Learning Detectors from Large Datasets for Object Retrieval in Video Surveillance

Rogerio Feris, Sharath Pankanti
IBM T. J. Watson Research Center
{*rsferis,sharat*}*@us.ibm.com*

Behjat Siddiquie
SRI International
behjat.siddiquie@sri.com

***Abstract*—We address the problem of learning robust and efficient multi-view object detectors for surveillance video indexing and retrieval. Our philosophy is that effective solutions for this problem can be obtained by learning detectors from huge amounts of training data. Along this research direction, we propose a novel approach that consists of strategically partitioning the training set and learning a large array of complementary, compact, deep cascade detectors. At test time, given a video sequence captured by a fixed camera, a small number of detectors is automatically selected per image location. We demonstrate our approach on the problem of vehicle detection in challenging surveillance scenarios, using a large training dataset composed of around one million images. Our system runs at an impressive average rate of 125 frames per second on a conventional laptop computer.**

***Keywords*-Large-scale learning, object retrieval, video surveillance**

I. Introduction

The number of surveillance cameras monitoring public places is growing worldwide at an alarm rate. The United Kingdom has installed more than 4 million security cameras over the past decade. In New York City, the number of cameras has grown so rapidly that recently the *New York Times* observed that the "Big Apple" is slowly turning into the "Big Eyeball". The key problem faced by law-enforcement agencies is that there are not enough human eyes to watch video footage from such a vast camera array in order to detect events of interest. The process of watching surveillance videos is resource-consuming and suffers from high costs of employing security personnel.

The field of intelligent visual surveillance seeks to address this problem by applying computer vision techniques to automatically detect specific events in long video streams. In this context, we are particularly interested in enabling automatic object retrieval based on attributes from surveillance videos, with focus on vehicles. We have built a search infrastructure similar to the work of Feris et al [1], involving video processing, database ingestion, and web service interface to support user queries such as "Show me all blue vehicles traveling at high speed northbound, last Saturday, from 2pm to 4pm". This paper addresses an important component of this framework: how to develop a robust and efficient approach for vehicle detection in surveillance videos, under varying pose and lighting changes.

Traditional surveillance systems use background modeling for detecting moving objects in the scene [2]. They have serious limitations to handle environmental conditions such as rain, reflections, shadows, and also crowded scenes, as multiple objects close to each other are often merged into a single blob. Appearance-based object detectors [3], [4] arise as a natural alternative or complementary solution to deal with these issues, but current approaches are still limited in terms of accuracy and computational speed. Modeling appearance is indeed a difficult problem due to dramatic non-linear variations in the object appearance manifold incurred by pose and lighting changes.

Our approach to address this challenge is to exploit huge amounts of training data to learn robust and efficient multi-view object detectors. Different from other multimedia retrieval tasks, where a large number of object classes are involved, and few examples may be available for many of them, surveillance is mostly concerned about vehicles and people, fo which many training images can be easily obtained. In this paper we focus on vehicle detection, leveraging a dataset of around 1 million vehicle images, automatically collected from 50+ cameras, using the technique proposed in [5].

Figure 1 shows the architecture of our proposed system. We use motionlets [1] to automatically split our training dataset into semantic partitions related to vehicle pose. For each partition, we create a set of compact, complementary detectors, each trained in a deep cascade structure, using hundreds of thousands of selected negative examples. As a result, we form a large pool of compact detectors tuned to work on specific submanifolds of object appearance. Given a test surveillance video, with a particular scene geometry, only few detectors are automatically selected per image location according to the scene geometry and photometric conditions.

We summarize below the key **contributions** of this paper:

- We propose a novel framework for learning robust and extremely efficient multi-view object detectors from huge amounts of training data.
- We show the importance of using hundreds of thousands of positive and negative samples to effectively model the highly non-linear object appearance manifold and improve classification results.
- A novel visual object representation is proposed, cor-

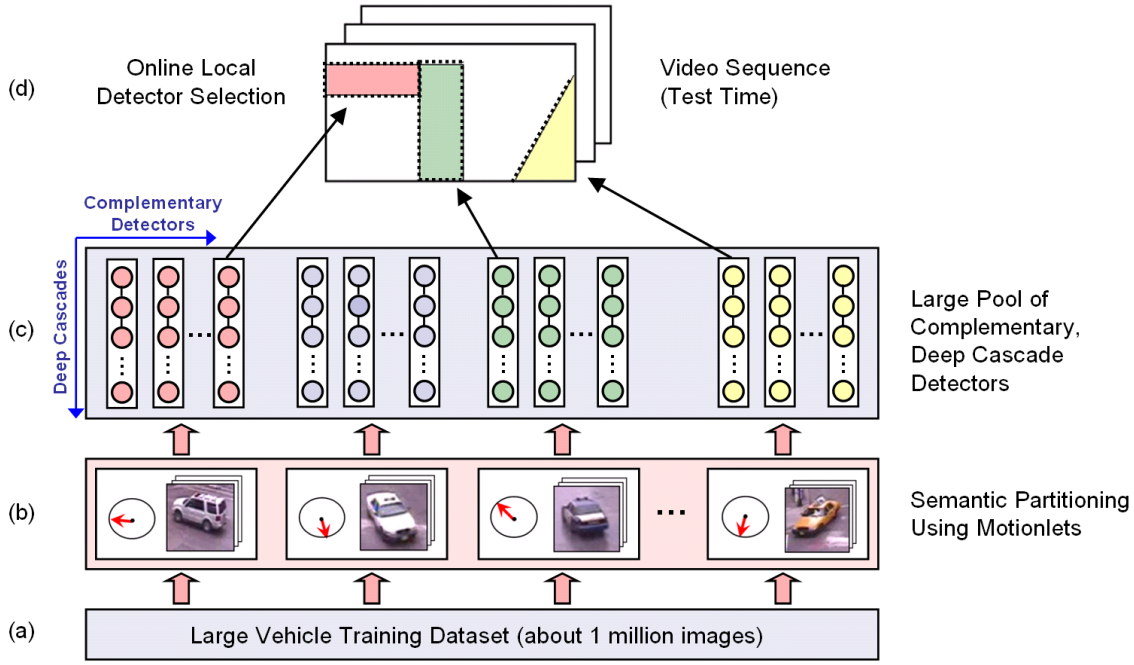

Figure 1. System Architecture. (a) Huge training dataset as input for the learning engine. (b) Automatic dataset partitioning based on motionlets. (c) A large array of compact, complementary, deep cascade detectors is created. (d) At test time, given a fixed-camera surveillance video, few detectors are selected per image position according to the scene geometry.

responding to a large pool of compact, complementary, deep cascade detectors.

- We propose a novel classifier selection method, specifically designed for fixed-camera surveillance videos, to select few classifiers per image location from the large pool of detectors.

II. Related Work

Various models and methods have been proposed for appearance-based object detection, in particular vehicle detection [3], [4], [6], [7], [8]. Although significant progress has been made over the past decades, most techniques are not designed to handle large amounts of data and usually run below 15 frames per second on conventional machines. Higher frame rate is a must for large-scale surveillance systems that run many video channels per server.

Large-scale learning is an emerging research topic in multimedia and computer vision. Recent methods have been proposed to deal with a large number of object classes [9], [10]. Even though the total number of training images processed by these methods is large (order of millions), the number of examples per object class is not (few thousands at most). Our method is unique in the sense that it considers a huge number of training examples for a specific object category.

Online Learning methods [11], [12] are suitable for processing large amounts of streaming data. They are tipically used for online classifier adaptation as new data comes in. A common limitation of these techniques is the inaccuracy in capturing online data to correctly update the classifier.

Traditional detector algorithms based on SVM or Adaboost classifiers [3], [6] usually consider few thousands of training examples for learning a classifier. As the number of training images grow to millions, training a single classifier becomes infeasible due to computational requirements and convergence issues. Convolutional neural networks [13] offers more advantages to handle large datasets, but are difficult to tune and, depending on the complexity of the data, may require a large number of filters. Our approach instead breaks up the complexity of the training data by learning efficient detectors on small strategically selected data partitions.

The work of Feris et al [1], [5] also splits the training data into motionlet clusters to better deal with non-linearities in the dataset. Training in each cluster is done with large-scale feature selection, but only few thousands of training examples are considered. We show in this paper that additionally exploiting large amounts of data leads to significant improvements in multi-view object detection

III. Learning Stage

We use a huge vehicle dataset containing around one million images for learning our object representation. The images in this dataset were automatically collected from more than 50 surveillance cameras, using the technique proposed in [5]. The images contain significant variation in

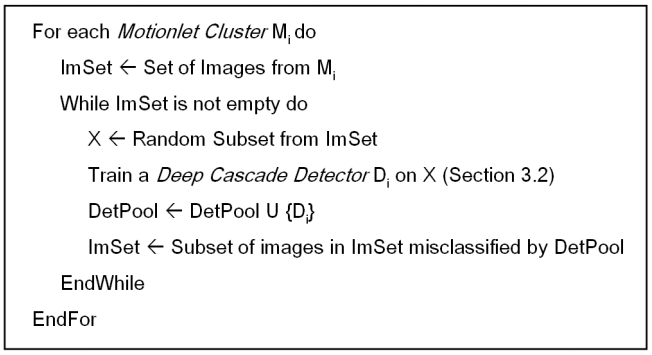

Figure 2. Algorithm for creating a large pool of complementary detectors from a large training set.

vehicle pose and different lighting conditions.

The appearance manifold of vehicle images under varying pose and lighting is complex and highly non-linear. Learning a monolithic detector on such a complex manifold leads to serious issues, including convergence problems and computational speed issues, not only for learning, but also for testing, as a monolithic detector trained on a large and diverse dataset would not be compact, requiring more feature computations. We instead adopt a different object representation, comprised of a large pool of complementary, compact, deep cascade detectors, as described next.

A. Pool of Complementary Detectors

The first step of our learning algorithm consists of automatically partitioning the dataset into motionlet clusters [1], i.e., clusters of vehicle images that share similar 2D motion direction. The motion information of a vehicle is directly related to its pose, therefore this operation provides a *semantic* partitioning of the dataset. It has been shown that motionlet clusters lead to more discriminative power when compared to appearance-based clustering [1]. Similar to [1], we used 12 motion clusters to segment the dataset.

Each motionlet cluster may still contain a fairly large number of images (e.g., tens of thousands). We further split this data by training a set of complementary detectors using the algorithm illustrated in Figure 2. For each motionlet cluster, we start by randomly sampling a a smaller set of positive samples - in our implementation, 5000 images - and train a deep cascade detector as described in Section III-B. We then apply the detector tuned to have very few or none false alarms (and all other already trained detectors in the pool) to the set of positive images of the motionlet cluster, and select those that are misclassified for training another complementary detector. Therefore we eliminate many redundant samples that are explained by previous detectors. This process is repeated until all the images in the cluster have been correctly classified by the current set of complementary detectors.

In our implementation, we used 12 motionlet clusters, and two complementary detectors per cluster, for a total of 24 deep cascade detectors. We are currently increasing this number and hope to have a pool containing hundreds of detectors soon. We stress that each detector is trained on a small subset of positive images of the training set, representing a particular submanifold of object appearance. Therefore the detectors are compact, requiring fewer features for discrimination.

B. Deep Cascade Detectors

We use the name "deep cascade detector" to refer to a cascade of classifiers trained using the algorithm described in [6], however with a much larger number of stages (bootstrap rounds). We train deep cascade detectors using a relatively small set of positive examples (few thousands), and a large number of selected negative samples (hundreds of thousands) to reduce false alarms. Our algorithm is based on the work of Viola and Jones [6]. It consists of a cascade of Adaboost classifiers, where the weak learners are simple thresholds over Haar-like features. Each stage of the cascade is tuned to minimize false negatives at the expense of a larger number of false positives - this allows fast inference by quickly discarding background image patches. Bootstrapping is employed by selecting negatives examples where the previous stages have failed. For details, see [6]. Our choice of employing Adaboost detectors rather than more sophisticated cascade classifiers [3] was mainly due to computational requirements.

We notice that a large number of bootstrap rounds, involving hundreds of thousands of selected negative samples, significantly improves performance, as supported by our experimental results (see Section V). Each stage of our deep cascade detector is trained with 5000 positive samples and 5000 negative samples, where the negative samples are image patches misclassified by the previous stages. We created deep cascades containing 40 stages, for a total of 200k selected negative examples. As illustrated in Figure 3, negative samples are initially selected from non-vehicle web images. After that, we collect negative samples from surveillance videos, usually in pedestrian areas. Finally, we add false positives related to vehicle parts or groups of vehicles, which can be collected automatically by a simple system which enables the user to collect image patches of detector firings with user-defined minimum and maximum patch sizes for specific locations of the video.

IV. Testing Stage

In certain traffic scenes, such as the one depicted in Figure 4, vehicles may drive in a single direction, with a well-defined pose. Even in more complex traffic intersections, vehicles may appear in few specific poses most of the time and be allowed to turn only at specific image locations. The minimum and maximum expected vehicle sizes can also be

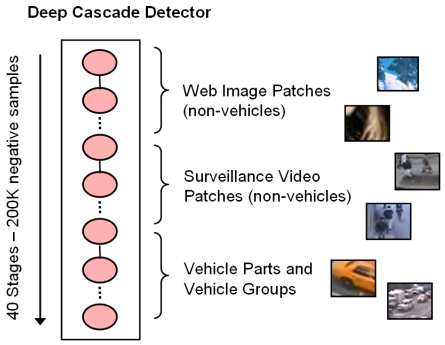

Figure 3. A deep cascade detector is trained with few thousands of positive samples and a large amount (hundreds of thousands) of selected negative examples.

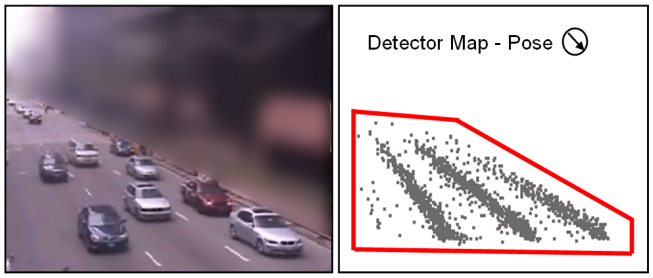

Figure 4. Example of a surveillance scene where only one detector is selected from the large pool and applied only inside the detector map depicted in the figure. The classifier selection process and application of the selected detectors only on specific portions of the image frame lead to high efficiency. The image was intentionally blurred due to confidentiality reasons.

predicted at each image location. In this section, we exploit these scene geometry constraints, and show how we use the large pool of deep cascade detectors to effectively capture vehicles in surveillance videos.

A. Online Classifier Selection

Given a fixed-camera surveillance video, our system browses through the large pool of detectors and selects only those that are suitable for each location of the video. The selection process occurs during an auto-calibration mode of the system. In this mode, we initially interleave all the detectors across the temporal domain, i.e., we run detector 1 in frame 1, detector 2 in frame 2, and so on, in a *Round Robin* fashion. The detectors are applied using multi-scale sliding windows over foreground regions obtained by background modeling [2]. For each detector, we keep the information of the number of firings per image location, as well the size of the detected vehicles per location.

This data collection scheme happens until a certain number of firings is reached, when all detectors that don't have a sufficient number of firings according to a threshold are immediately killed. For those detectors that remain alive, we limit their operation to specific regions defined by the convex envelope of their firing locations, which we call a *detector map*, as shown in Figure 4. As a result, only few detectors are selected per video, and they run only at specific regions of the image frames, making the system very efficient.

In addition to creating a detector map for the selected detectors, we also create a *size map* which indicates a range of possible vehicle sizes for each image location. This is done by fitting a plane using least squares with the size data collected when the system is operating in auto-calibration mode. The size map constrains the sliding window search over multiple scales, leading to more efficiency and more accuracy.

B. Application

We are mainly interested in attribute-based object retrieval from surveillance videos. In addition to the novel approach proposed for appearance-based object detection, we have also built higher-level analytics and the search infrastructure to support queries such as "Show me all blue vehicles traveling at high speed northbound, last Saturday, from 2pm to 4pm", similar to the work of Feris et al [1], [5].

More specifically, we use our object detection approach as complementary information to tracking based on foreground regions obtained by background subtraction. As objects are tracked, attributes like color, speed, size, time, and location are extracted and ingested into a database. User queries are then performed using a web-services interface. We refer to [5] for more details about tracking, attribute extraction, and the search infrastructure for this application.

V. Experimental Results

We collected a challenging test dataset containing 700 static images captured from more than 20 surveillance cameras (different from those used for training), showing vehicles in quite different poses and lighting conditions. The images were captured in different months, covering different weather conditions including sunny and rainy days, different lighting effects, such as shadows and specularities, and different periods of time such as morning and evening.

In this experiment, our goal is to evaluate the effectiveness of our proposed representation, as well as the importance of using large amounts of training data for object detection. Therefore we apply the detectors over the entire image in the test set, and run all of them in parallel. This enables us to better quantitatively evaluate the detectors independently of other tasks that improve accuracy and efficiency, such as constraining them to be applied only on foreground regions. As ground-truth, we have manually drawn bounding boxes around vehicles in all 700 images of our challenging test set. A vehicle is considered to be correctly detected if the Euclidean distance between the center of the detected box and the ground-truth is less than 30% of the width of the ground-truth box, and also if the width (i.e., size) of the detected vehicle box is within 50% of the width of the ground-truth box.

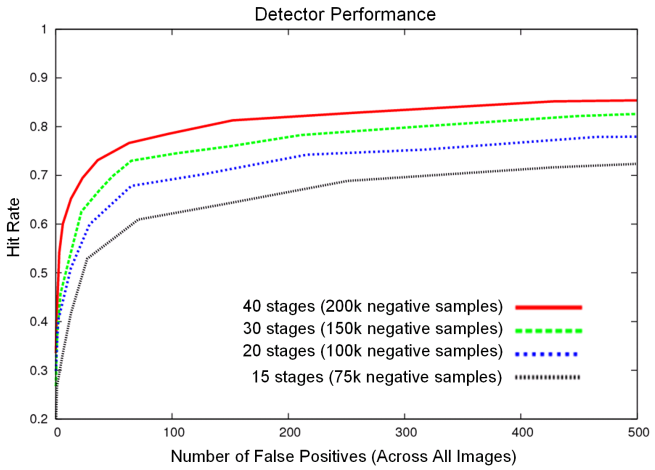

Figure 5. ROC curves of detectors trained with an increasingly large number of bootstrap rounds, ranging from relatively shallow cascades to very deep cascades trained with hundreds of thousands of negative samples.

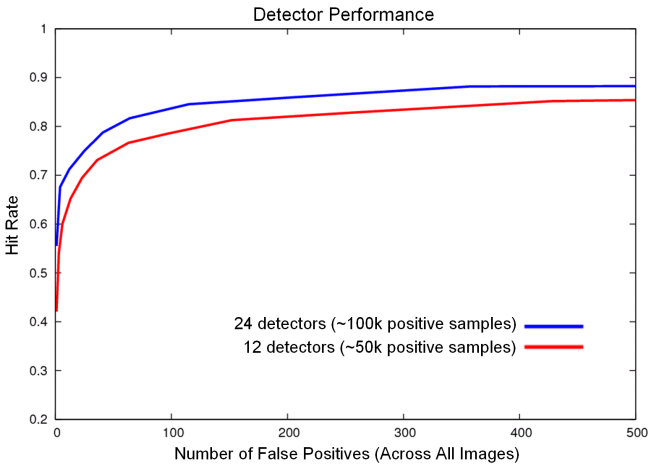

Figure 6. The use of hundreds of thousands of positive examples, through the learning of additional complementary detectors, improve the performance of our system.

We start by showing the importance of training deep cascade detectors with a large amount of selected negative samples. Figure 5 shows the ROC curves of detectors trained with an increasingly large number of bootstrap rounds. The accuracy keeps improving even with very deep cascades containing 40 stages and trained with hundreds of thousands of negative samples. We believe that training even deeper detectors would lead to more accuracy improvement. These ROC curves were generated by running 12 deep cascade detectors in parallel, one for each motionlet cluster. The work of Feris et al [5] corresponds to the curve associated with 20 stages. We got significantly higher performance by using more data and training deeper cascades.

Now we show the importance of training complementary detectors, while using a large amount of positive examples. Figure 6 shows a comparison of 12 detectors ($\sim$50,000 positive training samples; one detector per motionlet cluster) with adding another 12 complementary detectors, for a total of 24 detectors ($\sim$100,000 positive training samples; two detectors per motionlet cluster). Notice the significant performance improvement, especially in the left part of the curve which corresponds to few false alarms.

Finally, we have applied our method to video sequences, as part of our surveillance indexing and retrieval system. Object detection is performed at an impressive average rate of 125 frames per second on a conventional laptop machine (2.3GHz, 3GB of RAM), considering 320x240 resolution. This high speed is justified by several reasons. First, only few detectors are automatically selected per camera view, and interleaved across the temporal domain. Second, the detector sliding windows are applied over foreground regions and constrained by the corresponding detector maps and size maps. Third, the deep cascade detectors are compact, as they model specific submanifolds of the object appearance, and hence require less feature computations.

Figure 7(a) show examples of our vehicle detection results in video frames containing vehicles in different poses and lighting conditions. Figure 7(b) illustrates the interface of our retrieval system, which enables the user to search for vehicles based on semantic attributes. An example of a search for "yellow vehicles from time X to time Y " is demonstrated in Figure 7(c). Our system has been commercialized and deployed in several cities worldwide. We hope to present a live demo of the integrated vehicle detection and retrieval system in the conference.

We note that our system is limited to capture buses, bikes and large trucks, as we have not yet trained detectors for these object classes. In addition, the detector/pose map may prevent capturing anomalous events such as vehicles driving in the opposite direction of the normal flow. Both issues are alleviated in our system by using object detectors just as complementary information for the tracking module to beter carve out foreground blobs obtained by background modeling [5].

VI. Conclusions

We have described a novel framework for learning multi-view object detectors from large datasets, designed specifically for surveillance video indexing and retrieval. Our approach relies on learning a large pool of complementary, compact, deep cascade detectors, using hundreds of thousands of positive and negative examples. At test time, only few detectors are selected per camera view according to the scene geometry. We demonstrated our approach in vehicle detection, using a dataset containing around one million vehicle images.

As future work, we plan to extend our approach to other object classes, such as pedestrians and boats.

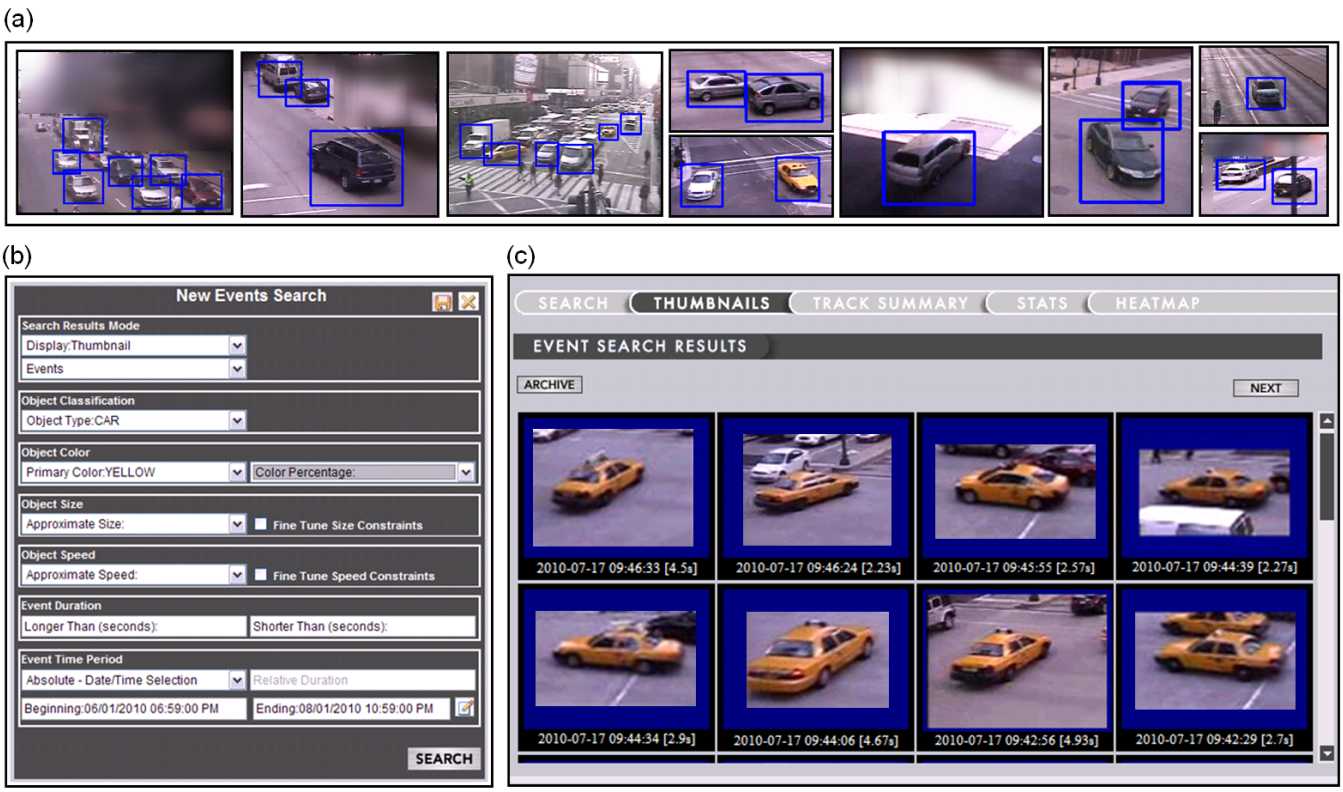

Figure 7. (a) Sample detection results (Some images were intentionally blurred due to confidentiality reasons). Note the large variations in vehicle pose and lighting conditions. (b) Search interface, which allows the user to search for vehicles based on semantic attributes. (c) Example of search results for yellow cars from a specific time period.

REFERENCES

[1] R. S. Feris, B. Siddiquie, Y. Zhai, J. Petterson, L. Brown, and S. Pankanti, "Attribute-based vehicle search in crowded surveillance videos," in *ICMR*, 2011.

[2] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *CVPR*, 1998.

[3] P. Felzenszwalb, R. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *CVPR*, 2010.

[4] B. Wu and R. Nevatia, "Cluster boosted tree classifier for multi-view, multi-pose object detection," in *ICCV*, 2007.

[5] R. S. Feris, B. Siddiquie, J. P. andY. Zhai, A. Datta, L. Brown, and S. Pankanti, "Large-scale vehicle detection, indexing, and search in urban surveillance," *IEEE Transactions on Multimedia*, 2011.

[6] P. Viola and M. Jones, "Robust Real-time Object Detection," in *International Journal of Computer Vision*, 2004.

[7] W. Zheng and L.Liang, "Fast car detection using image strip features," in *CVPR*, 2009.

[8] H. Schneiderman and T. Kanade, "A statistical approach to 3D object detection applied to faces and cars," in *CVPR*, 2000.

[9] O. Russakovsky and L. Fei-Fei, "Attribute learning in large-scale datasets," in *ECCV 2010 Workshop on Parts and Attributes*, 2010.

[10] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: a large dataset for non-parametric object and scene recognition," *IEEE Transactions on PAMI*, vol. 30, no. 11, 2008.

[11] S. A. O. Javed and M. Shah, "Online detection and classification of moving objects using progressively improving detectors," in *CVPR*, 2005.

[12] P. Roth, H. Grabner, D. Skocaj, H.Bischol, and Leonardis, "On-line conservative learning for person detection," in *PETS Workshop*, 2005.

[13] Y. LeCun, K. Kavukvuoglu, and C. Farabet, "Convolutional networks and applications in vision," in *ISCAS*, 2010.