A Review of Computer Vision Techniques for the Analysis of Urban Traffic

Norbert Buch, *Member, IEEE*, Sergio A. Velastin, *Member, IEEE*, and James Orwell

***Abstract*—Automatic video analysis from urban surveillance cameras is a fast-emerging field based on computer vision techniques. We present here a comprehensive review of the state-of-the-art computer vision for traffic video with a critical analysis and an outlook to future research directions. This field is of increasing relevance for intelligent transport systems (ITSs). The decreasing hardware cost and, therefore, the increasing deployment of cameras have opened a wide application field for video analytics. Several monitoring objectives such as congestion, traffic rule violation, and vehicle interaction can be targeted using cameras that were typically originally installed for human operators. Systems for the detection and classification of vehicles on highways have successfully been using classical visual surveillance techniques such as background estimation and motion tracking for some time. The urban domain is more challenging with respect to traffic density, lower camera angles that lead to a high degree of occlusion, and the variety of road users. Methods from object categorization and 3-D modeling have inspired more advanced techniques to tackle these challenges. There is no commonly used data set or benchmark challenge, which makes the direct comparison of the proposed algorithms difficult. In addition, evaluation under challenging weather conditions (e.g., rain, fog, and darkness) would be desirable but is rarely performed. Future work should be directed toward robust combined detectors and classifiers for all road users, with a focus on realistic conditions during evaluation.**

***Index Terms*—Closed-circuit television (CCTV), intersection monitoring, road user counting, road users, traffic analysis, urban traffic, vehicle classification, vehicle detection, visual surveillance.**

I. Introduction

In RECENT years, there has been an increased scope for the automatic analysis of urban traffic activity. This case is due, in part, to the additional numbers of cameras and other sensors, enhanced infrastructure, and consequent accessibility of data. In addition, the advancement of analytical techniques for processing the video (and other) data, together with increased computing power, has enabled new applications. We define video analytics as computer-vision-based surveillance algorithms and systems to extract contextual information from video. The main concept is to aid human operators in observing video data. Video cameras have been deployed for a long time for traffic and other monitoring purposes, because they provide a rich information source for human understanding. Video analytics may now provide added value to cameras by automatically extracting relevant information. This way, computer vision and video analytics become increasingly important for intelligent transport systems (ITSs). This paper aims at introducing computer vision and video analytics and familiarizing the reader with techniques that are more progressively used in ITSs. To make this paper manageable, we concentrate here on infrastructure-side monitoring and, thus, do not consider the vehicle side and other mobile devices.

In urban environments, several monitoring objectives can be supported by the application of computer vision and pattern recognition techniques, including the detection of traffic violations (e.g., illegal turns and one-way streets) and the identification of road users (e.g., vehicles, motorbikes, and pedestrians). For the latter task, the currently most reliable approach is either through the recognition of number plates, i.e., automatic number plate recognition (ANPR), which is also known as automatic license plate recognition (ALPR), or radio frequency transponders, which may not be as easily acceptable for pedestrians or bicycles. Nevertheless, ANPR tends to be effective only for specialized camera views (zoomed on plates) and cannot provide wide-area observation or the measurement of the interactions between road users. This case may be possible with computer vision using standard cameras. Thus, for the aforementioned monitoring objectives, the detection and classification of road users is a key task. However, using general-purpose surveillance cameras (i.e., monocular), this challenge is demanding. The quality of surveillance data is generally poor, and the range of operational conditions (e.g., night time, inclement, and changeable weather) requires robust techniques. Traffic analysis on highways appears to be less challenging than in the urban environment. This case can be observed from detection and classification performance figures in the urban domain, i.e., 82.8% [95], 61% [49], 85% [92], and 65% [109], compared with the highway domain, i.e., 91% [58], $>$ 95% [73], and 89% [9]. Road users in urban environments also include pedestrians and bicycles, which are usually not present on highways. This paper will cover methods that are applicable for all urban road users; a survey specifically for pedestrian detection can be found in [38]. Note that, although monitoring

Manuscript received February 1, 2010; revised July 29, 2010 and December 23, 2010; accepted January 30, 2011. Date of publication March 17, 2011; date of current version September 6, 2011. This work was supported in part by the Directorate of Traffic Operations, Transport for London. The work of N. Buch was supported in part by Transport for London. The Associate Editor for this paper was S. S. Nedevschi.

N. Buch is with Kristl, Seibt & Co GmbH, 8054 Graz, Austria (e-mail: n.buch@theiet.org).

S. A. Velastin is with the Digital Imaging Research Centre, Kingston University, KT1 2EE Kingston upon Thames, U.K. (e-mail: sergio.velastin@ieee.org).

J. Orwell is with Faculty of Computing, Information Systems, and Mathematics and the Digital Imaging Research Centre, Kingston University, KT1 2EE Kingston upon Thames, U.K. (e-mail: james@kingston.ac.uk).

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.2011.2119372

using closed-circuit television (CCTV) systems has extensively been studied, there is relatively little research on situations where pedestrians and other road users are mixed or interact with one another. Even more so is the case for monitoring two wheelers such as motorbikes and bicycles, where much more research is needed.

The significant difference between traffic surveillance and, for example, generic object recognition is important for the understanding of the methods used. Object recognition tasks typically focus on high-resolution images (megapixel range), with few constraints on the viewing angle. The Visual Object Classes (VOC) challenge [39] gives precise definitions for classification, detection, and segmentation problems. The challenge provides image databases from the web (e.g., Flickr[1]), where an example scenario labels images to belong to generic classes such as person, horse, car, and house. A development kit is provided for evaluation according to the specification. Results can be uploaded to provide direct comparison between methods in academic workshops. The computational cost for classifying unconstrained images in such scenarios is usually higher than for traffic-monitoring applications.

In contrast to the aforementioned approaches, traffic surveillance systems deal with low camera resolution and, therefore, a limited amount of visual detail of road users. The monitoring objectives generally require real-time (RT) processing, which further constrains the complexity of the proposed approaches. The scenes are usually more constrained than object recognition problems, with cameras mounted on poles above roads. However, stringent reliability requirements are defined by operators. Cameras are typically assumed to be stationary, on a "home" position, unless operators take control over a camera. Several algorithms use this assumption to extract information when no operator observes the camera and information may otherwise be lost. In general, this surveillance task is not as well defined as image retrieval, and no benchmarking challenge has taken place so far. Fig. 1 shows some example camera views from the i-LIDS data set [55] provided by the U.K. Home Office. The i-LIDS data set aims at providing a benchmark for surveillance systems, with a focus on event detection. Video sequences under a range of realistic conditions are provided for sterile-zone monitoring (intrusion), parked-vehicle detection, abandoned-baggage detection (in metro stations), doorway surveillance, and multiple-camera person tracking.

Note that ITSs use and will use a variety of sensors in addition to video cameras. Other sensory modes for traffic-monitoring systems can include inductive loops, weather stations, radar scanners, and laser scanners. There is great potential in fusing information from different sensor sources to provide robustness in a combined method [52], [65]. It is beyond the scope of this paper to discuss details on the fusion mechanisms and on applications. However, the work on the literature on video analysis discussed here can serve as one input to such approaches. Surveillance cameras deployed for human operators are usually monocular, but stereo cameras can improve scene analysis by providing some depth information, in addition

Fig. 1. Example frames from the i-LIDS parked car data set [55]. (a) and (b) Sunny conditions with shadows and reflections on cars. (c) Image saturation in the upper part of the image. (d) Detail of a light car in the saturated area, where only dark elements remain visible. (e) Interlacing artifacts are commonly dealt with by removing every second video line and, therefore, halving the resolution. (f) Raining condition with reflections. (g) Rain during dusk. (h) Headlight reflections at night.

to the video, to disambiguate objects. There will be a short discussion on the use of stereo in currently deployed systems in the following sections, whereas details on stereo algorithms and methods can be found in [17] and [121].

This paper will specifically focus on recent approaches for monocular road-side cameras in urban environments used by human operators to provide automated solutions to the aforementioned monitoring problems. Information fusion with other data sources, e.g., radar, light detection and ranging (LiDAR), and inductive loops, may be applied to the result generated with the methods described in this paper; however, a detailed discussion is beyond the scope here. A previous survey [76] focused on highway surveillance and on-vehicle systems. A more comprehensive review of on-vehicle vision systems for driver assistance and autonomous driving can be found in [130]. A review of general surveillance systems is provided in [79] and [139], with a particular focus on distributed surveillance systems.

The remainder of the paper is organized as follows. We will first consider the deployment of video analytics in Section II to show where commercial (off-the-shelf) systems are in use.

[1]http://www.flickr.com/

A review of computer vision techniques used in traffic analysis systems is presented in Section III to analyze the underlying algorithms and methods. The state of the art for prototype and academic systems is analyzed in Section IV. For this analysis, the full surveillance task from reading a video stream to classifying road users and event recognition is described based on the techniques in Section III. In Section V, detailed discussions and an outlook to future research are provided.

II. Video Analytics Deployed in the Traffic Domain

This section reviews applications and existing commercial systems for traffic monitoring. The first part in Section II-A will focus on vehicle counting, which is mainly applied to highway scenes. ANPR is a very specialized application that is typically used for tolling and is discussed in Section II-B. The most challenging and least solved problem that holds the highest research potential is incident detection in Section II-C.

A. Vehicle Counting

The problem of vehicle counting is most commonly solved by deploying inductive loops. These loops provide high precision but are very intrusive to the road pavement and, therefore, come with a high maintenance cost. Most video analytics systems on highways focus on counting and, possibly, classification to allow for more detailed statistics [8], [24], [30], [59], [75], [137]. Some systems [8], [24], [75], [137] have also been adopted for urban environments, with cameras mounted on high poles. These poles are higher than standard CCTV poles and are difficult to install. The extra height provides a better viewing angle, which limits the occlusion between densely spaced vehicles, which results in similar conditions to highways. However, these highly mounted cameras are particularly designed for video analytics, because standard CCTV cameras for human operators are mounted lower. Following from this anonymous analysis of traffic statistics, the next section will investigate the identification of vehicles based on number plates.

B. ANPR

ANPR is a very specialized well-researched application for video analytics. There is a vast range of companies, e.g., [30], [75], and [144], that provide solutions for tolling, congestion charging, vehicle identification, or vehicle tax verification. Cameras are highly zoomed to provide a high-resolution image of the number plate but therefore lose the context of the scene. Active infrared lighting is often used to exploit the reflective nature of the number plate. The task is simplified by the fact that the number plate is intended to be communicated and uniquely identifiable. Toll stations of freeways have dedicated lanes with cameras, where registered users can slowly pass without stopping. In contrast, inner city congestion charge systems (e.g., Stockholm, Sweden; London, U.K., and Singapore) have to be less intrusive and operate on the normal flow of passing traffic (free-flow tolling). Point-to-point travel-time statistics are obtained from the reidentification of vehicles with time stamps across the road network. The next section will move away from the analysis of individual road users to a mere conceptual understanding of the traffic situation to identify incidents.

C. Incident Detection

Work on incident detection focuses on a higher level of scene understanding than the aforementioned two approaches. Examples for highways include the detection of accidents [8], [24], [30], [59], [137] and stopped vehicles. Tunnel surveillance also focuses on smoke detection for warning of tunnel fires such as the system in [24], which is based on the background estimation method described in [7]. Hard-shoulder running has been rolled out as a pilot project in the U.K., including video analytics in [59]. The hard shoulder of a motorway is turned into a running lane during peak time, which requires reliable inspection for obstacles and monitoring for incidents during operation. A similar pilot project was started in January 2010 on the Swiss motorway A1 near Geneva using stereo cameras. With the second specifically installed camera, depth information can be calculated to more robustly analyze scenes than with a single camera. Specific road-traffic-related methods of using stereo are introduced in [37]. Stereo is also used in [75] to confirm vehicle types for tolling applications. For details on stereo, see the survey in [17] and [121] and an RT implementation for intelligent vehicles in [141]. The remainder of this paper will focus on monocular cameras to provide added value to already widely deployed CCTV cameras.

Urban environments involve a much wider range of incident detection systems than highway surveillance and require an even higher level of scene understanding. Congestion detection is rolled out in London [27] based on existing CCTV cameras, including systems in [59]. Existing systems could not currently demonstrate acceptable results for practical deployment for other scenarios. The detection of illegal parking is the objective for one data set from i-LIDS [55]. A high level of tracking accuracy is required for illegal turning, bus lane monitoring, and box junctions, because the actual lane of vehicles has to reliably be identified. These target applications also require the classification of vehicles as discussed in [20] and significant context information from a zoomed-out camera. A system for detecting "car park surfing" is available in [59], which monitors if pedestrians move from one car to another. This case is regarded as usual behavior before a theft to identify target vehicles. The next section will focus on the algorithmic aspects of systems, therefore describing the computer vision methods in detail. Section IV will then describe research prototype systems published in the literature. The analysis of these prototype systems can indicate which direction the development of commercial systems described here may go.

III. Elements of Traffic Analysis Systems

In this section, we introduce generic elements required in a traffic analysis system. These methods and algorithms are useful to understand the workings of video analytics. To structure the presentation, we have grouped the literature into top–down and bottom–up approaches. This section is structured to mirror

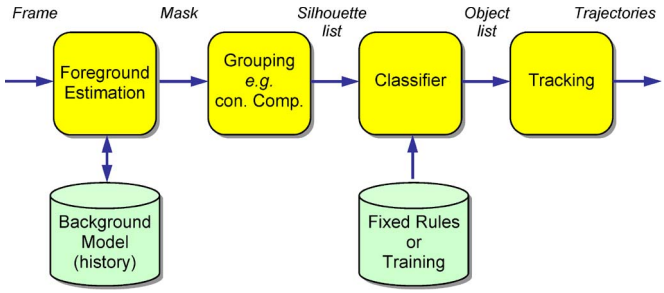

Fig. 2. Block diagram for a top–down surveillance system. The grouping of pixels in the foreground mask into silhouettes that represent objects is done early with a simple algorithm without knowledge of object classes.

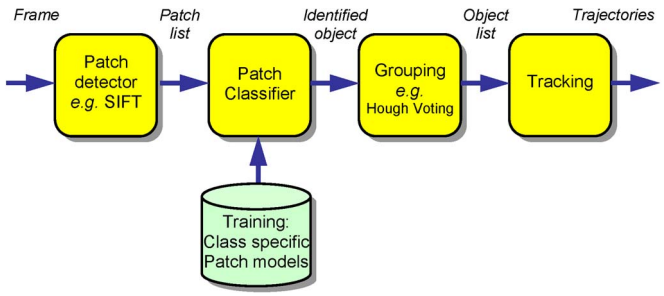

Fig. 3. Block diagram for a bottom–up surveillance system. Local image patches are first extracted from the input image and classified as a specific part of a trained object class. The identified parts are combined into objects based on the class through a grouping or voting process. Advanced tracking concepts [83] allow this grouping to be performed in the spatial–temporal domain, which directly produces an object trajectory rather than frame-per-frame object detections.

the processing pipeline of a typical video analytics application: foreground estimation (see Section III-A), classification (see Section III-B), and tracking (see Section III-D). See Fig. 2 for a block diagram. A statistical model typically estimates foreground pixels, which are then grouped with a basic model (e.g., connected regions) and propagated through the system until the classification stage; for example, see [14], [21], [29], [45], [51], [56], and [98]. Classification then uses prior information (previously learned or preprogrammed) about the object classes to assign a class label. For the remainder of this paper, we will refer to this class of algorithms as "top–down" or "object based," because pixels are grouped into objects early during the processing.

In contrast, we define a "bottom–up" approach as an approach that first detects and classifies parts of an object (see Fig. 3). This initial classification of the parts uses learned prior information (e.g., training) about the final object classes (e.g., an image area is classified to be a car wheel or a pedestrian head based on previously learned appearances of wheels and heads). The combination of these parts into valid objects and trajectories is the final step of the algorithm; for example, see [83], [85], and [104]. This type of approach is typically used in generic object recognition.

In the next section, we will first describe the top–down approach in more detail, including *foreground segmentation* and *top–down vehicle classification*. This approach is followed by relevant *bottom–up* classification approaches for traffic surveillance. The last section considers *tracking*, which can equally be applied after both classification methods.

A. Foreground Segmentation

Foreground estimation and segmentation is the first stage of several visual surveillance systems. The foreground regions are marked (e.g., mask image) for processing in the subsequent steps. The foreground is defined as every object, which is not a fixed furniture of a scene, where fixed could normally mean months or years. This definition conforms to human understanding, but it is difficult to algorithmically implement. There are two main different approaches to estimate the foreground, which both use strong assumptions to comply with the aforementioned definition. First, a background model of some kind can be used to accumulate information about the scene background of a video sequence. The model is then compared to the current frame to identify differences (or "motion"), provided that the camera is stationary. This concept lends itself well for computer implementation but leads to problems with slow-moving traffic. Any car should be considered foreground, but stationary objects are missed due to the lack of motion. The next five sections discus different solutions for using motion as the main cue for foreground segmentation.

A different approach performs segmentation based on whole object appearances and will be discussed in Section VI. This approach can be used for moving and for stationary cameras but requires prior information for foreground object appearances. This way, the review moves from the simple frame difference method in the next section to learning based methods in Section VI.

1) Frame Differencing: Possibly, the simplest method for foreground segmentation is frame differencing. A pixel-by-pixel difference map is computed between two consecutive frames. This difference is thresholded and used as the foreground mask. This algorithm is very fast; however, it cannot cope with noise, abrupt illumination changes, or periodic movements in the background such as trees. In [110], frame differencing is used to detect street-parking vehicles. Special care is taken in the algorithm to suppress the influence of noise. Motorcycles are detected in [102] based on frame differencing. However, using more information than only the last frame for subtraction is preferable. This approach leads to the background subtraction techniques described in the next sections.

2) Background Subtraction: This group of background models estimates a background image (i.e., fixed scene), which is subtracted from the current video frame. A threshold is applied to the resulting difference image to give the foreground mask. The threshold can be constant or dynamic, as used in [51]. The following methods differ in the way the background picture is obtained, resulting in different levels of image quality for different levels of computational complexity.

a) Averaging: In the background averaging method, all video frames are summed up. The learning rate specifies the weight between a new frame and the background. This algorithm has little computational cost; however, it is likely to produce tails behind moving objects due to the contamination of the background with the appearance of the moving objects.

The *instantaneous background* is used in [51] and [58]. This is the current frame but with detected objects removed. The regions of detected objects are filled with the old background pattern. By averaging the *instantaneous background* instead of the current frame, the tails generated by moving objects are reduced. The feedback of the motion mask could, however, lead to erroneous background estimations if the threshold is poorly set. A dynamic threshold is applied to reduce this problem of never updating a region detected as the foreground. The use of averaging, usually for computational reasons, is reported in [23] and [72]–[74].

b) Single Gaussian: To improve robustness compared to averaging, a temporal single Gaussian model can be used for every pixel of the background. Instead of using only the mean value for averaging, the variance of the background pixels is additionally calculated. This approach results in a *mean image* and a *variance image* for the background model. A new pixel is classified, depending on the position in the Gaussian distribution, which is the statistical equivalent to a dynamic threshold. A single Gaussian background model is used in [80], [97], [98], and [128].

c) Mode estimation: The mode of the temporal histogram of every pixel to estimate the background image is used in [164]. The mode estimation takes place in a constant time window. Robustness to illumination changes and long-term operation are not demonstrated in this paper. The algorithm described took 230 s for processing 600 frames on a Pentium 4 at 3 GHz and 1-GB RAM. For the mode in the histogram to correctly represent the background, the background has to dominantly be visible during the observation period to produce a dominant peak in the histogram. This condition is a similar assumption for a Gaussian mixture model (GMM) and holds for typical traffic surveillance applications but fails for parked vehicles or heavy congestion. However, the algorithm is sensitive to the bin size of the histogram. If the size is very small and the input pixel values vary over several bins, no distinct peak would appear. The GMM (see Section III) in comparison models the width of the distributions to tackle this parameter selection problem.

d) Kalman filter: A Kalman filter can be used to estimate the background image, where the color of each pixel is temporally modeled by one filter. The foreground can be interpreted as noise for the filter state. However, illumination changes are non-Gaussian noise and violate basic assumptions for the use of Kalman filters. In [94], a Kalman filter approach is proposed, which can deal with illumination changes. The illumination distribution over the image is estimated and used to adjust the individual Kalman filter states. The foreground estimation was tested in [95], indicating superior performance compared with the Kalman-filter-based algorithm proposed in [15].

e) Wavelets: A wavelet-based background model is introduced in [45] in the context of urban traffic surveillance. The evaluation indicates better performance than the GMM [125]; however, the test data are very limited in size. Following from the background subtraction that essentially estimates a single background image, more sophisticated statistical models will be introduced in the next sections.

3) GMM: The GMM was introduced in the seminal paper [125] and in [126]. Each pixel is temporally modeled as a mixture of two or more Gaussians and is updated online. The stability of the Gaussian distributions is evaluated to estimate if they are the result of a more stable background process or a short-term foreground process. Each pixel is classified to be a background if the distribution that represents it is stable above a threshold. The model can deal with lighting changes and repetitive clutter. The computational complexity is higher than standard background subtraction methods. Two images per Gaussian distribution used (typically three to five) have to be kept in the memory, which leads to 50 MB for a 720×576 color frame. The GMM for observing an intersection is used in [142], and [91] extends the GMM to deal with shadows (see Section V). For an introduction to GMMs, see [112]. The implementation in [64] is available in the OpenCV library [108] and is commonly used in research. Many researchers have adopted this model for traffic analysis [14], [21], [61], [142], [148], [161]. The limitation of the approach remains to be the computational complexity and, therefore, higher time requirements compared with the simpler approaches in Sections III-A1 and A2.

One alternative to the GMM is given in [133]. A probability density function (pdf) is used to detect objects in the picture. No explicit background image is kept. A pixel process is estimated for every pixel. Based on the estimated pdf, the probability for an observed pixel to occur is calculated. If the probability is high, no unexpected incident happened, and the pixel is assumed to be the background. If the probability is low, the pixel is assumed to be the foreground. The algorithm is very cost effective, because only an estimation for a GMM is calculated. The computation for every frame involves only an update and not a recalculation of the model. At a resolution of 320×240, the algorithms takes less than 80 ms on a Pentium 4 at 3.3 GHz and 2.5-GB RAM to segment a new video frame.

4) Graph Cuts: The foreground segmentation problem can be represented as a graph of a Markov random field (MRF). Every pixel of the images is represented by a node in the graph. The vertices between nodes and sources are set to a weight related to the data (data constraint). Sources represent the labels for a pixel (in this case, the foreground and the background). Vertices between nodes are used to introduce a smoothing constraint to avoid very small foreground or background regions. The graph cut completely separates the source and sink nodes and leaves the nodes connected to either source or sink node to indicate that this pixel corresponds to the respective label. The advantage of graph cuts is that the solution for this optimization problem can be found in polynomial time. A general introduction to graph cuts is given in [16]. Applications for image restoration, stereo imaging, and video blending are mentioned. Reference [136] is a tutorial for applications of graph cuts. Recent applications use graph cuts for scene understanding from moving vehicles [127]. A new more general marginal probability field (MPF) has been introduced in [155]. MRF is a special linear case of this new MPF. Having extracted foreground regions, the next section will show how unwanted detections such as shadows may be removed.

5) Shadow Removal: An evaluation of moving-shadow detection is given in [113]. The authors grouped the literature into four different categories. The first category is statistical nonparametric (SNP), which considers the color consistency of the human eye to detect shadows. One example of this approach is discussed in [31], which is used in several traffic systems [19], [22], [61], [161]. The statistical parametric (SP) approach imposes additional spatial constraints to SNP. Two different deterministic nonmodel-based approaches are described, which use a combination of statistical and knowledge-based assumptions. No single approach performs best; furthermore, the type of applications determines the best suited algorithm. Deepcast shadow positions are predicted in [61] based on their Global Positioning System (GPS) location, time information, and 3-D vehicle models. With this additional prior information, qualitative improvements are demonstrated, but no quantitative evaluation is provided.

A shadow removal technique using GMMs is introduced in [91]. Instead of using color consistency, the authors use the stability of states in the GMM to determine shadows. In contrast to the two groups of states in [125], one background state, several shadow states, and several foreground states are used. The concept assumes that shadow states are less stable (i.e., less frequent) than background states but are more stable than foreground states. Converged shadow states are copied into a Gaussian mixture shadow model (GMSM) to prevent them from being overridden by foreground states. This model calculates the shadow volume in the RGB space rather than assuming it to be a cylinder, e.g., for the color consistency assumption in [31].

All the aforementioned methods performed pixelwise reasoning to generate a binary foreground mask. The next section will look into considering larger image regions to directly segment whole objects.

6) Object-Based Segmentation: Object-based segmentation relies on object detection to identify the foreground. In this section, methods that detect objects in a holistic way by searching for full objects are considered. Reference [129] converts the wireframe of a 3-D vehicle into a gradient image by assigning a triangular gray-level profile to every edge. The projected image is compared to the gradient image of the camera to find a match. This work has been followed up in [57], [87], [117], and [163]. Optical flow is used in addition to wireframes in [109] to segment vehicles in the image.

Different methods are proposed to find correspondences between 3-D model projections and new images. Reference [95] generates the convex hull for 3-D vehicle models in the image. The ratio between the convex hull overlap of the model and the image normalized by the union of both areas generates a matching score. Similar 3-D vehicle models are matched with a motion-segmented input video in [124] for detection and in [21] for classification. An extension is provided in [61], which also adopts the size of vehicles. A method for rendering 3-D vehicle models for matching at new viewing angles is proposed in [50].

An approach with edges is used in [78]. Horizontal and vertical edges are grouped into vehicles using a probabilistic framework. The grouped vehicles are used for tracking in a highway surveillance application. All methods that employ 3-D modeling trade off additional prior information for higher computational complexity. A constant background color is assumed for highway scenes in [157]. This approach allows vehicle detection by simply taking the difference between the mean color and a pixel. The approach would not work in urban environments with street clutter. Whole objects were detected and directly segmented from the input image here. The next section will look into top–down classification, which uses the output of the foreground estimation to identify the class of foreground objects.

B. Top–Down Vehicle Classification

Classification is the task of assigning a new instance to a group of previously seen instances called the *class*. The classifier needs information about a new instance, which is usually referred to as *features*. Features are extracted from the whole object according to the aforementioned top–down methods. In Section III-B1, a selection of possible features is described. A machine-learning algorithm is trained with instances of known classes (hence, this approach is referred to as supervised learning) to extract discriminative information from the features (see Section II). The classifier then uses this learned information to assign a class label to a new instance.

1) Features: Classification and tracking relies on a feature extraction process, which ideally produces similar values for the instances of a given class throughout the video stream. This section gives an overview of different kinds of features, grouped by the support in the image as either a binary foreground region, the contour of this region, or larger image patches.

a) Region based: Region-based features are usually extracted from the whole image region of an object. In video sequences, this area is mainly the foreground silhouette extracted by the foreground segmentation algorithm. Image moments are often used to generate a feature vector for the silhouette. Without any feature generation, the convex hull of the silhouette (binary mask) can be used for comparison. Such an approach for region matching is used in [19], [21], [61], [95], and [124], and [51] and [159] use length and height to classify vehicles on a highway. Rule-based approaches are common. For example, [56] uses size and a *linearity* feature for vehicle classification. The *linearity* feature is a measure for the roughness of the vehicle silhouette. Size area, and length with a set of rules to classify vehicles in a highway scene are used in [58]. Occlusions between vehicles can produce similar effects (dents) on the silhouette, which is demonstrated in [161], where a similar *linearity* measure is used for occlusion reasoning. For vehicle classification, [97] and [98] use 17 different region features, including seven moments for seven classes. A comparison between image-based (IB) features, e.g., pixels and image measurement (IM) features, e.g., region size is given. Both feature types are used with the principal component analysis (PCA) and linear discriminant analysis (LDA) as the dimensionality reduction techniques. IM with LDA was used for the final algorithm, because it gave the best performance. The features are classified using a weighted k-nearest neighbor algorithm Section III-B2.4. A Kalman filter (see Section III-D1) is used to track the foreground regions based on the centroids.

The evaluation on a 24-h test sequence recorded by the authors shows a classification accuracy of 74.4% for independent tracking and classification. The accuracy can be increased to 88.4% by combining tracking and classification and, therefore, rejecting single misclassification. Further work of the authors incorporates histograms of oriented gradients (HOG) features for in-vehicle systems [43]. In [3], initial bounding boxes for vehicles are generated based on edges, which assumes that street clutter does not exhibit similar edge patterns. The bounding boxes are verified by symmetry and corner detection inside this region.

b) Contour based: Contour-based features only take the edge of a silhouette into account. The distance between contour points is used as a similarity measure. Processing is performed on closed contours as extracted from video sequences. The contour including edges is used in [57], [87] and [117].

One common problem when dealing with contours is occlusion between vehicles, particularly in urban environments. An algorithm in [161] can resolve occlusions between two vehicles by considering the convexity of the shape. The convex outline of a contour of two vehicles will have dents, which can be identified to separate two vehicles if the occlusion is not severe. The contour signature is used in [158] for vehicle classification from side views. Features discussed in this section will be passed to machine-learning algorithms for the training of feature models and to subsequently classify (i.e., compare) new features with those models.

2) Machine Learning: Machine-learning techniques are used to generate a discriminative classifier from training data and to assign class labels to unseen data. One important property of the learning technique is the supervision during learning. This approach describes the amount of labeling information required of the training data. Labeling can range from simply tagging an image with a class to completely manually segmenting the image and labeling individual parts of objects. Ground truth is similar information and required for evaluation. The classifier output for test data is then compared to this manually generated ground truth. Large amounts of ground truth are required to provide evaluation with high statistical confidence. Section V-B will look into common data sets, which is important to share the effort in generating this ground truth. A good overview of machine-learning techniques can be found in [90]. In the next sections, distance measures and clustering for training are introduced before discussing different classifier architectures.

a) Distance measures: Features are commonly represented as vectors in an N-dimensional feature space. This representation allows the definition of a distance (i.e., difference) between two vectors, which can be used during clustering and, in particular, classification to measure similarity between features. Several distance measures are available with various properties. First, the Manhattan distance calculates the sum of the absolute difference along every coordinate axis between the vectors. This condition results in the least computational effort but complex mathematics. Second, the Euclidean distance [44] returns the geometric distance between two vectors. Due to the square and square root, the computational complexity is increased. It is possible to normalize the Euclidean distance along every axis to reduce the effect of nonspherical data point clouds. The Mahalanobis distance is similar to a normalized Euclidean. The variance of the data along a coordinate axis is used for normalization. The covariance matrix of the data needs to be calculated for this reason. This normalization transforms the data cloud into a spherical shape. This distance is used for the training in [20]. For histogram comparison, the Bhattacharyya distance is used in [1]. The chi-square distance is used in [89] as a distance measure. It has similar properties to the Mahalanobis distance; however, it does not require the calculation of the covariance matrix. This paper describes a system for distinguishing between two classes of vehicles. The vehicles are presented centered at the image and at the same scale at high resolution to the algorithm, which is already a high degree of supervision. A set of modified scale-invariant feature transform (SIFT) features (see Section III-C1b) is calculated on edge points to give a rich representation for the image. Generated feature vectors are labeled according to the training vector with the smallest chi-square distance. A constellation model is used to find the most probable vehicle class based on the positions of the observed features vectors. This condition was evaluated for two separate cases of binary classification. In the first case, 50 cars and 50 minivans were randomly chosen from the sample pool for training. The testing was performed on 200 samples from each class taken from their own data. About 98% accuracy is reported for this case. The test between sedans and taxis resulted in a slightly lower accuracy. More detailed results are given for different shape models of the probabilistic framework (refer to Section III-C3 for shape models).

b) Dimensionality reduction: For feature vectors, not all dimensions are necessarily statistically independent. Dimensionality reduction can be applied to reduce the data to the signification dimensions and, this way, speed up processing or simplify classification. The classic method is PCA. This technique performs an orthogonal coordinate transformation of the feature space. The eigenvectors of the covariance matrix of the training data with the highest eigenvalues are used as new coordinate axes. This transformation ensures that the largest data variance is represented along the coordinate axes. Neglecting small eigenvalues that correspond to less significant deviations in the data reduces the dimensionality of the feature space. Reference [162] uses this concept with SIFT feature vectors to generate PCA–SIFT features. PCA has directly been applied on candidate images for vehicle detection at night time in [135], [118], and [119]. LDA, which is a similar concept, is used for vehicle classification in [98], and [23] uses independent component analysis (ICA), which separates the data into independent sources in addition to the orthogonal coordinate base of PCA. This paper introduces a vehicle classification algorithm. Standard foreground segmentation is performed to get bounding boxes of vehicles. The pixel values inside the bounding boxes form the feature vector. ICA is performed on the training images to reduce the dimension of the feature space. This approach is similar to the IB feature described in [97] and in Section III-B1a. To assign one of the three class labels to new feature vectors during operation, three one-class support vector machines (SVMs) are used (see Section III-B2d). The SVMs are trained with 50 vehicles each. Three tests are

conducted with 150 sample vehicles that were randomly chosen from the author's own sample pool. The reported performance is 65% recall at 75% precision. The ICA-based algorithm is shown to outperform a PCA-based baseline algorithm; however, [118], [119], and [135] report much higher performance with their PCA-based approach.

The following nonlinear embedding methods are compared in a review [140]:

1) isomap;
2) maximum variance unfolding;
3) kernel PCA;
4) diffusion maps;
5) locally linear embedding (LLE);
6) Laplacian eigenmaps;
7) Hessian LLE;
8) local tangent space analysis;
9) locally linear coordination (LLC);
10) manifold charting.

The distance measures and dimensionality reduction are used on original training data. The next section will discuss clustering, which can provide more meaning to the data by grouping data points together. The training data could, for example, be reduced by retaining only cluster centers for later classification steps.

c) Clustering: Clustering is performed on the training data. If the training data only contain object features, unsupervised clustering would need to identify the number of classes or clusters in the data and the correspondence of the training samples to those clusters. Because this general clustering problem is not satisfactorily solved, k-means clustering is commonly performed. This clustering technique groups the training samples into a specified number of groups based on the distance between features. This clustering technique for vehicle classification is used in [5] and [29]. Lighting conditions are clustered in [1]. Hierarchical clustering builds a cluster tree, which allows cutting off branches at different levels and sizes. Metrics other than the final cluster number can be used for this cutting, which allows more flexibility.

One related technique is the generation of a codebook or alphabet for object classification. This approach is usually applied if several local feature vectors are used to specify an object. The class label for every feature vector is known from supervision or from a previous clustering of the objects. The distance between feature vectors is used to group them together. Every group of feature vectors is replaced by one codebook entry that holds all class labels of the individual vectors. This approach can increase the speed of the final classifier and reduce the amount of training and data storage, as shown in [83], [85], and [105] for bottom–up object detection. The same concept is termed *visual dictionary* in [122] and is used for vehicle classification in [29], [152], and [154].

d) Classifiers: Classifiers map a new unknown object instance with an extracted feature vector to a known class or, perhaps, no class. This mapping process depends on what was previously learned from the training data. Different ways of generating and performing this mapping are outlined as follows.

Nearest neighbor classifier: The nearest neighbor classifier is the simplest nonparametric classifier for a feature vector. The distance between a new feature vector and every vector of the training set is calculated. Any distance measure can be used for this purpose. The class label of the closest training vector is assigned to the new vector. To improve robustness, the k-nearest neighbor algorithm can be used. The class label for the new class is determined by the k-nearest training vectors. Both methods require several distance calculations and do not scale very well for large training sets in terms of computational complexity and memory requirements. There is no time requirement for training; however, the classification time increases with the training size. This method to classify vehicles based on binary foreground features is used in [56] and [98]. In the seminal paper for SIFT [88], corresponding interest points are found using the nearest neighbor algorithm in the feature space. A further extension is the weighted k-nearest neighbor algorithm. For this case, the class membership is defined by weights, which results in a softer decision boundary. This algorithm to improve robustness to outliers in used in [97].

SVMs: An introduction and review of kernel-based learning used for SVMs can be found in [99]. SVM performs classification using linear decision hyperplanes in the feature space. During training, the hyperplanes are calculated to separate the training data with different labels. An SVM for vehicle classification is used in [23], [29], [34], [122], [135], and [154]. If the training data are not linear separable, a kernel function is used to transform the data into a new vector space. The data have to be linear separable in the new space. SVMs scale well for large training sets. The complexity for training increases with the number of training samples; however, the classification is independent of it. The generic approach does not provide a confidence measure for the classification. There are extensions that derive confidence based on the distance of a feature vector to the hyperplanes, which is not always reliable.

Probabilistic frameworks: Given that real-world measurements have uncertainty, probabilistic frameworks estimate the (posterior) probability based on observed data and prior knowledge. For example, the posterior probability of a vehicle that belongs to class A is calculated from the image data and the prior knowledge of how frequent vehicles of class A are observed. The vehicle detection system presented in [124] uses a Bayesian framework with Markov chain Monte Carlo (MCMC) sampling. First, a foreground map is computed using background subtraction. A proposal map is computed from the foreground map, indicating likely vehicle centroids. The distance of points from the boundary of the foreground map indicates the likelihood in the proposal map. A Bayesian problem is formulated for the vehicle positions. The proposal eliminates overlapping vehicles in 3-D space and is evaluated by the match between the foreground map and the projection silhouettes of the 3-D models. A MCMC algorithm is used to search for several good solutions. The MCMC generates new states by changing the number of vehicles, the positions, and the orientations. Tracking between frames is performed by a Viterbi optimization algorithm [146], which finds the optimal track through the set of solutions for every frame. Other works

[56], [78] use probabilistic frameworks for vehicle detection and tracking.

The aforementioned methods dealt with *top–down* classification, which analyzed objects as a whole. The next section will introduce *bottom–up* classification, where individual object parts are detected. This approach requires reliable part detections and classifications.

C. Bottom–Up Classification

This section discusses the literature for bottom–up approaches. An introduction to this concept, which is traditionally used for generic object recognition, is given in [111]. As discussed at the beginning of Section III, this approach involves detecting parts of objects and classifying them before they are grouped to objects. Section III-C1 introduces interest-point descriptors, which are used to extract discriminative features from images patches. Section III-C2 covers the learning technique of boosting, which has proved to be very powerful when used with interest points. Sections III-C3 and 4 introduce spatial models for interest points.

1) Interest-Point Descriptors: Interest points (also referred to as keypoints) are image positions, from which local features are extracted. These points may uniformly be sampled in the image space [34], [35] in a 3-D surface space [20] or defined by a saliency detector as in Harris corners, difference of Gaussians [88], and Hessian [11]. A comprehensive comparison of local patch features can be found in [96] and [160], including a temporal extension in [147], where it is shown that the performance of interest-point descriptors is mostly independent of interest-point detectors.

a) Basic patch based: The simplest patch-based feature vector is the collection of values of the image pixels. In [2], this approach is used to generate an alphabet of patches for object classification. The distance measure between patches is defined by their cross correlation. The correlation function is very sensitive to the size and illumination changes of the image. This fact encourages other feature transformations, which are more invariant, and therefore can deal with changing conditions. The following paragraphs introduce several solutions, followed by specific implementations in Sections III-C1b to e.

Using a histogram rather than pixel values allows for more spatial invariance. The seminal paper for those concepts is [88], followed up by several other algorithms [11], [20], [34], [96].

Binary edges can provide normalized input for feature descriptors. Illumination conditions are mostly removed during edge detection. The Canny edge detector to generate features is used in [78], [89], [105], and [106].

b) SIFT: SIFT was introduced in the seminal paper [88]. The local features generated are invariant to image scaling, translation, and rotation and are partially invariant to illumination changes and affine projection changes. The feature vectors are generated at the maxima of the scale space of the gradient input image. In addition to the 160-D feature vector, the characteristic scale and orientation of every interest point is calculated. Conceptually, a SIFT feature uniquely describes the appearance of salient points in the image, which will remain salient, even if the image is resized, rotated, or the illumination is changed. The SIFT features can be used to find point-to-point correspondences in two different images of the same object. SIFT features and other local features for generic object recognition are combined in [106], and [162] uses a derivation of SIFT, i.e., the PCA–SIFT, for generic object recognition. The local features are used in combination with global edge features in an adaptive boost (AdaBoost) classifier. Modified SIFT descriptors are used in [89] to generate a rich representation of vehicle images. Reidentified SIFT interest points between frames for tracking vehicles in urban scenes is used in [44].

c) SURF: The speeded-up robust features (SURF) descriptors are introduced in [11]. The descriptor aims for applications of correspondence finding between images, similar to SIFT and similar descriptors. However, the design focuses on computational speed, hence allowing for the loss of performance. The use of box filters instead of Gaussian filters in [88] reduces computational complexity. Haar wavelet responses in subregions around an interest point are used to generate the feature vector, which can efficiently be calculated with integral images this way.

d) HOGs: The concept of grids of HOG was introduced in [34]. To calculate the feature vector, the gradient input image window is divided into a grid of cells. For every cell, a HOG in pixels is calculated. The histogram represents an 8-D local feature vector. The vectors of all cells are concatenated to give one global feature vector for the image window. In the original paper, this vector is used to detect pedestrians. This concept is extended to vehicle detection in [20] by introducing 3-D *histograms of oriented gradients* (3-DHOG), which uses 3-D model surfaces rather than 2-D grids of cells. This approach allows the algorithm to resolve scale and use a single model for variable viewpoints of road users.

e) Other descriptors: There are a wide range of other descriptors introduced in the literature. The boundary fragment model (BFM) is introduced in the seminal paper [104]. The model uses only segments of contours for generic object recognition. The idea of local interest-point features as used in [28], [85], [88], and [106] and is extended to boundary elements. The Chamfer distance measure is used to generate a codebook of fragments in training and to classify newly seen boundary fragments to codebook entries. The use of a Canny edge detector to generate the boundary fragments allows the model to be used with still images. The concept of edgelets draws on similar ideas and is introduced in [156] for pedestrian detection.

Another extension of the SIFT descriptor is gradient location and orientation histogram (GLOH) in [96]. A larger feature vector with finer quantization than SIFT is extracted, and the dimensionality is reduced using PCA based on a large training set. All the aforementioned feature descriptors have to be used in conjunction with a classifier. The next section will introduce boosting, which has successfully been used to combine simple classifiers and to improve object detection performance by selecting "good" descriptors as discussed earlier.

2) Boosting: Boosting is a method of improving the performance of a simple (possibly poor) classifier. It is very popular in conjunction with local feature descriptors to also improve

computational speed by selecting an optimal subset of input features. AdaBoost was first introduced in [41] as an extension to boosting. An introduction to AdaBoost is given in [42]. AdaBoost uses weak classifiers, which only need to perform better than random. Weights for weak classifiers are learned during training. Every round of training changes the weights of training images to force the classifier to be trained on more difficult examples. The weighted weak classifiers result in a final strong classifier. The basic AdaBoost algorithm performs binary classification and is robust to overfitting. The original paper [143] uses a cascade of AdaBoost classifiers with underlying Haar filters for face detection. The success of this face detector increased the popularity of AdaBoost for computer vision. The same authors used a temporal extension of their algorithm for pedestrian detection in road surveillance [62]. Generic object recognition with a binary multilayer AdaBoost network is performed in [162]. In [104] and [106], binary AdaBoost is used for generic object recognition, where boosting automatically performs the feature selection. An extension to multiple classes and incremental learning is introduced in [105] and [107]. Boosting of gradient features to detect vehicles in road scenes is used in [77], and [1] uses a boosted classifier for illumination condition detection (e.g., day and night). After having selected good descriptors, the next section will introduce ways of modeling their spatial relationship.

3) Explicit Shape: Explicit shape implies directly modeling the spatial relationship between parts of objects detected. Various different models for the shape are introduced here, with relevance for traffic surveillance.

a) k-fans: The k-fan model was first introduced in [28] to schematize part-based object recognition. The parts of an object are divided into reference nodes and regular nodes of a graph. The parameter k represents the number of reference nodes. Every reference node has a spatial relation to every other node in the graph. By changing k from 0 to the total number of nodes, the spatial prior can be changed from no shape modeled to a full rigid structure. Most shape models are related to k-fans. Reference [78] use a one-fan model to group edges of a highway scene into vehicles. The camera calibration is used to model the 3-D appearance of vehicles. A constellation model similar to one fan for vehicle detection based on SIFT features is used in [89]. The HOG [34] and 3-DHOG [20] algorithms use a fully connected graph.

b) ISM: The implicit-shape model (ISM; one fan) is introduced in [85] and is explained in more detail in [82]. Image patches at key points of objects are learned during training. In addition to the object label, a pdf for the relative position in the object is provided. The evidence for an object position is accumulated based on these positions through generic Hough voting. In the case of Hough transform for line detection, every pixel of the image contributes to possible lines in the angle and position space. If several pixels vote for one angle and one position, this line is detected. A similar concept is used for object voting in [88]. Every detected SIFT interest point votes for its corresponding object centroid in the xy voting space. The maximum in this space defines the detected and classified object at a position. This method is extended using different features and distance measures in [26], [81], [83]–[85], [104], [106], and [107].

A similar approach is used in [2]; however, the relations between detected parts are used to generate a feature vector. Both methods use pixel values of the image patches. One good example for bottom–up surveillance based on ISM is [83], where road users are tracked from a static urban surveillance camera. The framework was first introduced in [81] based on a generic object detector [85] with ISM for vehicle detection from a moving camera. This work shows how bottom–up object detection approaches can be used for traffic analysis. The algorithm is demonstrated to perform in urban environments, similar to the state of the art on moving stereo, whereas most foreground segmentation methods discussed in Section III-A would not work for such a scenario. The limitations of this approach include lower detection ratios compared to typical bottom–up approaches and higher computational complexity.

c) Alphabets: The concept of alphabets is introduced to reduce the number of training samples. Instead of using every single feature vector from training, similar vectors are combined. The resulting entry holds a list of class labels and could take several positions in a shape model. This concept is used in [29], [84], [89], [105], [152], and [154].

4) Object Classification Without Explicit Shape Structure: A solution for generic object recognition without shape structure is given in [106] and is commonly referred to as a *bag of words* (zero fan). A large set of different key point features are extracted from images. An AdaBoost classifier is trained with these features. This training procedure automatically selects the most discriminative features for the final classifier. An additional boosting layer is introduced in [162]. This second layer uses global features to improve the classification.

a) Object recognition with hierarchy: The introduction of hierarchy in object recognition is mainly related to biological research. For example, [138] discusses the structure of the human visual cortex and derives a tree-style object hierarchy. The features of objects are based on image patches. Reference [122] presents something that is more relevant for computer vision applications. In this paper, a complete object recognition and segmentation system is implemented using a visual cortex structure. Four layers are used, which perform simple filtering, complex searching, and a repetition of these two steps. Comparable results to state-of-the-art computer vision are achieved with this biologically inspired system. This concept is used in traffic surveillance in [29] and [152]–[154]. A standard foreground estimation method and a motion tracker (no details are provided) generate vehicle images, which are passed through a sequence of simple and complex layers represented by Gabor filters and a SVM classifier. A different appearance classifier is trained for every 90° of viewing angle. In contrast, the algorithm in [20] can operate on arbitrary viewing angles. On the same data set, [153] outperforms [89]. The authors have moved this concept toward a bottom–up approach in [29], [152], and [154]. Feature vectors are now extracted from interest-point locations rather than a uniform density over the whole image patch.

Having covered methods for the detection and classification of road users from a *top–down* and a *bottom–up* perspective,

the next section will show how tracking can be used to estimate road user trajectories.

D. Tracking

Tracking is used to measure vehicle paths in video sequences. This approach is performed in the following two steps: 1) features for the object or foreground regions are generated in every video frame (see Section III-B1 and 2) a data association step has to provide correspondences between the regions of consecutive frames based on the features and a dynamic model. Temporal consistency constraints are required to avoid confusion of tracks and to smooth noisy position outputs of detectors. The data association step can use the same distance measure as machine-learning algorithms (see Section III-B2a). The classification result and location in the image is typically included in the feature for this association. The next sections discuss motion models for tracking in traffic applications and possible data association based on prediction.

a) Kalman filter: The Kalman filter was originally introduced in [66] and has successfully been used in several applications, including missile tracking. The optimal state of a linear time-invariant motion model is estimated, assuming a Gaussian process and measurement noise. The prediction stage of the Kalman filter is used to extrapolate the position of objects in a new frame based on a constant velocity constraint. The prediction can be associated with new measurements or can be used to trigger detectors. A correction step uses the detection as measurement and updates the filter state. A good introduction is provided in [151]. This concept is used in [14], [61], [63], [95], [97], [116], and [124] for tracking. In [22], the sample rate of the Kalman filter is altered to account for video frames where there is no detection. Kalman filters propagate a single object state between frames compared to multiple hypotheses for particle filters (PFs) in the next section. The extended Kalman filter (EKF) can facilitate nonlinear models.

b) PF: The PF is a generalization of the Kalman filter introduced in [48]. A recent tutorial [36] reviews the filter and relevant concepts. It allows for multiple hypotheses to be propagated between frames by modeling arbitrary pdf's by sample particles. This approach overcomes the constraint of a single Gaussian distribution of Kalman filters. The PF into the computer vision domain is introduced in [60]. The filter is used for traffic videos in [9], [44], [93], [102], [103], and [148].

c) S–T MRF: The spatial–temporal Markov random field (S–T MRF) is introduced in [68]–[70] for vehicle tracking in urban traffic scenes. The input image of a resolution of 640×480 is divided into blocks of 80×60 pixels. Every block is represented by a node in a S–T MRF, which is modeled as a graph similar to Section III-A4. The S–T MRF is used to generate vehicle labels for the blocks. Adjacent blocks and blocks in consecutive frames are considered neighbors for the model. A solution for the object map (nodes of the S–T MRF) of the current frame is found based on the current image, the previous image, and the previous object map. The result is used in a hidden Markov model (HMM) to detect events such as vehicle passes or collisions. References [67] and [71] are extensions to the earlier work that introduced incident detection in tunnels.

d) Graph correspondence: A system for region tracking based on graph correspondence is introduced in [51] for vehicle tracking. Every region in a frame is represented by a node in the graph similar to MRF. One vertex that leaves every node is generated for two consecutive frames. The destination node of the vertex is determined by the best overlap score of the image regions. Due to this bidirectional structure of the graph, the splitting and merging of region during tracking can be handled. To avoid conflicts in the graph, adding conflicting vertexes is suppressed. The graph correspondence for vehicle tracking and classification is used in [58], [86], and [142]. In [131], vehicle and pedestrian tracking is evaluated on the CLEAR data set [25] and uses greedy graph correspondence tracking based on [123]. Dynamic programming approaches can be used to find an optimal path through nodes of several frames. Reference [124] uses the Viterbi algorithm [40], [146] to find the optimal vehicle constellation over several frames.

e) Event cones: The concept of event cones for finding space–time trajectories is introduced in [81]. Every object observation in a frame is assigned an event cone, which, in turn, represents a volume of possible object positions in the future and the past. The shape of the cone is determined by the dynamic model of the object, similar to Kalman filters. Object detections of all frames are accumulated to allow a probabilistic framework with an optimization step to select the optimal set of trajectories to explain the full history of observations. This condition allows tracks to retrospectively be split, which is traded off against the optimization of a growing data set for long video sequences. In addition, a RT scene-understanding system might be presented with a continuously changing interpretation of the past video. The performance of this approach is demonstrated for an urban surveillance task in [26] and [83]. Having discussed individual parts of video analytics systems, the next section will bring them together by a discussion on whole prototype systems.

IV. Complete Traffic Analysis Systems

This section covers traffic surveillance systems that could be used in a control room environment for traffic management. By distinguishing between urban and highway scenes, a higher coverage of highway applications in the literature is shown similar to the deployment discussed in Section II. This case is partly due to the easier conditions on a highway with usually more homogeneous and constant flow than in urban areas. In addition, the distance between vehicles is larger and reduces the amount of occlusion. Fig. 1 shows some challenging examples from an urban environment. An overview and grouping by the complexity of systems is provided in Table I.

A. Urban

The challenge for monitoring urban traffic is the high density of vehicles and the low camera angle. The combination of both factors leads to a high degree of occlusion. In addition, the clutter on the streets increases the complexity of scenes. The literature is divided into 2-D approaches, which operate in the domain of the camera view, and 3-D approaches

TABLE I
OVERVIEW OF VIDEO ANALYTICS SYSTEMS (ANY SYSTEM NAME IS SET IN BOLDFACE)

	Real-time	Ref	Algorithms	Performance
Urban	Yes	[95]	3D convex hull matching 8 vehicle classes **(SCOCA)**	91.5% classification accuracy on 45 minutes video
		[21]	3D model matching against GMM silhouettes	89.8% classification accuracy on 1 hour video from i-LIDS [55]
		[142]	Graph correspondence tracking of oriented motion bounding boxes	12-15 fps on low resolution images
		[110]	Perspective normalised frame difference	24 hour video to detect 94.7% of parking events
		[4]	3D model matching against motions mask from multiple camera with calibration [6]	35 fps detecting 85% of vehicles with road coordinates
		[102]	Particle filter tracking of motion from background subtraction	20 fps for tracking 99% of motorcycles in 2 minutes video
	No	[124]	3D model matching against motion mask for single car	96.8% and 88% detection rate on two test videos
		[20]	Appearance model: 3DHOG to classify road users (5 classes)	92.1% classification accuracy on 1 hour video from i-LIDS [55]
		[68, 70, 71]	Spatio-temporal Markov Random Field	90% of events (vehicle passing, near miss) detected in 25 minutes of low resolution video
		[150]	Support vector regression for motion estimation. Infrared input	98% road user detection on two hours of video
		[109]	Tracking of vehicles from optical flow and wire frame **(Motris, Xtrack)**	65% correct tracks on strict assessment on video from [100]
		[120]	Feature points of vehicles are tracked	85% to 90% tracking performance on data from [100]
		[61]	GMM background estimation with shadow removal; 3D models are matched and shadows predicted	No information about speed or quantitative performance
		[49]	GMM with agent based tracking with occlusion reasoning **(W4)**	61% correct tracks on 30 minutes of video

(see Section II), which employ some degree of 3-D modeling or reconstruction. Both approaches demonstrate comparable performance.

1) Analysis in the Camera Domain: This section deals with systems that directly work in the camera coordinate domain. An early RT monitoring system for intersections is proposed

TABLE I
(*Continued*). OVERVIEW OF VIDEO ANALYTICS SYSTEMS (ANY SYSTEM NAME IS SET IN BOLDFACE)

Highway	Yes	[98]	Single Gaussian background model with size based vehicle classifier	10-15 fps to classify 2 hours of video with 88.4% accuracy
		[116]	*Instantaneous backgroud* like vehicle detection and tracking with Kalman filter	11 fps to process 16 seconds of video with 5.4% tracking error
		[129]	Matching of 3D wire frames with gradient images	100% correct detection on 1 minute of motorway video
		[78]	Probabilistic edge grouping to detect cars	10 fps for 85% detection rate of vehicle
		[148]	Particle filter vehicle tracking	15 fps for a three lane highway
		[9]	Markov chain Monte Carlo Particle Filter for tracking	20 fps to track 89% of vehicle correctly for 5 seconds of video
		[73]	Vehicle detection through height constraint	32ms per frame resulting in tracking accuracy in excess of 90%
	No	[51]	Instantaneous background silhouette tracked with graph correspondence	15 fps to successfully track 90% of vehicle in 20 minutes of video
		[93]	Particle filter for appearance tracking of whole vehicles	Qualitative evaluation only; low resolution, high vantage point video
		[89]	Vehicle classification with SIFT features, which relies on previous successful tracking	98% classification accuracy for 500 vehicle images
		[58]	Optical flow based vehicle detection and classification	91% classification accuracy on 5 minutes of video
		[56]	Estimation of highway lane centres and vehicle tracking	82% detection and 93% classification accuracy on 10 minutes video
		[128]	Collaborative background extraction and rule based shadow removal with tracking	Recall of 95.6% for detection over 500 frames

in [142]. A standard GMM is used for foreground segmentation. The tracking of foreground regions is done with graph correspondence. The tracked objects are classified into pedestrians and vehicles based on the main orientation of the bounding box. Example images for different weather conditions are shown; however, there is no quantitative evaluation of the performance of the system. A support-vector-regression-based background model is used in [150]. Shape-based data association in tracking feeding back to the detection is shown to significantly improve the results. A multiagent framework performs tracking under occlusions in [49]. Tracking performance of 61% is reported on 30 min of the authors' surveillance video.

A system for detecting parked vehicles is introduced in [110]. Camera homography is used to generate a normalized ground plane view. Based on a frame-differencing motion map, *parking-in* and *parking-out* conditions are calculated. A state machine is used to track the speed changes of vehicles until stopping to generate these conditions. The system is evaluated with 24 h of video data from two different sites. A detection rate of 94.7% is reported on their own data for parking.

Interest points are independently tracked at urban intersections in [120]. This condition provides robustness to errors in the background estimation and can deal with changing viewing angle, because no prior assumption about the constellation of feature points is made. The tracking performance is between 85% and 94%, depending on the data set. Whole vehicle parts rather than individual points are tracked with PFs in [93], operating on very low-resolution images.

Finally, there are two papers that look at specialized urban traffic applications. Reference [102] focuses on motorcycle tracking with multimodal PFs. A recall rate for counting of 99% is demonstrated for videos from Vietnam. In an urban setting in Venice, boats are tracked in [14]. GMM is combined with optical flow and a Kalman filter to track and count boats along the Grand Canal. The counting accuracy is 94% for a 2-h sequence, which is particularly challenging due to waves on the water. The next section will focus on methods that use 3-D information to improve robustness, which potentially requires more prior knowledge (e.g., camera calibration) than camera-domain methods.

2) Three-Dimensional Modeling: Systems in this section use explicit 3-D modeling. A RT system is introduced in [95] to track and classify vehicles at intersections. Three-dimensional models are used to initialize an object list for every fifth frame based on the convex hull overlap of model projection and motion map. Camera calibration is required for this operation. A feature tracker follows the detected objects along some frames before a new initialization takes place. The tracker is used to speed up the operation, because the 3-D operation would not be fast enough to operate on every frame in RT. The objects are classified into eight classes based on a two-stage classifier. The first stage evaluates the convex hull, whereas the second layer uses pixel appearance (color) for classes with similar convex hull. The performance is evaluated on 45 min of video data from two different sites. The total classification rate is given with 91.5% for the test data of the authors.

The work in [19], [21], and [22] builds on this approach and uses a GMM to compare motion silhouettes with 3-D models for the classification of five types of road users. An appearance model of 3-DHOG is introduced in [18] and [20]. This approach integrates local patch features from the bottom–up object recognition domain with 3-D models from the top–down domain.

The use of 3-D wireframe models for vehicle detection and classification was proposed in [129] and [132]. First, a hypothesis for a vehicle position is generated in a search window. For this approach, 1-D profiles along three axes of cars are correlated with trained templates. The hypothesis is verified by correlating the gradient input image with the wireframe image. The wireframe image is generated using the camera calibration to project the wireframe and replace every line with a 3-pixel-wide triangular gray-level profile. This line of research is followed up in [57], [87], and [117]. A similar work using optical flow to find detection regions is presented in [109], with previous work in [32] and [33]. The 3-D wireframes of vehicles are used in a Hough transform to provide additional cues for vehicle detection. Only four-vehicle models are provided, which leads to a low detection rate of 65% on video data in [100].

A Bayesian framework with MCMC sampling is used in [124]. A proposal map is computed from the foreground map, indicating likely vehicle centroids based on a constant-size vehicle model. The evaluation on two video sequences shows detection rates of 96.8% and 88%. The method is extended to predict (and, hence, remove) shadow projections. However, only shadows from known lighting conditions can be dealt with, e.g., sunlight. This work is extended in [61] by incorporating multiple-vehicle models but requires manual setup of vehicle orientation. The performance of this algorithm is not quantitatively evaluated. Automatic generation of 3-D vehicle models has been investigated in [47].

A completely different 3-D approach was presented in [78]. An edge detector is applied in an entry window to the side-view highway image to retrieve horizontal and vertical lines of vehicles. These line features are grouped together, using a probabilistic method, to form vehicles based on a 3-D line model. Once vehicles have been detected in the entry window of the scene, they are tracked using cross correlation between frames. The detection rate compared to hand counting is reported to be 85%.

The problem of collision detection in urban intersections is tackled in [4] and [6]. Multiple cameras, calibrated according to [92] with road primitives, are used to identify 3-D ground plane locations of vehicles by projecting all foreground masks to the road plane. Based on the authors' data, 85% of 273 vehicles are successfully detected. The next section will briefly introduce the latest developments for highway surveillance as shown in the light of the work done for the urban domain. At the end of the section, Table I will then provide an overview of traffic analysis methods with respect to their complexity.

B. Highways

Observing highway scenes usually gives the advantage of high camera angle and homogeneous traffic flow. A comprehensive review [76] focuses on this topic. Newer references are discussed here and divided into detection (see Section IV-D1) and classification (see Section IV-D2).

1) Detection: A region-based vehicle detection and classification system is proposed in [51]. The main focus of this paper is on detection, with effort put in a fast background estimation using the *instantaneous background* to get good segmentation. Tracking is performed using graph correspondence based on motion silhouette overlaps. The proposed classifier uses two classes (cars and noncars) with size-based features. The camera calibration is required to normalize these features. On the 20-min validation sequence, 70% of vehicles are correctly classified. Tracking using PFs is introduced in [148]. The system is motivated by generic surveillance, but results are shown for their own highway video sequence. A MCMC PF is used in [9] to track vehicles detected with simple frame differencing as a background model. On a short 50-frame sequence, up to 89% of vehicles was correctly tracked.

The traffic system proposed in [56] allows vehicle detection and classification into four classes. The camera is assumed to be in axis with the highway. This assumption allows the estimation of the lane centers by using the tracks (by Kalman filter) of

vehicle centroids. The lane centers are used to calculate the lane division lines. These lines are used to separate vehicle blobs that are merged due to shadows. The vehicles detected are classified based on size and the *linearity* feature. This *ad hoc* feature is a measure of the roughness of the blob. A Bayes classifier based on the Mahalanobis distance between feature vectors with constant prior is used for classification. The performance is evaluated on 10 min of video data from three different sites. The reported detection accuracy is 82%. Out of the detected vehicles, 93% is correctly classified using cues from multiple frames. Higher detection accuracy is reported in [149]; however, the test video exhibits less occlusion. A rule-based framework that deals with shadows and occlusions is introduced in [128]. A recall of 95.6% is reported on 500 frames from a proprietary video of nonoccluded vehicles on a highway.

In contrast, vehicle detection from cameras on the roadside using height features is introduced in [74] and followed up in [72] and [73]. With camera calibration, the height of interest points is estimated throughout the video based on a foot point constraint of the bottom of a motion silhouette. This approach allows effective grouping of points into cars and trucks. The segmentation and tracking performance exceeds 90%.

2) Classification: A system for tracking and classifying vehicles on highways is proposed in [116]. Vehicles are first classified into three classes based on the width of the bounding box and the traveling speed. The classified bounding boxes are tracked using a Kalman filter. The reported tracking error rate is 5.4%.

A motion segmentation and classification algorithm is described in [58]. Seven vehicle types are classified from side-view motorway images. Blob features such as length and compactness are used with a rule-based classifier. The *instantaneous background* update model is used. Merged blobs of different vehicles are separated using dense optical flow fields. However, this method only works if there is a speed difference between occluding vehicles. The performance is evaluated with a test sequence that lasts for 463 s, which results in a 91% overall classification rate.

V. Discussion

This section will discuss challenges in traffic surveillance, particularly in the urban domain. One major aspect is common data sets, which are analyzed in Section V-B. Future research directions are given in Section V-C.

Classical visual surveillance approaches of background modeling and tracking have successfully been applied for highway surveillance [9], [73], [97]. There are attempts to overcome the problem of occlusion and shadows for this type of scene. Urban environments are more challenging due to denser traffic, variable orientation of vehicles at intersections, and lower camera positions. More advanced approaches have been suggested, including 3-D models [21], [87], [95], shadow prediction [32], [124], and appearance models [78], [89]. Algorithms that were developed for the generic object recognition domain have been applied and show promising results in the urban traffic domain [20], [83], [153], [154].

A. Challenges

From an application perspective, the main technical challenge is the diversity of camera views and operating conditions in traffic surveillance. In addition, a large variety of observation objectives, e.g., vehicle counting, classification, incident detection, or traffic rule enforcement, can be useful. This condition has generated a large diverse body of work, where it is difficult to perform direct comparison between the proposed algorithms. It would be beneficial for the community to define a set of clear tasks as done in object recognition with PASCAL [115]. The main contribution of a challenge such as this paper is a public data set, which would be very beneficial for the community. The next section introduces a few available data sets. One possible reason for the lack of a common framework is the diversity of traffic rules and car classes around the world. Research always seems tailored to local environments, even if it only means adopting vehicle classes according to local traffic regulations. There is very limited literature in vehicle surveillance that deals with night time [13], [46], [118] and difficult light [53], [61]. There has been little work to cover wide ranges of weather conditions and lighting for this application. However, there is a significant body of work that deals with image restoration under these conditions [10], [54], [134]. Having input data available for weather scenarios may encourage researchers to use such methods in the context of traffic surveillance. There is little work dedicated to specific conditions, and algorithms are rather generically designed, with the hope that they would work for hard examples. To cover all possible situations and conditions, there might be the requirement for a bank of detectors, which are switched based on illumination as shown in [1] and [135].

The main technical challenge in urban environments includes occlusions and dense traffic. There are several solutions for occlusion handling in highway scenes [56], [73], [128] for relatively sparse traffic, which cannot necessarily be transferred to urban environments. Shadows also cause motion silhouettes of vehicles to merge. Without clear wide lanes, the splitting of silhouettes cannot as easily be performed as done in [56]. The introduction of 3-D models shows promising results and allows occlusion prediction or, at least, the modeling of a nonoverlapping 3-D constellation of vehicles for a given scene. The next section will discuss public data sets, which may be used to evaluate algorithms for the aforementioned challenges.

B. Data Sets

Public data sets and evaluation would allow researchers to objectively compare algorithms. In addition, labeled training data are essential for the training of machine-learning algorithms discussed in Section III-B2). Unfortunately, most authors use their proprietary data, which is rarely made available on the web. Even with videos available, ground truth is scarcer and very often application dependent. The i-LIDS data set [55] is an attempt by the U.K. Home Office to benchmark visual surveillance systems based on the requirements of users. One scenario deals with illegally parked cars in urban roads and

consists of 24 h of video. There is only event-based ground truth, which is of limited use for the evaluation of low-level algorithms. Tracking ground truth is available for parts of videos through [25], with a vehicle and pedestrian tracker evaluated in [131]. The data set has also been used in [20]–[22]. Grayscale images of urban intersection from a long-distance high-vantage point view are provided in [100] and are used in [32] and [33]. Image patches in [89] are available[2] as Matlab data files. Similar image patches are repeatedly used in [29], and [152]–[154], but no direct download is provided. Data for more general visual surveillance with some traffic related scenes are available in [145].

In general, all the data lack challenging weather and lighting conditions. Evaluation data for those conditions are essential to benchmark systems for full 24/7 operation as it may be required by traffic managers. Setting a well-defined challenge in conjunction with realistic well-balanced video data would be a valuable contribution to the field.

C. Future Research

There is a larger body of work that deals with vehicle detection than with classification. For several applications, knowing the class of road users is essential. Some combined detectors and classifiers have been proposed [20], [83], [87], [154]. Future classifiers should take tracking prediction into account. According to several studies [98], [150], the combination of both approaches improves the results. The recent 3-DHOG algorithm [20] is a good example for an appearance-based classifier, which can incorporate tracking predictions as initial hypotheses for new frames. The task of classification of vehicles should be pursued to increase the capabilities to the level of detection and tracking.

After the low-level detection and tracking has been tackled, there is great potential for traffic rule enforcement. Current systems mainly focus on basic counting in highway and urban scenes. More sophisticated analysis of road user interaction is desirable in urban environments, particularly including cyclists and pedestrians. Intelligent traffic light timing could benefit from a measurement of the state (e.g., position, velocity, and class) of all road users at an intersection. The currently common installations of inductive loops in several cities cannot provide such comprehensive data.

One emerging application area to consider is the communication of vehicles (C2X). This approach could either be vehicle to vehicle or vehicle to infrastructure, as outlined in [12] and [114]. The latter approach is of relevance for the technology surveyed in this paper. On the one hand, information from vehicles (location and speed) could be fed to video analytics for fusion and subsequent increase in performance. This way, information directly from the vehicles constitutes a further input, in addition to other possible sensors such as LiDAR and inductive loops. On the other hand, (meta) information or raw images from the cameras could be passed to nearby vehicles. This approach would allow intelligent vehicle applications, e.g., as reviewed in [130], but image data could be available earlier and outside the line of sight. Prototypes and demonstrators for such a system have been developed and successfully tested in Japan [101].

[2]http://people.csail.mit.edu/xiaoxuma/proj/

VI. Conclusion

We have presented a comprehensive review of computer vision techniques for traffic analysis systems, with a specific focus on urban environments. There is increasing scope in intelligent transport systems to adopt video analysis for traffic measurement. The research expands from the highway environment to the more challenging urban domain. This condition opens many more application possibilities with traffic management and enforcement. Traditional methods use background estimation and perform top–down classification, which can raise issues under challenging urban conditions. Methods from the object recognition domain (bottom–up) have shown promising results, overcoming some of the issues of traditional methods, but are limited in different ways. Clearer definitions of scenarios and applications are required to generate a more consistent body of work, which uses common data for comparable evaluation and better fusion of top–down and bottom–up algorithms. New application areas are likely to emerge from vehicle-to-vehicle and vehicle-to-infrastructure communication, where videos from traffic cameras could be passed to cars for processing.

References

[1] D. Acunzo, Y. Zhu, B. Xie, and G. Baratoff, "Context-adaptive approach for vehicle detection under varying lighting conditions," in *Proc. IEEE ITSC*, Sep. 30–Oct. 3, 2007, pp. 654–660.
[2] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse part-based representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1475–1490, Nov. 2004.
[3] D. Alonso, L. Salgado, and M. Nieto, "Robust vehicle detection through multidimensional classification for on-board video-based systems," in *Proc. IEEE ICIP*, Sep. 16–Oct. 19, 2007, vol. 4, pp. 321–324.
[4] S. Atev, H. Arumugam, O. Masoud, R. Janardan, and N. P. Papanikolopoulos, "A vision-based approach to collision prediction at traffic intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 4, pp. 416–423, Dec. 2005.
[5] S. Atev, G. Miller, and N. P. Papanikolopoulos, "Clustering of vehicle trajectories," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 647–657, Sep. 2010.
[6] S. Atev and N. Papanikolopoulos, "Multiview 3-D vehicle tracking with a constrained filter," in *Proc. IEEE ICRA*, May 2008, pp. 2277–2282.
[7] D. Aubert, F. Guichard, and S. Bouchafa, "Time-scale change detection applied to real-time abnormal stationarity monitoring," *Real-Time Imag.*, vol. 10, no. 1, pp. 9–22, Feb. 2004.
[8] Autoscope. [Online]. Available: http://www.autoscope.com
[9] F. Bardet and T. Chateau, "MCMC particle filter for real-time visual tracking of vehicles," in *Proc. IEEE 11th ITSC*, Oct. 2008, pp. 539–544.
[10] P. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *Int. J. Comput. Vis.*, vol. 86, no. 2/3, pp. 256–274, Jan. 2010.
[11] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded-up robust features," in *Proc. ECCV*, vol. 3951, Lect. Notes Comput. Sci., 2006, pp. 404–417.
[12] M. Bechler, T. M. Bohnert, S. Cosenza, A. Festag, M. Gerlach, and D. Seeberger, "Evolving the European its architecture for car-to-x communication," in *Proc. 16th ITS WC*, Stockholm, Sweden, Sep. 2009, pp. 1–8.
[13] L. Bi, O. T. Liu, and Y. Liu, "Using image-based metrics to model pedestrian detection performance with night-vision systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 155–164, Mar. 2009.
[14] D. Bloisi and L. Iocchi, "Argos—A video surveillance system for boat traffic monitoring in Venice," in *Proc. IJPRAI*, 2009, pp. 1477–1502.

[15] M. Boninsegna and A. Bozzoli, "A tunable algorithm to update a reference image," *Signal Process.: Image Commun.*, vol. 16, no. 4, pp. 353–365, Nov. 2000.

[16] Y. Boykov and O. Veksler, *Mathematical Models in Computer Vision: The Handbook—Graph Cuts in Vision and Graphics: Theories and Applications*. New York: Springer-Verlag, 2005, pp. 100–119.

[17] M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in computational stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 993–1008, Aug. 2003.

[18] N. Buch, M. Cracknell, J. Orwell, and S. A. Velastin, "Vehicle localization and classification in urban CCTV streams," in *Proc. 16th ITS WC*, Stockholm, Sweden, Sep. 2009, pp. 1–8.

[19] N. Buch, J. Orwell, and S. A. Velastin, "Detection and classification of vehicles for urban traffic scenes," in *Proc. Int. Conf. VIE*, Jul. 2008, pp. 182–187.

[20] N. Buch, J. Orwell, and S. A. Velastin, "Three-dimensional extended histograms of oriented gradients (3-DHOG) for classification of road users in urban scenes," in *Proc. BMVC*, London, U.K., Sep. 2009.

[21] N. Buch, J. Orwell, and S. A. Velastin, "Urban road user detection and classification using 3-D wireframe models," *IET Comput. Vis.*, vol. 4, no. 2, pp. 105–116, 2010.

[22] N. Buch, F. Yin, J. Orwell, D. Makris, and S. A. Velastin, "Urban vehicle tracking using a combined 3-D model detector and classifier," in *Knowledge-Based and Intelligent Information and Engineering Systems KES*. Santiago, Chile: Springer-Verlag, Sep. 2009, pp. 169–176.

[23] X. Chen and C. C. Zhang, "Vehicle classification from traffic surveillance videos at a finer granularity," in *Advances in Multimedia Modeling*. Berlin, Germany: Springer-Verlag, 2007, pp. 772–781.

[24] Citilog. [Online]. Available: http://www.citilog.com

[25] CLEAR, *Classification of Events, Activities and Relationships (CLEAR) Evaluation and Workshop*, 2007. [Online]. Available: http://isl.ira.uka.de/clear07/

[26] N. Cornelis, B. Leibe, K. Cornelis, and L. Van Gool, "Three-dimensional urban scene modeling integrating recognition and reconstruction," *Int. J. Comput. Vis.*, vol. 78, no. 2/3, pp. 121–141, Jul. 2008.

[27] M. Cracknell, "Image detection in the real world—A progress update," in *Proc. ITS WC*, New York, 2008.

[28] D. Crandall, P. Felzenszwalb, and D. Huttenlocher, "Spatial priors for part-based recognition using statistical models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2005, vol. 1, pp. 10–17.

[29] I. Creusen, R. Wijnhoven, and P. H. N. de With, "Applying feature selection techniques for visual dictionary creation in object classification," in *Proc. Int. Conf. IPCV Pattern Recog.*, Jul. 2009, pp. 722–727.

[30] CRS, *Computer Recognition Systems*. [Online]. Available: http://www.crs-traffic.co.uk/

[31] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti, "Improving shadow suppression in moving object detection with HSV color information," in *Proc. IEEE Intell. Transp. Syst.*, 2001, pp. 334–339.

[32] H. Dahlkamp, A. Ottlik, and H. Nagel, "Comparison of edge-driven algorithms for model-based motion estimation," in *Proc. 1st Int. Workshop SCVMA*, vol. 3667, Lect. Notes Comput. Sci., 2006, pp. 38–50.

[33] H. Dahlkamp, A. E. C. Pece, A. Ottlik, and H. Nagel, "Differential analysis of two model-based vehicle tracking approaches," in *Proc. Pattern Recog.*, vol. 3175, Lect. Notes Comput. Sci., 2004, pp. 71–78.

[34] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. CVPR*, Jun. 2005, vol. 1, pp. 886–893.

[35] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. ECCV*, 2006, pp. 428–441.

[36] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," in *The Oxford Handbook of Nonlinear Filtering*. New York: Oxford Univ. Press, 2009.

[37] J. Douret and R. Benosman, "A multicamera 3-D volumetric method for outdoor scenes: A road traffic monitoring application," in *Proc. Int. Conf. Pattern Recog.*, vol. 3. Los Alamitos, CA: IEEE Comput. Soc., 2004, pp. 334–337.

[38] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2179–2195, Dec. 2009.

[39] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, 2010.

[40] G. D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.

[41] Y. Freund, "Boosting a weak learning algorithm by majority," *Inf. Comput.*, vol. 121, no. 2, pp. 256–285, Sep. 1995.

[42] 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.

[43] T. Gandhi and M. M. Trivedi, "Video-based surround vehicle detection, classification and logging from moving platforms: Issues and approaches," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2007, pp. 1067–1071.

[44] T. Gao, Z. G. Liu, W. C. Gao, and J. Zhang, "Moving vehicle tracking based on SIFT active particle choosing," in *Proc. Adv. Neuro-Inf. Process.*, vol. 5507, Lect. Notes Comput. Sci., 2009, pp. 695–702.

[45] T. Gao, Z. G. Liu, W. C. Gao, and J. Zhang, "A robust technique for background subtraction in traffic video," in *Advances in Neuro-Information Processing*. Berlin, Germany: Springer-Verlag, 2009, pp. 736–744.

[46] J. Ge, Y. Luo, and G. Tei, "Real-time pedestrian detection and tracking at nighttime for driver-assistance systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 2, pp. 283–298, Jun. 2009.

[47] N. Ghosh and B. Bhanu, "Incremental unsupervised three-dimensional vehicle model learning from video," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 423–440, Jun. 2010.

[48] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Proc. Inst. Elect. Eng., F. Radar Signal Process.*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[49] P. Guha, A. Mukerjee, and K. S. Venkatesh, "Appearance-based multiple-agent tracking under complex occlusions," in *Proc. PRICAI—Trends in Artificial Intelligence*, vol. 4099, Lect. Notes Comput. Sci., 2006, pp. 593–602.

[50] Y. Guo, C. Rao, S. Samarasekera, J. Kim, R. Kumar, and H. Sawhney, "Matching vehicles under large pose transformations using approximate 3-D models and piecewise MRF model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[51] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 37–47, Mar. 2002.

[52] D. L. Hall and S. A. H. McMullen, *Mathematical Techniques in Multisensor Data Fusion*. Norwood, MA: Artech House, 2004.

[53] N. Hautiere, J.-P. Tarel, and D. Aubert, "Mitigation of visibility loss for advanced camera-based driver assistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 474–484, Jun. 2010.

[54] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2010.168

[55] *Imagery Library for Intelligent Detection Systems, i-LIDS*, Home Office Scientific Development Branch. [Online]. Available: http://www.ilids.co.uk/

[56] J. W. Hsieh, S. H. Yu, Y. S. Chen, and W. F. Hu, "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 175–187, Jun. 2006.

[57] W. Hu, X. Xiao, D. Xie, T. Tan, and S. Maybank, "Traffic accident prediction using 3-D model-based vehicle tracking," *IEEE Trans. Veh. Technol.*, vol. 53, no. 3, pp. 677–694, May 2004.

[58] C. L. Huang and W. C. Liao, "A vision-based vehicle identification system," in *Proc. 17th Int. Conf. Pattern Recog.*, 2004, vol. 4, pp. 364–367.

[59] Ipsotek. [Online]. Available: http://www.ipsotek.com/

[60] 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.

[61] B. Johansson, J. Wiklund, P. Forssén, and G. Granlund, "Combining shadow detection and simulation for estimation of vehicle size and position," *Pattern Recognit. Lett.*, vol. 30, no. 8, pp. 751–759, Jun. 2009.

[62] M. J. Jones and D. Snow, "Pedestrian detection using boosted features over many frames," in *Proc. ICPR*, Dec. 2008, pp. 1–4.

[63] Y. Jung, K. Lee, and Y. Ho, "Content-based event retrieval using semantic scene interpretation for automated traffic surveillance," *IEEE Trans. Intell. Transp. Syst.*, vol. 2, no. 3, pp. 151–163, Sep. 2001.

[64] P. KaewTraKuPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," in *Proc. 2nd Eur. Workshop Adv. Video-Based Surv. Syst.*, London, U.K., Sep. 2001.

[65] N. Kaempchen and K. Dietmayer, "Fusion of laserscanner and video for advanced driver assistance systems," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2004, pp. 1–8.

[66] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME-J. Basic Eng.*, ser. D, vol. 82, pp. 35–45, 1960.

[67] S. Kamijo, M. Harada, and M. Sakauchi, "Incident detection based on semantic hierarchy composed of the spatiotemporal MRF model and statistical reasoning," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2004, vol. 1, pp. 415–421.

[68] S. Kamijo, K. Ikeuchi, and M. Sakauchi, "Vehicle tracking in low-angle and front-view images based on spatiotemporal Markov random field model," in *Proc. 8th World Congr. Intell. Transp. Syst.*, 2001.
[69] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Traffic monitoring and accident detection at intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 2, pp. 108–118, Jun. 2000.
[70] S. Kamijo, K. Ikeuchi, and M. Sakauchi, "Event recognitions from traffic images based on spatiotemporal Markov random field model," in *Proc. 8th World Congr. Intell. Transp. Syst.*, Sep. 2001.
[71] S. Kamijo and M. Sakauchi, "Classification of traffic events based on the spatiotemporal MRF model and the Bayesian network," in *Proc. 9th World Congr. Intell. Transp. Syst.*, 2002.
[72] N. K. Kanhere, "Vision-based detection, tracking and classification of vehicles using stable features with automatic camera calibration," Ph.D. dissertation, Clemson Univ., Clemson, CA, 2008.
[73] N. K. Kanhere and S. T. Birchfield, "Real-time incremental segmentation and tracking of vehicles at low camera angles using stable features," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 148–160, Mar. 2008.
[74] N. K. Kanhere, S. J. Pundlik, and S. T. Birchfield, "Vehicle segmentation and tracking from a low-angle off-axis camera," in *Proc. IEEE Comput. Soc. Conf. CVPR*, 2005, vol. 2, pp. 1152–1157.
[75] Kapsch TrafficCom. [Online]. Available: http://www.kapsch.net/en/ktc/
[76] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image Vis. Comput.*, vol. 21, no. 4, pp. 359–381, 2003.
[77] A. Khammari, F. Nashashibi, Y. Abramson, and C. Laurgeau, "Vehicle detection combining gradient analysis and AdaBoost classification," in *Proc. IEEE Intell. Transp. Syst.*, 2005, pp. 66–71.
[78] Z. Kim and J. Malik, "Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, 2003, vol. 1, pp. 524–531.
[79] P. Kumar, A. Mittal, and P. Kumar, "Study of robust and intelligent surveillance in visible and multimodal framework," *Informatica*, vol. 32, pp. 63–77, 2008.
[80] P. Kumar, S. Ranganath, and W. M. Huang, "Bayesian-network-based computer vision algorithm for traffic monitoring using video," in *Proc. IEEE Intell. Transp. Syst.*, 2003, vol. 1, pp. 897–902.
[81] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool, "Dynamic 3-D scene analysis from a moving vehicle," in *Proc. IEEE Conf. CVPR*, Jun. 2007, pp. 1–8.
[82] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *Int. J. Comput. Vis.—Special Issue on Learning for Recognition and Recognition for Learning*, vol. 77, no. 1-3, pp. 259–289, May 2008.
[83] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool, "Coupled object detection and tracking from static cameras and moving vehicles," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1683–1698, Oct. 2008.
[84] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian detection in crowded scenes," in *Proc. IEEE Comput. Soc. Conf. CVPR*, 2005, vol. 1, pp. 878–885.
[85] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *Proc. ECCV Workshop Stat. Learn. Comput. Vis.*, May 2004, pp. 17–32.
[86] C. Y. Liu, "Scale-adaptive spatial appearance feature density approximation for object tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 1, pp. 284–290, Feb. 2011.
[87] J. Lou, T. Tan, W. Hu, H. Yang, and S. J. Maybank, "Three-dimensional model-based vehicle tracking," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1561–1569, Oct. 2005.
[88] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. ICCV*, Los Alamitos, CA, 1999, vol. 2, pp. 1150–1157.
[89] X. Ma and W. E. L. Grimson, "Edge-based rich representation for vehicle classification," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, 2005, vol. 2, pp. 1185–1192.
[90] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[91] N. Martel-Brisson and A. Zaccarin, "Learning and removing cast shadows through a multidistribution approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 7, pp. 1133–1146, Jul. 2007.
[92] O. Masoud and N. P. Papanikolopoulos, "Using geometric primitives to calibrate traffic scenes," *Transp. Res. Part C, Emerging Technol.*, vol. 15, no. 6, pp. 361–379, Dec. 2007.
[93] T. Mauthner, M. Donoser, and H. Bischof, "Robust tracking of spatial related components," in *Proc. 19th ICPR*, Dec. 2008, pp. 1–4.
[94] S. Messelodi, C. M. Modena, N. Segata, and M. Zanin, "A Kalman-filter-based background updating algorithm robust to sharp illumination changes," in *Proc. 13th Int. Conf. Image Anal. Process.*, vol. 3617, Lect. Notes Comput. Sci., F. Roli and S. Vitulano, Eds., 2005, pp. 163–170.
[95] S. Messelodi, C. M. Modena, and M. Zanin, "A computer vision system for the detection and classification of vehicles at urban road intersections," *Pattern Anal. Appl.*, vol. 8, no. 1/2, pp. 17–31, Sep. 2005.
[96] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.
[97] B. Morris and M. Trivedi, "Robust classification and tracking of vehicles in traffic video streams," in *Proc. IEEE ITSC*, 2006, pp. 1078–1083.
[98] B. Morris and M. Trivedi, "Improved vehicle classification in long traffic video by cooperating tracker and classifier modules," in *Proc. IEEE Int. Conf. AVSS*, 2006, p. 9.
[99] K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–202, Mar. 2001.
[100] H.-H. Nagel, Image Sequence Server. [Online]. Available: http://i21www.ira.uka.de/image_sequences/
[101] A. Nakagawa, T. Nakano, and Y. Okamoto, "Demonstration experiments of driving safety support systems using vehicle-to-infrastructure communications systems," *Toshiba Rev.—Special Reports on New Stage of Intelligent Transport Systems*, vol. 64, no. 4, 2009.
[102] P. V. Nguyen and H. B. Le, A multimodal particle-filter-based motorcycle tracking system," in *PRICAI 2008: Trends in Artificial Intelligence*, Lect. Notes Comput. Sci. Berlin, Germany: Springer-Verlag, 2008, pp. 819–828.
[103] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image Vis. Comput.*, vol. 21, no. 1, pp. 99–110, Jan. 2003.
[104] A. Opelt, A. Pinz, and A. Zisserman, "A boundary fragment model for object detection," in *Proc. Eur. Conf. Comput. Vis.*. Berlin, Germany: Springer-Verlag, 2006, pp. 575–588.
[105] A. Opelt, "Generic Object Recognition," Ph.D. dissertation, Graz Univ. Technol., Styria, Austria, Mar., 2006.
[106] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer, "Generic object recognition with boosting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 416–431, Mar. 2006.
[107] A. Opelt, A. Pinz, and A. Zisserman, "Incremental learning of object detectors using a visual shape alphabet," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Los Alamitos, CA, 2006, vol. 1, pp. 3–10.
[108] OpenCV, Open Source Computer Vision Library OpenCV. [Online]. Available: http://sourceforge.net/projects/opencvlibrary
[109] A. Ottlik and H. H. Nagel, "Initialization of model-based vehicle tracking in video sequences of inner city intersections," *Int. J. Comput. Vis.*, vol. 80, no. 2, pp. 211–225, Nov. 2008.
[110] K. Park, D. Lee, and Y. Park, "Video-based detection of street-parking violation," in *Proc. Int. Conf. Image Process. CVPR*, 2007.
[111] A. Pinz, "Object categorization," in *Foundations and Trends in Computer Graphics and Vision*. Hanover, MA: Now, 2005, pp. 255–353.
[112] P. W. Power and J. A. Schoonees, "Understanding background mixture models for foreground segmentation," in *Proc. Image Vis. Comput. New Zealand*, 2002, pp. 267–271.
[113] A. Prati, I. Miki, R. Cucchiara, and M. M. Trivedi, "Comparative evaluation of moving shadow detection algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 7, pp. 918–923, Jul. 2003.
[114] PRE-DRIVE C2X Project, *Preparation for Driving Implementation and Evaluation of Car-2-x Communication Technology*. [Online]. Available: http://www.pre-drive-c2x.eu
[115] PASCAL, *The PASCAL Visual Object Classes Homepage*. [Online]. Available: http://pascallin.ecs.soton.ac.uk/challenges/VOC/
[116] R. Rad and M. Jamzad, "Real-time classification and tracking of multiple vehicles in highways," *Pattern Recognit. Lett.*, vol. 26, no. 10, pp. 1597–1607, Jul. 2005.
[117] P. Remagnino, S. Maybank, R. Fraile, K. Baker, and R. Morris, "Automatic visual surveillance of vehicles and people," *Advanced Video-Based Surveillance Systems*, pp. 97–107, Hingham, MA, 1998.
[118] K. Robert, "Night-time traffic surveillance: A robust framework for multivehicle detection, classification and tracking," in *Proc. IEEE Conf. Adv. Video Signal Based Surv.*, 2009, pp. 1–6.
[119] K. Robert, "Video-based traffic monitoring at day and night time," in *Proc. IEEE 12th Int. Conf. Intell. Transp. Syst.*, 2009, pp. 1–6.

[120] N. Saunier and T. Sayed, "A feature-based tracking algorithm for vehicles in intersections," in *Proc. 3rd Can. Conf. Comput. Robot Vis.*, Jun. 2006, p. 59.
[121] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, no. 1, pp. 7–42, Apr. 2002.
[122] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortexlike mechanisms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 411–426, Mar. 2007.
[123] K. Shafique and M. Shah, "A noniterative greedy algorithm for multiframe point correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 1, pp. 51–65, Jan. 2005.
[124] X. Song and R. Nevatia, "Detection and tracking of moving vehicles in crowded scenes," in *Proc. IEEE WMVC*, 2007, p. 4.
[125] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, Jun. 1999, vol. 2, pp. 246–252.
[126] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.
[127] P. Sturgess, K. Alahari, L. Ladicky, and P. H. S. Torr, "Combining appearance and structure from motion features for road scene understanding," in *Proc. Brit. Mach. Vis. Conf.*, 2009, pp. 1–11.
[128] X. Su, T. M. Khoshgoftaar, X. Zhu, and A. Folleco, "Rule-based multiple object tracking for traffic surveillance using collaborative background extraction," in *Advances in Visual Computing*. Berlin, Germany: Springer-Verlag, 2007, pp. 469–478.
[129] G. D. Sullivan, K. D. Baker, A. D. Worrall, C. I. Attwood, and P. R. Remagnino, "Model-based vehicle detection and classification using orthographic approximations," in *Proc. 7th Brit. Mach. Vis. Conf.*, Sep. 1996, vol. 2, pp. 695–704.
[130] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 694–711, May 2006.
[131] M. Taj, E. Maggio, and A. Cavallaro, "Objective evaluation of pedestrian and vehicle tracking on the clear surveillance dataset," *Multimodal Technol. Perception Humans*, vol. 4625, Lect. Notes Comput. Sci., pp. 160–173, 2008.
[132] T. N. Tan, G. D. Sullivan, and K. D. Baker, "Model-based localization and recognition of road vehicles," *Int. J. Comput. Vis.*, vol. 27, no. 1, pp. 5–25, Mar. 1998.
[133] T. Tanaka, A. Shimada, D. Arita, and R. Taniguchi, "A fast algorithm for adaptive background model construction using parzen density estimation," in *Proc. IEEE Conf. Adv. Video Signal Based Surv.*, 2007, pp. 528–533.
[134] J.-P. Tarel and N. Hautiere, "Fast visibility restoration from a single-color or gray-level image," in *Proc. IEEE 12th Int Conf. Comput. Vis.* , 2009, pp. 2201–2208.
[135] T. H. Thi, K. Robert, S. Lu, and J. Zhang, "Vehicle classification at nighttime using eigenspaces and support vector machine," in *Proc. CISP*, May 2008, vol. 2, pp. 422–426.
[136] P. H. S. Torr, "Graph cuts and their use in computer vision," in *Proc. Int. Comput. Vis. Summer Sch.*, 2007.
[137] Traficon. [Online]. Available: http://www.traficon.com
[138] S. Ullman, "Object recognition and segmentation by a fragment-based hierarchy," *Trends Cognitive Sci.*, vol. 11, no. 2, pp. 58–64, Feb. 2007.
[139] M. Valera and S. A. Velastin, "Intelligent distributed surveillance systems: A review," *Proc. Inst. Elect. Eng.—Vis. Image Signal Process.*, vol. 152, no. 2, pp. 192–204, Apr. 2005.
[140] L. J. P. van der Maaten, E. O. Postma, and H. J. van den Herik, "Dimensionality reduction: A comparative review," *J. Mach. Learn. Res.*, 2009, Submitted for publication.
[141] W. van der Mark and D. M. Gavrila, "Real-time dense stereo for intelligent vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 38–50, Mar. 2006.
[142] H. Veeraraghavan, O. Masoud, and N. Papanikolopoulos, "Vision-based monitoring of intersections," in *Proc. IEEE 5th Int. Conf. Intell. Transp. Syst.*, 2002, pp. 7–12.
[143] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.
[144] Virage. [Online]. Available: http://www.virage.com/
[145] VISOR, *Video Surveillance Online Repository*. [Online]. Available: http://www.openvisor.org/
[146] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inf. Theory*, vol. 12, no. 2, pp. 260–269, Apr. 1967.
[147] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatiotemporal features for action recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2009.
[148] J. Wang, Y. Ma, C. Li, H. Wang, and J. Liu, "An efficient multiobject tracking method using multiple particle filters," in *Proc. World Congr. WRI*, 2009, vol. 6, pp. 568–572.
[149] J. M. Wang, Y. C. Chung, S. C. Lin, S. L. Chang, S. Cherng, and S. W. Chen, "Vision-based traffic measurement system," in *Proc. 17th ICPR*, Aug. 2004, vol. 4, pp. 360–363.
[150] J. Wang, G. Bebis, and R. Miller, "Robust video-based surveillance by integrating target detection with tracking," in *Proc. CVPRW*, Jun. 2006, p. 137.
[151] G. Welch and G. Bishop, "An introduction to the Kalman filter," Univ. North Carolina, Chapel Hill, NC, Tech. Rep. 95-041, 2004.
[152] R. Wijnhoven, P. H. N. de With, and I. Creusen, "Efficient template generation for object classification in video surveillance," in *Proc. 29th Symp. Inf. Theory Benelux*, May 2008, pp. 255–262.
[153] R. G. J. Wijnhoven and P. H. N. de With, "Experiments with patch-based object classification," in *Proc. IEEE Conf. Adv. Video Signal Based Surv.*, Sep. 2007, pp. 105–110.
[154] R. G. J. Wijnhoven and P. H. N. de With, "Comparing feature matching for object categorization in video surveillance," in *Proc. Adv. Concepts Intell. Vis. Syst.*, vol. 5807, Lect. Notes Comput. Sci., 2009, pp. 410–421.
[155] O. J. Woodford, C. Rother, and V. Kolmogorov, "A global perspective on map inference for low-level vision," in *Proc. IEEE ICCV*, 2009, pp. 2319–2326.
[156] B. Wu and R. Nevatia, "Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet-based part detectors," *Int. J. Comput. Vis.*, vol. 75, no. 2, pp. 247–266, Nov. 2007.
[157] A. Yoneyama, C.-H. Yeh, and C.-C. Jay Kuo, "Robust vehicle and traffic information extraction for highway surveillance," *EURASIP J. Appl. Signal Process.*, pp. 2305–2321, 2005.
[158] D. Zhang, S. Qu, and Z. Liu, "Robust classification of vehicle based on fusion of TSRP and wavelet fractal signature," in *Proc. IEEE ICNSC*, Apr. 2008, pp. 1788–1793.
[159] G. Zhang, R. P. Avery, and Y. Wang, "Video-based vehicle detection and classification system for real-time traffic data collection using uncalibrated video cameras," *Transp. Res. Rec.*, vol. 1993, pp. 138–147, 2007.
[160] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int. J. Comput. Vis.*, vol. 73, no. 2, pp. 213–238, 2007.
[161] W. Zhang, Q. M. J. Wu, X. Yang, and X. Fang, "Multilevel framework to detect and handle vehicle occlusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 161–174, Mar. 2008.
[162] W. Zhang, B. Yu, G. J. Zelinsky, and D. Samaras, "Object class recognition using multiple-layer boosting with heterogeneous features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog*, Jun. 2005, vol. 2, pp. 323–330.
[163] Z. Zhang, M. Li, K. Huang, and T. Tan, "Boosting local feature descriptors for automatic objects classification in traffic scene surveillance," in *Proc. ICPR*, 2008.
[164] J. Zheng, Y. Wang, N. L. Nihan, and M. E. Hallenbeck, "Extracting roadway background image: Mode-based approach," *Transp. Res. Rec.*, vol. 1944, pp. 82–88, 2005.

Norbert Buch (SM'07–M'10) received the M.Sc. degree in electrical engineering from the University of Technology (TUG), Graz, Austria, in 2006 and the Ph.D. degree from Kingston University, Kingston upon Thames, U.K., in 2010. His Ph.D. dissertation focused on computer vision traffic analysis and vehicle classification.

Prior to this research, he has worked on electromagnetic oil prospecting. He is currently developing automotive test equipment with Kristl, Seibt & Co GmbH, Graz, Austria. He has published journal papers and conference proceedings about computers.

Dr. Buch is a member of the Institution of Engineering and Technology and the British Machine Vision Association. He received three academic excellence scholarships at TUG, two Best Research Poster awards, and the Peer Mentor of the Year Award from the Faculty of Computing, Information Systems and Mathematics, Kingston University, and the Best Presentation Award at the 2009 British Computer Society Doctoral Consortium.

Sergio A. Velastin (M'90) received the B.Sc. and M.Sc. (Research) degrees in electronics and the Ph.D. degree from the University of Manchester, Manchester, U.K., in 1978, 1979, and 1982, respectively, for his research on vision systems for pedestrian tracking and road-traffic analysis.

In October 1990, he worked in industrial R&D and project management before joining, in October 2001, King's College London, University of London, London, U.K., and Kingston University, Kingston upon Thames, U.K., where he is currently a Professor of applied computer vision and the Director of the Digital Imaging Research Centre. His team is well known for work on robust computer vision techniques to support the monitoring of public spaces, especially under crowded and congested conditions, and to detect situations that can endanger personal security. He conceived and worked in the European Union (EU) project CROMATICA, worked on the EU project ADVISOR and EU project CARETAKER, and was the Technical Coordinator of the EU Project PRISMATICA. His work in the EU project MIRACLES led to the deployment of video analytics in the metro of Rome, Italy. He is the author or a coauthor of more than 100 international publications. His main research interests include distributed visual monitoring systems, computer vision algorithms for pedestrian and vehicular monitoring, and real-time computer architectures. He is an Associate Editor for the *Institution of Engineering and Technology (IET) Computer Vision Research Journal*. He is working in close collaboration with potential beneficiaries such as public transport operators.

Dr. Velastin is a member of the IET, the British Machine Vision Association, and the Board of Governors of the IEEE Intelligent Transportation Society.

James Orwell received the B.S. degrees in physics and philosophy from Oxford University, Oxford, U.K., and the Ph.D. degree in image processing from King's College London, London, U.K.

He is currently a Reader with the Faculty of Computing, Information Systems, and Mathematics, Kingston University, Kingston upon Thames, U.K., where he teaches programming to undergraduates and works with postgraduates on digital imaging research projects. He is also a Member of the Digital Imaging Research Centre (DIRC), Kingston University. His research interests include detection and tracking algorithms for visual surveillance and sports applications and the representation of extracted visual semantics. He has worked on numerous projects related to image processing, including projects for the Defence Evaluation and Research Agency (at King's College) and research contracts in vehicle tracking and recognition (at Kingston University) and as a Short-Term Research Fellow with BTExact. He was the Principal Investigator for the EU INMOVE Project (2002–2004), which developed a software toolkit for developing intelligent audio-visual applications for mobile phone networks, and the EU CARETAKER (2005–2008), which developed a monitoring system for town centers, railway stations, or other public space using video and audio devices. Under the Grand Challenge Programme, he was funded by the Ministry of Defence to evaluate DIRC visual surveillance technology for the protection of armed forces in hostile environments (2007).

Dr. Orwell has received two Engineering and Physical Sciences Research Council-funded Industrial Cases Awards with BAe Systems and Overview and two Department for Business, Enterprise and Regulatory Reform Knowledge Transfer Partnership awards from Pharos and Infoshare. He is an active Member of the IST 37 Committee and has provided contributions to MPEG standardization activities, in particular the MPEG-A Part 10 (Visual Surveillance Application Format). He has provided numerous media interviews on the topic of visual surveillance, including the Guardian and BBC Radio 4.