. The change of motion vector

Another factor that needs to be taken into consideration is the distances among vehicles as it is a good indication for multi-vehicle accidents. For each vehicle, we record its minimum distance from its nearest vehicle – *mdist* at each sampling point.

As mentioned earlier, some heuristics need to be established in order to process the initial queries. This heuristic model is built based on the observation that the sudden change of velocity and driving direction may indicate an accident. Further, the closer the vehicle is to the other vehicles, the higher the chance of an accident. At the i^{th} sampling point, the property vector of a TS is α_i = [1/$mdist_i$, $vdiff_i$, θ_i]. A series of such vectors $\alpha = [\alpha_1,...,\alpha_n]$ represent the entire TS in a VS. It is worth mentioning that this event model may also be adjusted to detect U-turns, speeding and any other event that involves the abnormal behavior of a vehicle.

5. Semantic event retrieval

5.1. Data collecting and problem definition

Both VS and TS are time series data in that their values change over time. The analysis of time series data shall not only focus on each individual data point separately but shall also look into the continuity within such kind of data. In time series model of neural network, there is a commonly used method called sliding window, which slides over the whole set of time series data to extract consecutive yet overlapped data sequences i.e. windows. This method is also adopted in this framework. Figure 4 shows an example of sliding window for time series data. In this example, a 6-tuple sequence is extracted from time series data by sliding a window of size 6 one step a time along the time axis *t*.

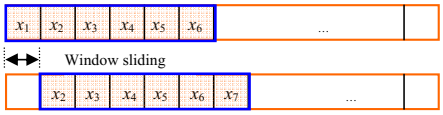

Figure 4. An example of sliding window

In the proposed framework, we use sliding window to extract VSs. This raised a question – what would be an appropriate window size? The advantage of video data over other typical time series data is that it can be visualized, thus one can have a concrete idea of how the data is organized and flows over time. Therefore, in our framework, the size of the window can be simply decided by the typical length of an event as it can be acquired by counting the number of frames covering this event. Take car crash as an example, the typical length in terms of number of frames for this kind of events is very short i.e. about 15 frames. Given a sampling rate of 5 frames per point, 3 sampling points are needed to depict a car crash event. Thus, the window size is 3 in our case.

As mentioned in Section 1, each extracted VS contains one or more TSs. With RF, the VS label is known. In a traffic accident query, if a returned VS is labeled "relevant" i.e. the user confirms it contains an accident scene, then at least one vehicle (TS) demonstrates abnormal behavior in that VS. That is, there exists at least one contained TS whose label is "relevant". On the other hand, if the VS label is "irrelevant", the labels of all the contained TSs are "irrelevant". With L_i representing the label of the i^{th} bag (VS) and l_{ij} representing the label of the j^{th} instance (TS) in the i^{th} bag, the scenario can be formally defined below:

$$L_i = 1 \quad iff \quad \exists_{j=1}^{m} l_{ij} = 1 \tag{3}$$

$$L_i = 0 \quad iff \quad \forall_{j=1}^{m} l_{ij} = 0 \tag{4}$$

5.2. Learning and retrieval mechanism

The goal of MIL in our case is to find out the labels of all TSs based on VS labels, according to which the most "relevant" VSs are returned to the user in the retrieval phase. With One-class classification, we consider all "relevant" TSs are alike in similar ways while "irrelevant" TSs deviate from the query target in their own ways and do not necessarily belong to the same class. Therefore, "irrelevant" TSs are simply treated as outliers in One-class classification. One-class SVM tries to assess whether a test point is likely to belong to the distribution underlying the training data set, which is composed of "relevant" samples only.

One-Class SVM has so far been studied in the context of SVMs [18]. The objective is to create a binary-valued function that is positive in those regions of input space where the data predominantly lies and negative elsewhere. The idea is to model the dense region as a "ball". In our MIL problem, "relevant" instances are inside the "ball" and "irrelevant" instances are outside. If the origin of the "ball" is $\vec{o}$ and the radius is *r*, an instance $\vec{x_i}$ is inside the "ball" *iff* $\left\| \vec{x_i} - \vec{o} \right\| \le r$. This is shown in Figure 5 with triangles inside the circle being the "relevant" instances.

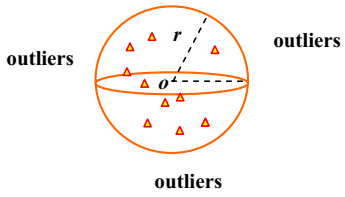

Figure 5. One-class classification

This "ball" is actually a hyper-sphere. The goal is to keep this hyper-sphere as "pure" as possible and include most of the "relevant" objects. Since this involves a non-linear distribution in the original space, the strategy of Schölkopf's One-Class SVM [18] is first to do a mapping θ to transform the data into a feature space F corresponding to the kernel K:

$$\theta(u)\cdot\theta(v) \equiv K(u,v) \tag{5}$$

where u and v are two data points. In this study, we choose to use Radial Basis Function (RBF) Machine below.

$$K(u,v) = \exp\left(\|u-v\|/2\sigma\right) \tag{6}$$

Mathematically, One-Class SVM solves the following quadratic problem:

$$\min_{w,\xi,\rho} \frac{1}{2}\|w\| - \rho + \frac{1}{\delta n}\sum_{i=1}^{n}\xi_i \tag{7}$$

subject to

$$(w\cdot\theta(x_i)) \geq \rho - \xi_i, \quad \xi_i \geq 0 \text{ and } i = 1,...,n \tag{8}$$

where ξ_i is the slack variable, and $\delta \in (0,1)$ is a parameter that controls the trade off between maximizing the distance from the origin and containing most of the data in the region created by the hyper-sphere. It actually corresponds to the ratio of "outliers" in the training dataset. When it is applied to the MIL problem, Equation (8) is also subject to Equations (3) and (4). If w and ρ are a solution to this problem, then the decision function is $f(x) = sign(w\cdot\theta(x) - \rho)$ and it will be 1 for most examples x_i contained in the training set.

5.3. Interactive event learning and retrieval process

In the initial query, the user specifies an event of interest as the query target. The ultimate goal is to retrieve those video sequences that contain similar semantic events. At this point, no relevance feedback information is provided by the user. Therefore, no training sample set is available. In order to provide an initial set of video sequences for the user to provide relevance feedback, for each video sequence in the database, we calculate its relevance (or similarity score) to the target query video event according to some event-specific search heuristics.

Suppose the user wants to query accidents, then the feature vector of a TS at sampling point i -- $\alpha_i = [1/mdist_i, vdiff_i, \theta_i]$ from Section 4 is used. In the initial retrieval, the relevance score of a VS is represented by the highest score of its contained TSs i.e. $S_v = max(S_{T1}, S_{T2}, ..., S_{Tn})$. S_v is the score of a VS and S_{Ti} is the score of TS_i contained in that VS. The score of a TS is the highest score of its sampling points i.e. $S_{Ti} = max(S_{\alpha 1}, S_{\alpha 2}, ..., S_{\alpha n})$, where $S_{\alpha i}$ is the score of a sampling point in TS_i. $S_{\alpha i}$ is calculated as the square sum of all the three features in the feature vector $\alpha_i = [1/mdist_i, vdiff_i, \theta_i]$. It is assumed that a big velocity change, a sudden change of driving direction, and a short distance between two vehicles are indications of possible accidents. The retrieval results are returned in the descending order of the VSs' relevance scores.

After the initial query, a certain amount of results are presented to the user. In our experiments, the top 20 VSs are returned for the user's feedback. The user identifies a returned VS as "relevant" if it is of his/her interest; otherwise the user labels it "irrelevant". With this information at hand, we set up the training set by collecting the highest scored TSs in the "relevant" VSs. These training samples are then fed into the One-class SVM, which further learns the user's interest and refines the retrieval result in the following iterations. Note that the One-class SVM learns from the entire trajectory sequence (TS) within the window (VS) i.e. $\alpha = [\alpha_1,...,\alpha_n]$, but not only the highest scored sampling point α_i in the TS. This is different from the way of calculating the similarity score in the initial query. By analyzing the entire trajectory sequence, the continuity of the data is well kept in the learning process of One-class SVM.

In this way, most of the "relevant" TSs can be identified. However, it is not a rare case that an accident can involve two or more vehicles. Therefore, some "relevant" VSs may contain more than one "relevant" TSs. Suppose the number of "relevant" VSs is h and the number of all TSs in the training set is H. Then the ratio of "outlier" TSs in the training set is set to:

$$\delta = 1 - (\frac{h}{H} + z) \tag{9}$$

z is a small number used to adjust the δ in order to alleviate the problem mentioned in Section 5.2. Our experimental results show that z= 0.05 works well. It is also shown in our experiment that, with this technique, the retrieval results are improved through iterations.

6. Experiments

6.1. System overview

Figure 6 shows the overall flow of the whole system proposed in this paper. The raw video is analyzed by segmenting and tracking semantic objects (vehicles) in it. After tracking, the object trajectories are modeled with the curve fitting technique. In this 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 for accidents, the system performs an initial query based on some heuristics as discussed in Section 5.3, and returns the initial retrieval results to the user. The user responds to each returned VS by giving his/her feedbacks. The learning mechanism in the system will then learn from these feedbacks and refine the retrieval results in the next iteration. The whole process goes through several iterations until a satisfactory result is obtained.

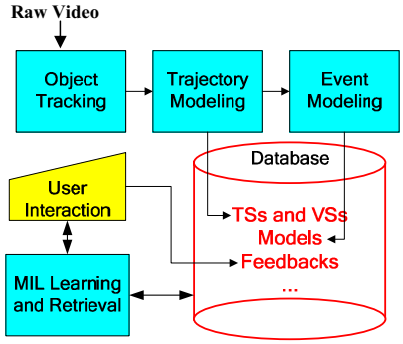

Figure 6. The system overview

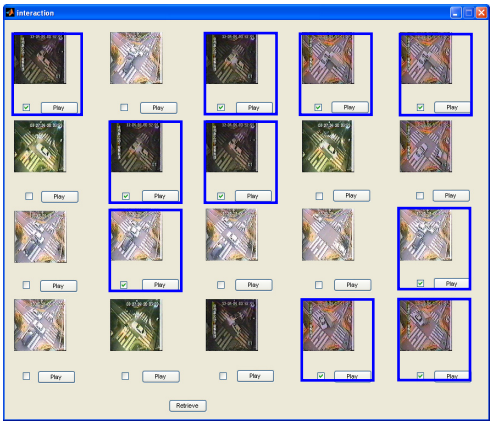

Figure 7. The user interaction interface

Figure 7 shows the interface for the user to provide feedback information. The top 20 VSs are returned to the user at each iteration. The user can play the retrieved VSs. If the user thinks a VS contains an accident scene, that VS will be selected. This is equal to labeling the video sequence (VS) "relevant". As shown in the interface, 10 VSs (in blue rectangles) are labeled "relevant" given a car accident query.

6.2. System performance

The proposed framework is tested on two video clips. The first one is taken in a tunnel and contains 2504 frames in total. The second one is taken by a real-life traffic surveillance video at a road intersection in Taiwan and contains 592 frames. The sampling rate is 5 frames/checkpoints and the window size is 3. After sampling and window sliding, there are altogether 109 TSs (15 frames each) extracted from the first clip and 168 TSs from the second clip. The reason that we observe more TSs in the second clip is that more vehicles are present in this video.

The proposed framework is compared with 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. That is to say, the initial weights of the three features are all 1s and the square sum of the features is computed as the relevance score. With the user's relevance feedback, the feature vectors of all relevant trajectory sequences are gathered. The inverse of the standard deviation of each feature is computed and used as the updated weight for this feature in the next round. In our experiment, we found that some large weights can introduce bias in computing relevance scores and hence affect the retrieval accuracy. Therefore, it is necessary to normalize these weights. We first tried to linearly normalize these weights to the range of [0 1]. However, the problem with this method is that a weight that equals zero will always eliminate the corresponding feature. We then tried another method i.e. the percentage of each weight among the total weight is used as the normalized weight. In our experiment, it is found that the latter outperforms both the linear normalization and no normalization at all.

Five rounds of relevance feedback are performed - Initial (no feedback), First, Second, Third, and Fourth. In each iteration, the top 20 video sequences are returned to the user. In a large-scale information mining and retrieval system, since there is no prior knowledge as to the total number of "correct" results given a user's query, it is not applicable to use traditional data mining measurements such as precision and recall. Instead, we use the "accuracy" measure for such a purpose, which is defined as the percentage of all the "relevant" VSs within the top

n (e.g. *n*=20) returned VSs. Figure 8 shows the retrieval accuracies within the top 20 video sequences for the first video clip after Initial, First, Third, and Fourth round of iterations.

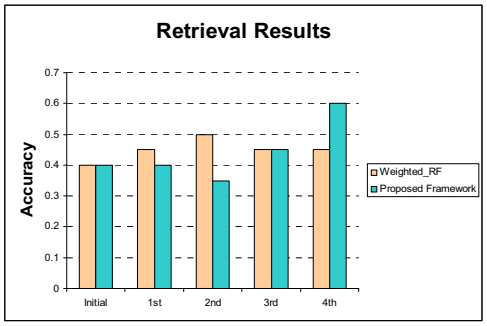

Figure 8. The retrieval accuracies for the 1st clip

It can be gleaned from Figure 8 that the initial accuracies of the two methods are the same since the same retrieval algorithm is used in the initial round. After that, the proposed framework performs much better in that the accuracy values increase steadily from 40% to 60%. Although the "Weighted_RF" method performs slightly better at the 2nd and the 3rd iterations, its overall accuracy gain over all 4 iterations is only 10% i.e. from 40% to 50% in the third iteration. After that, its accuracy keeps bouncing around between 35% and 50% and does not show any further performance gain.

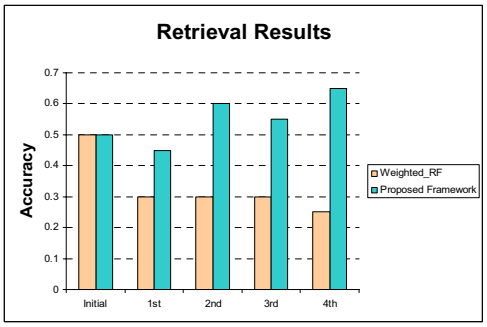

Figure 9. The retrieval accuracies for the 2nd clip

Most of the accidents in the first clip only involve a single vehicle. The video is taken in a tunnel and features some accident scenes where speeding vehicles lost control and hit on the sidewalls of the tunnel. In the second clip, all the accidents occurred at a road intersection and often involve two or more vehicles. The retrieval results are compared with that of the weighted RF in Figure 9. Although the accuracy gains with the proposed framework is 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.

Ideally, all the video clips in a transportation surveillance video database shall be mined and retrieved as a whole. However, in order to do that, it requires that we normalize all the video clips taken at different locations with different camera parameters. Those parameters, such as camera angle and camera position, are necessary for normalization. Unfortunately, these metadata are missing in our experimental videos. Therefore, at the current stage, the retrieval is performed independently for each group of videos taken by the same camera at the same location. Our next immediate step is to collect our own transportation surveillance videos and normalize them before storing them into the database.

7. Conclusions and future work

In this paper, an interactive Multiple Instance Learning (MIL) framework for semantic video mining and retrieval is proposed. The framework is tested on Transportation Surveillance videos to find out user-interested semantic events such as car accidents. 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 spatio-temporal event models are then constructed. In the learning and retrieval phase, for the top returned Video Sequences at each iteration, the user provides feedback to the relevance of each such sequence. The user only needs to give feedback to the whole Video Sequence and the learning algorithm will analyze the contained Trajectory Sequences in order to find out the spatio-temporal patterns of user-interested moving vehicle behaviors. Therefore, we map this to a MIL problem. One-class SVM is used as the learning algorithm that refines the retrieval results with the user's feedbacks. This framework successfully incorporates the Relevance Feedback technique and MIL in mining spatio-temporal video data, which is a well-studied topic in Content Based Image Retrieval but needs significant extensions when applied to video data retrieval. The framework shows its effectiveness as demonstrated by our experimental results on real-life transportation surveillance videos.

In our future work, more generic event models will be constructed and tested with the proposed framework. Currently, the framework only supports the user's query by specified event types. We will extend this to include query by example, query by sketches, and allow a customized combination of different query types.

8. Acknowledgement

The work of Chengcui Zhang was supported in part by SBE-0245090 and the UAB ADVANCE program of the Office for the Advancement of Women in Science and Engineering and IBM UIMA Faculty Award.

9. References

[1] J. Fernyhough, A. G. Cohn, and D. C. Hogg, "Constructing Qualitative Event Models Automatically from Video Input", *Image and Vision Computing,* Vol. 18, No. 2, pp. 81-103, 2000.

[2] Y. Rui, T.S. Huang, and S. Mehrotra, "Content-based Image Retrieval with Relevance Feedback in MARS", in *Proceedings of the International Conf. on Image Processing*, 1997, pp. 815-818.

[3] C. Zhang, X. Chen, M. Chen, S.-C. Chen, and M.-L. Shyu, "A Multiple Instance Learning Approach for Content-Based Image Retrieval Using One-class Support Vector Machine", in *Proc. of IEEE International Conference on Multimedia & Expo (ICME)*, pp. 1142-1145, July 6-8, 2005, Amsterdam, The Netherlands.

[4] 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 Transactions on Image Processing*, Vol. 12, No. 8, pp. 924-937, 2003.

[5] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez, "Solving the Multiple-Instance Problem with Axis-Parallel Rectangles", *Artificial Intelligence Journal*, vol. 89, pp. 31-71, 1997.

[6] O. Maron and T. Lozano-Perez, "A Framework for Multiple Instance Learning", *Advances in NaturalInformation Processing System 10*, Cambridge, MA, MIT Press, 1998.

[7] Q. Zhang and S. A. Goldman, "EM-DD: An Improved Multiple-Instance Learning Technique", *Advances in Neural Information Processing Systems (NIPS)*, 2002.

[8] C. Buckley, A. Singhal, and M. Miltra, "New Retrieval Approaches Using SMART:TREC4", *Text Retrieval Conference*, sponsored by National Institute of Standard and Technology and Advanced Research Projects Agency, pp. 25-24, 1995.

[9] M. Minsky, "The Society of Mind," *Simon and Schuster,* 1986.

[10] J. Wang and J.-D. Zucker, "Solving the Multiple Instance Learning Problem: A Lazy Learning Approach", in *Proceedings of the 17th International Conference on Machine Learning*, pp. 1119-1125, 2000.

[11] 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 *Proceedings of the 14th Biennial Conference of the Canadian Society for Computational Studies of Intelligence*, AI 2001, pp. 204-214, 2001.

[12] S. Tang and H. Gao, "Traffic-Incident Detection Algorithm Based on Nonparametric Regression", *IEEE Transactions on Intelligent Transportation Systems,* Vol. 6, No. 1, March 2005.

[13] C. Zhang, X. Chen, and W.-B. Chen, "A PCA-based Vehicle Classification Framework", in *Proc. of IEEE International Workshop on Multimedia Databases and Data Management, in conjunction with IEEE International Conference on Data Engineering (ICDE 2006)*, April 8, 2006, Atlanta, Georgia, USA.

[14] S Atev, H. Arumugam, O. Masoud, R.Janardan, and N.P. Papanikolopoulos, "A Vision-Based Approach to Collision Prediction at Traffic Intersections", *IEEE Transactions on Intelligent Transportation Systems,* Vol. 6, No. 4, December 2005.

[15] J. Ramon and L. De Raedt, "Multi-Instance Neural Networks", in *Proceedings of the ICML 2000 Workshop on Attribute-value and Relational Learning*, 2000.

[16] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support Vector Machines for Multiple-Instance Learning", *Advances in Neural Information Processing Systems 15*, pp. 561-568. Cambridge, MA:MIT Press, 2003.

[17] H. Veeraraghavan, O Masoud, and N.P. Papanikolopoulos, "Computer Vision Algorithms for Intersection Monitoring", *IEEE Transactions on Intelligent Transportation Systems,* Vol. 4, No. 2, June 2003.

[18] Schölkopf, B., Platt, J.C. et al., "Estimating the Support of a High-dimensional Distribution", *Microsoft Research Corporation Technical Report MSR-TR-99-87*, 1999.

[19] M. Nakazato, C. Dagli, and T. S. Huang, "Evaluating Group-based Relevance Feedback for Content-based Image Retrieval", in *Proceedings IEEE International Conference on Image Processing (ICIP'03)*, Spain, 2003, Vol. 2, pp. 599-602.

[20] 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 Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, pp. 154-167, September 2003.

[21] Y. Rui, T.S. Huang, M. Ortega, and S. Mehrotra, "Relevance Feedback: A Power Tool in Interactive Content-based Image Retrieval", *IEEE Transactions on Circuits and Systems for Video Technology,* Special Issue on Segmentation, Description, and Retrieval of Video Content, Vol.18, No. 5, pp.64 4-655, 1998.

[22] Y. Rui and T.S. Huang, "A Novel Relevance Feedback Technique in Image Retrieval", in *Proceedings of the 7th ACM International Conference on Multimedia (Part 2),* pp. 67-70, 1999.

[23] J.J. Rocchio, *Relevance Feedback in Information Retrieval. The Smart System Experiments in Automatic Document Processing*, Englewood Cliffs, NJ: Prentice Hall Inc. 1971, pp. 313-323.

[24] G. Salton and M.J. McGill, *Introduction to Modern Information Retrieval,* McGraw-Hill Book Company, 1983.

[25] S. Aksoy and R.M. Haralick, "A Weighted Distance Approach to Relevance Feedback", in *Proceedings of the International Conference on Pattern Recognition,* pp. 812-815, 2000.

[26] Y. Ishikawa, R. Subramanya, and C. Faloutsos, "Mindreader: Query Databases Through Multiple Examples", in *Proceedings of the 24th International Conference on Very Large Databases,* pp. 218-227, 1998.

[27] C.-H. Chang and C.-C. Hsu, "Enabling Concept-based Relevance Feedback for Information Retrieval on the WWW", *IEEE Transactions on Knowledge and Data Englineering,* Vol. 11, No. 4, July/August, pp. 595-609, 1999.

[28] F.M. Porikli and X. Li., "Traffic Congestion Estimation Using HMM Models without Vehicle Tracking", *IEEE Intelligent Vehicles Symposium (IV)*, pp. 188-193, June 2004.

[29] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, "Automatic Symbolic Traffic Scene Analysis Using Belief Networks", in *Proceedings of National Conference on Artificial Intelligence,* 1994.

[30] H. Buxton and S. Gong, "Visual Surveillance in a Dynamic and Uncertain World", *Artificial Intelligence*, 78:431-459, 1995.

[31] D. Xie, W. Hu, T. Tan, and J. Peng, "Semantic-based Traffic Video Retrieval Using Activity Pattern Analysis", in *IEEE International Conference on Image Processing (ICIP)*, 2004.

[32] S. Kamijo, Y. Matsushita, and I. Katsushi, "Traffic Monitoring and Accident Detection at Intersections", *IEEE Transactions on Intelligent Transportation Systems,* Vol. 1, No. 2, pp. 108-118, June 2000.

[33] W. Hu, X. Xian, D. Xie, and T. Tan, "Traffic Accident Prediction Using 3-D Model-Based Vehicle Tracking", *IEEE Transactions on Vehicular Technology,* Vol. 53. No. 3, May 2004.

[34] S. Dance and T. Caelli, "On the Symbolic Interpretation of Traffic Scenes", in *ACCV93 Proceedings of the Asian Conference on Computer Vision*, pp. 798-801, Osaka Japan, November 1993.

[35] Y.-K. Jung, K.-W. Lee, and Y.-S. Ho, "Conent-Based Event Retrieval Using Semantic Scene Interpretation for Automated Traffic Surveillance", *IEEE Transactions on Intelligent Transportation Systems*, Vol 2, No. 3, September 2001.