Traffic Surveillance: A Review of Vision Based Vehicle Detection, Recognition and Tracking

Ma'moun Al-Smadi
Faculty of Science and Technology, Universiti Sains Islam Malaysia, Negeri Sembilan, Malaysia.
E-mail: Malaysia ma_smadi@bau.edu.my

Khairi Abdulrahim
Faculty of Science and Technology, Universiti Sains Islam Malaysia, Negeri Sembilan, Malaysia.
E-mail: khairiabdulrahim@usim.edu.my

Rosalina Abdul Salam
Faculty of Science and Technology Islamic Science Institute (ISI), Universiti Sains Islam Malaysia, Negeri Sembilan, Malaysia.
E-mail: rosalina@usim.edu.my

Abstract
Video-based analysis of traffic surveillance is an active area of research, which has a wide variety of applications in intelligent transport systems (ITSs). In particular, urban environments are more challenging than highways due to camera placement, background clutter, and vehicle pose or orientation variations. This paper provides a comprehensive review of the state-of-the-art video processing techniques for vehicle detection, recognition and tracking with analytical description. In this survey, we categorize vehicle detection into motion and appearance based techniques, varying from simple frame differencing and adaptive median filtering, to more sophisticated probabilistic modeling and feature extracting. We also discuss vehicle recognition and classification utilizing vehicle attributes like color, license plate, logo and type, provide a detailed description of the advances in the field. Next we categorize tracking into model, region and features based tracing. Finally tracking algorithms including Kalman and particle filter are discussed in terms of correspondence matching, filtering, estimation and dynamical models.

Keywords: Computer Vision, Vehicle Detection, Vehicle Tracking, Traffic Surveillance.

Introduction
In recent years, there have been an extensive use of video cameras for traffic surveillance systems, since it can be considered as a rich source of information about traffic flow [1]. Moreover, the fast progress in computer vision, computing and camera technologies, together with the advancement on automatic video analysis and processing have raise the interest in video-based traffic surveillance applications [2].

The application of computer vision techniques in traffic surveillance become increasingly important for intelligent transportation system (ITS) [3]. These systems make use of visual appearance in vehicle detection, recognition and tracking that is useful for incident detection, behavior analysis and understanding [2], [4]. Also, it provide traffic flow parameters that include vehicle class, count, trajectory etc. Although, a significant research effort have been dedicated to improve video-based traffic surveillance systems, various challenges still facing practical ITS applications [5]. Typical traffic scenes includes straight highways, urban road section, intersections, turns and tunnels, which impose additional challenges that include scale and pose variations, traffic congestion, weather and lighting conditions [1]. On the other hand, the variability in vehicle types, size color and pose limits vehicle recognition and tracking to specific scenes [6]. Traffic congestion and camera placement affect performance, since it raise the probability of occlusion [3]. Several computer vision techniques have been proposed in the literature to address the aforementioned problems. However, a universal method that can be applied to all types of vehicles and environments does not exist in real world.

A recent survey [1] presented the state of the art vehicle surveillance architecture from the prospective of hierarchal and networked surveillance, with detailed discussion on special computer vision issues. A survey on vehicle detection, tracking and on-road behavior analysis can be found in [2]. A review of computer vision techniques for urban traffic analysis [3], which concentrate on infrastructure-side. In [4] the key computer vision and pattern recognition have been reviewed with detailed description of technical challenges and comparison of various solutions. Old review is found in [5].

This paper provide a comprehensive review of various techniques involved in video-based traffic surveillance from computer vision perspective. It include various techniques used in vehicle detection, recognition and tracking. The review also include improvements, modifications, highlight the advantages and disadvantages.

The remainder of this paper is organized as follows. In the next section we provide a review of the state of the art research on vehicle detection. Section III reviews the literature about vehicle recognition and classification. In section IV, vehicle tracking is analyzed. Detailed discussion is presented in section V. Finally, section VI summarizes and conclude this paper.

Vehicle Detection
Vehicle detection form the first step in video based analysis for different ITS applications [2]. Accuracy and robustness of vehicle detection have a great importance in vehicle recognition, tracking, and higher level processing [3]. The research effort in this field was divided into: motion based and appearance based techniques [1]. Motion segmentation techniques use the motion cues to distinguish moving vehicles.
from stationary background. On the other hand, appearance based techniques employ appearance features of the vehicle like color, shape and texture to isolate the vehicle from the surrounding background scene. This section will review vehicle detection literature starting from simple frame differencing to the complex machine learning techniques. Vehicle detection techniques with a list of selected publications in each category is shown in Table 1.

Table 1: Representative work in vision based vehicle detection.

| Motion Segmentation | | |
|---------------------|------------------|
| Frame Differencing: | [7], [10], [11]. |
| Background Subtraction | | |
| • Parametric | | |
| Single Gaussian: | [21], [22], [23], [24]. |
| Sigma-Delta: | [8], [17], [18]. |
| GMM: | [17], [19], [26], [27], [28]. |
| • Nonparametric | | |
| KDE: [16], [41], [42], [44]. |
| Codebook: [45], [46], [47], [48], [49]. |
| • Predictive | | |
| Kalman Filter: | [50], [53], [54], [56]. |
| Eigenbackground: | [51], [57], [58]. |
| Optical Flow | [9], [59,60], [61]. |
| Appearance Based | | |
| Feature Based | | |
| • SIFT: | [62], [64], [68], [69], [70], [71], [72], [73], [74], [75]. |
| • SURF: | [65], [76], [77], [78], [79]. |
| • HOG: | [66], [81], [82], [83], [84], [85], [86]. |
| • Haar-like: | [67], [87], [88], [89], [90], [91], [92], [93]. |
| Part Based: | [6], [83], [94], [95], [96], [97], [149]. |
| 3-D Model: | [99], [100], [101], [102], [103]. |

A. Motion Segmentation

Motion detection and segmentation use motion cues to distinguish moving vehicles from stationary background, it can be classified into: temporal frame differencing [7] that depends on the last two or three consecutive frames, background subtraction [8], which require frame history to build background model and finally optical flow [9] is based on instantaneous pixel speed on image surface.

i. Frame differencing

Temporal frame differencing is the simplest and fastest method, in which pixel-wise difference is computed between two consecutive frames. The moving foreground regions are determined using a threshold value [7]. Street-parking vehicles were detected using frame differencing in [10], with noise suppression. Motorcycles were detected in [11]. However, using more information is preferable, the use of three consecutive frames improves detection as in [7]. In which dual inter-frame subtraction are calculated and binarized followed by a bitwise AND to extract the moving target region.

The fusion of frame differencing with other background subtraction techniques was discussed in [12], combined with Gaussian mixture model in [13], [14] and used with corner features extraction in [15]. Temporal difference is highly adaptive with a fast performance. However, it cannot cope with noise, rapid illumination variations, or periodic movements in background. Also its performance degrade on slow and fast motion and it cannot extract all the relevant motion pixels [3].

ii. Background subtraction

These techniques are based on accumulating information about the background scene to produce a background model [8]. After that frames are compared with the background model to identify moving regions, provided that the camera is fixed. It can be categorized into parametric, non-parametric and predictive techniques [16].

a- Parametric background modelling

Parametric background modelling uses a unimodal probability density function to model each pixel, and update the distribution’s parameters. The running Gaussian average [12] is an example of such technique that use Gaussian density function recursively to represent each pixel. Another common techniques with better performance is based on temporal median filter [17] or the approximate median [18], Sigma delta estimation [19] and Gaussian mixture model [20]. However, these approaches remain challenging for slow or temporary stopped vehicle, sudden illumination variations and complex backgrounds.

- Frame averaging: In the conventional averaging technique, a set of N frames are summed up and divided by the number of frames [21]. The resulting model will be subtracted from the consecutive frames. Due to the computational efficiency this technique was used in [22], [21]. However, it has tail effects and its accuracy depends on N that increase memory requirements.

- Single Gaussian: Temporal single Gaussian is used to model background recursively, which improve robustness and reduce memory requirement. To achieve more adaptive background model pixels variance was additionally calculated [23]. The model is computed recursively in the form of cumulative running average and standard deviation [24]. Based on its position in the Gaussian distribution, each pixel is classified either a background or a foreground pixel. Thus single Gaussian model can be considered as the statistical equivalent of dynamic threshold [24]. This model has limited computational cost; yet it still produce tail effects.

- Median filter: The non-recursive median filtering is a common technique, in which the background is estimated by finding the median value for each pixel from a set of frames stored in a buffer. This technique is based on the assumption that the background pixels will not vary dramatically over a time period. For colored frames filtering was accomplished using median [17]. A recursive approximation of the temporal median was
proposed in [18]. It estimate the median through a simple recursive filter that increases or decreases by one if the input pixel is greater or less than the estimate respectively, it is not changed if it is equals. In addition to the high computational complexity of the non-recursive median filtering, its memory requirement is high. In contrast the major strengths of the approximate median filter is its computational efficiency, robustness to noise, and simplicity, moreover it can handle slow movements [8]. A notable limitation is that it does not model the variance of pixels.

- **Sigma-delta:** similar to the recursive median filter, Manzanera and Richefeu used a simple recursive nonlinear operator based on sigma delta filter to estimate two orders of temporal statistics (mean and variance) for every pixel [19]. Thus, it can successfully adjust the intensity of the background. Stability was improved in [25] using selective update with relevance feedback through updating only background pixels. Spatiotemporal processing proposed in [19], improves the detection by removing non-significant pixels. The additional processing improves detection, yet adaption to complex scenes still an eventual problem. To overcome this problem, multiple-frequency sigma–delta was introduced in [19]. Weighted sum of multiple models with different updating periods was computed. Another multi-model was introduced in [26], using a mixture of three distributions. Confidence measurement was proposed in [25], and enhanced in [27]. They tied each pixel with a numerical confidence level that is inversely proportional to the updating period. In [28], a b-level sigma-delta filtering was proposed, which includes conditional temporal and spatial updates. Selective and partial updates using global variance in [29], make a good balance between sensitivity and reliability at the expense of high computation.

The main drawback of this technique is that it cannot handle complex environments with multiple objects of variable motion [27]. Moreover, it quickly degrades under slow or congested traffic conditions [26]. Various improvement on this technique came at the expense of memory requirement and computational cost.

- **Gaussian mixture model:** Gaussian mixture model (GMM) was introduced by Chris stauffer and W.E.L. Grimson in 1999 [20]. It models each pixel as a mixture of two or more Gaussians temporally with online updated. These distributions are estimated as either a stable background process or short-term foreground process by evaluating its stability. If the pixel distribution is stable above threshold, then it is classified as background pixel.

The speed and adaption rate of the GMM was improved in [30], [31] through extending the standard update equations. All of them use a fixed number of distributions. An improved GMM model using recursive computation was presented in [32], which update GMM parameters continuously. The number of distributions was chosen adaptively on-line from a Bayesian perspective. Self-adaptive GMM was proposed in [33] for real-time background subtraction. In this technique a description mixture was learnt to describe the first video frame and used to initialize the new frames. To suppress illumination variation, GMM was extended in [34] to the spatial relations by modeling the joint color of each pixel pair. Color and gradient information were employed to reduce the foreground false detection rate in [35]. Shadows were managed in [36]. In [37], GMM and motion energy image were combined to introduce temporal information in foreground detection. It was combined with inter-frame difference in [14]. GMM was used extensively for traffic analysis [32], [38], [39] with various adaption to improve the original technique.

GMM can handle multi-model background distribution, since it maintains a density function for each [20]. Thus, it is adaptive to light variations and repetitive clutter with higher computational complexity [40]. Sudden variation in global illumination affects the background model dramatically [39]. GMM parameters require carful tuning. Moreover, learning rate affects sensitivity.

b- **Non-parametric background modelling**

These techniques use pixel history to build a probabilistic representation of the observation using recent samples of pixel’s values [16], without the need to consider pixel’s values as a particular distribution. Kernel density estimation (KDE) and codebook model are example of such techniques [41].

- **Kernel Density Estimation (KDE):** The nonparametric KDE technique is used to characterize a multimodal probability density function as proposed in [42]. In this perspective, the probability of each background pixel is estimated from many recent samples using a Parzen-window. Pixels that are unlikely to come from this distribution based on a predefined threshold are labeled as foreground.

The KDE background modeled in [41] utilized two components for the optical flow and three components for the intensity in normalized color space. In [13], they used KDE technique over a joint domain-range representation of image pixels, multimodal spatial uncertainties were directly modelled. The bandwidth was chosen adaptively in [16], by utilizing color and gradients as features for change detection. A KDE technique was used in [43] to represent the spatial-temporal background and a single Gaussian function to represent the spatial foreground.

KDE adapt quickly to changes in the background process and detect targets with high sensitivity. However, due to memory constraints it cannot be used for long-time background sampling [16]. This technique overcomes the problem of fast variations in background [44].

- **Codebook model:** Another nonparametric approach is based on codebook model [45], in which a set of dynamically handled codewords are used to replace parameters represented by probabilistic function to model each background pixel. After that quantization/clustering technique is required. Each codebook may contain a variable number of codewords that models a cluster of samples to construct a part of background [45]. The new pixels are classified as background if its value is within the range of any codeword otherwise it is classified as foreground. Additional improvement to the algorithm were presented; layered modeling/detection for model maintenance and adaptive codebook updating for global
illumination variations. Pseudo layers along with a codebook were used in [46], to represent different objects in the background.

In [47], the proposed codebook employ diverse cues: pixel texture, pixel color and region appearance. Conventional codebook is used to cluster the texture information of the scene, and utilized to detect initial foreground regions. Codebook-GMM Model was proposed in [48], the background was constructed and maintained using codebook technique, and the foreground were detected using the GMM distribution whose parameters were calculated from the codebook clusters. A modified codebook technique with better performance was introduced and compared in [49].

c- Predictive background modelling

In these techniques predictive procedures are employed in modeling the state dynamic of each pixel like Kalman filtering [50] and eigenbasis reconstruction or eigenbackground [51]. Additional predictive techniques used Wiener filter or autoregressive models, Wavelet transform that are based on spatial information, Markov random field (MRF) and dynamic conditional random field (DCRF) consider both temporal and spatial information [27], [23], [21].

- kalman filter background modeling: The background model can be estimated using Kalman filter [52], in which one filter is used for temporally modeling each pixel color. The foreground can be interpreted as noise for the filter state. It was firstly introduced by Karmann and von Brandt in [50]. In which, the internal state of the system is described by the background intensity and its temporal derivative, which are updated recursively.

Several variations on kalman filter background modeling have been proposed, differing mainly in the state spaces used. The simplest version uses only the luminance intensity [53], others used the intensity and its spatial derivatives [54]. However, the illumination variations are non-Gaussian and violate Kalman filters assumptions. The technique proposed in [55] is able to deal with both gradual and sudden illumination variations. Individual states of Kalman filter are adjusted using an estimate of the illumination distribution over the whole frame. In [56] pixel intensity was tracked using one-dimensional Kalman filter. Prediction/correction equations are used to update the standard deviation and prior intensity values.

- Eigenbackground: Based on an eigenvalue decomposition, Oliver et al. proposed a background modelling technique that perform offline learning and online classification [51]. In the learning phase the average of a set of sample frames is computed and subtracted from all frames. Then the covariance matrix is computed and the eigenvector matrix is composed from the best eigenvectors. In the classification phase, each new frame is projected into the eigenspace. Then it is back projected onto the image space to give the background model.

In [57] a block-level eigenbackground was proposed, in which the original frame was divided into blocks to perform independent block training and subtraction. The algorithm was extended in [58] to select the best eigenbackground for each pixel. The modification include selective training, model initialization, and pixel-level reconstruction.

iii. Optical flow

Optical-flow-based motion segmentation use flow vectors characteristics of moving objects over time to detect moving regions in video. It is the instantaneous pixel speed on the image surface that corresponds to object motion in 3-D space. The generated field represent the velocity and direction of each pixel or sub-pixel as a motion vector [9]. There are many methods for computing optical flow among which few are partial differential equation based methods, gradient consistency based methods and least squared methods. Merged vehicle blobs was separated in [59] using dense optical flow. Optical flow and 3-D wireframes have been used to segment vehicles in [60]. In [9], it was used to deal with vehicle scale variations and color similarity, and in [61] for vehicle detection and speed estimation.

This technique is less susceptible to occlusion. It provide an accurate subpixel motion vectors that is best suited in presence of camera motion, light variation and complex or noisy background. However, iterative calculation increase its computational complexity.

B. Appearance Based Techniques

The use of visual information like color, texture and shape in detecting vehicles require prior information [62]. Feature extraction is used to compare the extracted 2-D image features with the true 3-D features in the real world. In contrast to motion segmentation techniques that detect motion only, appearance based techniques detect stationary objects in images or videos [1].

i. Feature Based Techniques

Representative features use coded descriptions to characterize the visual appearance of the vehicles. A variety of features have been used in vehicle detection such as local symmetry edge operators [63]. It is sensitive to size and illumination variations, thus a more spatial invariance edge based histogram was used in [64]. In recent years, these simple features evolves into more general and robust features that allow direct detection and classification of vehicles. Scale Invariant Feature Transformation (SIFT) [62], speeded up Robust Features (SURF) [65], Histogram of Oriented Gradient (HOG) [66] and Haar-like features [67] are extensively used in vehicle detection literature.

- SIFT: Scale Invariant Feature Transformation (SIFT) was first introduced in 1999 [62]. Features are detected through a staged filtering approach, which identifies local edge orientation around salient keypoints in scale space. The generated features are invariant to image scaling, translation, and rotation, also it is partially invariant to illumination changes and affine or 3D projection. Thus, it can describe the appearance of salient points uniquely. In addition to the feature vector, the characteristics scale and orientation of every keypoint is calculated. It can be used to find correspondence of object points in different frames [62].

A modified SIFT descriptor was used in [64], by introducing a rich representation for vehicle classes. In [68], SIFT interest points were re-identified as the initial particles to improve
tracking performance. SIFT-based template matching technique was used in [69], to locate special marks in the license plate. SIFT and Implicit Shape Model (ISM) were combined in [70] to detect a set of keypoints and generate feature descriptors. In [71] an SIFT based mean shift algorithm was proposed. To compress the length of SIFT, Principal Component Analysis (PCA)–SIFT was introduced in [72], through combining local features with global edge features using an adaptive boost classifier. However, it was slow and less distinctive [65]. Based on an enhanced SIFT feature-matching technique vehicle logo recognition algorithm was proposed in [73]. The SIFT matching algorithm was combined with SVM in [74], for multi-vehicle recognition and tracking. It perform tracking well in complex situations. Still, it consumes a lot of time, which restricts practical applications. The proposed method in [75], combined the advantages of SIFT and CAMSHIFT to track vehicle.

Due to its distinctive representation, SIFT has wide applications. However, the high dimensionality and the use of Gaussian derivatives to extract feature points are time-consuming and do not satisfy the real-time requirement [76]. Its low adaption to illumination variation is another drawback.
- **SURF**: Speeded Up Robust Features (SURF) is a scale and rotation invariant interest point detector and descriptor that was introduced in [65]. Compared to SIFT its computational complexity was reduced by replacing Gaussian filter with a box of filters, which slightly affects the performance. SURF algorithm uses a Hessian matrix approximation on an integral image to locate the points of interest. The second-order partial derivatives of an image describe its local curvatures [77].

In [76], symmetrical SURF descriptor was proposed for vehicle detection with make and model recognition. Recently, symmetrical SURF was used in [77] for vehicle color recognition and in [78] to detect the central line of the vehicles. The proposed technique can process one vehicle per frame with 21 fps. A GPU based multiple camera system in [79] used Gabor filter as a directional filter with SURF matching for unique representation of vehicles. An on road vehicle detection in [80], uses cascade classifier and Gentle AdaBoost classifier with Haar-SURF mixed features.

Because of its repeatability, distinctiveness, robustness and real-time capability, it has become one of the most commonly used features in computer vision [76]. Nevertheless, it is not stable under rotation and illumination variations.
- **HOG**: The grid of Histogram of Oriented Gradient (HOG) [66] compute the image gradient directional histogram, which is an integrated presentation of gradient and edge information. It was originally proposed to detect pedestrian, then in [81], it was introduced for vehicle detection by using 3-D model surface instead of 2-D grid of cell to generate 3-D histogram of oriented gradient (3-DHOG).

HOG symmetry feature vectors was proposed in [82] and used together with the original HOG in hypothesis verification. A combination of a latent support vector machine (LSVM) and HOG was used in [83] to combines both local and global features of the vehicle as a deformable object model. HOG was combined with Disparity Maps in [84] to detect Airborne Vehicle in Dense Urban Areas. In [85] a relative discriminative extension to HOG (RDHOG) was proposed to enhance the descriptive ability.

- **Haar-like Features**: Haar-like features [67] are formed of sum and differences of rectangles over an image patch to describe the grey-level distribution of adjacent regions. The filters used to extract the features consists of two three or four that can be at any position and scale. The output of the filter is calculated by adding the pixel values for the grey region and white region separately, then the difference between the two sums is normalized. It represents horizontal or vertical intensity difference, intensity difference between the middle region and aside areas, diagonal intensity differences and the difference between the center and surrounding areas.

Haar features was used in [87] to detect vehicles and in [88],[89] it was employed to train a cascaded AdaBoost classifier. Haar-like and motion features were used to detect highway vehicles in [90], and on urban in [91]. In addition to Haar-like features, local binary pattern (LBP) feature were used in [92] to train the boosting classifiers for the detection of vehicle license plate. Vehicle detection with multiple layer perceptron’s (MLP) ensemble was implemented using Haar-like features in [93].

In addition to the high computation efficiency, Haar-like features are sensitive to vertical, horizontal and symmetric structure, which make them well suited for real time application. Moreover, it require a relatively limited training data [80].

### ii. Part based detection models

In this technique the vehicle is divided into a number of parts modeled by the spatial relation between them [63]. Many recent studies employ this technique in vehicle recognition [6]. They consider the vehicle to be separated into front, side and rear parts which contains window, roof, wheels, and other parts [6]. The distinct parts are detected based on their appearance, edge and shape feature [94]. After that spatial relationship, motion cue and multiple models are used to identify vehicles.

In [95] part labelling was defined to cover the object densely. To ensure consistent layout of parts while allowing deformation they used Layout Consistent Random Field model. The method was expanded to 3-D models in [96] to learn physically localized part appearances. Also they combine object-level descriptions with pixel-level appearance, boundary, and occlusion reasoning. Deformable part based modelling was employed in [83] through the combination of a latent support vector machine (LSVM) and histograms of oriented gradients (HOGs). The algorithm combines vehicle global and local features as a deformable model composed of root filter and five parts filters to detect front, back, side, and
front, back truncated.
Deformable part based model was used in [83], it consists of a global “root” filter, six part filters and a spatial model to detect and track vehicles on road using part-based transfer learning (PBTL). Vehicle detection by independent parts (VDIP) was introduced in [97] for urban driver assistance. Front, side, and rear parts were trained independently using active learning. Part matching classification using a semisupervised approach form vehicles sideview from independently detected parts.A rear view vehicle detection was considered in [6] based on multiple salient parts that includes license plate and rear lamps. For part localization distinctive color, texture and region features were used. Then Markov random field model was used to construct probabilistic graph of the detected parts. Vehicle detection was accomplished by inferring the marginal posterior of each part using loopy belief propagation.

iii. Three Dimensional modeling
In this technique, vehicle detection can be achieved through the use of computer generated 3-D models with appearance matching. The use of 3D models for vehicle detection and classification was proposed in [98].
The classification of vehicles into 8 classes in [99] relies on a set of three dimensional models, each one providing a coarse description of various vehicle shapes. In [100] a fixed size 3D rectangular box was adopted to reduce the computation at the expense of matching accuracy. Vehicle detection using 3D models was proposed in [101] and used in [102] for urban vehicle tracking. Motion silhouettes were extracted and compared with a projected model silhouette to identify the ground plane position of vehicle. Based on edge-element and optical-flow association a 3D model was used in [60] for automatic initialization. A deformable 3D model was proposed in [103] that deforms to match various passenger vehicles. The main drawback of 3-D modelling is how to achieve an accurate 3-D model, which make it limited to few number of vehicle class. Moreover the representation, extraction and matching complicate as the number of models increase.

Vehicle Recognition and Classification
Detected foreground regions may correspond to different objects in natural scenes. For instance, the scene may include vehicles of different types and classes, humans, and other moving objects such as animals, motorcycles, etc. thus it is necessary to isolate, distinguish and recognize the object of interest (i.e. vehicle).
Vehicle recognition aims at identifying correspondence between real-world and its projection in two dimensional image space. Which may involves extracting vehicle static attributes that includes color, license plate, logo and type as shown in Table 2. Feature extraction, representation and matching are the main challenges. Vehicle representation is based on visual cues such as edges, boundaries, junctions, brightness or color [64].

<table>
<thead>
<tr>
<th>Color</th>
<th>RGB:</th>
<th>Hasegawa, et al., [120]</th>
</tr>
</thead>
<tbody>
<tr>
<td>License Plate</td>
<td>Texture:</td>
<td>Anagnostopoulos, et al., [110]</td>
</tr>
<tr>
<td></td>
<td>Color:</td>
<td>Wan, [109]</td>
</tr>
<tr>
<td></td>
<td>Features:</td>
<td>Chen, et al., [111]</td>
</tr>
<tr>
<td>Logo</td>
<td>Edge:</td>
<td>Wang, et al., [117]</td>
</tr>
<tr>
<td></td>
<td>Features:</td>
<td>Psyillos, et al., [73]</td>
</tr>
<tr>
<td>Type</td>
<td>Shape:</td>
<td>Negri, et al., [121]</td>
</tr>
<tr>
<td></td>
<td>Appearance:</td>
<td>Chen, et al., [122]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thakoor, et al., [123].</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Huang, et al., [126]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Feres, et al., [127]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Zhang, [128]</td>
</tr>
</tbody>
</table>

A. Color Recognition
Color of vehicle is an essential attribute that have wide applications in ITS, such as security and crime prevention issues. The variation in illumination and camera view point in outdoor scene affects the color classification dramatically [104]. In [105] k nearest neighbor like classifier was used to classify vehicle color into six groups, each of them contains similar colors like black, dark blue and dark gray. HSV color space was used in [106]. They classify vehicle color into red, blue, black, white and yellow using 2-D histogram of H and S channels together with SVM. Multiple classification methods (K-NN, ANNs and SVM) along with two region of interest were used in [107] to recognize vehicle color in a 16 color space. In [108] bag of word technique was used to select region of interest for color recognition.

B. License Plate Recognition
Automatic recognition of license plate number is generally performed in three major steps: license plate localization, plate character segmentation and recognition. Accurate localization of license plate require the use of edge, color [109], texture [110] or features combination [111]. Character segmentation varies according to the issuing country due to variation in color, size and aspect ratio [112]. While character recognition is affected by the camera zoom factor and require the use of a single classifier such as Artificial Neural Network (ANN) [113], Hidden Markov Model (HMM) [114] or Support Vector Machine (SVM) [115]. Some research use multistage or parallel classifiers [116].

C. Logo Recognition
Vehicle logo provide important information about vehicle make and model, thus it plays a major role in vehicle classification and identification. Logo detection is a critical prerequisite step for logo recognition. Some techniques used edge detection and morphological filtering as in [117], or using coarse to fine vehicle logo localization step [118]. Others detect the frontal vehicle logo using license plate
D. Vehicle Type Classification
Vehicle type classification gain a great deal of research interest recently. It follow two major directions either based on the shape or the appearance of the vehicle. Shape features that are used to classify vehicle types include: size, silhouette dimension, aspect ratio etc. The number of vehicle classes varies according to the used features and the classification technique. The curves associated with the 3D ridges of vehicle surface were used in [120] with 88% accurate rate, they classify vehicles into SUV, car and minibus. Oriented-contour point model was proposed in [121] to represent vehicle type. The edges in the four vehicle orientation from the front view were used together with a voting algorithm and Euclidean edge distance with classification rate of 93.1%. 3D model based classification were used in [122] for car, van, bus and motorcycle classification with accuracy of 96.1 and 94.7 respectively. Structural Signatures feature that captures the relative orientation of vehicle surfaces and the road Surface was used in [123] to classify passenger vehicles into sedans, pickups, and minivans/sport utility vehicles in highway videos with limited accuracy below 90%.

Appearance based classification techniques use appearance features like edge, gradient and corner. It can distinguish between a wide variety of vehicle type with different time complexity and accuracy depending on the selected features and classifier. Edge features was also used with K-means classifier to distinguish between 5 vehicle classes in [124]. A 2-D linear discriminant analysis technique was proposed in [125] to classify 25 vehicle types with 91% accuracy. High recognition accuracy of 94.7% was obtained for 20 vehicle types in [126]. Simple blob measurements was used in [136] to classify eight different vehicle types using VECTOR system with accuracy over 80%. Multiple feature plan that contains RGB colors, gradient magnitude and other features were used in [127] to train a single detector for multiple vehicle types (buses, trucks and cars). They introduce the term shape free appearance space to denote the image space of the vehicle. Another high recognition accuracy of 98.7% for 21 vehicle types was obtained in [128].

Many challenging issues still exist in vehicle recognition and classification. That include: illumination variations, road environment, camera field view, similarity in vehicle appearance and the large number of vehicle types.

Vehicle Tracking
Afar detection and recognition of vehicle, tracking aims to obtain vehicle trajectory through identifying motion dynamic attributes and characteristics to locate its position in every frame [1]. Vehicles tracking can be merged with the detection process or performed separately. In the first case detected vehicles and correspondence are jointly estimated by updating location iteratively using information obtained from previous frames. In the latter case, vehicle detection is performed in every frame, and data association is used to provide correspondence between vehicles in consecutive frames [3]. Current trends in vehicle tracking can be classified into three categories: model-based (multi-view or deformable), region-based (shape or contour), and feature-based tracking as shown in Figure 2 with a list of selected publications in each category presented in Table 3.

Table 3: Representative work in vehicle tracking categories.

<table>
<thead>
<tr>
<th>Model-Based Tracking</th>
<th>Multi-view Model</th>
<th>Deformable Template</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Koller et al., [54]</td>
<td>Tan, &amp; Baker, [131]</td>
</tr>
<tr>
<td></td>
<td>Lou, et al., [129]</td>
<td>Z. Zhang, et al., [132]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Y.-L. Lin, et al., [133]</td>
</tr>
<tr>
<td>Region Based Tracking</td>
<td>Shape Based</td>
<td>Contour-Based</td>
</tr>
<tr>
<td></td>
<td>Mandellos, et al., [134]</td>
<td>Zhang, et al., [38]</td>
</tr>
<tr>
<td>Feature-Based Tracking</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Buch, et al., [81]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bouttefroy, et al., [140]</td>
<td></td>
</tr>
</tbody>
</table>

A. Model-Based Tracking
Model-based tracking use prior knowledge to create a geometric model for the target of interest (i.e. vehicle), which can be 2-D or 3-D appearance model. These models are used to match with moving regions and describe vehicle motion [60]. This accurate and robust technique costs high computation, since exact model is difficult to obtain. Different methods were proposed to model vehicles, which can be categorized into multi-view model and deformable template [60], [129], [130]. Multiview 3-D model is constructed using 2-D geometrical features. In [54], vehicle 3-D model was built using edges and correspondence features. The technique proposed in [129] evaluate the distance between extracted edge points and the projected model. The similarity of the projected 2-D contour was evaluated in [40] to extract the 3-D vehicle pose.

On the other hand the 3-D template model is projected into image through evaluating image intensity or gradient. Deformable vehicle model with 29 parameters was combined with principle component analysis in [98] for vehicle tracking. Gradient vectors and intensity values were used in [131] to estimate and track vehicle pose and orientation. A dynamin 3-D vehicle model was proposed in [132] for deformable model based tracking. Vehicle tracking was achieved using complex deformable 3-D model in [103], and 3-D deformable model fitting with weighted Jacobian in [133]. In [83] deformable object model was combined with particle filter to improve likelihood estimation for on-road multi-vehicle tracking. 3-D model-based vehicle localization was used for constrained multiple-kernel tracking in [130]. Multiview based techniques are sensitive to noise and occlusion, while Deformable template based techniques focus on shape fitting and discard color information. In addition both techniques are time consuming.
B. Region Based Tracking
Tracking based on region detects vehicles silhouette as connected regions within rectangular, oval or any simple geometric shape, which can be characterized by area, coordinates, centroids, edges, contour or intensity histogram etc. Data association between region characteristics within consecutive frame is used to perform tracking. Region based tracking search for the vehicle using shape matching or it evolve an initial contour to its new position using contour tracking. Contour representation use a closed curve that is updated automatically [1].
In [134] shape based tracking with Kalman filtering were used to match simple region. In [135] graph-based region tracking was used for highway vehicles by finding the maximal weight graph. The computational complexity and its failure in crowded situation are the main drawbacks of this technique. In [40] length and height of the convex hull were used to track vehicle.
Vehicle centroid and velocity were used in Kalman filtering framework in [136]. In [137] vehicle appearance was modeled using hue-saturation color histogram with Markov Chain Monte-Carlo Particle Filters tracking paradigm. In [54] the contour that represent the moving vehicles were detected and tracked using snakes. Vehicle position was defined by the center of the snake convex contour with linear motion pattern. In [38] the contour of two vehicles was used to resolve occlusion. Vehicle-contour-tracking method was used in [138] to handle visual clutter and partial occlusions.

C. Feature-based Tracking
The detected vehicle features are used to perform matching in consecutive frames. Thus, vehicle features are tracked in a transformed space instead of pixels space. Earlier techniques used corners and edges to represent vehicles [139]. Motion constrains are used to group vehicle sub-features together. Several techniques propose the combination of corners, edges or interest points with feature descriptors like SIFT [71], SURF [79], HOG [81],[85] and Haar-like [87] for vehicle tracking. Other techniques perform tracking based on color histogram, which is more robust to noise and invariant to vehicle rotation and translation [140].
The concept of HOG was extended to 3D (3-DHOG) in [81], which uses 3D model surfaces rather than 2D grids of cells. This technique allows to resolve the scale variation and use a single model for variable viewpoints of road users. Region-based tracking was combined with Scale Invariant Feature Transform (SIFT) features based tracking in [75]. Feature-based technique can perform well in relatively crowded circumstances. But the main challenge in this technique is to choose appropriate set of features which can effectively represent the moving object (i.e. vehicle).
Feature-based tracking perform well in relatively crowded circumstances. But the main drawback of this technique is to choose appropriate features which can effectively represent the moving vehicle.

D. Tracking Algorithms
All tracking techniques require prediction and data association process that can be performed using tracking algorithms that include Kalman filter and Particle filter [87].

i. Kalman filter tracking
In vehicle tracking Kalman filtering is used to estimate the object position in the new frame [52], assuming that the dynamics of the moving object can be modeled and that the noise effect is stationary with zero mean. The current states of the Kalman filter are estimated recursively using the previously estimated states and current measurements. The state vector contains the variables of interest, which represent the state of the dynamic system. It can be position, velocity, orientation angles, etc. In the case of the moving vehicles, it has two degrees of freedom, the position and the velocity.
In [54] Koller et. al., propose the application of Kalman filter concept in contour vehicle tracking. Extended Kalman filtering was used in [129] to track the 3-D vehicle model, which improves accuracy and stability. Projective Kalman filter was combined with mean-shift algorithm in [140] to perform vehicle tracking. They integrate the non-linear projection of the vehicle trajectory in its observation function to provide accurate estimation of vehicle position. Variable sample rate Kalman filter proposed in [102] track 3D model vehicle on the ground plane. Kalman filter was used in [141] to predict the possible location of the vehicle, then accurate estimation was achieved by predicted point matching using Gabor wavelet features. In [134] Kalman filter was used to track vehicle shape based on its location, speed and length.
Sivaraman &Trivedi used Kalman filtering in [97] to integrate tracking of vehicle parts in the image plane. Position and dimension of the target are used with the constant velocity model. In [14] Kalman filtering was adopted using vehicle coordinates and unit displacement of center of mass together with the dimensions and unit displacement of tracking region. Detection by tracking technique was used in [6], they estimate vehicles trajectories by Kalman filter. The state vector was defined by the license plate center and the vehicle speed.

ii. Particle filter tracking
The particle filtering technique has many applications in visual tracking. It is a sequential Monte Carlo sampling technique that estimates the latent state variables of a dynamical system based on a sequence of observations [142]. Basically, particle filter uses a set of random samples with associated weights and estimation to represent the posterior probability density. When the number of particles is large enough, the group of particles with associated weight can completely describe a posteriori probability distribution to give optimal Bayesian estimation of particle filter.
Vehicle contour tracking in [138] is based on particle filter condensation algorithm. In [140] particle filter was used to track the color histogram of the vehicle. A hybrid mean-shift (MS) and particle-filtering approach was developed in [143], which aims to deal with partial occlusions and the background clutter. Color histogram and edge-based shape features were combined in [144], the particle filter performs well, even with significant color variations, poor lighting, and/or background clutter edges. Another approach that use 3-D scene information in vehicle tracking is based on the Lucas–Kanade tracker algorithm [21].
The work in [145] employed particle filter method in Bayesian estimation for vehicle tracking in urban environments, and they claim that it performs better than EKF.
in multimodal distributions. The center of the rectangle that encloses the vehicle was used in [146] to initialize the particle filter algorithm, with zero weight assigned to particles that fall outside of the rectangle area. Vehicle tracking in [147] fuse several cues in particle filter, which include color, edge, texture and motion constrained, which provide accurate tracking. The tracking technique in [148] is based on spatial and temporal coherence of particles. Particles are grouped according to their spatial positions and motion vectors. Vehicle tracking in [37] uses the similarity between color histogram to identify vehicle particle. Markov Chain Monte-Carlo Particle Filters was used in [137] for real time tracking. In [85], RDHOG was integrated with particle filter framework (RDHOGPF) to improve the tracking robustness and accuracy.

**Discussion**

This section will provide a discussion, analyses and perspectives of challenges and future research directions on video-based traffic surveillance. Most of the work achieved so far deal with highway rather than urban environments. The main technical challenge from the application perspective lies in the camera view and operating condition, which impose many additional limitations [3]. Vehicle surveillance systems undergo various difficulties especially in urban traffic scenarios such as road sections and intersection in which dense traffic, vehicle occlusion, pose and orientation variation and camera placement highly affect their performance. In road sections vehicles usually travels in a uni-direction in which heavy traffic and congestion may affect vehicle detection due to slow or temporary stopped vehicles. Vehicle pose and orientation with respect to the camera often varies while moving within intersections due to lane change and turn left, right and round [130]. This will vary the appearance and scale of vehicle within consecutive frames affecting tracking and classification dramatically. On the other hand different vehicle types varies in size shape and color. All of that will increase the complexity of recognition and tracking process and affect the real time performance. Nighttime is a dramatic challenge for traffic surveillance, in which headlight and taillights are used to represent the vehicle [149]. Despite the significant progress that have been made in vehicle surveillance during the last years, many challenging issues still need further research and development especially in urban environment, in which vehicle pose and orientation varies dramatically at road turns and intersections.

**Conclusion**

In this paper, we have provided an extensive review of the state-of-the-art literature addressing computer vision techniques used in video based traffic surveillance and monitoring systems. These systems perform three major operations that is vehicle detection, tracking and recognition. Vehicle detection was divided into two main categories based on vehicle representation, namely, techniques based on motion cues and techniques that employ appearance features. Both techniques can be used to isolate vehicles from the background scene with different computational complexity and detection accuracy. We provide detailed summaries on vehicle color, license plate and logo recognition together with vehicle shape and appearance type classification. Vehicle tracking was categorized into model, region and feature based tracking with a discussion on motion and parameter estimation schemes employed like Kalman and Particle filtering. We believe that, this paper provides a rich bibliography content regarding vehicles surveillance systems, which can provide valuable insight into this important research area and encourage new research.

**Acknowledgments**

This paper presents a work that is supported by the ministry of higher education (MOHE), under the Research Acculturation Grant Scheme (RAGS/2013/USIM/ICT07/2) and Science Fund Grant Scheme (USIM/SF/FST/30/30113).

**References**


