Framework for Real-Time Behavior Interpretation From Traffic Video

Pankaj Kumar, *Member, IEEE*, Surendra Ranganath, Huang Weimin, and Kuntal Sengupta

***Abstract*—Video-based surveillance systems have a wide range of applications for traffic monitoring, as they provide more information as compared to other sensors. In this paper, we present a rule-based framework for behavior and activity detection in traffic videos obtained from stationary video cameras. Moving targets are segmented from the images and tracked in real time. These are classified into different categories using a novel Bayesian network approach, which makes use of image features and image-sequence-based tracking results for robust classification. Tracking and classification results are used in a programmed context to analyze behavior. For behavior recognition, two types of interactions have mainly been considered. One is interaction between two or more mobile targets in the field of view (FoV) of the camera. The other is interaction between targets and stationary objects in the environment. The framework is based on two types of *a priori* information: 1) the contextual information of the camera's FoV, in terms of the different stationary objects in the scene and 2) sets of predefined behavior scenarios, which need to be analyzed in different contexts. The system can recognize behavior from videos and give a lexical output of the detected behavior. It also is capable of handling uncertainties that arise due to errors in visual signal processing. We demonstrate successful behavior recognition results for pedestrian–vehicle interaction and vehicle–checkpost interactions.**

***Index Terms*—Bayesian network, behavior analysis, camera calibration, classification, context, event detection, three-dimensional (3-D) tracking, tracking, video.**

I. Introduction

It HAS been projected that the number of vehicles in the industrialized world will double to 1 billion by 2050, while a 12-fold increase in the developing world is expected, from 200 million to 2.5 billion vehicles. Such an enormous increase in vehicles would definitely require more sophisticated and intelligent handling of traffic resources. In dense traffic situations, the cost of an accident can be high with respect to loss of human life and disruptions on road networks working at or near full capacity. The world-wide economic cost resulting from traffic accidents is estimated to be about U.S. $518 billion a year. It is expected that, by the year 2020, road accidents could become the world's third leading cause of death and disability. Any reduction in the number or severity of such incidents would have large social and economic benefits. With this motivation, we consider a framework for detecting various traffic behaviors in video streams from stationary cameras, which are usually installed for visually monitoring traffic activities. Such a system can help in intelligent route guidance and event recognition for accident warning. Traffic congestion and accidents can be reported in real time to vehicles so that they can consider alternate routes to their destination.

A comprehensive video-based surveillance system performs the following functions:

- detects mobile objects;
- tracks them through the image sequence;
- classifies the tracked targets;
- analyzes their behaviors.

There has been a significant amount of work in the area of detecting moving vehicles [1]–[4]. Foreground detection schemes usually have the problem of detecting shadows of moving objects as foreground. This problem has been addressed in previous works, such as [5]–[7]. There have been several works on the tracking of vehicles and classifying them into different types [8]–[12]. Others, along with tracking, have estimated important traffic parameters [13], [14] by using image-processing techniques. With success in moving vehicle segmentation, tracking, and classification, the next stage in building a complete visual surveillance system is behavior detection for interaction between vehicles and between vehicles and pedestrians. Some examples of work done in detecting vehicle behavior from videos are [10] and [15]–[19]. In [15], Herzog designed and constructed an integrated knowledge-based system that is capable of translating visual information into natural language descriptions. The focus was on high-level scene analysis, i.e., from geometrical representations, as might be provided by a vision system, into linguistic descriptions of object motions. In a combined vision and natural language system aimed at simultaneous natural language descriptions of dynamic imagery, the recognition of motion events has to be done incrementally so that they can be described even while they are in progress. Medioni *et al.* [10] presented a complete system for event detection and behavior recognition in videos taken from a single airborne moving camera. The event recognition involves humans and vehicles and uses optical flow to segment the mobile object from the background. Optical flow methods of segmentation are computationally very intensive and difficult to realize in real time without extra hardware support. Furthermore, there is no robust method for classifying the targets into different categories. Remagnino *et al.* [19] showed an

Manuscript received December 1, 2003; revised July 15, 2004 and August 1, 2004. This work was supported by The Enterprise Challenge under Grant P00291/475 and the National University of Singapore Research Grant under Grant R-263-000-214-112. The Associate Editor for this paper was F.-Y. Wang.

P. Kumar and H. Weimin are with the Institute of Infocomm Research, Singapore 119613, Singapore (e-mail: kumar@i2r.a-star.edu.sg; wmhuang@i24.a-star.edu.sg).

S. Ranganath is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576, Singapore.

K. Sengupta is with AuthenTec Inc., Melbourne, FL 32902-2719 USA (e-mail: kuntal.sengupta@authentec.com).

Digital Object Identifier 10.1109/TITS.2004.838219

elegant solution to the parking lot monitoring task. This work described a pedestrian and car surveillance system that models the interaction between any two agents using a small belief network. Behaviors and situations for object interaction were modeled using Bayesian networks. These probabilistic models have advantages and disadvantages. Advantages include the ability to handle uncertain and incomplete data, which are asynchronously input to the network. Disadvantages include a fixed topology, where prior and conditional probabilities for root nodes and links have to be known or learned. Training the network for optimal results requires a large data set for each behavior type, which may be difficult to obtain, especially for behaviors related to accidents. In [20], Ivanov *et al.* presented an automatic surveillance system that labels events of human–car interactions in an open car park. Miura *et al.* [17] proposed a novel "intelligent navigator" based on an in-vehicle camera system. The two main components of the intelligent navigator are an advice-generation system and a road-scene recognition system. The advice-generation system is based on a layered reasoning architecture, with a vision system to detect lanes and vehicles. Here, only the behaviors that are of interest to a driver are discussed, not behaviors related to traffic management.

We present a complete real-time surveillance system with a rule-based behavior and event-recognition module for traffic videos. The behavior-recognition module of the system is similar in approach to the work of [10], but here the system is designed for real-time operation and can be used for data acquired from existing video cameras on the sides of the roads and/or overpasses. Thus, the system does not require the deployment of new cameras. Furthermore, we have used the approach of translating measurements in image space to physical parameters in the world coordinate space by using camera calibration techniques, in conjunction with a Bayesian network-based target-classification scheme. This gives a more robust and reliable detection of events and, hence, of behaviors. The novel Bayesian network-based target classifier uses both image features and video-based tracking results. It is easy to get sufficient data to train a Bayesian network for classifying target types, but it is difficult to get sufficient training data for behavior recognition, especially those related to accidents. Therefore, our system uses Bayesian network for target classification, but a rule-based method for event and behavior analysis.

Henceforth, this paper is organized as follows. Section II gives an overview of the system and its different modules. Section III briefly discusses the mobile object-detection and tracking algorithms. Features such as position, velocity, etc. are translated to the world coordinate system by using camera calibration, as discussed in Section IV. Section V discusses some of the target features selected for representation of targets. Section VI discusses the Bayesian network used for classifying the different target types. Following this is the development of a framework for behavior analysis. Behavior analysis is context sensitive, where the context is usually formed by the different stationary objects and moving targets in the FoV of the camera. Section VII explains how the contextual information is programmed and used. Events and behaviors are discussed in Sections VIII and IX, respectively. In Section X, we show the successful working of our system on some example traffic videos. Finally, Section XI concludes this paper.

II. System Overview

Fig. 1 shows a schematic diagram of the various components of the complete behavior-recognition module and the flow of data. A statistical background model of the camera's FoV is computed from the image sequence. Background subtraction is used to segment the moving foreground objects that are then tracked with the tracking module. The behavior recognition system uses two types of inputs:

1) extracted features such as shape, position, motion, and track of the targets obtained from video analysis;
2) *a priori* knowledge of the spatial context of the various objects present in the FoV of the camera.

A priori knowledge of the relation between context and behaviors and description of behaviors in terms of events is programmed into the system. Based on contextual information, a decision is made about which scenarios to analyze. For a given context, only a subset of scenarios are analyzed, because we do not expect all behaviors to occur in a context. For example, there would be no behavior related to a checkpost in the FoV of a camera where there are no checkposts. Once the context is known, the different types of behaviors that need to be considered are significantly reduced. The output of the behavior recognition module is the recognized behavior and the frames in which the specific behavior took place.

III. Motion Detection and Tracking

There have primarily been three classes of techniques for the segmentation of moving foreground objects in videos: 1) frame differencing, as used in [3], [21]; 2) background subtraction, as used in [22]–[25]; and 3) optical flow, as used in [10], [26]. Frame differencing does not yield good results when the objects are not sufficiently textured and optical flow computations are very intensive and difficult to realize in real time. We propose a background subtraction technique to segment the moving objects into image sequences. This technique is capable of modeling the background, even in the presence of foreground objects, and of updating the model as new video frames are acquired. Here, the background pixels are modeled with a single Gaussian distribution; this can be easily extended to a mixture of Gaussians, if desired.

Let N frames of a color image sequence be used for modeling the background (we use $N = 200$). We use the YC_rC_b color space for background modeling because the empirical results of [7] show YC_rC_b to be optimal for foreground segmentation and shadow suppression among the various standard color spaces studied there. Let $\mathbf{p}_{ijk}$ be a pixel at image coordinate i, j in frame k. Since each pixel $\mathbf{p}_{ijk}$ has three components, Y, C_r, and C_b, their histograms are modeled by three Gaussians. We find the histograms $H_{ij_Y}(u), H_{ij_{C_r}}(u)$, and $H_{ij_{C_b}}(u)$ of the pixels in the N frames, at each spatial location i, j and each

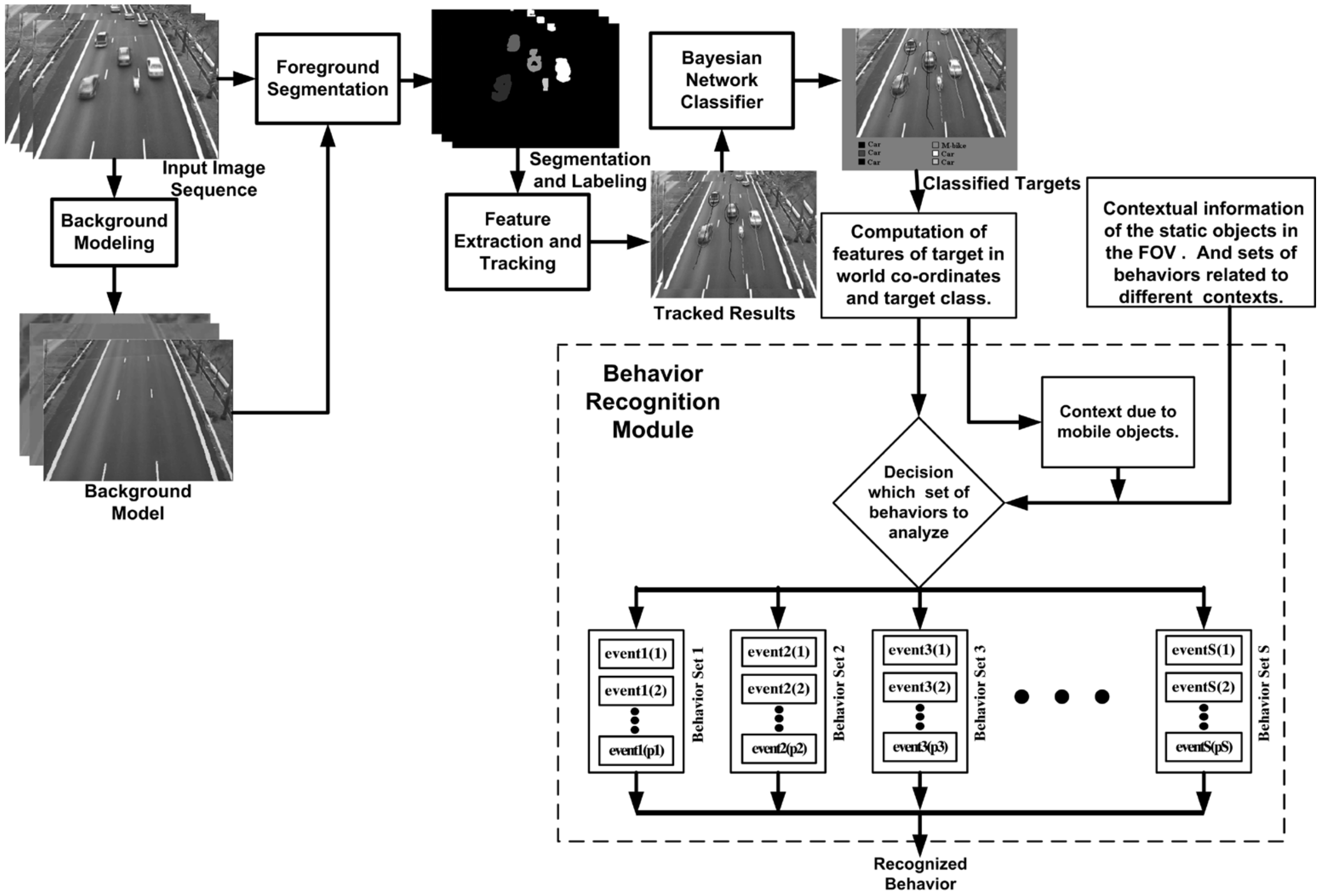

Input Image Sequence
Foreground Segmentation
Segmentation and Labeling
Bayesian Network Classifier
Car
Car
Car
M-bike
Car
Car
Classified Targets
Background Modeling
Feature Extraction and Tracking
Tracked Results
Computation of features of target in world co-ordinates and target class.
Contextual information of the static objects in the FOV . And sets of behaviors related to different contexts.
Background Model
Behavior Recognition Module
Context due to mobile objects.
Decision which set of behaviors to analyze
event1(1)
event1(2)
event1(p1)
Behavior Set 1
event2(1)
event2(2)
event2(p2)
Behavior Set 2
event3(1)
event3(2)
event3(p3)
Behavior Set 3
eventS(1)
eventS(2)
eventS(pS)
Behavior Set S
Recognized Behavior

Fig. 1. Schematic diagram of the behavior-recognition system. Video input is used to model the background and also to segment the moving foreground objects. The inputs to the behavior-recognition system are the features of targets, spatial context, and the different behaviors for different spatial contexts. Output is the recognized behavior and the frames in which it occurred.

channel. The peak of each histogram is the intensity or chrominance value most frequently found at the corresponding pixel location and channel and, thus, is expected to be the background. Using a window of width $2W$ centered on the mode of each histogram, we compute the mean and variance of the Gaussian distribution using

$$\mu_{ij_Y} = \frac{1}{\sum_{u=(u_{ij_Y}^{\max})-W}^{(u_{ij_Y}^{\max})+W} H_{ij_Y}(u)} \times \sum_{u=(u_{ij_Y}^{\max})-W}^{(u_{ij_Y}^{\max})+W} u \times H_{ij_Y}(u) \tag{1}$$

$$\sigma_{ij_Y}^2 = \frac{1}{\sum_{u=(u_{ij_Y}^{\max})-W}^{(u_{ij_Y}^{\max})+W} H_{ij_Y}(u)} \times \sum_{u=(u_{ij_Y}^{\max})-W}^{(u_{ij_Y}^{\max})+W} (u - \mu_{ij_Y})^2 \times H_{ij_Y}(u). \tag{2}$$

In our computations, we use $W = 5$. Equations (1) and (2) are for the Y channel; the computation for other channels is similar.

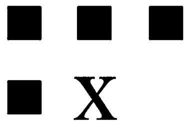


Fig. 2. Dots are the eight connected causal neighbors of the pixel $\mathbf{x}$ for a left-to-right and top-to-bottom raster scan.

We use hysteresis thresholding to classify each pixel as being foreground or background. The classification rule is as follows:

if (*any of the causal 8 connected neighbors of* $\mathbf{p}_{ijk}$, *as shown in Fig. 2, is foreground*)
then *use the lower threshold for classifying the pixel*
else *use the higher threshold.*

Each channel of each pixel $\mathbf{p}_{ijk}$ has its own lower and upper thresholds obtained as a product of the corresponding standard deviation with a constant factor γ, which has smaller and larger values for obtaining the lower threshold and higher threshold, respectively, for use in hysteresis thresholding. Different threshold values are used for luminance and chrominance channels and are denoted as $\gamma_{Y_{\text{bg}}}$ and $\gamma_{C_{\text{bg}}}$, respectively. The work of Prati [6] on shadow detection in hue saturation and

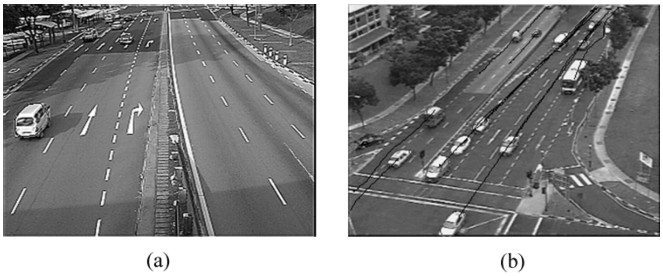


Fig. 3. (a) Result of shadow suppression and foreground extraction algorithms and (b) tracking of many targets simultaneously by the multibody tracking algorithm used in our system.

value (HSV) color space has shown that luminance values of the shadow pixels are always less than the mean and usually lie in a range of values below the mean. There are negligible change in the chromacity channels due to shadows. Therefore, in our algorithm we use two thresholds based on constants $\gamma_{Y_{\mathrm{bg}}}$ and $\gamma_{Y_{\mathrm{sh}}}$ to detect the shadow pixels as

```
**if** $((|p_{ijk_Y} - \mu_{ij_Y}| < \gamma_{Y_{\mathrm{bg}}} \times \sigma_{ij_Y})\&$
$(|p_{ijk_{Cb}} - \mu_{ij_{Cb}}| < \gamma_{C_{\mathrm{bg}}} \times \sigma_{ij_{Cb}})\&$
$(|p_{ijk_{Cr}} - \mu_{ij_{Cr}}| < \gamma_{C_{\mathrm{bg}}} \times \sigma_{ij_{Cr}}))$
**then** $\mathbf{p}_{ijk}$ *is background*
**else if** $((\mu_{ij_Y} - p_{ijk_Y} > \gamma_{Y_{\mathrm{bg}}} \times \sigma_{ij_Y})\&$
$(\mu_{ij_Y} - p_{ijk_Y} < \gamma_{Y_{\mathrm{sh}}} \times \sigma_{ij_Y})\&$
$(|p_{ijk_{Cb}} - \mu_{ij_{Cb}}| < \gamma_{C_{\mathrm{bg}}} \times \sigma_{ij_{Cb}})\&$
$(|p_{ijk_{Cr}} - \mu_{ij_{Cr}}| < \gamma_{C_{\mathrm{bg}}} \times \sigma_{ij_{Cr}}))$
**then** $\mathbf{p}_{ijk}$ *is shadow*
**else** $\mathbf{p}_{ijk}$ *is foreground.*
```

This segmentation algorithm is a part of the hysteresis thresholding pseudo code. If the causal neighbors of pixel $\mathbf{p}_{ijk}$ is foreground, then a lower value is used for $\gamma_{Y_{\mathrm{bg}}}, \gamma_{Y_{\mathrm{sh}}}$, and $\gamma_{C_{\mathrm{bg}}}$. However, when the causal neighbors are not foreground, then higher values are used for these parameters. After segmentation the foreground pixels are grouped to form eight connected blobs. The convex hull of each blob is then approximated by an ellipse. We use Kalman filters and a dynamic programming-based pattern-matching technique to achieve robust tracking [27]. Fig. 3 shows shadow detection and multibody tracking results. In this paper, our main focus is behavior analysis; hence, we dispense with the details of feature extraction and tracking, which can be found in [28].

IV. Camera Calibration

Working in world coordinates is better than image coordinates, as many ambiguities can be resolved. For example, perspective foreshortening gives an erroneous perception of target motion in the image plane. Targets that are closer to the camera appear to move more quickly than targets that are farther away, even if their ground speeds are similar. To translate the measurements in image coordinates to measurements in world coordinates, the camera parameters are required. This needs to be done only once for a camera setup and image coordinate space.

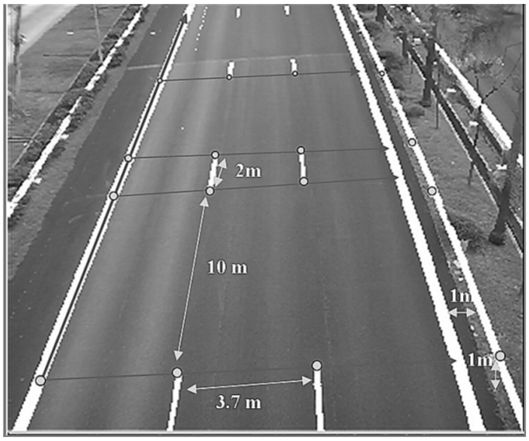


Fig. 4. Some of the points and their world coordinate measurements used for computing the perspective transformation matrix P. Some of the points chosen here have nonzero Z coordinate values.

TABLE I
Standard Height Values Used for Different Target Classes Considered in the System

Pedestrian	M-bike	Car	Truck	Heavy Truck
1.7m	1.7m	1.5m	3.0m	4.0m

The world coordinate axes are chosen so that the XY plane is aligned to the ground plane of the scene and the Z axis is perpendicular to the ground plane. The perspective transformation equation for a pin-hole camera model

$$\begin{aligned} \begin{bmatrix} x_i \\ y_i \\ \lambda \end{bmatrix} &= \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \\ \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} &= P \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \end{aligned} \tag{3}$$

is computed using manually selected points in the image space and their coordinates in the world coordinate space. Here, the subscripts w and i are used to indicate coordinate values of a point in the world and image space, respectively. To compute P, the 3×4 perspective projection matrix we need a minimum of six point matches. We pick more than this minimum number of corresponding points and use least squares to solve the overconstrained linear equations and filter out noise due to errors in measurements. To obtain the values of the world coordinates, we have used the standard dimensions on the road markings. Therefore, there is no need to take the actual ground measurements on the road. Fig. 4 shows some of the points chosen for camera calibration and their location in three dimensions for one of the videos.

From (3) it can be easily shown that if the three-dimensional (3-D) height of a point is known along with its image coordinates, then its unique 3-D location in world coordinate space can be computed as shown in (4) and (5) at the bottom of the next

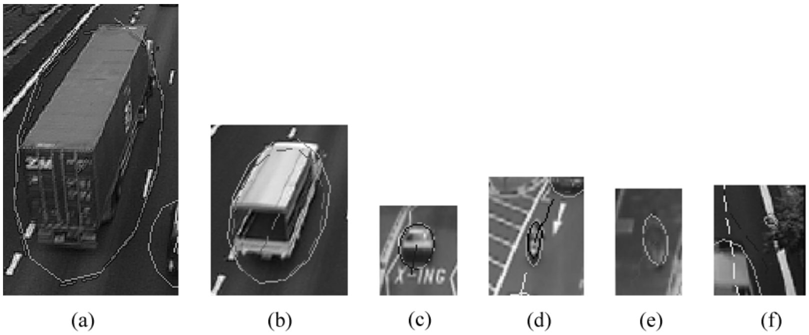


Fig. 5. Example images of targets from different classes. The size of the images here are proportional to the way in which they appeared in the original video. (a) Heavy truck, (b) truck, (c) car, (d) motorbike, (e) pedestrian, and (f) noise.

page. The classification of the object into one of the following classes—*pedestrian, motorbike, cars, trucks, heavy trucks*, and *noise*—using a Bayesian network (Section VI) allows us to represent the object's height by the values shown in Table I. These values were obtained by measuring typical heights of the objects in different classes. Fig. 5. show some images of the different classes of targets used for classification. A point that lies on top of the target will have its Z_w coordinate equal to the height of the target as we have initially aligned the XY axes of the world-coordinate system with the ground plane of the scene. To ensure that the point chosen in the segmented foreground region is a good approximation to the top of the target, the following heuristics are used, based on the ellipse that approximates the target in the image space.

- For pedestrians and motorbikes, we select the point that lies on the major axis of the ellipse and is 10% into the perimeter of the ellipse from the front of the car. Here, the implicit assumption is that the pedestrians are standing or walking.
- For cars, the selected point lies on the major axis. It is midway between the centroid of the ellipse and the point where the major axis intersects the ellipse perimeter from the top of the image.
- For trucks and heavy trucks, the point we choose is on the major axis of the ellipse approximating the target and is 10% inside the ellipse boundary in the direction of motion of the target.

Experiments showed that the different heuristics used for locating the top of cars and trucks was less prone to errors. We also show later (in Section X) that small errors in height estimate of the targets does not significantly affect the position and speed estimates.

Using this technique of translating measurements from image coordinates to world coordinates, we could detect vehicle speeds within an error range of $\pm 5\%$. This error range was obtained by comparing the vehicle's speedometer reading with estimated speed. This relatively high accuracy of speed estimation makes it possible to reliably infer the acceleration and deceleration of targets.

V. Target Features

The targets are represented by spatial and temporal features in two-dimensional (2-D) image space and also in 3-D world coordinate space. Some of the features used in representation of targets are as follows.

1) *Size*: The major and minor axis of the ellipse that approximates the convex hull of the target.
2) *Position*: This is the centroid of the target.
3) *Velocity*: which is obtained from the Kalman filter tracking the target centroid.
4) *Target type*: At present, we have five types of targets: 1) pedestrians; 2) motorbikes; 3) cars; 4) buses and trucks; and 5) heavy trucks and double-decker buses. We have also included an additional category for noise. This classification is done based on the size, shape, velocity, and position of the target using a Bayesian network.
5) *Target track*: which is the trajectory of the target, obtained as the position of the target in previous frames.

These target features are used for event detection and to define the context for interaction with other mobile objects. Fig. 6 shows different levels of target features and descriptors. The numerical values of high-level descriptors are computed from lower level image features. The world coordinate velocity and acceleration are computed using the temporal information of the

$$X_w = \frac{(p_{32}y_i - p_{22})}{\{(p_{31}x_i - p_{11})(p_{32}y_i - p_{22}) - (p_{12} - p_{32}x_i)(p_{21} - p_{31}y_i)\}} \left\{ \frac{p_{12} - p_{32}x_i)(p_{23} - p_{33}y_i)Z_w}{(p_{32}y_i - p_{22})} + \frac{(p_{12} - p_{32}x_i)(p_{24} - p_{34}y_i)}{(p_{32}y_i - p_{22})} \right\} \tag{4}$$

$$Y_w = \frac{(p_{21} - p_{31}y_i)X_w}{(p_{32}y_i - p_{22})} + \frac{(p_{23} - p_{33}y_i)Z_w}{(p_{32}y_i - p_{22})} + \frac{(p_{24} - p_{34}y_i)}{(p_{32}y_i - p_{22})}. \tag{5}$$

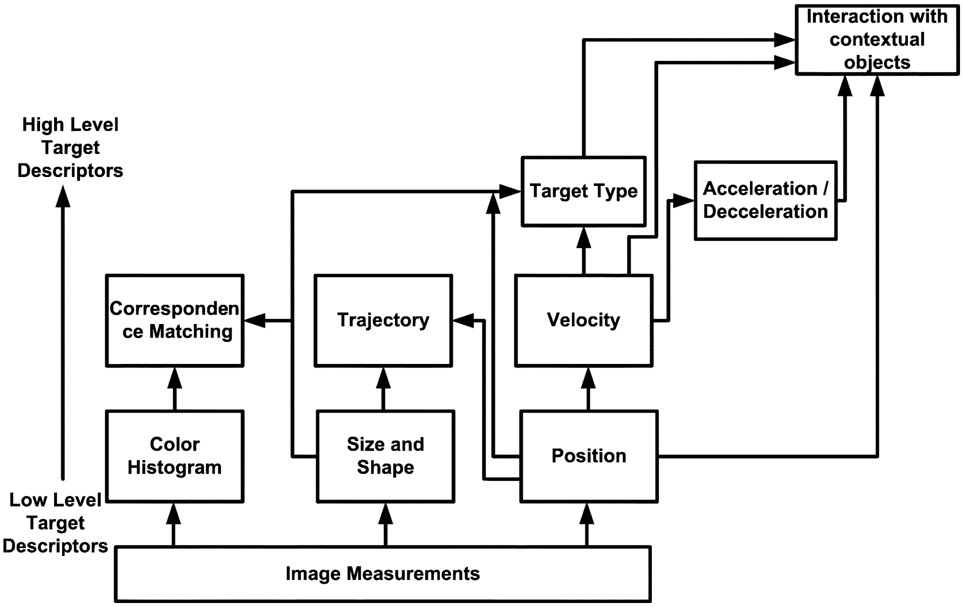

Interaction with contextual objects
High Level Target Descriptors
Target Type
Acceleration / Decceleration
Correspondence Matching
Trajectory
Velocity
Color Histogram
Size and Shape
Position
Low Level Target Descriptors
Image Measurements

Fig. 6. Different levels of target features and descriptors. The high-level target attributes are obtained from low-level image measurements, tracking, and target classification.

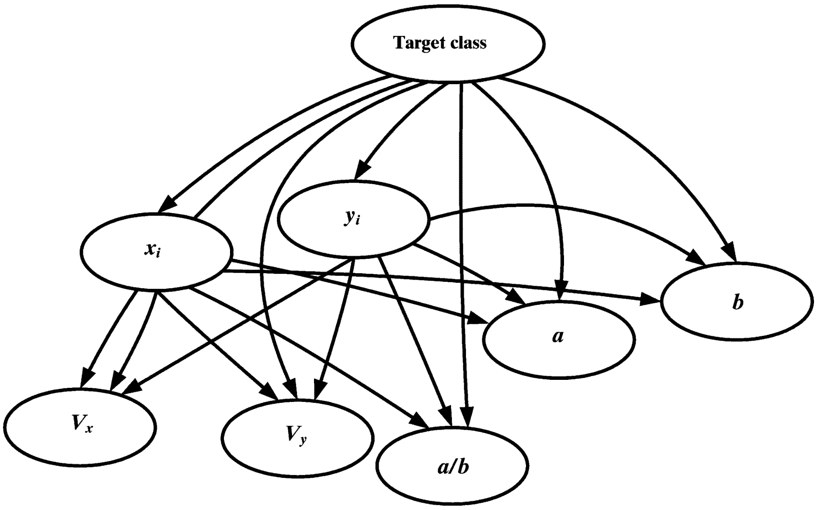


Fig. 7. Network structure used for target classification. Here, the velocity variables V_x and V_y and the size measures a and b are dependent on both target type and image position of the target. The aspect ratio of the target is dependent on the target type and position of the target.

frames, camera calibration, and the Kalman filter estimates of the classified target's velocity in the image plane.

VI. Bayesian Network Classifier

Bayesian networks (BNs) are useful for combining evidence in vision problems, particularly when the information is diverse, dependent, both causal and diagnostic (deductive and abductive), and the inference procedure is best posed in probabilistic terms [29], [30]. BNs have been used in many applications, such as audio-visual speaker detection [31] and content-based image and video indexing [32]. Huang *et al.* [16] used a BN for automatic traffic scene analysis. Here, we present a BN for the classification of targets in video obtained from a fixed camera. The camera is usually placed above the road and looking downward onto the traffic. In this situation, when there is perspective foreshortening, it is difficult to build a deterministic functional relationship to map the size, shape, position, and motion features to the target class. For example, a car close to the camera may be the same size as a truck further from the camera. Similarly, a pedestrian passing by close to the camera may have the same apparent motion in image space as a quickly moving car far from the camera. Furthermore, there are internal dependencies in the features themselves. For example, the aspect ratio (shape parameter) of an object may depend upon its position in the image. Therefore, to establish a relationship between the various image features of a target and its class and also to model the conditional dependencies of the features, we propose a new BN classifier for inferring target class from measurements from each frame and tracking results.

Fig. 7 shows the BN used for target classification. Each node is a variable and the target class is the root node. Here we use a

TABLE II
ATTRIBUTES OF A STATIC CONTEXTUAL OBJECT
(CHECKPOST 1, AS SEEN IN Fig. 8)

name	Checkpost
function	to temporarily stop vehicles
normal interaction time	5 seconds
geometry	rectangle [(115 137), (116 160)]

supervised training approach where the network parameters are learned for optimal classification performance. The seven measurement nodes are x_i, y_i (the x, y coordinates of the target in image space), V_x and V_y (the x, y components of the targets motion in image space obtained from tracking), a and b (the major and minor axis of the ellipse modeling the target), and a/b (the aspect ratio of the ellipse). An efficient inference algorithm is used to compute the distribution of the target class node given the measurements [29]. The network structure in Fig. 7 has been manually specified using the knowledge of the pin-hole camera model. The velocities V_x and V_y are dependent upon both the target class and the image position of the target, (x_i, y_i). Similarly the size of the target represented by a and b is made dependent upon the position of the target and its type. The aspect ratio, a/b, measured for a target is dependent on its position and target type. In future development of our work we will consider network structure learning algorithms for better classification of the targets.

VII. CONTEXT

Context plays a very important role in the detection of events. The contextual information of the scene is provided by the operator and needs to be done once for a given surveillance setup. Context is defined by the spatio-temporal properties of static objects in the environment and by the zone of influence (ZoI) of mobile targets. Contextual information governs the different types of predefined scenarios of events that need to be analyzed for recognizing different behaviors. An example is the context of a checkpost, which checks the entrance of unauthorized vehicles. The behavior of interest would be improper access to the restricted area or detection of a malfunctioning checkpost.

Static objects that form a part of the context are defined geometrically by polygons and attributes such as *name, function, time of normal interaction, status*, etc. Table II gives the attributes of checkpost 1, as shown in Fig. 8. This figure shows an example of a scene with static contextual objects, which form the context for recognition of behaviors at a checkpost. The objects are:

1) checkpost 1 for vehicles entering the restricted area and checkpost 2 for exiting vehicles;
2) areas for interaction 3 and 4 with checkposts 1 and 2, respectively;
3) cash card machine 5.

When a target enters the area for interaction (AFI) of a checkpost, the system analyzes the scenarios of events related to the context of checkpost. There are different possible behaviors in this context and each is defined by a temporal sequence of events.

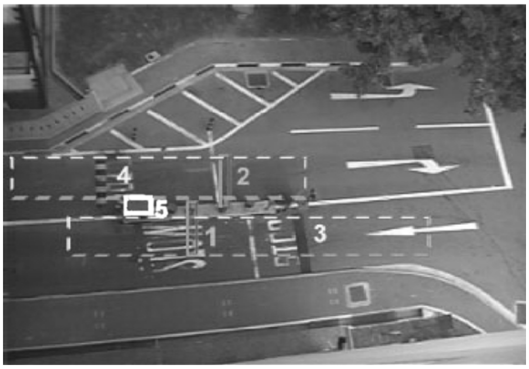


Fig. 8. Different contextual objects in the camera's FoV for description of static context that underlies behavior recognition are highlighted. Points 3 and 4 are AFI of checkposts 1 and 2. The checkposts are represented by thin rectangular regions. Object 5 is a cash card machine used for paying the parking fee.

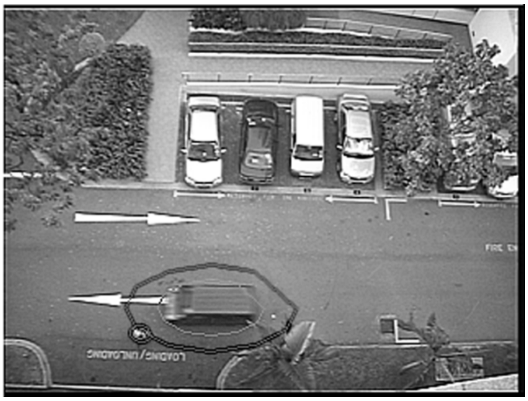


Fig. 9. Context representation for recognizing behaviors related to moving targets. The context here is formed by the proximity of the two interacting targets, a pedestrian and a vehicle. There is overlap between the outer ellipses, which are the ZoIs of the two targets. The overlap of ZoIs is indicative of the targets proximity to each other.

To recognize behaviors that involve the interaction of two or more targets, we define a context that arises when two or more targets come in proximity with each other. Proximity of targets is determined by the normalized area of overlap of ZoI of the targets. The ZoI of a target is defined as the outer ellipse whose center and orientation is the same as the target's, but whose major and minor axes are 1.6 times that of the approximating ellipse. This value is heuristically chosen by experimentation. Fig. 9 shows a pedestrian in proximity to a car moving at high speed, simulating a potential accident behavior. Here, the context for analyzing accident behavior has been formed by the overlap of the ZoI of the pedestrian and the van shown with the larger ellipse around the targets. When two or more targets are close to each other, the system looks for events in which their relative velocity is dangerously high. The relative velocity of the targets is obtained by vector subtraction of the 3-D estimated velocities of the interacting targets.

VIII. EVENTS

Events are usually described by the spatio-temporal relationship between targets and contextual elements or with other targets. Events are also defined in terms of constraints on or properties of the high-level target descriptors. For example, if we want to detect "speeding of cars," then its measured speed is compared with the upper speed limit provided by the operator. If the measured speed is greater than the speed limit provided by the operator, then the event "car is speeding" is detected.

Measurements from visual sensors are usually erroneous; therefore, the system should be robust to errors. To do this, we look for temporal consistency in detected events, which is measured by a confidence factor κ. In a given context, all the events that can take place are associated with the target using an initial value of $\kappa = 0$. When a specific event is detected, its κ is increased by 0.2 and κ for other nondetected events is decremented by 0.2 The confidence factor has a floor value of 0 and maximum value of 1, i.e., once κ reaches a value 1 or 0, then it is not further incremented or decremented. The following are some examples of events we considered in our experiments.

1) **Moving toward the checkpost**: This event is detected when the current distance between the target and checkpost is greater than the distance between the target and checkpost in the next frame.
2) **Stopped in front of the checkpost**: The target is in the AFI of a checkpost and the speed of the target is less than a threshold.
3) **Crossing the checkpost**: The distance between target and checkpost is almost zero, but the speed is above a threshold.
4) **Moves away from the checkpost on the other side of the checkpost**: The direction of velocity is same as before, but the distance between the target and checkpost is increasing.
5) **Moves away from the checkpost on the same side of the checkpost**: The direction of velocity is reversed and the distance between the target and checkpost is increasing.
6) **Moves out of the AFI of a checkpost**: The current position of the target is within the AFI of a checkpost, but the velocity is directed away from the AFI of the checkpost.
7) **Crosses the checkpost outside the AFI of the checkpost**: The target is outside the AFI of a checkpost and is crossing the checkpost. The protocol for recognition of the event of crossing the checkpost is the same as 3).

IX. Behavior Analysis

A behavior is defined as a sequence of events, with or without temporal constraints on the order of event occurrence. Behavior analysis can be as simple as the detection of a single event, e.g., a car is speeding, or can be a complex sequence of multiple events, e.g., a car is entering a restricted area and violating the checkpost norms. Given the context of the vehicle, different behaviors can be analyzed. For example, consider a vehicle entering the AFI of a checkpost. In this context, the following behaviors are possible and are analyzed by defining each of these behaviors by a sequence of events as follows.

1) **Normal crossing of checkpost**:
   a) Target moves toward the checkpost;
   b) Target stops in front of the checkpost;
   c) Target moves toward the checkpost;
   d) Target crosses the checkpost;
   e) Target moves away from the checkpost on the other side of the checkpost;
   f) Target leaves the AFI of the checkpost.
2) **Breakdown of the checkpost or breakdown of a vehicle in front of a checkpost**:
   a) Target moves toward the checkpost;
   b) Target stops in front of the checkpost for more than the normal time of interaction with the checkpost;
   c) There are more vehicles stopping in the AFI of the checkpost.
3) **Target avoids the checkpost and backs off**:
   a) Target moves toward the checkpost;
   b) Target stops before the checkpost;
   c) Target moves away from the checkpost on the same side of the checkpost;
   d) Target leaves the AFI of the checkpost.
4) **Vehicle is trying to gain illegal access to the restricted area by moving in pedestrian walkway**:
   a) Target moves toward the checkpost;
   b) Target moves out of the AFI of the checkpost, i.e., outside the road region onto pedestrian's walkway;
   c) Target crosses the checkpost outside the AFI of the checkpost;
   d) Target moves away from the checkpost on the other side of the checkpost.

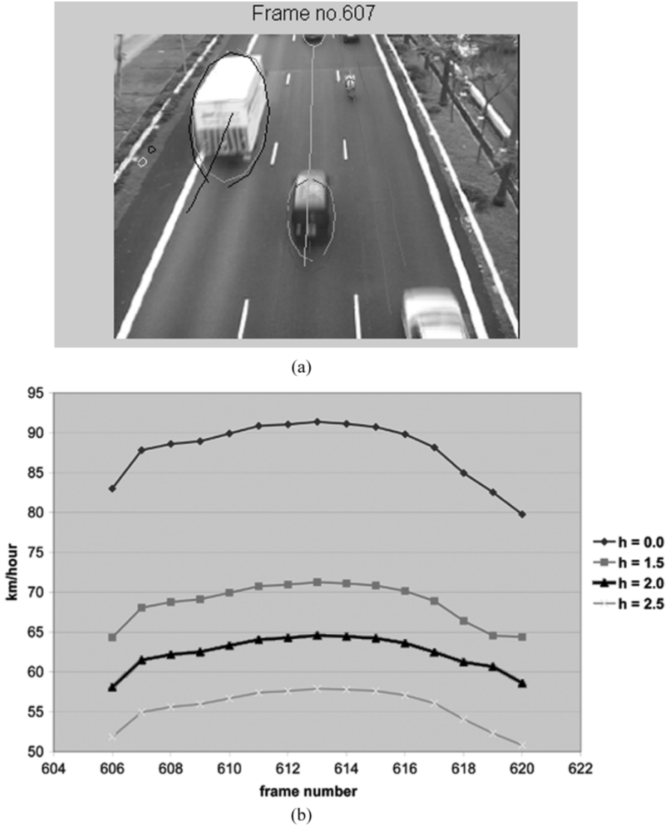
For behavior recognition, we compute a recognition factor Φ for each behavior of different targets, which is the sum of the confidence factors κ of each event indexed by i in the behavior j, divided by N_j the total number of events in that behavior

$$\Phi_j = \frac{\sum_{i=1}^{i=N_j} \kappa_i}{N_j}. \tag{6}$$

The behavior that yields the highest value of Φ is considered to be the recognized behavior. To increase the discrimination of behavior recognition, a higher weight can be given to more crucial events and lower weights to the less significant events. For example, in the case of the behavior, "vehicle avoiding the checkpost and backing off," the most crucial event is, "target moves away from the checkpost on the same side of the checkpost." An example of a common and, hence, less significant event is "target moves toward the checkpost"; this event is common to all behaviors in the context of a checkpost.

X. Results

This system has been tested on several videos of traffic scenes, which include pedestrians and other vehicles. For obtaining classification results using the BN proposed in this paper, over 1000 occurrences of different targets were identified and target tracking was performed for every one of them. Table III shows the average classification results of the BN-based scheme discussed previously. Very high recognition



(a)

(b)

Fig. 10. (a) Tracking results in frame 607 show a van in the center along with its track. The van was moving with a constant speed of 65 km/h, as read from its speedometer. (b) Plots show the estimated speed of the vehicle for different height values denoted by the parameter "h" and expressed in meters. The estimated speeds from the proposed system, which uses $h = 2.0$ m, are in the range of 58.1–64.5 km/h in frames 606–620, as shown by the third plot from the top. This is quite accurate considering the low-level image-processing errors on video data.

results have been obtained for cars and pedestrians. Table III also shows the accuracy of identifying segmentation error as noise. Lipton *et al.* [21] showed recognition results of 86.8% and 82.8% for vehicles and humans using Mahalanobis distance-based clustering. Here, we have a higher correct classification rate, even though the number of classes is six as compared to three in [21]. Fig. 10 shows the results of the 3-D motion estimation for different height values of a target. When the estimated height of the vehicle is taken to be zero, then there is significant error in the speed estimates. The speed estimates are in the range of 80–92 km/h when the actual speed is 65 km/h. The speed estimates for other values of height, such as 1.5, 2, and 2.5 m are close to the actual speed of 65 km/h. The initial frames when the target just enters the FoV show larger errors in speed estimates because the Kalman filter parameters take some time to settle to the correct value. Later, when the target starts moving out of the FoV of the camera, the errors in speed estimate may be attributed to the error in choosing the point that represents the height of the target. However, the speed estimates are quite accurate. This accurate measurement of speed allows detecting whether a vehicle is accelerating or decelerating. We show results of rule-based behavior recognition in two different contexts. One is for the interaction between two mobile targets and another is for the interaction between mobile targets and static objects in the environment. In the results, the different targets have been successfully classified into their respective classes. In these videos, there are four classes of targets pedestrians, motorbikes, cars, and trucks denoted by P, M, C, and H, respectively, in the results. The behaviors of the targets have been correctly annotated with textual remarks. Fig. 11 shows the correct detection of a dangerous interaction between a pedestrian and a vehicle. The targets are in close proximity to each other and their relative velocity is high. The system correctly analyzes this behavior to be dangerous and all such behaviors in the video stream were correctly detected. In Fig. 12, we show the recognition of behaviors of vehicles at a checkpost. All the possible behaviors at the checkpost as discussed in Section IX were correctly analyzed and classified by the recognition factor Φ. Figure captions give further details of the results.

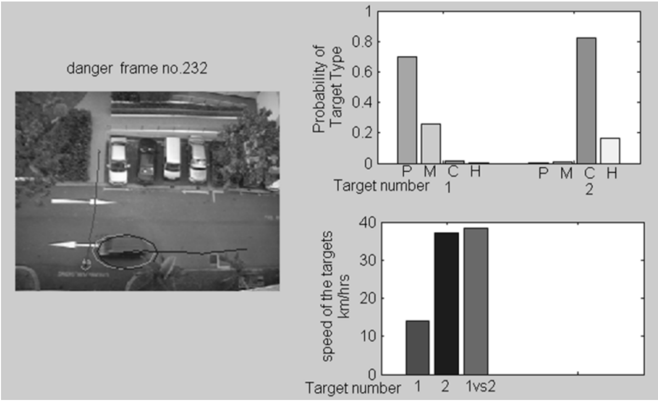


Fig. 11. Detection of dangerous behavior between two targets. The bar graph on the top right shows the degree of match for classification of the target type. There are two targets in the FoV. Target 1 is a pedestrian and 2 is a van that has been classified to be of type car. The bar graph on the bottom right shows the measured speeds of targets 1, 2, and their relative speeds (1vs2). The relative speed of the targets is very high and they are in close proximity to each other. Therefore, it has been detected as a dangerous interaction in frame 232.

TABLE III
CLASSIFICATION RESULTS FOR THE BN SHOWN IN Fig. 7

Object	Total	Correct Classification
Pedestrian	63	86.7%
Motor bike	105	92.5%
Cars	695	96.3%
Truck	92	93.3%
Heavy Truck	45	93.6%
Noise	163	88.6%

XI. CONCLUSION

This paper describes a complete real-time rule-based behavior-recognition system for traffic videos. This system will be useful for better traffic rule enforcement by detecting and signaling improper behaviors. This system is capable of detecting potential accident situations and is designed for existing camera setups on road networks; thus, no new camera installation will be required. The system is based on the analysis of 2-D image

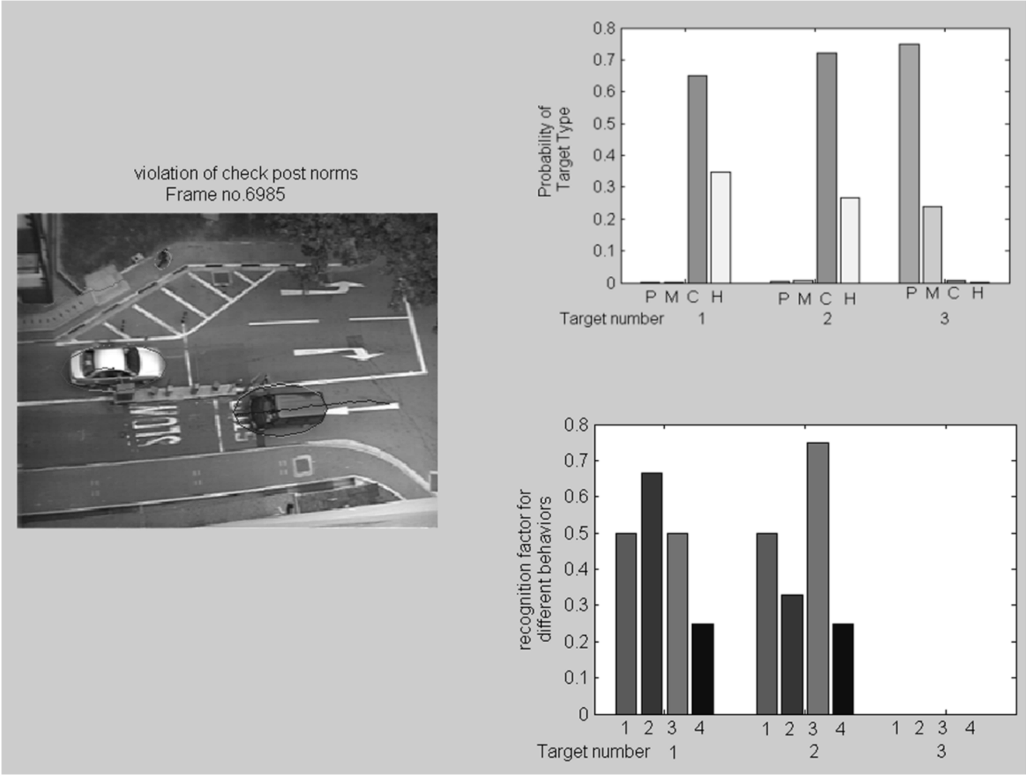


Fig. 12. Results of detecting different behaviors of vehicles at a checkpost. Targets 1,2, and 3 shown in the frame have been correctly classified as car, car, and pedestrian, as shown by the top right bar graph in this figure. The results of behavior analysis of the targets is shown by the bottom right bar graph. The behavior types 1, 2, 3, and 4 are the same as those discussed in Section IX. Target 1, the white car, has stopped at the check post for an unusually long time. This is correctly detected by the system as the recognition factor for behavior 2 is highest of all. Target 2, the black van, is avoiding the checkpost by backing off. This behavior has also been correctly detected as the recognition factor, in this case is highest for behavior 3. Target 3 is detected to be a pedestrian and, hence, not analyzed for behavior recognition in the context of the checkpost.

features and derived 3-D position and motion features. We have described a moving target-segmentation scheme that is dynamically updated and gives good shadow-detection results. The segmentation results are used to obtain 2-D image features of the target.

A novel approach to target classification in traffic videos using BNs has been proposed. This classifier has yielded very good classification results. Using the tracking results and the results of classification, world coordinate estimates of target position and velocity are obtained, which are accurate to within a small error of $\pm 5\%$ of ground truth.

The 3-D position and speed estimates are used for behavior recognition yielding robust behavior interpretations. The *a priori* knowledge of context and predefined scenarios is used for behavior recognition. The problem of imprecision and uncertainty due to errors in signal processing and image feature measurements have been handled by introducing a new parameter κ as a confidence measure. This confidence factor is based on the temporal consistency of detected events. We have demonstrated successful high-accuracy target classification and robust behavior recognition on real-life traffic videos. This system works well for changing illumination and rainy weather, as well.

This system's performance can be further improved by having context descriptions made more specific to the target. For example, for the detection of proximity among targets, the ZoI of a target can be made more specific to the target type. The ZoI of pedestrians can be larger than vehicles, because pedestrians are more negatively affected by accidents. The ZoI can also be made a function of the speed of the target. Targets with greater speed have larger ZoI than targets with slower speeds. Such changes would further improve the system's performance and reliability.

References

[1] N. Friedman and S. Russell, "Image segmentation in video sequences: A probabilistic approach," in *Proc. 30th Conf. Uncertainty in Artificial Intelligence (UAI)*, vol. 1–3, Aug. 1997, pp. 175–181.

[2] C. Gentile, O. Camps, and M. Sznaier, "Segmentation for robust tracking in the presence of severe occlusion," *Comp. Vision Pattern Recogn.*, vol. II, pp. 483–489, 2001.

[3] M.-P. Dubuisson and A. K. Jain, "Contour extraction of moving objects in complex outdoor scenes," *Int. J. Comp. Vision*, vol. 14, pp. 83–105, 1995.

[4] C. Stauffer and W. Grimson, "Adaptive background mixture models for real time tracking," in *Proc. Computer Vision and Pattern Recognition*, June 1999, pp. 246–252.

[5] M. Kilger, "Shadow handler in a video-based real-time traffic monitoring system," in *IEEE Workshop Application of Computer Vision*, 1992, pp. 11–18.

[6] A. Prati, I. Mikic, C. Grana, and M. M. Trivedi, "Shadow detection algorithms for traffic flow analysis: A comparative study," in *Proc. 4th IEEE Int. Conf. Intelligent Transportation Systems*, Oakland, CA, Aug. 2001.

[7] P. Kumar, K. Sengupta, A. Lee, and S. Ranganath, "A comparative study of different color spaces for foreground and shadow detection for traffic monitoring system," in *Proc. IEEE 5th Int. Conf. Intelligent Transportation Systems*, Singapore, Sept. 2002, pp. 100–105.

[8] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russell, "Toward robust automatic traffic scene analysis in real-time," in *Proc. Int. Conf. Pattern Recognition*, Jerusalem, Israel, Nov. 1994, pp. 126–131.

[9] A. J. Lipton and N. Haering, "Commode: An algorithm for video background modeling and object segmentation," in *Proc. 7th Int. Conf. Control, Automation, Robotics and Vision*, Singapore, Dec. 2002, pp. 1603–1608.

[10] G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia, "Event detection and analysis from video streams," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 873–889, Aug. 2001.

[11] P. Kumar, S. Ranganath, and H. Weimin, "Bayesian network based computer vision algorithm for traffic monitoring using video," in *Proc. IEEE 6th Int. Conf. Intelligent Transportation Systems*, Shanghai, China, Oct. 2003, pp. 897–904.

[12] H. Tao, H. S. Sawhney, and R. Kumar, "Object tracking with Bayesian estimation of dynamic layer representations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, pp. 75–89, Jan. 2002.

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

[14] K. Yamada and M. Soga, "A compact integrated visual motion sensor for ITS applications," *IEEE Trans. Intell. Transport. Syst.*, vol. 4, pp. 35–42, Mar. 2003.

[15] G. Herzog, "Utilizing interval-based event representation for incremental high-level scene analysis," in *Proc. 4th Int. Workshop Semantics of Time, Space, and Movement Spatio-Temporal Reasonning*, Chateau de Bonas, France, 1992, pp. 425–435.

[16] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, "Automatic symbolic traffic scene analysis using belief networks," in *Proc. 12th Nat. Conf. Artificial Intelligence*, Seattle, WA, 1994, pp. 966–972.

[17] J. Miura, M. Itoh, and Y. Shirai, "Toward vision-based intelligent navigator: Its concept and prototye," *IEEE Trans. Intell. Transport. Syst.*, vol. 3, pp. 136–146, June 2002.

[18] B. Neumann, *Semantic Structures: Advances in Natural Language Processing*. Hillsdale, NJ: N.J. Lawrence Erlbaum, 1989, ch. 5, pp. 167–206.

[19] P. Remagnino, T. Tan, and K. Baker, "Agent oriented annotation in model based visual surveillance," in *Proc. Int. Conf. Computer Visions (ICCV'98)*, 1998, pp. 857–862.

[20] Y. Ivanov, C. Stauffer, A. Bobick, and W. Grimson, "Video surveillance of interactions," in *Proc. 2nd Int. Workshop Visual Surveillance*, Fort Collins, CO, June 1999, pp. 82–89.

[21] A. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving target detection and classification from real-time video," in *Proc. IEEE Workshop Application of Computer Vision*, 1998, pp. 8–14.

[22] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Who, when, where, what: A real time system for detecting and tracking people," in *Proc. 3rd Int. Conf. Automatic Face and Gesture Recognition (FG'98)*, Apr. 1998, pp. 222–227.

[23] O. Javed and M. Shah, "Tracking and object classification for automated surveillance," in *Proc. Eur. Conf. Computer Vision*, 2002, pp. 343–357.

[24] I. Haritaoglu, D. Harwood, and L. Davis, "A fast background scene modeling and maintenance for outdoor surveillance," in *Proc. Int. Conf. Pattern Recognition*, Sept. 2000, pp. 179–183.

[25] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," *Comp. Vision Image Understand.*, vol. 80, pp. 42–56, 2000.

[26] M. C. A. Giachetti and V. Torre, "The use of optical flow for road navigation," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 34–49, Feb. 1998.

[27] P. Kumar, S. Ranganath, K. Sengupta, and W. Huang, "Co-operative multi-target tracking and classification," in *Proc. Eur. Conf. Computer Vision*, May 2004, pp. 376–389.

[28] P. Kumar, "Multi-body tracking and behavior analysis," Ph.D. dissertation, Elect. Comp. Eng. Dept., Nat. Univ. Singapore, July 2004.

[29] F. V. Jensen, *Lecture Notes on Bayesian Networks and Influence Diagrams*. New York: Springer-Verlag, 1999.

[30] A. Stassopoulou and T. Caelli, "Building detection using Bayesian networks," *Int. J. Pattern Recogn. Art. Intell.*, vol. 14, no. 6, pp. 715–734, 2000.

[31] T. Choudhury, J. M. Rehg, V. Pavlovic, and A. Pentland, "Boosting and structure learning in dynamic baysian networks for audio-visual speaker detection," in *Proc. Int. Conf. Pattern Recognition*, Quebec City, PQ, Canada, Aug. 2002, pp. 789–794.

[32] A. Mittal and C. L. Fah, "Addressing the problems of Bayesian network classification of video using high-dimensional features," *IEEE Trans. Knowl. Data Eng.*, vol. 16, pp. 230–244, Feb. 2004.

Pankaj Kumar (S'01–M'04) received the B.Tech degree in electrical and computer engineering from the Indian Institute of Technology, Delhi, India, and the M.Eng. and Ph.D. degrees from the National University of Singapore, Singapore.

Currently he is an Associate Scientist with the Institute of Infocomm Research, Singapore. His research interests include computer vision, artificial intelligence, behavior analysis, multicamera tracking, and surveillance.

Surendra Ranganath received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, the M.E. degree in electrical communication engineering from the Indian Institute of Science, Bangalore, India, and the Ph.D. degree in electrical engineering from the University of California, Davis.

From 1982 to 1985, he was with the Applied Research Group, Tektronix, Inc., Beaverton, OR, where he was working in the area of digital video processing for enhanced and high-definition television. From 1986 to 1991, he was with the Medical Imaging Group, Philips Laboratories, Briarcliff Manor, NY. In 1991, he joined the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, where he currently is an Associate Professor. His research interests include digital signal and image processing, computer vision, and neural networks and are focused on human–computer interaction and video surveillance applications.

Huang Weimin received the B.Eng. degree in automation and the M.Eng. and Ph.D. degrees in computer engineering from Tsinghua University, Beijing, China, in 1989, 1991, and 1996, respectively.

He is a Research Scientist with the Institute for Infocomm Research, Singapore. He has worked on research on handwriting signature verification, biometrics authentication, and audio/video event detection. His current research interests include pattern recognition, image processing, computer vision, human–computer interaction, and statistical learning.

Kuntal Sengupta received the B.Tech. degree from the Indian Institute of Technology, Kanpur, India, in 1990 and the M.S. and Ph.D. degrees from The Ohio State University, Columbus, in 1993 and 1996, respectively.

From 1996 to 1998, he was a Researcher with the Advanced Telecommunications Research (ATR) Laboratories, Kyoto, Japan. From 1998 to 2002, he was an Assistant Professor in the Electrical and Computer Engineering Department, National University of Singapore, Singapore. Currently, he is a Senior Algorithm Scientist with AuthenTec, Melbourne, FL. His present research interests include biometrics, human–computer interaction, video analysis, and multimodal fusion.

Dr. Sengupta received the Siemens Best Paper Award at IEEE Computer Vision and Pattern Recognition in 1993.