A General Active-Learning Framework for On-Road Vehicle Recognition and Tracking

Sayanan Sivaraman, *Member, IEEE*, and Mohan Manubhai Trivedi, *Fellow, IEEE*

***Abstract*—This paper introduces a general active-learning framework for robust on-road vehicle recognition and tracking. This framework takes a novel active-learning approach to building vehicle-recognition and tracking systems. A passively trained recognition system is built using conventional supervised learning. Using the query and archiving interface for active learning (QUAIL), the passively trained vehicle-recognition system is evaluated on an independent real-world data set, and informative samples are queried and archived to perform selective sampling. A second round of learning is then performed to build an active-learning-based vehicle recognizer. Particle filter tracking is integrated to build a complete multiple-vehicle tracking system. The active-learning-based vehicle-recognition and tracking (ALVeRT) system has been thoroughly evaluated on static images and roadway video data captured in a variety of traffic, illumination, and weather conditions. Experimental results show that this framework yields a robust efficient on-board vehicle recognition and tracking system with high precision, high recall, and good localization.**

***Index Terms*—Active safety, computer vision, intelligent driver-assistance systems, machine learning.**

I. Introduction

WORLDWIDE automotive accidents injure between 20 and 50 million people each year, and at least 1.2 million people die as a result of them. Between 1% and 3% of the world's domestic product is spent on medical care, property damage, and other costs that are associated with auto accidents [38]. As a result, over the years, there has been great interest in the development of active safety systems among vehicle manufacturers, safety experts, and academics.

The design of active safety systems presents many difficult challenges. A key requirement of active safety systems is that they accurately, reliably, and efficiently identify dangerous conditions [32]. Often, we would like an active safety system to help avoid collisions by detecting lane departures [21], pedestrians [10], or other vehicles [11].

It is widely recognized that computer vision is a critical technology for the development of intelligent vehicles [32]. The development of vision-based on-road vehicle detectors for active safety is an open area of research [30]. Vision-based vehicle recognition has the potential to deliver information in an intuitive way for human drivers to understand [33].

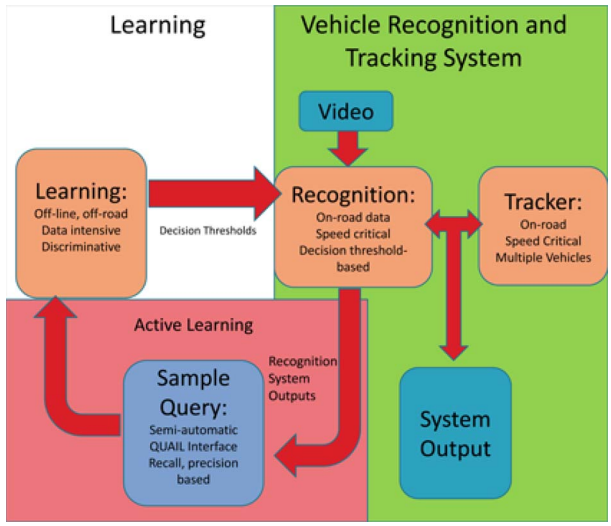


Fig. 1. ALVeRT. The general active learning framework for vehicle recognition and tracking systems consists of (white and red) an offline learning portion and (green) an online implementation portion. (Red) Prior works in vehicle recognition and tracking have not utilized active learning.

Robust recognition of other vehicles on the road using vision is a challenging problem. Highways and roads are dynamic environments, with ever-changing backgrounds and illuminations. The ego vehicle and other vehicles on the road are generally in motion, and therefore, the sizes and the locations of vehicles in the image plane are diverse. There is high variability in the shape, size, color, and appearance of vehicles found in typical driving scenarios [30].

Although active learning for object recognition has been an area of great recent interest in the machine-learning community [15], no prior research study has used active learning to build an on-road vehicle recognition and tracking system. In this paper, a general framework for robust active-learning-based vehicle recognition and tracking is introduced. The vehicle-recognition system has been learned in two iterations using the active-learning technique of selective sampling [6] to query informative examples for retraining. Using active learning yields a significant drop in false positives per frame and false-detection rates, while maintaining a high vehicle-recognition rate. The robust on-road vehicle-recognition system is then integrated with a condensation [13] particle filter, which is extended to multiple-vehicle tracking [17], to build a complete vehicle-recognition and tracking system. A general overview of the complete framework can be seen in Fig. 1.

Manuscript received May 9, 2009; revised July 16, 2009 and September 28, 2009; accepted December 3, 2009. Date of publication February 17, 2010; date of current version May 25, 2010. This work was supported in part by the University of California Discovery Grant and in part by the Electronics Research Laboratory, Volkswagen. The Associate Editor for this paper was L. Li.

The authors are with the Laboratory for Intelligent and Safe Automobiles, University of California, San Diego, La Jolla, CA 92039 USA (e-mail: ssivaram@ucsd.edu; mtrivedi@ucsd.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2010.2040177

The main novelty and contributions of this paper include the following. A general active-learning framework for on-road vehicle recognition and tracking is introduced. Using the introduced active-learning framework, a full vehicle-recognition and tracking system has been implemented, and a thorough quantitative performance analysis has been presented. The vehicle-recognition and tracking system has been evaluated on both real-world video and public-domain vehicle images. In this paper, we introduce new performance metrics for assessing on-road vehicle recognition and tracking performance, which provide a thorough assessment of the implemented system's recall, precision, localization, and robustness.

The rest of this paper is organized as follows. In Section II, we present a brief overview of recent related works in vehicle recognition and active learning. In Section III, we review active-learning concepts. In Section IV, we detail the active-learning framework for vehicle recognition. In Sections V and VI, we detail the classification and tracking algorithms. In Section VII, we provide experimental studies and performance analysis. Finally, in Section VIII, we provide concluding remarks.

II. Related Research

Here, we present a brief overview of two categories of papers that are relevant to the research presented in this paper. The first set of papers deals with vehicle detection and tracking. The second set of papers deals with active learning for object recognition.

A. Vehicle Detection and Tracking

Vision-based vehicle detection is an active area of research in the intelligent transportation systems community [30]. In the literature, many studies have performed experimental validation on static images [12], [14], [25], [31], [36].

A statistical approach has been used in [36], performing vehicle detection using principal component analysis (PCA) and independent component analysis (ICA) to do classification on a statistical model and increased its speed by modeling the PCA and ICA vectors with a weighted Gaussian mixture model [36]. This methodology showed very strong performance on static images of parked vehicles but had slow execution times, and the study limited its scope to sedans.

A neural network approach has been used in [14]. In [14], a multilayer feedforward neural-network-based approach was presented for vehicle detection, with the linear output layer replaced by a Mahalanobis kernel. This paper showed strong fast results on precropped illumination-normalized 32×32 image regions.

A support-vector-machine (SVM) approach was used in [31]. Sun *et al.* [31] built multiple detectors using Gabor filters, Haar wavelets, PCA, truncated wavelets, and a combination of Gabor and wavelet features using neural networks and SVM classifiers. A thorough comparison of feature and classifier performance was presented, with the conclusion that the feature fusion of Haar and Gabor features can result in robust detection. Results were presented for precropped 32×32 pixel illumination-normalized image regions.

A similar feature analysis was performed in [25]. Negri *et al.* [25] compared the performance vehicle detectors with Adaboost classification trained using Haar-like features, a histogram of oriented gradient features, and a fusion of the two feature sets. In this paper, it was also found that a feature fusion can be valuable. Results were presented for static images.

Haselhoff *et al.* [12] tested the performance of classifiers that are trained with images of various resolutions, from the smallest resolution of 18×18 pixels to the largest of 50×50 pixels, using Haar-like feature extraction and Adaboost, discussing the tradeoff between classifier performance and training time as a function of training image resolution. Results were presented on static images. Such a signal-theoretic analysis of Haar features is invaluable to the study of on-road vehicle detection.

Khammari *et al.* [16] implemented a detector that first applied a three-level Gaussian pyramid and used control point features classified by Adaboost and tracked using an Adaboost-based cascaded tracking algorithm. Quantitative analysis was reported for the vehicle detector precropped image subregions. The full detection and tracking system was then implemented on-road.

A growing number of on-road vehicle studies are reporting results for video data sets [3]–[5]. Arrospide *et al.* [3] performed detection and tracking that evaluated the symmetry-based quality of the tracking results. While tracking based on symmetry metric sounds to be a promising idea, no quantitative performance analysis was provided in this paper.

Chan *et al.* [5] built a detector that used vertical and horizontal symmetry, as well as taillight cues in a statistical model for detection, and tracked detected vehicles on the road using particle filtering. This paper illustrated the great potential for the use of particle filters in robust on-road vehicle recognition and tracking. Results were presented for on-road video sequences, but false-alarm rates were not reported.

Cao *et al.* [4] implemented a monocular vehicle tracker based on optimized optical flow using a 3-D pulse-coupled neural network. As this method relies on optical flow, the performance of the system is heavily dependent on the speed difference between the ego vehicle and the target vehicle, as evidenced by the results. This paper did not report false-alarm rates.

Recently, visual detectors have been used in fusion schemes with radar and laser. Alessandretti *et al.* [2] implemented a preceding vehicle and guardrail detector using radar and video fusion. In [37], a fusion of laser scanning and vision-based vehicle detection was implemented. The laser scanner estimated the distance and the orientation of vehicles on the road, and the visual detector either ran a rear/front or side vehicle detector. The performance of the fusion scheme was reported to be better than either detection scheme by itself. In [24], the fusion of laser scanning and vision-based vehicle detection was also implemented. This paper also used an Adaboost-based tracker for more robust results.

B. Active Learning for Object Recognition

Active learning for object recognition has gotten more popular [1], [8], [15], [27], [29], [34]. Active learning has been used to improve classifier performance by reducing false alarms [1],

TABLE I
SELECTED ACTIVE-LEARNING-BASED OBJECT RECOGNITION APPROACHES

Research Study	Feature Extraction	Learning, Classification	Selective Sampling Query	Target Object
Abramson and Freund [1], 2005.	Control Points	Adaboost	SEVILLE Visual Interface	Pedestrians
Kapoor et al. [15] 2007	SIFT+PCA	SVM	Probabilistic Selective Sampling	Various Objects
Enzweiler and Gavrila [8], 2008.	Haar Wavelets	SVM	Probabalistic Selective Sampling	Pedestrians
Roth and Bischof [27], 2008.	Haar Wavelets	Online Boosting	Manual Initialization+ Tracking	Faces
Vijayanarasimhan and K Grauman [34], 2008.	Local features	SVM	Semi-automatic Annotation-based Selective Sampling	Various Objects
This study, 2009.	Haar Wavelets	Adaboost	QUery and Archiving Interface for active Learning [QUAIL] Visual Interface	Vehicles

[6], [29], to increase a classifier's recall [27], to semiautomatically generate more training data [1], [8], [29], and to perform training with fewer examples [6], [20], [27].

Recently, Enzweiler and Gavrila [8] have used active learning to train a pedestrian detector. A generative model of pedestrians was built, from which training examples were probabilistically selectively sampled for training. In addition, the generative pedestrian models were used to synthesize artificial training data to enhance the pedestrian detector's training set [8].

Roth and Bischof [27] used a manually initialized tracker to generate positive training samples for online active learning to develop a face detector that outperformed the face detector that is trained using passive learning.

Abramson and Freund [1] used selective sampling to drastically reduce the false-positive output by a pedestrian detector using iterative training with Adaboost and control point features. Table I contains a summary of recent active-learning-based object-recognition studies.

III. ACTIVE LEARNING: MOTIVATION

A. Overview

Passive learning consists of the conventional supervised learning of a binary classifier learned from labeled "positive" examples and random "negative" examples. To build a vehicle-recognition system in this manner, the positive training examples consist of vehicles, and the negative training examples consist of random nonvehicles.

Active learning is a general term that refers to a paradigm in which the learning process exhibits some degree of control over the inputs on which it trains. Active learning has been shown to be more powerful than learning from random examples in [6], [20], and [27]. In vision-based object recognition tasks, a common approach is selective sampling [8], [15], [34]. Selective sampling aims to choose the most informative examples train a discriminative model. Cohn *et al.* [6] demonstrated that selective sampling is an effective active-learning technique by sequentially training robust neural network classifiers. Li and Sethi [20] used a confidence-based sampling to train robust SVM classifiers.

Training aims to build a system that correctly recognizes vehicles in video frames by learning the decision threshold between vehicles and nonvehicles. Consider the concept of a vehicle $v(s) = 1$ as an image subregion, with s classified as a vehicle and $v(s) = 0$ as an image subregion classified as a nonvehicle. A concept is *consistent* with a training example s if $v(s) = t(s)$, which is the true class of s. Consider the set S^m consisting of m training examples that are used in the initial passive training. Assume that all the examples from S^m are consistent with concept v, i.e., the training error is zero. In classification tasks, there exists a region of uncertainty $R(S^m)$, where the classification result is not unambiguously defined. This is to say that the discriminative model can learn a multitude of decision thresholds for the given training patterns but disagrees in certain regions of the decision space [8]. Areas that are *not* determined by the model that is trained using S^m are of interest for selective sampling, as they constitute more informative examples [6]. Given a random test sample x with true class $t(x)$ and training data S^m, we define the region of uncertainty $R(S^m)$ as follows:

$$R(S^m) = \{x : \exists v, v \text{ is consistent with all } s \in S^m \text{ and } v(x) \neq t(x)\}. \quad (1)$$

It is of note that both the trained decision boundary and the region of uncertainty are a function of the training data S^m. In general, the decision boundary or threshold between positive examples and negative examples resides in $R(S^m)$, which is the region of uncertainty. If we use a point that lies outside of $R(S^m)$ to update the classifier, the classifier will remain unchanged. If we use a point inside the region of uncertainty, the region of uncertainty will be reduced [6]. As the region of uncertainty is reduced, the classifier becomes less likely to report false positives and gains precision.

B. Implementation

Representing $R(S^m)$ exactly is generally a difficult, if not impossible, task. A good approximation of the region of uncertainty is a superset, i.e., $R^+(S^m) \supseteq R(S^m)$, as we can selectively sample from $R^+(S^m)$ and be assured that we do

not exclude any part of the domain of interest [6]. This is the approach that is taken in many object-recognition studies [8], [15], [29], [34].

In general, active learning consists of two main stages: an initialization stage and a stage of query and retraining [20]. In the initialization stage, a set of training examples is collected and annotated to train an initial classifier. This is the same process as passive learning [6], [27]. Once an initial classifier has been built, a query function is used to query unlabeled examples, and a human or ground-truth mechanism is used to assign a class label to the queried examples. The newly labeled training examples are then used to retrain the classifier [20].

The query function is central to the active-learning process [20] and generally serves to select difficult training examples, which are informative in updating a decision boundary [6]. In [8], this was achieved by building a generative model of the classes, and samples with probabilities that are close to the decision boundary were queried for retraining. In [20], confidence outputs from SVM classifiers were used to map to error probabilities, and those examples with high error probabilities were queried for retraining.

IV. Active-Learning-Based Vehicle Recognition and Tracking

A. Initialization

The initial passively trained Adaboost [9] cascaded classifier was trained using 7500 positive training images and 20 500 negative training images. The passively trained cascade consisted of 30 stages. The positive training images were collected from many hours of real driving data on San Diego highways in the Laboratory for Intelligent and Safe Automobiles Infiniti Q45 (LISA-Q) testbed [22]; the testbed is described in Section VII-D.

B. Query and Retraining

Efficient visual query and archival of informative examples has been performed via the query and archiving interface for active learning (QUAIL). The interface is able to evaluate the vehicle recognition system on a given random video frame, marking all system outputs and allowing the user to quickly tag false positives and missed vehicles. The interface then archives missed vehicles and true positives as positive training examples, and false positives as negative training examples, performing selective sampling in an intuitively visual user-friendly efficient manner.

Strictly speaking, only missed detection and false positives are known to lie within the region of uncertainty, which we call $R(S^m)$. As it is not possible to exactly represent the region of uncertainty, including correctly identified vehicles in the archived training data ensures that the data comprise a superset of the region of uncertainty. Maintaining a superset $R^+(S^m)$ of the region of uncertainty helps protect against oversampling one part of the domain [6]. We are assured that we are not excluding examples from the domain of interest but acknowledge that we retrain on some positive examples that are not of interest.

(a) (b)

Fig. 2. (a) QUAIL. QUAIL evaluates the passively trained vehicle-recognition system on real-world data and provides an interface for a human to label and archive the ground truth. Detection is automatically marked green. Missed detection is marked red by the user. False positives are marked blue by the user. True detection is left green. (b) QUAIL outputs. A true detection and a false positive archived for retraining.

Fig. 3. Examples of false-positive outputs queried for retraining using QUAIL.

Fig. 4. Examples of true positives queried for retraining using QUAIL.

Fig. 2(a) shows a screenshot of the QUAIL, and Fig. 2(b) shows the corresponding cropped true detection and false-positive image regions. Fig. 3 shows false positives that were queried and archived for retraining using QUAIL. Fig. 4 shows true detection that was queried and archived for retraining using QUAIL.

For the retraining, we had 10 000 positive images and 12 172 negative images. The negative training images consisted exclusively of false positives from the passively trained detector output. Adaboost training was used to build a cascade of 20 stages. Fig. 5 shows a schematic of the general framework for training an active-learning-based vehicle detector.

V. Vehicle Recognition Using Rectangular Features and an Adaboost Classifier

For the task of identifying vehicles, a boosted cascade of simple Haar-like rectangular features has been used, as was introduced by Viola and Jones [35] in the context of face detection. Various studies have incorporated this approach into on-road vehicle-detection systems such as [12], [26], and [37]. The set of Haar-like rectangular features is well suited to

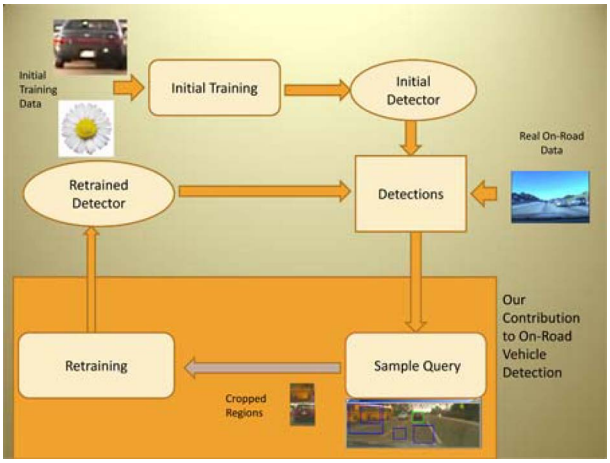
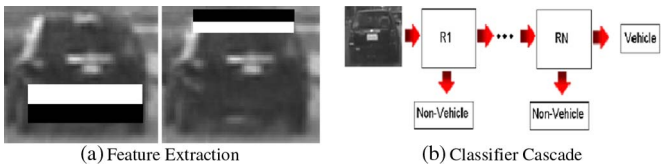


Fig. 5. Schematic of a framework for ALVeRT. An initial passively trained vehicle detector is built. Using QUAIL, false positives, false negatives, and true positives are queried and archived. A new classifier is trained using the archived samples.



(a) Feature Extraction (b) Classifier Cascade

Fig. 6. (a) Examples of Haar-like features used in the vehicle detector. (b) Cascade of boosted classifiers.

the detection of the shape of vehicles. Rectangular features are sensitive to edges, bars, vertical and horizontal details, and symmetric structures [35]. Fig. 6(a) shows examples of Haar-like rectangular features. The algorithm also allows for rapid object detection that can be exploited in building a real-time system, partially due to fast and efficient feature extraction using the integral image [35]. The resulting extracted values are effective weak learners [35], [37], which are then classified by Adaboost.

Adaboost is a discriminative learning algorithm, which performs classification based on a weighted majority vote of weak learners [9]. We use Adaboost learning to construct a cascade of several binary classifier stages. The earlier stages in the cascade eliminate many nonvehicle regions with very little processing [35]. The decision rule at each stage is made based on a threshold of scores that are computed from feature extraction. Each stage of the cascade reduces the number of vehicle candidates, and if a candidate image region survives until it is output from the final stage, it is classified as a positive detection [37]. Fig. 6(b) shows a schematic of the cascade classifier.

VI. Vehicle Tracking With a Condensation Filter

We integrate a particle filter for vehicle tracking. The probability densities of possible predictions of the state of the system are represented by a randomly generated set, and multiple hypotheses are used to estimate the density of the tracked object. The original condensation algorithm was designed to track one object. First, a random sample set of hypotheses is generated, based on the previous sample state and a representation of the

Fig. 7. (Left) Detector outputs for (top) a single vehicle and (middle and bottom) multiple vehicles. (Middle) Multiple-location hypotheses generated by the condensation filter. Note the multimodal distribution of the hypotheses when (bottom) tracking multiple vehicles. (Right) Best tracking results, as confirmed by detection in the consequent frame.

previous stage's posterior probability, i.e., $p(x_t|Z_{t-1})$, where x is the state, and Z is the set of all observed measurements. Then, the random samples are used to predict the state using a dynamic system model. Finally, a new measurement is taken, and each of the multiple position hypotheses is weighted, yielding a representation of the observation density $p(z_t|x_t)$ [13].

The basic condensation algorithm was not designed to track an arbitrarily changing number of objects. Koller-Meier and Ade [17] proposed extensions to the algorithm to allow for tracking of multiple objects and to track objects entering and leaving the camera's field of view (FOV) using one condensation tracker. Maintaining the tracks of multiple objects is achieved by including a representation of the probability distribution of all tracked objects in the condensation tracker itself, as given by the following:

$$p(x_t) = \sum_i \alpha^{(i)} p^{(i)}(x_t). \tag{2}$$

During tracking, all tracked objects are given equal weightings $\alpha^{(i)}$ to ensure that the sample set does not degenerate [17]. To track newly appearing objects into the condensation tracker, the observed measurements are directly integrated into the sampling. An initialization density $p(x_{t-1}|z_{t-1})$, which is a representation of the probability of the state at time $t-1$ given just one measurement, is calculated and combined with the posterior density from the previous step, as shown by

$$p'(x_{t-1}|Z_{t-1}) = (1-\gamma)p(x_{t-1}|Z_{t-1}) + \gamma p(x_{t-1}|z_{t-1}) \tag{3}$$

$$\gamma = \frac{M}{N}. \tag{4}$$

In this implementation of the condensation algorithm, $N-M$ samples are drawn from the representation of the previous stage's posterior density, and M new samples are drawn from the initialization density [17]. This method ensures that there is a reinitialization of the probability density every time there is a new measurement observed, allowing for very fast convergence of the tracks [5], [13]. Fig. 7 shows example detection, multiple particle-tracking hypotheses, and

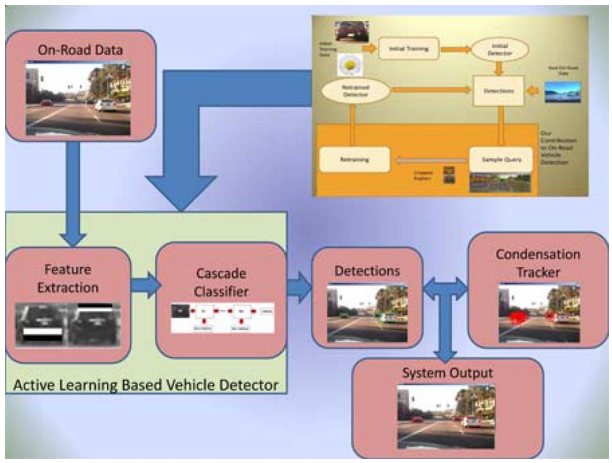


Fig. 8. Full ALVeRT system overview. Real on-road data are passed to the active-learning-based vehicle-recognition system, which consists of Haar-like rectangular feature extraction and the boosted cascade classifier. Detection is then passed to the condensation multiple-vehicle tracker. The tracker makes predictions, which are then updated by the detection observations.

TABLE II
DATA SETS USED IN THIS STUDY

Dataset	Description
Caltech Vehicle Image 1999	This dataset consists of 126 distinct static images of vehicles. The dataset is publicly available, and has been used in research studies such as [36] and [31]. The image dataset can be accessed at Caltech's Computational Vision website, http://www.vision.caltech.edu/archive.html
LISA-Q Front FOV Video Datasets 1-3	The LISA-Q Front FOV Datasets are videos taken from the LISA-Q testbed [22]. Dataset 1 consists of 1600 consecutive frames, captured during sunny evening rush hour traffic. Dataset 2 consists of 300 consecutive frames, captured on a cloudy morning on urban roads. Dataset 3 consists of 300 consecutive frames, captured on a sunny afternoon on highways. The datasets will be available for academic and research communities, pending approvals, at the Laboratory for Intelligent and Safe Automobiles website, http://cvrr.ucsd.edu/LISA/index.html

the best tracking hypotheses, confirmed by the consequent vehicle-detection observations. Fig. 8 shows a flowchart depicting the active-learning-based vehicle recognition and tracking (ALVeRT) system overview.

VII. EXPERIMENTAL EVALUATION

A. Experimental Datasets

Two main data sets were used to quantify the performance of vehicle recognition. The first data set, which is the publicly available Caltech 1999 data set, consists of 126 distinct static images of rear-facing vehicles.

To test the full ALVeRT system, video data sets are necessary. The second data set, which is the LISA-Q Front FOV data set, consists of three video sequences, consisting of 1600, 300, and 300 consecutive frames, respectively. Table II briefly describes the data sets that are used in this paper.

B. Performance Metrics

The performance considerations we present consist of the following: precision, recall, localization, robustness, efficiency, and scalability.

Many vehicle-detection studies test their classifiers on static images by first cropping down test samples into candidate subregions, normalizing the illumination and contrast, and then quantifying the performance of the classifier as a binary classifier [14], [31]. This evaluation procedure can report lower false-positive rates and higher detection rates than the system would output on full consecutive video frames because the system is presented with a fewer vehicle-like regions than if they are required to search through a video frame. This evaluation does not give information about localization or robustness because test image intensities and contrasts have been normalized [14], [31] and does not indicate scalability.

Other studies do not crop down test images for validation but quantify their true negatives as being all image subregions that are not identified as false positives [12], [36]. For an $n \times n$ image, there are potentially n^4 rectangular subregions [19]. Using this definition of a false positive, studies on vehicle detection present false-positive rates on the order of 10^{-5} [12], [36]. Such false-positive rates have little practical meaning. This evaluation method gives a practical assessment of recall, precision, and efficiency but not of robustness, localization, or scalability.

Recent vehicle tracking papers either do not offer numerical evaluation of their systems [3] or provide numerical values for successfully tracked vehicles but do not provide counts of erroneously tracked objects [4], [5]. Such evaluations do indicate recall and efficiency but do not provide information on precision, scalability, localization, and robustness.

In our research, the performance of a detection module is quantified by the following metrics: true positive rate, false detection rate, average false positives per frame, average true positives per frame, and false positives per vehicle.

The true positive rate TPR is the percentage of nonoccluded vehicles in the camera's view that are detected. TPR is assessed by dividing the number of truly detected vehicles by the total number of vehicles. This takes into account the vehicles preceding the ego-vehicle and those in adjacent lanes. This quantity measures recall and localization. TPR is defined by

$$TPR = \frac{\text{detected vehicles}}{\text{total number of vehicles}}. \tag{5}$$

The false detection rate FDR is the proportion of detection that were not true vehicles. We assess the FDR by dividing the number of false positives by the total number of detections. This is the percentage of erroneous detection. FDR is a measure of precision and localization; it is defined by

$$FDR = \frac{\text{false positives}}{\text{detected vehicles} + \text{false positives}}. \tag{6}$$

The average false positives per frame quantity, i.e., average $FP/Frame$, describes how susceptible a detection module is to false positives and gives an informative measure of the

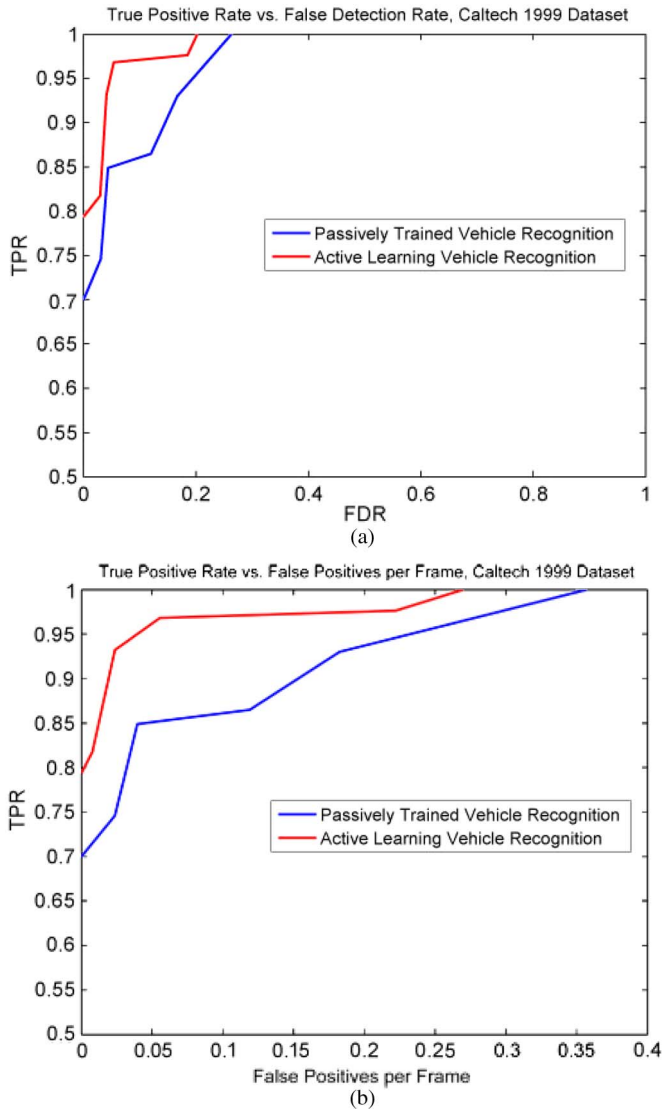


Fig. 9. (a) Plot of TPR versus FDR (Caltech 1999 data set) for the (blue) passively trained recognition and (red) the active-learning vehicle recognition. We note the improvement in performance due to active learning on this data set. (b) Plot of TPR versus the number of false detections per frame (Caltech 1999 data set) for (blue) the passively trained recognition and (red) the active-learning vehicle recognition. We note the reduction in false positives due to active learning.

Fig. 10. Sample vehicle recognition results from the Caltech 1999 data set.

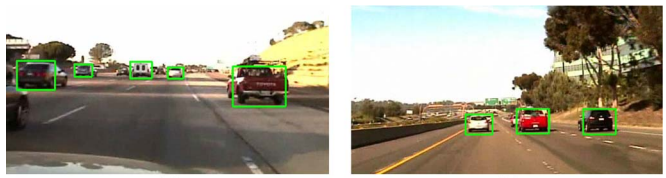
credibility of the system. This quantity measures robustness, localization, and scalability. $FP/Frame$ is defined by

$$\text{Average } FP/Frame = \frac{\text{false positives}}{\text{total number of frames processed}}. \quad (7)$$

(a) (b)

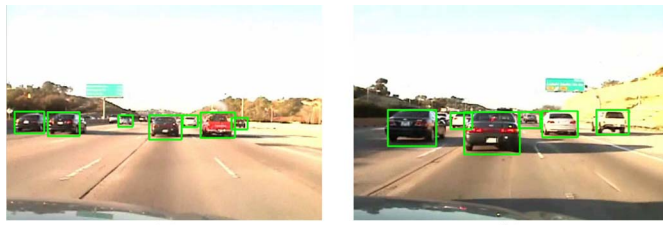
Fig. 11. (a) Recognition output in a nonuniformly illuminated scene. Six vehicles were detected. No false positives and no missed detections. (b) Recognition output in cloudy conditions. Note the reflections, glare, and smudges on the windshield.

Fig. 12. (Left) Vehicle in front was not detected in this frame on a cloudy day due to smudges and dirt on the windshield. (Right) Vehicle's track was still maintained in the following frame.



(a) (b)

Fig. 13. (a) Recognition output in shadows. Five vehicles were detected, and two were missed due to their distance and poor illumination. We note that poor illumination seems to limit the range of the detection system; vehicles farther from the ego vehicle are missed. (b) Vehicle recognition output in sunny highway conditions.



(a) (b)

Fig. 14. (a) Recognition output in sunny conditions. Note the vehicle detection across all six lanes of traffic during San Diego's rush hour. The recognizer seems to work best in even sunny illuminations, as such illumination conditions have less scene complexity compared with scenes with uneven illuminations. (b) Recognition output in dense traffic. We note that the vehicle directly in front is braking, and the system is aware of vehicles in adjacent lanes.

The average false positives per object $FP/Object$ describes, on average, how many false positives are observed. This quantity measures robustness. This performance metric was first used in [28]. It is defined by

$$\text{Average } FP/Object = \frac{\text{false positives}}{\text{true vehicles}}. \quad (8)$$

TABLE III
EXPERIMENTAL DATA SET 1: JANUARY 28, 2009, 4 P.M., HIGHWAY, SUNNY

Recognition/Tracking System	TPR	FDR	FP/Frame	TP/Frame	FP/Object
Passively Trained Vehicle Recognition	89.5%	51.1%	4.2	4.1	0.94
Active Learning Vehicle Recognition	93.5%	7.1%	0.32	4.2	0.07
ALVeRT	95.0 %	6.4%	0.29	4.2	0.06

TABLE IV
EXPERIMENTAL DATA SET 2: MARCH 9, 2009, 9 A.M., URBAN, CLOUDY

Recognition/Tracking System	TPR	FDR	FP/Frame	TP/Frame	FP/Object
Passively Trained Vehicle Recognition	83.5%	79.7%	4.0	1.0	3.3
Active Learning Vehicle Recognition	80.2%	41.7%	0.72	0.98	0.57
ALVeRT	91.7 %	25.5%	0.39	1.14	0.31

TABLE V
EXPERIMENTAL DATA SET 3: APRIL 21, 2009, 12.30 P.M., HIGHWAY, SUNNY

Recognition/Tracking System	TPR	FDR	FP/Frame	TP/Frame	FP/Object
Passively Trained Vehicle Recognition	98.1%	45.8%	2.7	3.16	0.83
Active Learning Vehicle Recognition	98.8%	10.3%	0.37	3.18	0.11
ALVeRT	99.8 %	8.5%	0.28	3.17	0.09

The average true positives per frame $TP/Frame$ describes how many true vehicles are recognized on average. This quantity indicates robustness. It is defined by

$$\text{Average } TP/Frame = \frac{\text{true positives}}{\text{total number of frames processed}}. \tag{9}$$

The overall performance of the system, including efficiency, indicates how scalable a system is. If the system performs with high precision, high recall, good localization, and in a robust and efficient manner, it is a viable system to be used as part of a larger more sophisticated framework.

The performance metrics give an informative assessment of recall, precision, localization, and robustness. We have formally defined the metrics in (5)–(9).

C. Static Images: Caltech 1999 Dataset

We first evaluate the passively trained vehicle recognition and active-learning-based vehicle recognition systems on the publicly available Caltech 1999 data set. The data set consists of 126 static images of vehicles. We note that using this data set, we are not able to evaluate the full ALVeRT system. Fig. 9(a) and (b) plots TPR versus FDR and $FP/Frame$. We note that the active-learning-based vehicle-recognition performance was stronger than the passively trained recognition system. Fig. 10 shows sample recognition results on this data set. It is of note that on this data set, our vehicle detection achieves a lower false positive per frame rate when compared with [36].

D. Video Sequences: LISA-Q Front FOV Datasets

In the LISA, on-road data are captured daily and archived as part of ongoing studies into naturalistic driving and intelligent driver-assistance systems. The data used in this paper were captured in the LISA-Q testbed, which has synchronized the capture of vehicle controller area network data, Global Positioning System, and video from six cameras [32]. The video from the front-facing camera comprises the LISA-Q Front FOV data sets.

LISA-Q Front FOV 1 was captured around 4 P.M. on January 28, 2009, during San Diego's rush hour. There were many vehicles on the road performing complex highway maneuvers. It was a sunny evening; however, at this time of the year, the sun was within an hour or so of setting. The time of day and geographical attributes like hills and canyons result in complex illumination and shadowing. Poor illumination can limit the range of detection, and complex shadows can result in false positives. The ALVeRT system robustly performed in this difficult environment.

LISA-Q Front FOV 2 was taken around 9 A.M. on March 9, 2009, on an urban road in La Jolla, CA. The combined early morning and clouds resulted in poor illumination conditions. In addition, the vehicle's windshield is smudged/dirty. The active-learning-based recognizer did not perform as well on this data set as on data sets 1 and 3. The full ALVeRT system performed significantly better than the vehicle recognition system alone because the condensation particle tracker maintains tracks from frame to frame. Inconsistent vehicle detection from the active-learning-based recognizer can still result in smooth vehicle tracks in the condensation tracking framework, without introducing false positives that result in erroneous tracks. Fig. 11(b) shows vehicle recognition outputs from data set 2, and Fig. 12 shows the tracking output.

LISA-Q Front FOV 3 was captured around 12:30 P.M. on April 21, 2009, on a highway. The weather was sunny and clear. These conditions can be thought of as ideal. We note that the active-learning-based recognizer had a much better false positive-per-frame rate than the passively trained recognizer. Fig. 13(b) shows detection results from this data set.

We note that the passively trained recognizer is more susceptible to false positives, with FDR of over 45% in each of the three data sets. This means that almost one out of every two detections is indeed erroneous. In addition, $FP/Frame$ between 2.7 and 4.2 false positives per frame indicates that the system is generally not credible. However, the passively trained

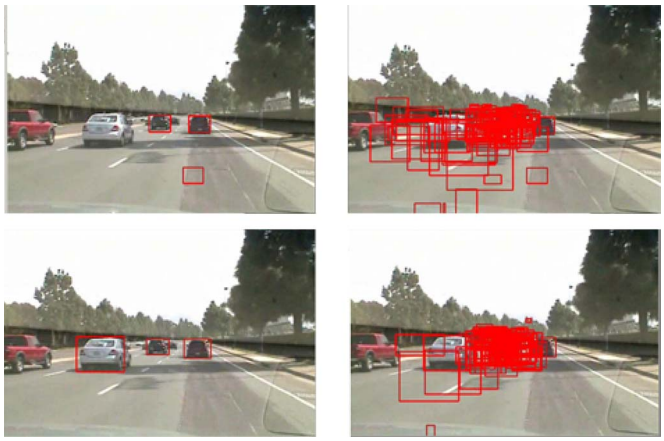


Fig. 15. (Top left) Vehicle-recognition system has output two true vehicles, one missed vehicle, and one false positive. (Top right) These detection outputs are passed to the tracker. We note that the missed vehicle still has a track maintained and is being tracked despite the missed detection. (Bottom left) In the following frame, all three vehicles are detected. (Bottom right) In the corresponding tracker output, we note that a track was not maintained for the false positive from the previous frame.

recognizer had quite a high detection rate, which indicates that the Haar-like wavelet feature set has potential use in effective on-road vehicle recognition and validates the initialization approach (see Fig. 14).

The active-learning-based recognizer had a higher detection rate than the passively trained recognizer and an impressive reduction in the false-detection rate. We note that the FDR value depends both on the detection rate and the false positives that are generated. This classifier produced about one false positive every three frames, achieving a significant reduction in $FP/Frame$. The dramatic reductions in false-positive rates are a result of selective sampling. We have trained the classifier using difficult negative training examples that are obtained using the QUAIL system detailed in Section IV.

Integrating multivehicle tracking improved the performance of the overall system, with increased detection rates and fewer false positives per frame than the recognizer alone. Tables III–V detail the performance of each system on each of the test data sets.

To discuss scalability, it is useful to use tracking as an example of scaling up a system. If a detector outputs between 2.7 and 4.2 false positives per frame, those false positives can create erroneous tracks. This means that the passively trained vehicle recognizer is not scalable. However, the active-learning-based recognizer outputs between 0.32 and 0.72 false positive per frame, which is not consistent enough for a tracker to create erroneous tracks. Thus, the active-learning-based recognition system is scalable, as shown by the performance of the full detection and tracking system. This is demonstrated in Fig. 15.

Fig. 16(a) shows plots of TPR versus FDR for each of the three systems per frame. Fig. 16(b) shows plots of TPR versus $FP/Frame$.

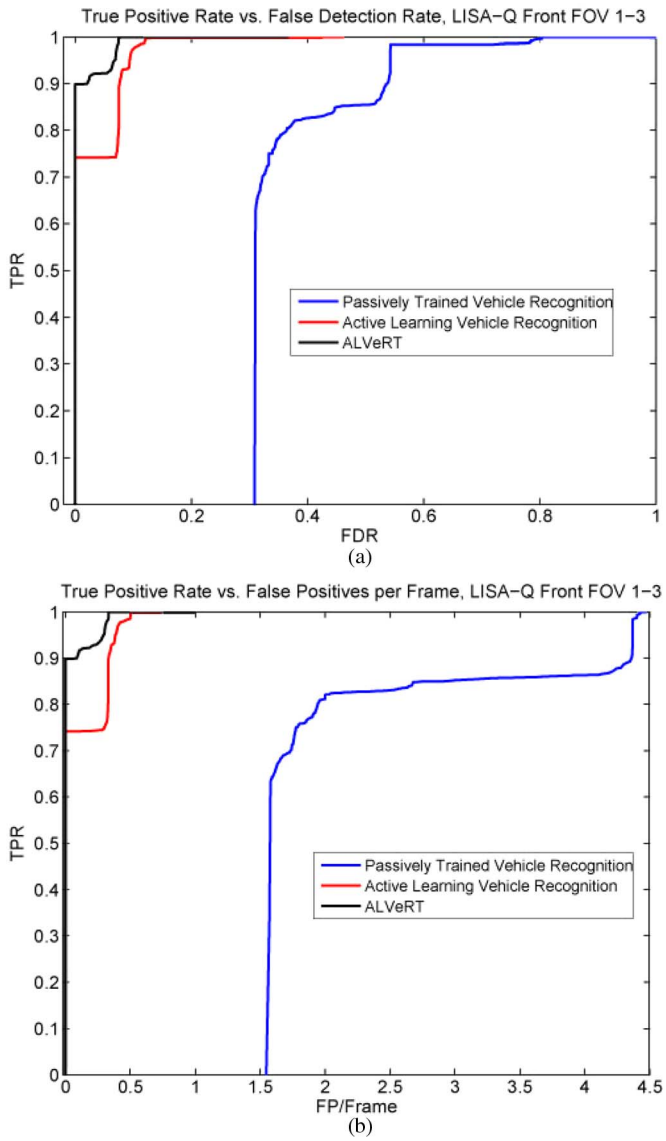


Fig. 16. (a) Plot of TPR versus FDR (LISA-Q Front FOV data sets 1–3) for (blue) the passively trained recognition, (red) the active-learning vehicle recognition, and (black) the ALVeRT system. We note the large improvement in performance due to active learning for vehicle detection. (b) Plot of TPR versus the number of false detections per frame (LISA-Q Front FOV data sets 1–3) for (blue) the passively trained recognition, (red) the active-learning vehicle recognition, and (black) the ALVeRT system. We note the reduction in false positives due to active learning.

VIII. Conclusion

A general active-learning framework for robust on-road vehicle recognition and tracking has been introduced. Using active learning, a full vehicle-recognition and tracking system has been implemented, and a thorough quantitative analysis has been presented. Selective sampling was performed using QUAIL, which is a visually intuitive user-friendly efficient query system. The system has been evaluated on both real-world video and public-domain vehicle images. We have introduced new performance metrics for assessing on-road vehicle-recognition and tracking performance, which provide a thorough assessment of the implemented system's recall, precision, localization, and robustness. Given the success of the ALVeRT framework, a number of future studies are planned. These research efforts include pedestrian-protection systems [10], [18], trajectory learning [23], and integrated active safety systems [7], [21].

ACKNOWLEDGMENT

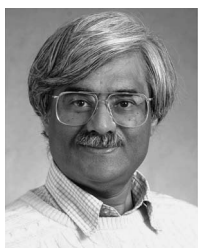
The authors would like to thank their colleagues B. Morris, A. Doshi, Dr. S. Krotosky, Dr. S. Cheng, Dr. J. McCall, and Dr. E. Murphy-Chutorian for their valuable contributions to testbed design and data collection. The authors would also like to thank the associate editor and reviewers for their valuable comments.

REFERENCES

[1] Y. Abramson and Y. Freund, "Active learning for visual object detection," Univ. Calif. San Diego, San Diego, CA, 2005.
[2] G. Alessandretti, A. Broggi, and P. Cerri, "Vehicle and guard rail detection using radar and vision data fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 95–105, Mar. 2007.
[3] J. Arrospide, L. Salgado, M. Nieto, and F. Jaureguizar, "On-board robust vehicle detection and tracking using adaptive quality evaluation," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 2008–2011.
[4] Y. Cao, A. Renfrew, and P. Cook, "Vehicle motion analysis based on a monocular vision system," in *Proc. Road Transport Inf. Control Conf.*, May 2008, pp. 1–6.
[5] Y. Chan, S. Huang, L. Fu, and P. Hsiao, "Vehicle detection under various lighting conditions by incorporating particle filter," in *Proc. IEEE Conf. Intell. Transp. Syst.*, Oct. 2007, pp. 534–539.
[6] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Mach. Learn.*, vol. 15, no. 2, pp. 201–221, May 1994.
[7] A. Doshi, S. Y. Cheng, and M. M. Trivedi, "A novel active heads-up display for driver assistance," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 85–93, Feb. 2009.
[8] M. Enzweiler and D. M. Gavrila, "A mixed generative-discriminative framework for pedestrian classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[9] Y. Freund and R. E. Schapire, "A short introduction to boosting," *J. Jpn. Soc. Artif. Intell.*, vol. 14, no. 5, pp. 771–780, Sep. 1999.
[10] T. Gandhi and M. M. Trivedi, "Computer vision and machine learning for enhancing pedestrian safety," in *Computational Intelligence in Automotive Applications*. Berlin, Germany: Springer-Verlag, May 2008, pp. 59–77.
[11] T. Gandhi and M. M. Trivedi, "Vehicle surround capture: Survey of techniques and a novel Omni video based approach for dynamic panoramic surround maps," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 3, pp. 293–308, Sep. 2006.
[12] A. Haselhoff, S. Schauland, and A. Kummert, "A signal theoretic approach to measure the influence of image resolution for appearance-based vehicle detection," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2008, pp. 822–827.
[13] M. Isard and A. Blake, "CONDENSATION: Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, Aug. 1998.
[14] O. L. Junior and U. Nunes, "Improving the generalization properties of neural networks: An application to vehicle detection," in *Proc. IEEE Conf. Intell. Transp. Syst.*, Oct. 2008, pp. 310–315.
[15] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Active learning with Gaussian processes for object categorization," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[16] A. Khammari, F. Nashashibi, Y. Abramson, and C. Laurgeau, "Vehicle detection combining gradient analysis and Adaboost classification," in *Proc. IEEE Conf. Intell. Transp. Syst.*, Sep. 2005, pp. 66–71.
[17] E. B. Koller-Meier and F. Ade, "Tracking multiple objects using the condensation algorithm," *Robot. Auton. Syst.*, vol. 34, no. 2/3, pp. 93–105, Feb. 2001.
[18] S. J. Krotosky and M. M. Trivedi, "On color-, infrared-, and multimodal-stereo approaches to pedestrian detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 619–629, Dec. 2007.
[19] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: Object localization by efficient subwindow search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[20] M. Li and I. K. Sethi, "Confidence-based active learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1251–1261, Aug. 2006.
[21] J. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, Mar. 2006.
[22] J. McCall, O. Achler, and M. M. Trivedi, "Design of an instrumented vehicle testbed for developing human centered driver support system," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2004, pp. 483–488.
[23] B. T. Morris and M. M. Trivedi, "Learning, modeling, and classification of vehicle track patterns from live video," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 425–437, Sep. 2008.
[24] F. Nashashibi, A. Khammari, and C. Laurgeau, "Vehicle recognition and tracking using a generic multisensor and multialgorithm fusion approach," *Int. J. Veh. Auton. Syst.*, vol. 6, no. 1/2, pp. 134–154, 2008.
[25] P. Negri, X. Clady, S. M. Hanif, and L. Prevost, "A cascade of boosted generative and discriminative classifiers for vehicle detection," *EURASIP J. Adv. Signal Process.*, vol. 2008, pp. 1–12, 2008.
[26] D. Ponsa, A. Lopez, F. Lumbreras, J. Serrat, and T. Graf, "3D vehicle sensor based on monocular vision," in *Proc. IEEE Conf. Intell. Transp. Syst.*, Sep. 2005, pp. 1096–1101.
[27] P. M. Roth and H. Bischof, "Active sampling via tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[28] M. V. Shirvaikar and M. M. Trivedi, "A neural network filter to detect small targets in high clutter backgrounds," *IEEE Trans. Neural Netw.*, vol. 6, no. 1, pp. 252–257, Jan. 1995.
[29] S. Sivaraman and M. M. Trivedi, "Active learning based robust monocular vehicle detection for on-road safety systems," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2009, pp. 399–404.
[30] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using evolutionary Gabor filter optimization," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 125–137, Jun. 2005.
[31] Z. Sun, G. Bebis, and R. Miller, "Monocular precrash vehicle detection: Features and classifiers," *IEEE Trans. Image Process.*, vol. 15, no. 7, pp. 2019–2034, Jul. 2006.
[32] M. M. Trivedi, T. Gandhi, and J. McCall, "Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 108–120, Mar. 2007.
[33] M. M. Trivedi and S. Cheng, "Holistic sensing and active displays for intelligent driver support systems," *Computer*, vol. 40, no. 5, pp. 60–68, May 2007.
[34] S. Vijayanarasimhan and K. Grauman, "Multi-level active prediction of useful image annotations for recognition," in *Proc. Neural Inf. Process. Syst. Conf.*, 2008, pp. 1705–1712.
[35] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Dec. 2001, vol. 1, pp. 511–518.
[36] C. Wang and J.-J. J. Lien, "Automatic vehicle detection using local features—A statistical approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 83–96, Mar. 2008.
[37] S. Wender and K. Dietmayer, "3D vehicle detection using a laser scanner and a video camera," *IET Intell. Transp. Syst.*, vol. 2, no. 2, pp. 105–112, Jun. 2008.
[38] World Health Org., "World Report on Road Traffic Injury Prevention," [Online]. Available: http://www.who.int/violence_injury_prevention/publications/road_traffic/world_report/factsheets/en/index.html

Sayanan Sivaraman (M'07) received the B.S. degree in electrical engineering in 2007 from the University of Maryland, College Park, and the M.S. degree in electrical engineering in 2009 from the University of California, San Diego, La Jolla, where he is currently working toward the Ph.D. degree with specialization in intelligent systems, robotics, and controls.

His research interests include computer vision, machine learning, intelligent vehicles, and transportation systems.



Mohan Manubhai Trivedi (F'09) received the B.E. degree (with honors) from the Birla Institute of Technology and Science, Pilani, India, and the Ph.D. degree from Utah State University, Logan.

He is currently a Professor of electrical and computer engineering and the Founding Director of the Computer Vision and Robotics Research Laboratory and the Laboratory for Intelligent and Safe Automobiles at the University of California, San Diego, La Jolla.

Dr. Trivedi is currently an Associate Editor for the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS. He will serve as the General Chair for the 2010 IEEE Intelligent Vehicles Symposium in San Diego.