Learning-Based Spatio-Temporal Vehicle Tracking and Indexing for Transportation Multimedia Database Systems

Shu-Ching Chen, *Member, IEEE*, Mei-Ling Shyu, *Member, IEEE*, Srinivas Peeta, and Chengcui Zhang, *Student Member, IEEE*

***Abstract*—One key technology of intelligent transportation systems is the use of advanced sensor systems for on-line surveillance to gather detailed information on traffic conditions. Traffic video analysis can provide a wide range of useful information to traffic planners. In this context, the object-level indexing of video data can enable vehicle classification, traffic flow analysis, incident detection and analysis at intersections, vehicle tracking for traffic operations, and update of design warrants. In this paper, a learning-based automatic framework is proposed to support the multimedia data indexing and querying of spatio-temporal relationships of vehicle objects in a traffic video sequence. The spatio-temporal relationships of vehicle objects are captured via the proposed unsupervised image/video segmentation method and object tracking algorithm, and modeled using a multimedia augmented transition network model and multimedia input strings. An efficient and effective background learning and subtraction technique is employed to eliminate the complex background details in the traffic video frames. It substantially enhances the efficiency of the segmentation process and the accuracy of the segmentation results to enable more accurate video indexing and annotation. The paper uses four real-life traffic video sequences from several road intersections under different weather conditions in the study experiments. The results show that the proposed framework is effective in automating data collection and access for complex traffic situations.**

***Index Terms*—Advanced traffic management systems (ATMS), advanced traveler information systems (ATIS), background learning and background subtraction, intelligent transportation systems (ITS), multimedia database indexing, segmentation, vehicle tracking, video analysis.**

I. Introduction

In RECENT years, intelligent transportation systems (ITS), which integrate advances in telecommunications, information systems, automation, and electronics to enhance the efficiency of existing road networks, have been identified as the new paradigm to address the growing mobility problems, and to alleviate congestion and augment the quality of vehicular flow. While a whole new generation of methodological and algorithmic constructs are being developed to exploit the powerful capabilities afforded by the ITS technologies, concurrent efforts needed to enable practical implementation are lacking in some crucial aspects. One such aspect with sparse focus is the ability to collect, analyze, and store large-scale multimedia traffic flow data for real-time usage. It implies capabilities to: 1) store and catalogue data in an organized manner for easy acces; 2) reconstruct traffic situations through off-line analysis for addressing traffic safety and control; and 3) automate the process of data indexing and retrieval by obviating the need for human intervention and supervision. While each of these capabilities significantly enhances operational feasibility, the last capability has critical implications for real-time implementation in terms of substantially reducing computational time for the associated control procedures. One key application domain that addresses these three capabilities is the ability to track video sequences both in time and space for easy and unsupervised access.

In this paper, our emphasis is on automatic traffic video indexing for capturing the spatio-temporal relationships of vehicle objects so that they can be catalogued for efficient access using a multimedia database system. Issues associated with extracting traffic movement and accident information from real-time video sequences are discussed in [1]–[4]. Two common themes exist in these studies. First, the moving objects (vehicles) are extracted from the video sequences. Next, the behavior of these objects is tracked for immediate decision-making purposes. However, these efforts do not have capabilities to: 1) index the data for on-line analysis, storage or off-line pattern querying, and 2) automate data processing. To enable the indexing of information at the object-level for video data, an intelligent framework should have the ability to segment the area of video frames into different regions, where each of them or a group of them represents a semantic object (such as a vehicle object in traffic video) that is meaningful to users. While most previous studies are based on some low-level global features such as color histograms and texture features, the unsupervised video segmentation method used in our framework focuses on obtaining object-level segmentation, objects in each frame, and their traces across frames. Thus, the spatio-temporal relationships of objects can be further elicited by applying object tracking techniques. Then, the mul-

Manuscript received July 20, 2001; revised September 22, 2003. The work of S.-C. Chen was supported in part by the National Science Foundation (NSF) under Grants EIA-0-220-562 and NSF HRD-0-317-692. The work of M.-L. Shyu was supported in part by the National Science Foundation (NSF) under Grant ITR-0-325-260 (Medium). The Guest Editor for this paper was P. A. Ioannou.

S.-C. Chen is with the Distributed Multimedia Information System Laboratory, School of Computer Science, Florida International University, Miami, FL 33199 USA (e-mail: chens@cs.fiu.edu).

M.-L. Shyu is with the Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL 33124 USA (e-mail: shyu@miami.edu).

S. Peeta is with the School of Civil Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: peeta@ecn.purdue.edu).

C. Zhang is with the Distributed Multimedia Information System Laboratory, School of Computer Science, Florida International University, Miami, FL 33199 USA (e-mail: czhang02@cs.fiu.edu).

Digital Object Identifier 10.1109/TITS.2003.821290

timedia augmented transition network (MATN) model and multimedia input strings [5] are used to capture and model the temporal and spatial relations of vehicle objects. In previous work, we have addressed unsupervised image segmentation and object tracking techniques, and applied these techniques to some application domains such as traffic monitoring [6]–[8]. In this paper, a learning-based object tracking and indexing framework is proposed to improve the vehicle identification process for object tracking and indexing for transportation multimedia database systems.

In this study, an effective background learning algorithm is proposed in our learning-based object tracking and indexing framework. By incorporating the background learning algorithm with the unsupervised segmentation method, the initial inaccurate background information can be refined and adjusted in a self-adaptive way throughout the segmentation. That is, the background learning process and the segmentation process will benefit each other symbiotically in an iterative way as the processes go further. Experiments are conducted using four real-life traffic video sequences from several road intersections under different weather conditions. The results indicate that almost all moving vehicle objects can be successfully identified at a very early stage of the processing, thereby ensuring that accurate spatio-temporal information of objects can be obtained through object tracking.

After the vehicle objects are successfully identified, the MATN model and multimedia input strings are proposed to represent and model their spatio-temporal relations [5]. The capability to represent the spatio-temporal relations of the objects is critical from several perspectives. First, the ability to store multimedia data efficiently provides significant insights on traffic data collection and monitoring vis-à-vis exploiting recent advances in sensor systems. This is especially important in the context of real-time data access for large-scale traffic system operation and control. Second, an ability to obtain and store spatio-temporal relationships provides powerful capabilities to analyze and/or address problems characterized by time-dependency. For example, such a capability can significantly aid the off-line analysis of traffic accidents to isolate their causes and identify potential design issues or operational conflicts. Third, it can significantly reduce manual effort and intervention by automating the data collection and processing. For example, it can aid in revising traffic warrants without the need for supervised analysis, which is a significant improvement over current labor-intensive approaches involving the painstaking manual examination of the video data collected. Fourth, the ability to store data from different media on the same traffic situation in an automatic and efficient manner simplifies data access and fusion. This has significant real-time implications as data storage and processing can constitute a substantial part of the real-time implementation of traffic control strategies.

This paper is organized as follows. Section II gives the literature review. Section III introduces the proposed learning-based object tracking and indexing framework. The experiments, results, and the analysis of the proposed multimedia traffic video data indexing framework are discussed in Section IV. Four real-life traffic video sequences are used for the experiments. Conclusions and future work are presented in Section V.

II. Literature Review

Over the past decade, ITS have been identified as the new paradigm to address the growing mobility problems. With the exponential growth in computational capability and information technology, traffic monitoring and large-scale data collection have been enabled through the use of new sensor technologies. One ITS technology, advanced traffic management systems (ATMS) [9]–[11], aims at using advanced sensor systems for on-line surveillance and detailed information gathering on traffic conditions. Another, advanced traveler information systems (ATIS), provides network-wide routing information to road users. In addition, advanced public transportation systems (APTS) target mass transportation systems to enable greater operational efficiency and travel convenience. Another example of ITS technologies is the use of advanced sensor systems for on-line surveillance, such as traffic video analysis.

Image processing and object tracking techniques have previously been applied to traffic video analysis to address queue detection, vehicle classification, and vehicle counting. In particular, vehicle classification and vehicle tracking have been extensively investigated in [4], [12]–[15]. In [13], optical-flow analysis was employed, while spatio-temporal Markov random field (MRF) for vehicle tracking with the occlusion effect among vehicles was proposed in [4]. Three methods for moving object detection within the video surveillance and monitoring (VSAM) testbed were developed in [14]. One of them uses temporal differencing to detect moving targets and train the template matching algorithm. These targets are then tracked using template matching. Another approach to moving object detection uses a moving airborne platform [15]. Though several approaches have been proposed for vehicle identification and tracking, to the best of our knowledge, none of them connect to databases. Some have limited capabilities to index and store the collected data. Therefore, they cannot provide organized, unsupervised, easily accessible, and easy-to-use multimedia information. Hence, there is a critical need to index the data efficiently in traffic multimedia databases for transportation operations.

For traffic intersection monitoring, digital cameras are fixed and installed above the area of the intersection. A classic technique to resolve the moving objects (vehicles) is background subtraction [16]. This involves the creation of a background model that is subtracted from the input images to create a difference image. Ideally, the difference image contains only the moving objects (vehicles). Various approaches to background subtraction and modeling techniques have been discussed in the literature [2], [12], [17], [18]. They range from modeling the intensity variations of a pixel via a mixture of Gaussian distributions, to simple differencing of successive images. [19] provides some simple guidelines for the evaluation of various background modeling techniques. There are two key problems in this context: 1) a complex learning model is highly time-consuming, and 2) a simple differencing technique cannot guarantee good segmentation performance.

III. Learning-Based Object Tracking and Indexing for Traffic Video Sequences

Traffic video analysis at intersections can provide a rich array of useful information such as vehicle identification, queue detection, vehicle classification, traffic volume, and incident detection. To the best of our knowledge, traffic operations currently either do not connect to databases or have limited capabilities to index and store the collected data (such as traffic videos) in their databases. Therefore, they cannot provide organized, unsupervised, conveniently accessible and easy-to-use multimedia information to the end users. The proposed learning-based object tracking and indexing framework includes background learning and subtraction, vehicle object identification and tracking, the MATN model, and multimedia input strings. The additional level of sophistication enabled by the proposed framework in terms of spatio-temporal tracking generates a capability for automation. This capability alone can significantly influence and enhance current data processing and implementation strategies for several problems vis-à-vis traffic operations.

In the proposed framework, an unsupervised video segmentation method called the simultaneous partition and class parameter estimation (SPCPE) algorithm is applied to identify the vehicle objects in the video sequence [20], [21]. In addition, the technique of background subtraction is introduced to enhance the basic SPCPE algorithm to help get better segmentation results, so that the more accurate spatio-temporal relationships of objects can be obtained. After the spatio-temporal relationships of the vehicle objects are captured, the MATNs and multimedia input strings are used to represent and model their temporal and relative spatial relations. In the following subsections, we will first discuss the motivation for the proposed framework. Then, an overview of the SPCPE algorithm and the object tracking techniques will be provided. This will be followed by an introduction to the background subtraction technique. Then, we will briefly describe how to use the MATNs and multimedia input strings to model traffic video frames. Two example video frames are used to demonstrate how video indexing is modeled through the MATNs and multimedia input strings.

A. Motivation

Image segmentation techniques have been used previously to extract the semantic objects from images or video frames, but in most cases the nonsemantic content (or background) in the images or video frames is very complex. For example, at a traffic intersection, there are nonsemantic objects such as the road pavement, trees, and pavement markings/signage in addition to the semantic objects (vehicles), which introduces complications for the segmentation methods. Therefore, an effective way to obtain background information can enable better segmentation results. This leads to the idea of background subtraction. Background subtraction is a technique to remove nonmoving components from a video sequence. The main assumption for its application is that the camera remains stationary. The basic principle is to create a reference frame of the stationary components in the image. Once created, the reference frame is subtracted from any subsequent images. The pixels resulting from new (moving) objects will generate a difference not equal to zero.

The traditional way to eliminate background details is to manually select video sequences containing no moving objects and then average them together. This is done through the construction of a reference background frame by accumulating and averaging images of the target area (e.g., a road intersection) for some time interval [4], [22]. However, the determination of the time interval is subjective, and is based on experience or estimation from experimental results. For the traditional method to work well, one key condition is that the video sequence should have approximately constant lighting conditions. Hence, it is not a robust technique as it is sensitive to intensity variations, as lighting conditions are not controlled [18]. That is, it can generate false positives by incorrectly detecting moving objects solely due to lighting changes. It can also generate false negatives by adding static objects to the scene that are not part of the reference background frame. In the proposed framework, instead of manually selecting the suitable frames to generate one reference background image at a time, an adaptive background learning method is used to achieve this goal. The idea is to first use the unsupervised segmentation method together with the object tracking technique to distinguish the static objects from the mobile objects. Then, these static objects are grouped with the already identified background area to form a new estimation of the background.

The basic workflow of the proposed framework is shown in Fig. 1. In the first step, a background learning method is applied on the first few video frames (for example, the first four frames) to obtain the initial reference background image. By applying the unsupervised segmentation method, the static and mobile objects are roughly determined, and then the static objects are grouped with the already identified background to form the initial reference background image. The second step involves the subtraction of the initial reference background image from the current frame to generate the difference image whose background is much cleaner compared to the original frame. Then, the unsupervised segmentation method is applied on the difference image to get the segmentation results. Using them, a new reference background image is generated in a self-adaptive way. The details are described in the following subsections. After the vehicle objects (such as cars and buses) are successfully identified, their relative spatio-temporal relationships are further indexed and modeled by the MATN model together with multimedia input strings. The proposed segmentation method can identify vehicle objects but cannot differentiate between them (into cars, buses, etc.). Therefore, *a priori* knowledge (size, length, etc.) of different vehicle classes should be provided to enable such classification. In addition, since the vehicle objects of interest are the moving ones, stopped vehicles will be considered as static objects and will not be identified as mobile objects until they start moving again. However, the object tracking technique ensures that such vehicles are seamlessly tracked though they "disappear" for some duration due to the background subtraction. This aspect is especially critical under congested or queued traffic conditions.

In a traffic video monitoring sequence, when a vehicle object moves out of the monitor area (intersection) or stops in the inter-

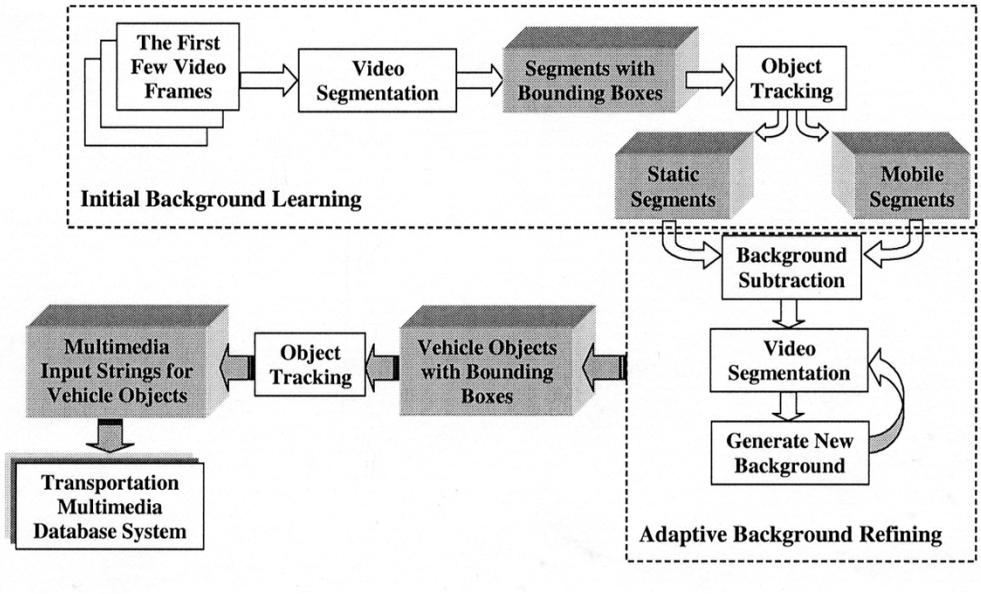

Fig. 1. Basic workflow of the proposed framework.

section area (including the approaches to the intersection), our framework may deem it as part of the background information. In the former case, tracking is not necessary as the vehicle is out of the monitoring area. Usually, in such a situation, the centroid of its bounding box will be very close to the boundary of the monitoring area. In the latter case, since the vehicle objects move into the intersection area before stopping, they are identified as moving vehicles before their stop due to the characteristics of our framework. In this situation, their centroids identified before they stop will be in the intersection area. For these vehicles, the tracking process is frozen until they start moving again and they are identified as "waiting" rather than "disappearing" objects. That is, the tracking process will follow the same procedure as before unless one or more new objects abruptly appear in the intersection area. Then, the matching and tracking of the previous "waiting" objects will be triggered to continue tracking the trails of these vehicles.

B. Unsupervised Video Segmentation Method (SPCPE)

The SPCPE algorithm is an unsupervised video segmentation method to partition video frames [20], [21]. A given class description determines a partition, and vice versa. Hence, the partition and the class parameter have to be estimated simultaneously. In practice, the class descriptions and their parameters are not readily available. An additional difficulty arises when images have to be partitioned automatically without the intervention of the user: we do not know *a priori* which pixels belong to which class. In the SPCPE algorithm, the partition and the class parameters are treated as random variables. The method for partitioning a video frame starts with an arbitrary partition and employs an iterative algorithm to estimate the partition and the class parameters jointly. Since the successive frames in a video do not differ much, the partitions of adjacent frames do not differ significantly. Each frame is partitioned by using the partition of the previous frame as an initial condition to speed up the convergence rate of the algorithm. A randomly generated initial partition is used for the first few frames during the initial background learning.

1) Class Parameter Estimation: The mathematical description of a class specifies the pixel values as functions of the spatial coordinates of the pixel. The parameters of each class can be computed directly by using a least squares technique. Suppose we have two classes. Let the partition variable be $c = \{c_1, c_2\}$ and the classes be parameterized by $\theta = \{\theta_1, \theta_2\}$. More precisely, an image is partitioned into two classes by dividing the image pixels into two subsets c_1 and c_2. The notation y_{ij} is used here to represent the value of a pixel at location (i, j). Also, suppose all the pixels y_{ij} (in the image data Y) belonging to class k $(k = 1, 2)$ are put into a vector $\boldsymbol{Y_k}$. Each row of the matrix Φ is given by $(1, i, j, ij)$ and a_k is the vector of parameters $(a_{k0}, \ldots, a_{k3})^T$

$$y_{ij} = a_{k0} + a_{k1}i + a_{k2}j + a_{k3}ij, \quad \forall (i,j)\, y_{ij} \in \boldsymbol{c_k} \tag{1}$$

$$\boldsymbol{Y_k} = \Phi a_k \tag{2}$$

$$\hat{a}_k = (\Phi^T \Phi)^{-1} \Phi^T \boldsymbol{Y_k}. \tag{3}$$

2) Class Partition Estimation: We estimate the best partition as that which maximizes the *a posteriori* probability (MAP) of the partition variable given the image data Y. The MAP estimates of $c = \{c_1, c_2\}$ and $\theta = \{\theta_1, \theta_2\}$ are given by

$$\begin{aligned}(\hat{c}, \hat{\theta}) &= \operatorname{Arg}\max_{(c,\theta)} P(c, \theta \,|\, Y) \\ &= \operatorname{Arg}\max_{(c,\theta)} P(Y \,|\, c, \theta) P(c, \theta).\end{aligned} \tag{4}$$

We assume that the pixel values and parameters are independent and that the parameters are uniformly distributed. We also assume that the error function of y_{ij} is represented by a Gaussian with mean 0 and variance 1. Let $J(c, \theta)$ be the functional to be minimized. With these assumptions the joint estimation can be simplified to the following form:

$$(\hat{c}, \hat{\theta}) = \operatorname{Arg}\min_{(c,\theta)} J(c_1, c_2, \theta_1, \theta_2) \tag{5}$$

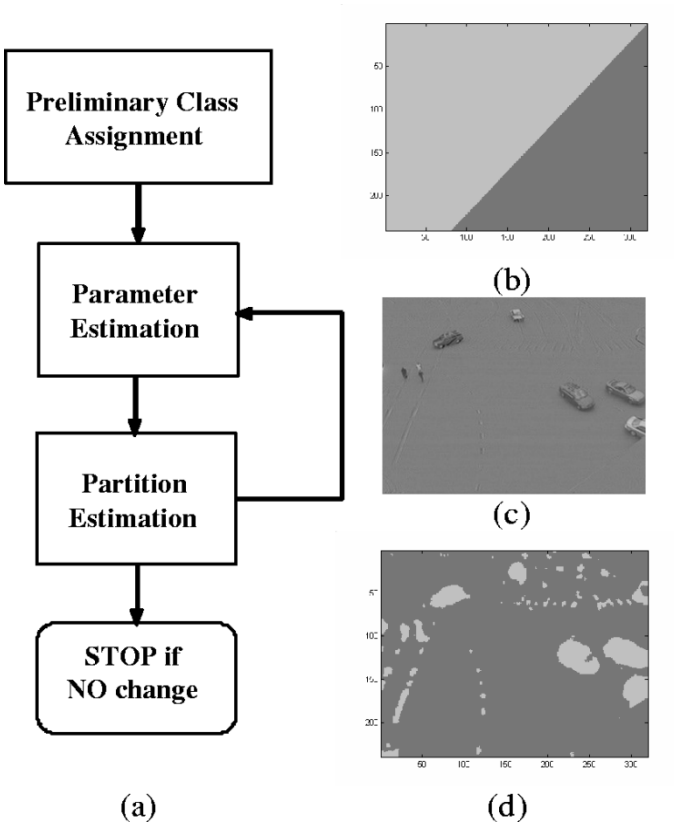

Fig. 2. (a) The flowchart of the SPCPE algorithm; (b) Initial random partition; (c) Original traffic video frame; (d) Object segmentation result.

$$J(c_1, c_2, \theta_1, \theta_2) = \sum_{y_{ij} \in c_1} -\ln p_1(y_{ij}; \theta_1) + \sum_{y_{ij} \in c_2} -\ln p_2(y_{ij}; \theta_2). \quad (6)$$

The minimization of J can be carried out alternately on c and θ in an iterative manner. Let $\hat{\theta}(c)$ represent the least squares estimates of the class parameters for a given partition c. The final expression for $J(c, \hat{\theta}(c))$ can be derived easily and is given by

$$J(c, \hat{\theta}(c)) = \operatorname{Arg} \min_{(c_1, c_2)} \left\{ \frac{N_1}{2} \ln \hat{\rho}_1 + \frac{N_2}{2} \ln \hat{\rho}_2 \right\} \quad (7)$$

where $\hat{\rho}_1$ and $\hat{\rho}_2$ are the estimated model error variances of the two classes and $\mathrm{N}_1, \mathrm{N}_2$ are the number of pixels in each class. As shown in Fig. 2(a), the algorithm starts with an arbitrary partition of the data in the first video frame and computes the corresponding class parameters. Using these class parameters and the data, a new partition is estimated. Both the partition and the class parameters are iteratively refined until there is no further change in them. We note here that the functional J is not convex. Hence its minimization may yield a local minimum, which guarantees the convergence of this iterative algorithm. Fig. 2(d) shows the segmentation result for the video frame in Fig. 2(c) given the initial partition in Fig. 2(b). Since the successive frames do not differ much due to the high temporal sampling rate, the partitions of the adjacent frames do not differ significantly. The key idea is to use the method successively on each frame of the video, incorporating the partition of the previous frame as the initial condition while partitioning the current frame. This can greatly reduce the computational cost substantially (up to 90%) because the number of iterations needed is much less than that of randomly initial partition.

C. Object Tracking

In order to index the vehicle objects, the proposed framework must have the ability to track the moving vehicle objects (segments) within successive video frames [6]. Since the tracking trail information can be obtained for each segment, it is possible to distinguish the static objects from the mobile objects in the frame, enabling the estimation of the background information. Furthermore, this tracking technique can be used to determine the trace tubes (trails) for vehicle objects, which enable the proposed framework to provide useful and accurate traffic information for ATIS and ATMS.

1) Identifying Static and Mobile Objects Using Object Tracking: After video segmentation, the segments (objects) with their bounding boxes and centroids are extracted from each frame. Intuitively, two segments that are spatially the closest in adjacent frames are connected. Euclidean distance is used to measure the distance between their centroids.

Definition 1: A bounding box B (of dimension 2) is defined by the two endpoints S and T of its major diagonal [16]

$$\mathrm{B} = (\mathrm{S}, \mathrm{T}), \text{where } \mathrm{S} = (s_1, s_2) \quad \text{and} \quad \mathrm{T} = (t_1, t_2) \text{ and } s_i \le t_i \text{ for } i = 1, 2.$$

Due to Definition 1, the area of B: $\mathrm{Area}_B = (t_1 - s_1) \times (t_2 - s_2)$.

Definition 2: The centroid ctd_B of a bounding box B corresponding to an object O is defined as follows:

$$\mathrm{ctd}_O = [\mathrm{ctd}_{O1}, \mathrm{ctd}_{O2}]$$

where

$$\mathrm{ctd}_{O1} = \left(\sum_{i=1}^{No} O_{xi} \right) \Big/ No; \qquad \mathrm{ctd}_{O2} = \left(\sum_{i=1}^{No} O_{yi} \right) \Big/ No; \quad (8)$$

where No is the number of pixels belonging to object O within bounding box B, O_{xi} represents the *x* coordinate of the ith pixel in object O, and O_{yi} represents the *y* coordinate of the ith pixel in object O.

Let ctd_M and ctd_N, respectively, be the centroids of segments M and N that exist in consecutive frames, and δ be a threshold. The Euclidean distance between them should not exceed the threshold δ if M and N represent the same object in consecutive frames

$$\mathrm{DIST}(\mathrm{ctd}_M - \mathrm{ctd}_N) = \sqrt{(\mathrm{ctd}_{M1} - \mathrm{ctd}_{N1})^2 + (\mathrm{ctd}_{M2} - \mathrm{ctd}_{N2})^2} \le \delta \quad (9)$$

In addition to the use of the Euclidean distance, some size restriction is applied to the process of object tracking. If two segments in successive frames represent the same object, the difference between their sizes should not be large. The details

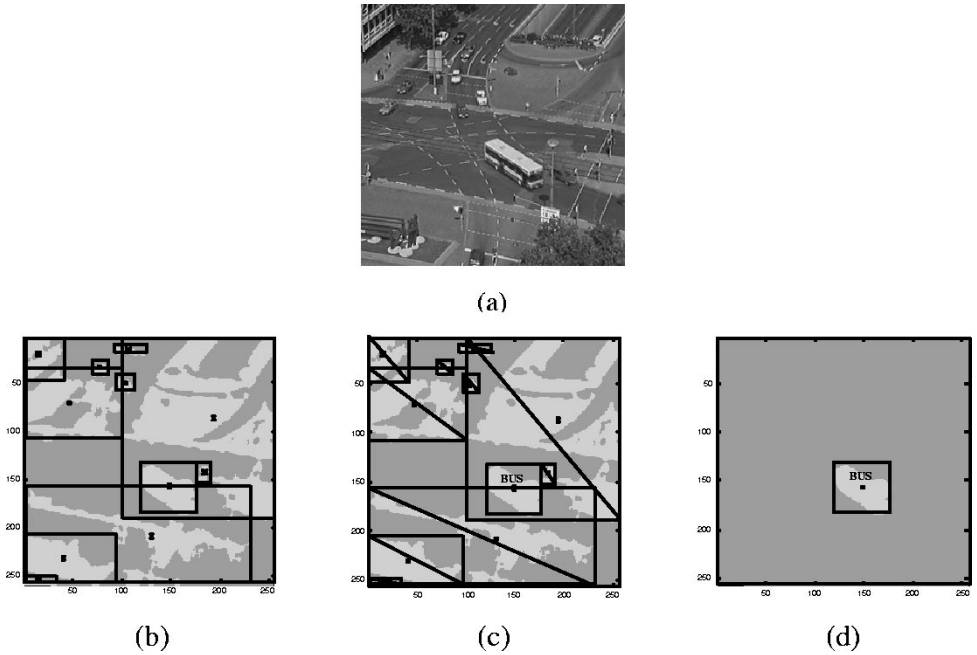

Fig. 3. (a) Original video frame 3, (b) segmentation result along with the bounding boxes and centroids for (a), (c) segments with diagonals are identified as "static segments," and (d) final segmentation result for frame 3 after filtering the "static segments."

of object tracking can be found in [8]. Fig. 3 illustrates the segmentation result for frame 3, where the dark gray area represents the background and the light gray segments (class 2) are supposed to correspond to the vehicle objects we want to extract. However, there are several segments that do not correspond to moving vehicles. For example, part of the road pavement, road lamps, and trees are also identified as objects though they are not vehicle objects of interest.

Since the location of the camera monitoring the intersection area is fixed above the ground, the centroids of static segments (road pavement, trees, etc.) should remain fixed throughout the video sequence. This is how "static segments" are differentiated from "mobile segments" (moving vehicles) in this application domain. Fig. 3(c) shows the "static segments" identified in frame 3. However, there is problem related to vehicles that move very slowly; they may be identified as static segments and thus become part of the background information learned till the current frame. For example, as shown in Fig. 3(d), except the bus object in the middle of the intersection area, the other 10 cars [eight of them are located in the upper left part, one white car is located in the upper right part, and a gray car is in front of the bus, as seen in Fig. 3(a)] are identified as static segments and are merged with the already identified background area (the dark gray area) even though they are moving slowly. As mentioned earlier, based on the object tracking results, an initial reference background image can be generated. However, the initial background area information obtained here is not very accurate because many slow moving vehicles are identified as part of the background. In Section III.D, we show that it is not necessary to obtain very accurate initial background information in order to achieve good segmentation results. By applying a self-adaptive background adjusting and subtraction method, the proposed framework can robustly capture the spatio-temporal relationships of vehicle objects in real-life traffic video sequences.

2) Handling Occlusion Situations in Object Tracking: As mentioned earlier, in most cases, the complexities associated with the background preclude good segmentation results and complicate the solution for object occlusion situations. However, by applying the background subtraction method discussed in Section III.D, substantially simpler difference images are obtained. This enables fast and satisfactory segmentation results, greatly benefiting the handling of object occlusion situations. A more sophisticated object tracking algorithm, namely the *backtrack-chain-update split* algorithm, is given in [8], [23], which can handle the situation of two objects overlapping under certain assumptions (e.g., the two overlapped objects should have similar sizes). It considers the situation when overlapping happens between two objects that separate from each other in a later/earlier frame. In this case, it can find the split object and use the information in the current frame to update the previous frames in a backtrack-chain manner. Our previous study suggests that this algorithm is effective in handling two-object occlusions.

However, there are cases where a large object overlaps with a small one. For example, as shown in Fig. 4, the large bus merges with the small gray car to form a new big segment in frame 20 though they are two separate segments in frame 19. In this scenario, the car object and the bus object that were separate in frame 19 cannot find their corresponding segments in frame 20 by centroid-matching and size restriction. However, from the new big segment in frame 20, we can reason that this is an "*overlapping*" segment that actually includes more than one vehicle object. For this purpose, a difference binary map reasoning method is proposed in this paper to identify which objects the "*overlapping*" segment may include. The idea is to obtain the difference binary map by subtracting the segment result of frame 19 from that of frame 20 and check the amount of difference between the segmentation results of the consecutive frames. As shown in the difference binary map in Fig. 4, the

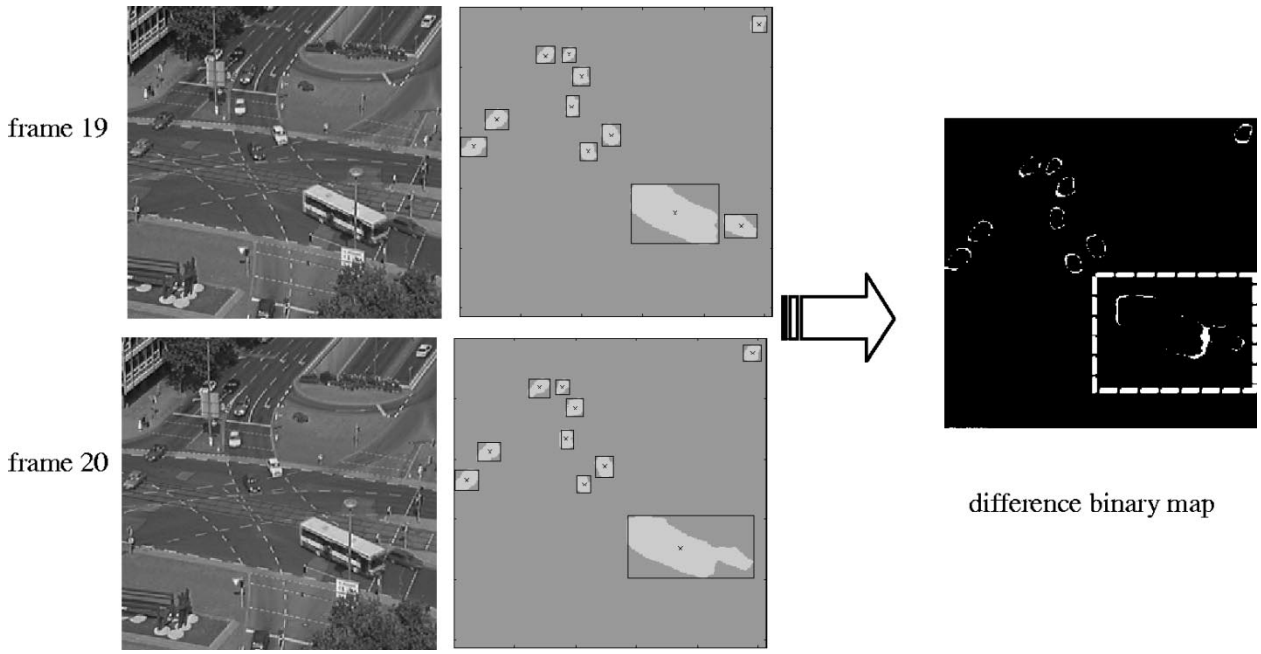

Fig. 4. Handling object occlusion in object tracking.

white areas in it indicate the amount of difference between the segmentation results of the two consecutive frames. Thereby, the car and bus objects in frame 19 can be roughly mapped into the area of the big segment in frame 20 with relatively small differences. Hence, the vehicle objects in the big segment in frame 20 can be obtained by reasoning that this segment is most probably related to the car and bus objects in frame 19. Therefore, for the big segment (the "*overlapping*" segment) in frame 20, the corresponding links to the car and bus objects in frame 19 can be created, which means that the relative motion vectors of that big segment in the following frames will be automatically appended to the trace tubes of the bus and car objects in frame 19.

D. Self-Adaptive Background Subtraction

After background learning using the first few video frames, most of the static segments can be identified and subtracted from the set of segments in the current frame [as shown in Fig. 3(d)]. In our experiment, we use the first 3 frames for initial background learning.

As shown in Fig. 5(a), an initial reference background image (background_image_3) is obtained by extracting the final segmentation result of frame 3 (segments_3) from the original frame 3. Then, this initial background image is subtracted from the next frame (frame 4) to obtain the difference image (difference_image_4). As illustrated in Fig. 5(a), the difference_image_4 not only stores the visual information for the bus object identified in frame 3, but also shows the motion difference information for each car object that has been identified as part of the background in frame 3. That is, from the segmentation result for difference_image_4, all 11 vehicle objects are successfully identified as separate segments in frame 4 no matter whether they moved fast or slow. Though, in difference_image_4, the visual information representing the motion difference of the slow moving vehicles is very obscure and much darker when compared with that of the bus object, the SPCPE method can successfully identify all the vehicles in frame 4, which provides a better estimation for a new background image. For comparison purposes, we also show the original segmentation result (original_segments_4) for frame 4 without any background learning and subtraction. There, the bus object merged with the road pavement to form a big segment, and most of other vehicles cannot be identified as separate segments.

A key point here is that if a new background image is always constructed based on the current frame's segmentation result, the construction error will accumulate and finally become unacceptable. This means that the trail of a moving vehicle will also be identified as part of the object, which causes inaccurate or even unacceptable segmentation results after processing a number of frames. This is because when an object moves, a small part of the background area will appear in the current frame though this area was identified as part of the vehicle object in the preceding frame. Without any adjustments, the accumulative construction error will lead to unacceptable segmentation results. In our framework, a simple but effective self-adaptive adjustment is applied in creating a new background image. The adjustment is done by shrinking the size of the bounding box of each segment before constructing a new background image for use in the next frame's segmentation based on the current segmentation results. This adjustment can be thought as the prediction of the changes in the background area. The key aspect of this self-adaptive process is the strong support derived from the robustness of the SPCPE segmentation method. Although the background area is enlarged a little as the result of the shrinking of the bounding boxes, the resulting difference image still includes the new information of the motion difference (if any) that can be identified as part of the moving object by the SPCPE segmentation method. In other words, due to the shrinking of bounding box, the object size will decrease in the newly created

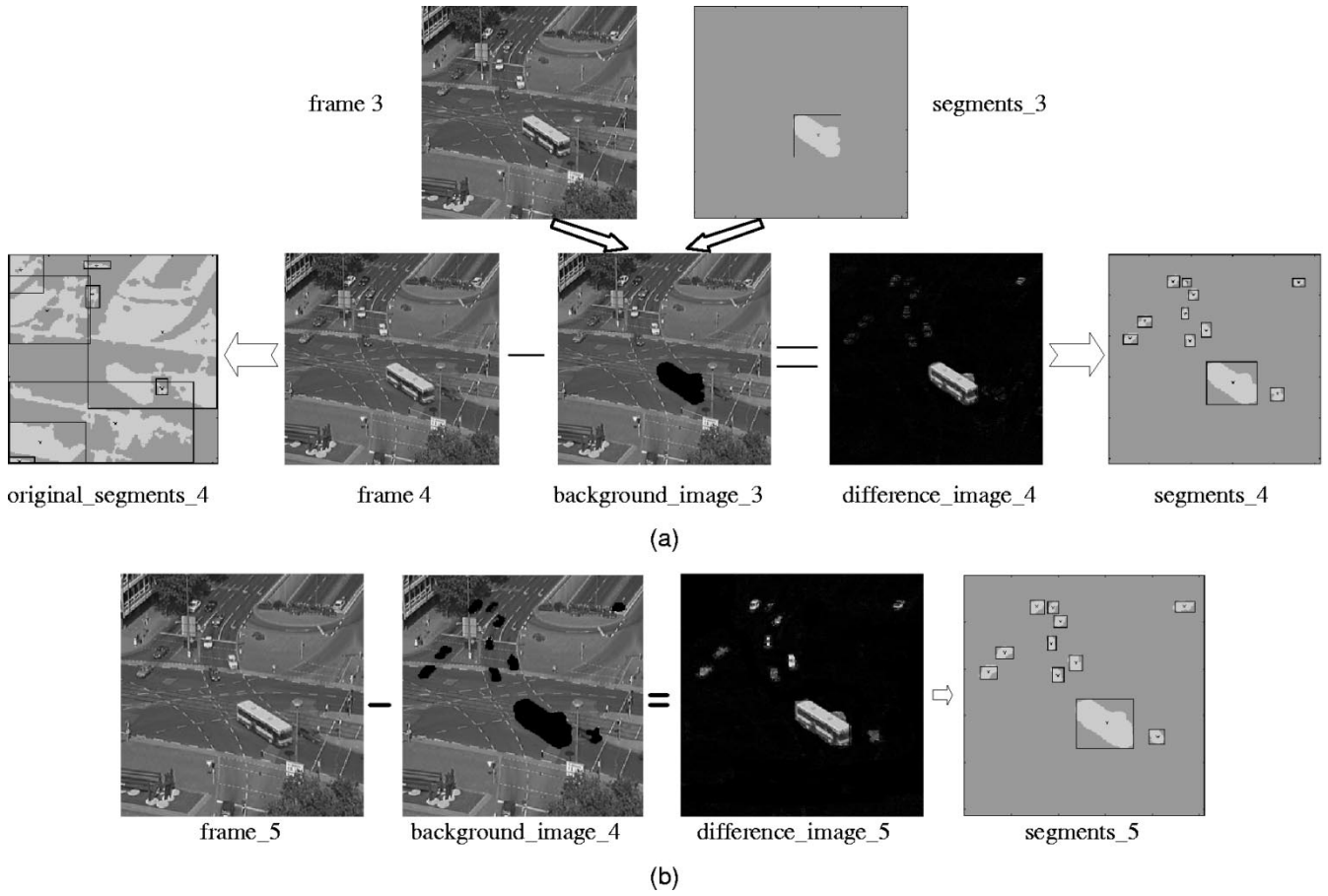

Fig. 5. Self-adaptive background learning and subtraction in the traffic video sequence. (a) Generating the initial reference background image (background_image_3) and subtracting it from the next frame (frame 4), then applying the segmentation on the obtained difference image (difference_image_4) and (b) self-adaptive adjustment in generating new background images.

background image. However, this will not affect the segmentation result of the next frame because the motion difference area will still appear in the difference image, and can be identified as part of the vehicle object to compensate the size loss. As a vehicle object moves from the current frame to the next frame, part of the area of the vehicle object identified in the current frame may become part of the background area in the next frame. Without shrinking, this part of background area may be misidentified as part of the current object. That is, the shrinking of the bounding box is used to achieve the motion prediction without losing any information in segmenting and identifying the moving vehicle objects. Fig. 5(b) shows the segmentation result for frame 5 by applying the proposed background adjustment method. The segmentation result accurately identifies the bounding boxes of all vehicle objects.

E. Using MATNs and Multimedia Input Strings

The spatio-temporal relations of the vehicle objects in the video sequence must be captured in an efficient way. In the proposed framework, the spatio-temporal relations are indexed and modeled using a MATN model and multimedia input strings [5]. A MATN can be represented diagrammatically as a labeled directed graph, called a *transition graph*. Multimedia input strings are used as inputs to an MATN, and are proposed to represent the spatio-temporal relations of the vehicle objects in the video sequences. Multimedia input strings adopt the notations from regular expressions [24]. A multimedia input string is accepted by the grammar if there is a path of transitions which corresponds to the sequence of symbols in the string and which leads from a specified initial state to one of a set of specified final states.

Fig. 6 illustrates the use of MATNs and multimedia input strings to model the spatio-temporal relations of the vehicle objects in traffic video frames. Assume three objects, the *ground*, *car*, and *bus* represented by $\boldsymbol{G}, \boldsymbol{C}$, and $\boldsymbol{B}$, respectively. As introduced in [5], one semantic object is chosen as the target semantic object in each video frame. The minimal bounding rectangle (MBR) concept in R-trees [25] is also used so that each semantic object is covered by a rectangle. Here, we choose the *ground* as the target object. In order to distinguish the 3-D relative spatial positions, 27 numbers are used [5]. In this example, each frame is divided into nine 2-D subregions with the corresponding subscript numbers shown in Fig. 6(a).

Each video frame is represented by an input symbol in a multimedia input string. The "&" symbol between two vehicle objects denotes that the vehicle objects appear in the same frame. The subscripted numbers are used to distinguish the relative spatial positions of the vehicle objects relative to the target object "*ground*" [as shown in Fig. 6(a)]. For simplicity, two example frames (frames 1 and 2) are used to explain how to construct the

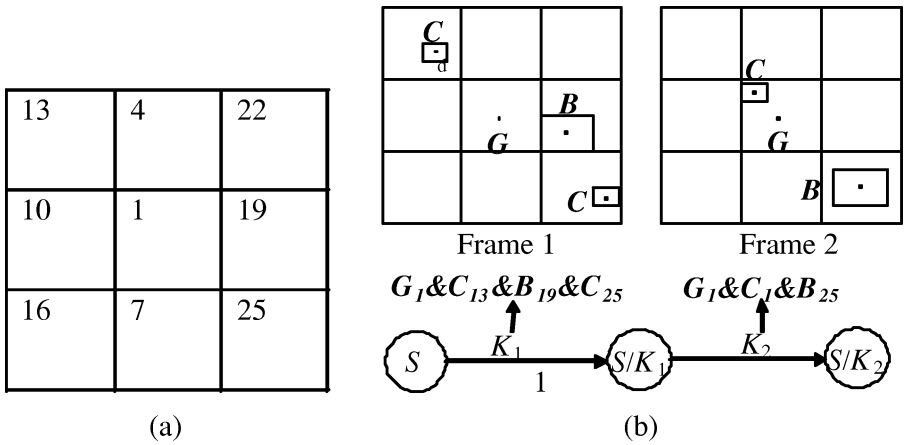

Fig. 6. MATN and multimedia input strings for modeling the key frames of traffic video shot S. (a) Nine subregions and their corresponding subscript numbers and (b) example MATN model.

TABLE I
OVERALL PERFORMANCE OF VEHICLE OBJECT IDENTIFICATION

Video #	Number of Frames	Frame Size	Quality	Correct	Missed	False	Precision	Recall
Video #1	50	512×512	Good	64	0	0	100%	100%
Video #2	300	268×251	Medium	83	12	1	99%	87%
Video #3	1733	353×473	Poor	621	66	14	98%	90%
Video #4	3000	240×320	Medium	1611	195	103	94%	89%
Overall	**5083**			**2379**	**273**	**118**	**95%**	**90%**

multimedia input strings and the MATN [as shown in Fig. 6(b)]. The multimedia input string to represent these two frames is as follows:

$$\underbrace{(G_1 \,\&\, C_{13} \,\&\, B_{19} \,\&\, C_{25})}_{K_1}\underbrace{(G_1 \,\&\, C_1 \,\&\, B_{25})}_{K_2}.$$

Two input symbols, K_1 and K_2 are used for this purpose. The appearance sequence of the vehicle objects in an input symbol is based on the relative spatial locations of the vehicle objects in the traffic video frame from left to right and top to bottom. For example, frame 1 is represented by input symbol K_1. $\boldsymbol{G_1}$ indicates that $\boldsymbol{G}$ is the target object. $\boldsymbol{C_{13}}$ implies that the first car object is to the left of and above $\boldsymbol{G}$, $\boldsymbol{B_{19}}$ denotes that the bus object is to the right of $\boldsymbol{G}$, and $\boldsymbol{C_{25}}$ means that the second car is to the right of and below $\boldsymbol{G}$. The multimedia input string for frame 2 is different from that of frame 1 in that the car $\boldsymbol{C_{25}}$ that appeared in frame 1 has already left the road intersection in frame 2, the car $\boldsymbol{C_{13}}$ in frame 1 moved into the same subregion as $\boldsymbol{G}$ in frame 2 and thus becomes $\boldsymbol{C_1}$, and the bus object $\boldsymbol{B_{19}}$ in frame 1 moved into the lower right corner in frame 2 and becomes $\boldsymbol{B_{25}}$. Hence, the spatial locations of vehicle objects change, and the number of vehicle objects decreases from three to two. This example illustrates the ability of a multimedia input string to represent the spatial relations of objects.

Fig. 6(b) is the MATN for the two frames in this example. The starting state name for this MATN is $S/$. As shown in Fig. 6(b), there are two arcs with labels K_1 and K_2. The different state nodes and the order of their appearance in the MATN is based on the temporal relations of the selected video frames. The multimedia input strings model the relative spatial relations of the vehicle objects.

IV. EXPERIMENTAL ANALYSIS

A. Experimental Results

Four real-life traffic video sequences are used to analyze spatio-temporal vehicle tracking using the proposed learning-based vehicle tracking and indexing framework. These video sequences are obtained from different sources, showing four different road intersections with different qualities and under different weather conditions. Videos #1 to #3 are grayscale videos downloaded from the research website of University Karlsruhe [26]. Video #3 was taken in the winter where there was snow on the lane and light snow was falling with a strong wind. The qualities of video #2 and video #3 are significantly worse than that of video #1. Video #4 is also a grayscale video captured with a common digital camera. The proposed new framework is fully unsupervised in that it can enable the automatic background learning process that greatly facilitates the unsupervised video segmentation process without any human intervention. Based on our experiments, by applying the background learning process, the computation time savings for the segmentation process are eighty percent of the original time cost. As shown in Table I, the overall

performance of vehicle object identification over the four video sequences is robust. The precision and recall values are approximately 95% and 90%, respectively.

A portion of the traffic video #1 is used to illustrate how the proposed framework can be applied to address spatio-temporal queries such as: "estimate the traffic flow at this road intersection approach from 5:00 PM to 5:30 PM." This requires the proposed framework to elicit information on the number of vehicles passing through that intersection approach in the stated time duration. Further, the collected information must be indexed and stored into a multimedia database in real-time or off-line. These aspects are addressed hereafter.

The enhanced video segmentation method is applied to the video frames by considering two classes. Consider a video frame like the one in Fig. 3(a). There could be several variations in the background such as road pavement, trees, pavement markings/signage, and ground. Since the interest is only in the vehicle objects, it is a two-class problem. The first frame is partitioned with two classes using random initial partitions. After obtaining the partition of the first frame, the partitions of the subsequent frames are computed using the previous partitions as the initial partitions since there is no significant difference between consecutive frames. By doing so, the segmentation process will converge fast, thereby providing support for real-time processing. The most time consuming part at the beginning is the background learning process because enough background information does not exist at that time. Hence, the segmentation process has to deal with the original video frames which include very complex backgrounds. The effectiveness of the proposed background learning process ensures that a long run is not necessary to fully determine the accurate background information. In our experiments, the preliminary background information can be obtained usually within 5 consecutive frames, and it is good enough for the future segmentation process. In fact, by combining the background learning process with the unsupervised segmentation method, our framework can enable the adaptive learning of background information.

Fig. 7 shows the segmentation results and the corresponding multimedia input strings for a few frames (19, 25, 28, and 34) along with the original frames, background images, and the difference images. As illustrated by the figure, the background of this traffic video sequence is very complex. Some vehicle objects (for example, the small gray vehicles in the upper left part of the video frames) can easily be ignored or confused with the road surface and surrounding environment. While there is an existing body of literature [27] that addresses relatively simple backgrounds, our framework can address far more complex situations, as illustrated here.

In Fig. 7, the video frames in the leftmost column represent the original frames. The second column shows the background images derived from the immediate preceding frames. The third column shows the difference images after background subtraction. The segmentation results are illustrated in the fourth column, and the rightmost column shows the bounding box and centroid for each segment in the current frame. As illustrated by the fourth column of Fig. 7, a single class can capture almost all vehicle objects, even those vehicles that look small and obscure in the upper left area of the video frames. Another class captures most part of the ground. Also, the fourth column of Fig. 7 shows that almost all vehicle objects are captured as separate segments. However, the bus and the gray car (in the lower right part of the intersection area) are identified as one big segment in frames 25, 28, and 34; while they are identified as separate segments in frame 19. As discussed in Section III, this occlusion situation can be detected by the proposed difference binary map method. In our indexing schema for a multimedia database, we use a special symbol to denote such an "*overlapping*" segment that has the corresponding links to the related vehicle segments in the preceding frame.

Fig. 7 also lists the multimedia input strings for the selected frames. As discussed in Section III.E, we use symbolic representations (multimedia input strings) to represent the spatial relationships of the objects in each frame. In the rightmost column of Fig. 7, the ground ($\boldsymbol{G}$) is selected as the target object, the segments are denoted by $\boldsymbol{C}$ for cars or $\boldsymbol{B}$ for buses, and the "*overlapping*" objects are denoted by symbol $\boldsymbol{O}$ which has the corresponding links to the related segments in the preceding frame. As shown in Fig. 7, there are 11 vehicle objects visible in frame 19—two gray cars ($\boldsymbol{C_{10}}$ & $\boldsymbol{C_{10}}$) are in the left middle area, one white car is located in the upper left area ($\boldsymbol{C_{13}}$), three cars are in the upper middle area ($\boldsymbol{C_4}$ & $\boldsymbol{C_4}$ & $\boldsymbol{C_4}$), two cars are located in the middle area($\boldsymbol{C_1}$ & $\boldsymbol{C_1}$), one bus ($\boldsymbol{B_{25}}$) and one dark gray car ($\boldsymbol{C_{25}}$) are in the lower right corner, and another white car driving toward northeast is located in the upper right area ($\boldsymbol{C_{22}}$). Frame 25 indicates that the white car (the second $\boldsymbol{C_4}$ in frame 19) is moving slowly into the middle area so that its symbol changes to $\boldsymbol{C_1}$, and the bus and dark gray car ($\boldsymbol{B_{25}}$ and $\boldsymbol{C_{25}}$ in frame 19) are identified as an "*overlapping*" object $\boldsymbol{O_{25}}$ in frame 25. In frame 28, the white car ($\boldsymbol{C_{22}}$ in frames 19 and 25) in the upper right corner disappeared from the intersection area while another white car appears ($\boldsymbol{C_{22}}$) in the upper right corner from the underneath tunnel. Also in frame 28, we can see part of a new car object entering into the intersection area from the upper bound, which is successfully identified as a new segment (the third $\boldsymbol{C_4}$ in frame 28) even though the small area it occupies in frame 28 is part of the background in the preceding frames. And in frame 34, one gray car (heading southwest, denoted by the first $\boldsymbol{C_{10}}$ in the preceding frames) disappeared from the intersection area. A point to note here is that though the white car (located in the upper left part) that is slowly heading southeast has part of its body becoming progressively invisible due to the rectangular poster in front of it in the frames, our framework can successfully identify it as a complete segment (denoted by $\boldsymbol{C_{13}}$ in all the selected frames) throughout the entire video sequence. As illustrated by Fig. 7, the multimedia input strings can model not only the number of vehicle objects, but also the relative spatial relations of the vehicle objects.

As mentioned in Section III, we apply the object tracking technique to track the trail of each vehicle object to the extent possible. Fig. 8 shows the tracking of the trail of the bus object in the video sequence. Fig. 8(a) shows the bounding boxes and centroids of the bus object from frame 4 to frame 34, while Fig. 8(b) shows the trail information of the bus object by applying the object tracking technique. In fact, in the proposed indexing schema, it is not necessary to record the position of the

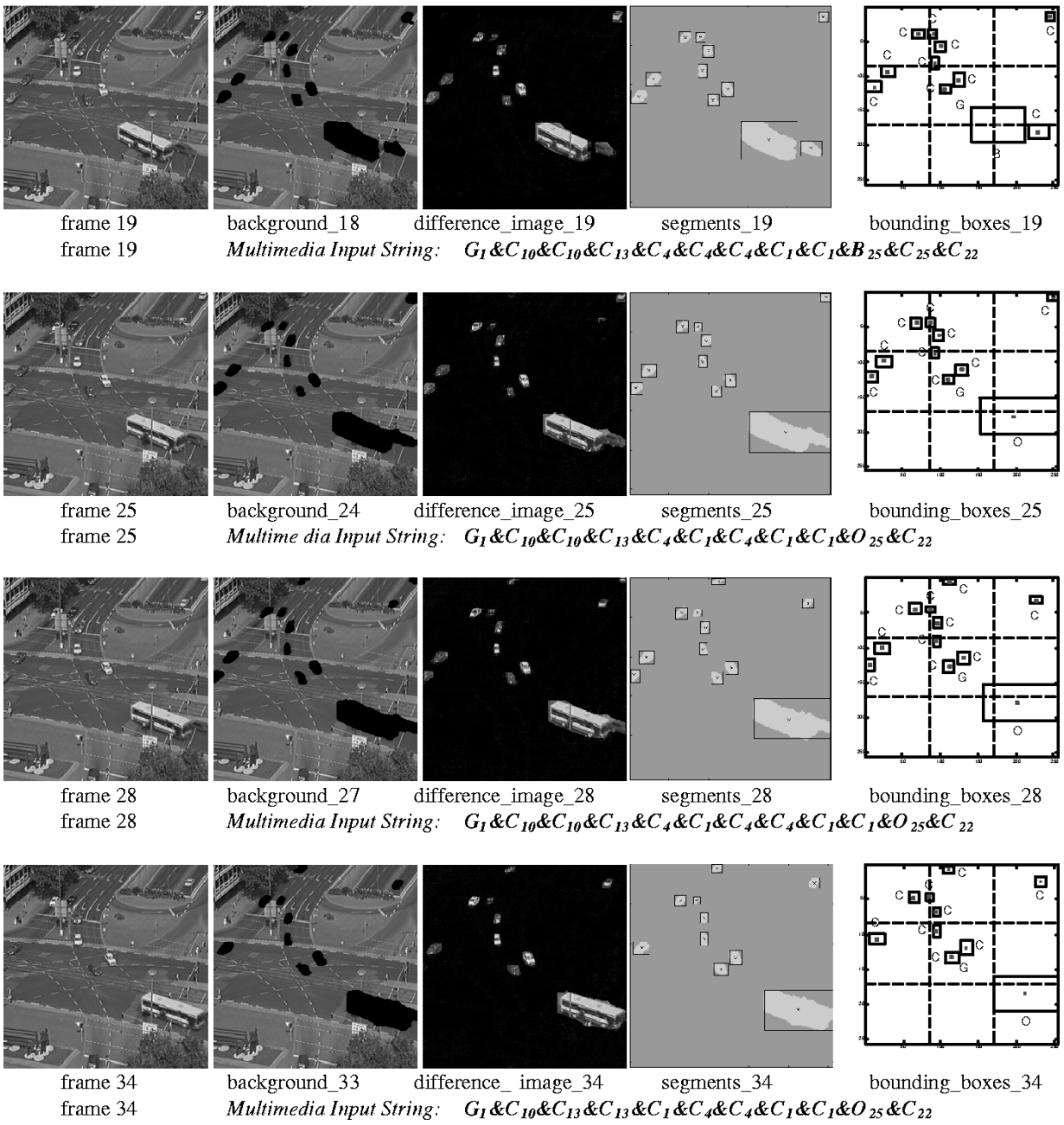

Fig. 7. Segmentation results and multimedia input strings for frames 19, 25, 28, and 34. The leftmost column shows the original video frames, the second column shows background reference images derived from the immediate preceding frames, the third column shows difference images obtained by subtracting the background reference images from the original frames, the fourth column shows the vehicle segments extracted from the video frames, and the rightmost column lists the bounding box and centroid for each segment in the current frame.

bus segment in each frame. Instead, it can be done when there is a *major move* in that object or based on a fixed frequency.

As described earlier, the framework can determine not only the indexes for the number of vehicle objects, but also the index information of relative spatial relations by recording the positions of the centroid of the segment throughout the video sequence. However, to address the traffic flow query mentioned at the beginning of this section, it is necessary to have a "judge line" in the frame so that the traffic flow in a specified direction can be estimated. This judge line can be provided by the end user. For example, it could be a line before vehicles go into or out of the intersection area. By using the centroid position information of each vehicle object, the traffic flow in a specified direction can be roughly estimated. Also, since vehicle classification may be important, the sizes of the bounding boxes are used to determine the vehicle types (such as "car" and "bus"). For "*overlapping*" segments, their links to specific vehicle segments can be used to correctly account for their contribution to the traffic flow. While the traffic flow query mentioned earlier is used as an example here, the proposed framework has the potential to address other (and more complex) spatio-temporal related database queries. For example, it can be used to reconstruct accidents at intersections in an automated manner to identify causal factors to enhance safety.

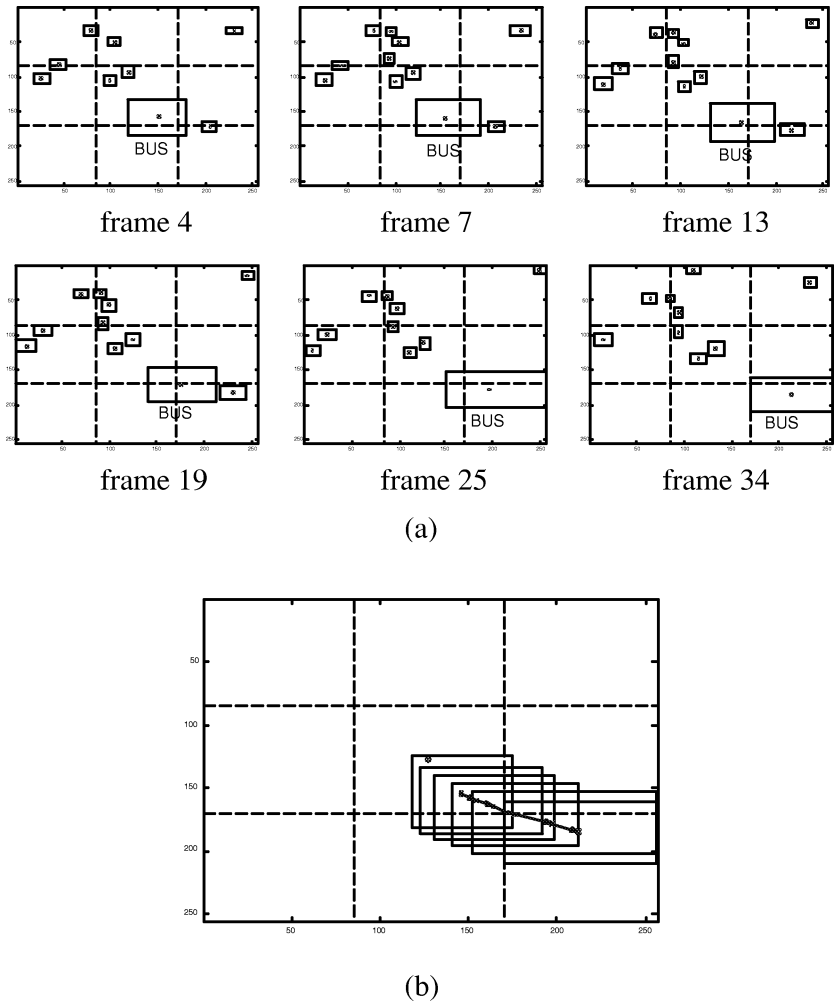

Fig. 8. Tracking the trail of a bus in the traffic video sequence. (a) Bounding boxes and centroids for the bus object in the video sequence and (b) the trail of the bus object from frame 4 to frame 34.

B. Insights

The experimental results demonstrate the effectiveness of vehicle identification and indexing using the proposed framework. The index information can be used to address spatio-temporal queries for traffic applications. In the study experiments, the backgrounds of the traffic video sequences are complex. Our framework can address such complex scenarios for intersection monitoring.

Based on our experiments, the false positives are mainly caused by camera motion. The temporal tracking over a set of frames allows us to reduce this kind of errors since segments identified due to noise and motion are not temporally coherent. By contrast, the typical reason to have the false negatives is because of the slow motion of the vehicles, in which the motion difference exists but is not significant. In such situations, our segmentation method tends to detect only part of the vehicle object. It is typically a small region and will likely be discarded through a noise-filtering phase. In our framework, the segments which are very small will be identified as noise and hence rejected. The selection of the threshold value for determining whether a segment should be identified as noise depends on prior knowledge such as the average size of the vehicle objects (in terms of pixels) under a specific shooting scale. Also, the thresholds selected for object tracking depend on the shooting scale and the average vehicle speed. The constancy for the shrinking of the bounding box for background update is selected as four to six pixels, and it works well in most scenarios.

V. Conclusions and Future Work

In this paper, a learning-based spatio-temporal vehicle tracking and indexing framework is presented for unsupervised video data storage and access for real-time traffic operations. It incorporates a unsupervised image/video segmentation method, background learning and subtraction techniques, object tracking, MATN model, and multimedia input strings. A self-adaptive background learning and subtraction method is proposed and applied to four real life traffic video sequences to enhance the object segmentation procedure for obtaining more accurate spatio-temporal information of the vehicle objects. The background learning process is relatively simple and very effective based on our experiment results. Almost all vehicle objects are successfully identified through this framework. The spatio-temporal relationships of the vehicle objects are captured via the unsupervised image/video segmentation method and the proposed object tracking algorithm, and modeled using the

MATN model and multimedia input strings. Useful information is indexed and stored into a multimedia database for further information retrieval and query. A fundamental advantage of the proposed background learning algorithm is that it is fully automatic and unsupervised, and performs the adjustments in a self-adaptive way. As illustrated by the experiments, the initial inaccurate background information can be iteratively refined as the procedure proceeds, thereby benefiting the segmentation process in turn. Hence, the proposed framework can deal with very complex situations vis-à-vis intersection monitoring.

The proposed research seeks to bridge the important missing link between transportation management and multimedia information technology. In order to develop a transportation multimedia database system (MDBS) with adequate capabilities, the following future work will be investigated: 1) to store and organize the rich semantic multimedia data in a systematic and hierarchical model; 2) to identify the vehicle objects in video sequences under different conditions; and 3) to fuse different types of media data.

REFERENCES

[1] R. Cucchiara, M. Piccardi, and P. Mello, "Image analysis and rule-based reasoning for a traffic monitoring system," in *Proc. IEEE Int. Conf. Intelligent Transportation Systems (IEEJ JSAI)*, vol. 1, Tokyo, Japan, June 2000, pp. 119–130.

[2] D. J. Dailey, F. Cathey, and S. Pumrin, "An algorithm to estimate mean traffic speed using uncalibrated cameras," *IEEE Trans. Intell. Transport. Syst.*, vol. 1, pp. 98–107, June 2000.

[3] T. Huang and S. Russell, "Object identification: A Bayesien analysis with application to traffic surveillance," *Artificial Intell.*, vol. 103, pp. 1–17, 1998.

[4] S. Kamijo, Y. Matsushita, and K. Ikeuchi, "Traffic monitoring and accident detection at intersections," *IEEE Trans. Intell. Transport. Syst.*, vol. 1, pp. 108–118, June 2000.

[5] S.-C. Chen and R. L. Kashyap, "A spatio-temporal semantic model for multimedia database systems and multimedia information systems," *IEEE Trans. Knowledge Data Eng.*, vol. 13, pp. 607–622, July/Aug. 2001.

[6] S.-C. Chen, M.-L. Shyu, and C. Zhang, "An intelligent framework for spatio-temporal vehicle tracking," in *Proc. 4th Int. IEEE Conf. Intelligent Transportation Systems*, Oakland, CA, Aug. 25–29, 2001, pp. 213–218.

[7] ——, "An unsupervised segmentation framework for texture image queries," in *Proc. 25th IEEE Computer Soc. Int. Computer Software and Applications Conf. (COMPSAC)*, Chicago, IL, Oct. 8–12, 2001, pp. 569–573.

[8] S.-C. Chen, M.-L. Shyu, C. Zhang, and R. L. Kashyap, "Object tracking and augmented transition network for video indexing and modeling," *Proc. 12th IEEE Int. Conf. Tools with Artificial Intelligence (ICTAI 2000)*, pp. 428–435, Nov. 13–15, 2000.

[9] S. Peeta, "System optimal dynamic traffic assignment in congested networks with advanced information systems," Ph.D. dissertation, The University of Texas at Austin, 1994.

[10] S. Peeta and H. S. Mahmassani, "System optimal and user equilibrium time-dependent traffic assignment in congested networks," *Annals Operations Res.*, pp. 81–113, 1995.

[11] ——, "Multiple user classes real-time traffic assignment for online operations: A rolling horizon solution framework," *Transport. Res.*, vol. 3, no. 2, pp. 83–98, 1995.

[12] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition*, 1999, pp. 246–252.

[13] S. M. Smith and J. M. Brady, "Asset-2: Real-time motion segmentation and shape tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 814–820, 1995.

[14] [Online]. Available: http://www-2.cs.cmu.edu/~vsam/research.html #COMPUS

[15] I. Cohen and G. Medioni, "Detecting and tracking objects in video surveillance," in *Proc. IEEE Computer Vision and Pattern Recognition 99*, Fort Collins, CO, June 1999, pp. 319–325.

[16] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, MA: Addison-Wesley, 1993.

[17] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, 1998, pp. 22–31.

[18] I. Haritaoglu, D. Harwood, and L. Davis, "W4—Who, where, when, what: A real-time system for detecting and tracking people," in *Proc. IEEE 3rd Int. Conf. Face and Gesture Recognition*, Nara, Japan, 1998, pp. 222–227.

[19] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *Proc. 7th Int. Conf. Computer Vision (ICCV'99)*, Island of Crete, Sept. 1999, pp. 255–261.

[20] S.-C. Chen, S. Sista, M.-L. Shyu, and R. L. Kashyap, "An indexing and searching structure for multimedia database systems," in *Proc. IS&T/SPIE Conf. Storage and Retrieval for Media Databases 2000*, San Jose, CA, Jan. 23–28, 2000, pp. 262–270.

[21] S. Sista and R. L. Kashyap, "Unsupervised video segmentation and object tracking," *Computers Industry*, vol. 42, no. 2–3, pp. 127–146, June 2000.

[22] I. Haritaoglu, D. Harwood, and L. Davis, "A fast background scene modeling and maintenance for outdoor surveillance," in *Proc. 15th IEEE Int. Conf. Pattern Recognition: Applications, Robotics Systems, and Architectures*, Barcelona, Spain, Sept. 2000, pp. 179–183.

[23] S.-C. Chen, M.-L. Shyu, C. Zhang, and R. L. Kashyap, "Identifying overlapped objects for video indexing and modeling in multimedia database systems," *Int. J. Artificial Intell. Tools*, vol. 10, no. 4, pp. 715–734, Dec. 2001.

[24] S. C. Kleene, *Representation of Events in Nerve Nets and Finite Automata, Automata Studies*. Princeton, N.J.: Princeton Univ. Press, 1956, pp. 3–41.

[25] A. Guttman, "R-tree: A dynamic index structure for spatial search," in *Proc. ACM SIGMOD*, June 1984, pp. 47–57.

[26] [Online]. Available: http://i21www.ira.uka.de/cgi-bin/download?bad

[27] N. Friedman and S. Russell, "Image segmentation in video sequences: A probabilistic approach," in *Proc. 13th Conf. Uncertainty in Artificial Intelligence (UAI 97)*, 1997, pp. 175–181.

Shu-Ching Chen (S'95–M'99) received Master's degrees in computer science, electrical engineering, and civil engineering from Purdue University, West Lafayette, IN, and the Ph.D. degree from the School of Electrical and Computer Engineering, at the same university, in December 1998.

He has been an Assistant Professor in the School of Computer Science, Florida International University (FIU) since August 1999. He has authored and coauthored more than 100 research papers in journals, refereed conference/symposium/workshop proceedings and book chapters. His main research interests include distributed multimedia database systems, data mining, and multimedia networking.

Dr. Chen was the General Co-Chair of the 2003 IEEE International Conference on Information Reuse and Integration, and Program Co-Chair of the 10th ACM International Symposium on Advances in Geographic Information Systems and the First ACM International Workshop on Multimedia Databases.

Mei-Ling Shyu (S'95–M'99) received three Master's degrees in computer science, electrical engineering, and restaurant, hotel, institutional, and tourism management from Purdue University, West Lafayette, IN, and the Ph.D. degree from the School of Electrical and Computer Engineering, at the same university, in 1999.

She has been an Assistant Professor at the Department of Electrical and Computer Engineering, University of Miami, Miami, FL, since January 2000. She has authored and coauthored more than 80 technical papers published in various prestigious journals, referred conference/symposium/workshop proceedings and book chapters. Her research interests include data mining, multimedia database systems, multimedia networking, and database systems.

Dr. Shyu was the Program Co-Chair of the First ACM International Workshop on Multimedia Databases.

Srinivas Peeta was an Assistant Professor in the Transportation and Infrastructure Systems Group, Purdue University, West Lafayette, IN, from 1994 to 2000. Since August 2000, he has been an Associate Professor of civil engineering at the same university. He has written more than 45 papers in archival journals and refereed conference proceedings. His research interests broadly include the use of operations research, control theory, computational intelligence techniques, and sensor technologies to model and evaluate the dynamics of large-scale transportation networks, especially in the context of advanced information systems.

Dr. Peeta's Ph.D. dissertation won the 1994 best dissertation award in the Transportation Science Section of the Institute for Operations Research and Management Science. He received a CAREER Award from the National Science Foundation in 1997.

Chengcui Zhang (S'00) received the Bachelor and Master degrees in computer science from Zhejiang University, China. Currently, she is a Ph.D. candidate at the school of Computer Science, Florida International University (FIU).

She has authored and coauthored more than 30 technical papers published in various prestigious journals, referred conference/workshop proceedings and book chapters. Her research interests include multimedia databases, multimedia data mining, image and video database retrieval, and GIS data filtering.

Ms. Zhang is the recipient of several awards, including the Presidential Fellowship and the Best Graduate Student Research Award at FIU.