Vehicle Detection Based on Video for Traffic Surveillance on road

Dr. Swapan Kumar Deb¹, Rajiv Kumar Nathr²

Civil Engineering Department, Indian Institute of Technology, New Delhi, India
¹skdeb@hotmail.com
²rajivknath@gmail.com

Abstract—Today most of the cities of the world have intelligent transport system which is equipped with electronics devices to communicate about the traffic condition with the moving vehicle and also monitor the traffic rules and regulation. Today most of the vehicles are equipped with on-board automotive driver assistance system aiming to alert drivers about driving environments, and possible collision with other vehicles has attracted a lot of attention lately. In these systems, robust and reliable vehicle detection is a critical step. This paper presents a review of recent vision-based on-road vehicle detection systems. Our focus is on systems where the camera is mounted on the vehicle and also being fixed such as in traffic/driveway monitoring systems. First, we discuss the problem of on-road vehicle detection using optical sensors followed by a brief review of intelligent vehicle research worldwide. Then, we discuss active and passive sensors to set the stage for vision based vehicle detection. Methods aiming to quickly hypothesize the location of vehicles in an image as well as to verify the hypothesized locations are reviewed next. Tracking of vehicle is also reviewed to illustrate the benefits of exploiting temporal continuity for vehicle detection.

Keywords—Vehicle-recognition, Traffic Control and Surveillance System, vehicle type.

1. Introduction

Traditional technology for traffic sensing, including inductive loop detectors and video cameras, are positioned at fixed locations in the transportation network. Data related to traffic flow is currently obtained from detectors embedded in pavements or pneumatic tubes stretched across roads. Such methods do not prove to be time-efficient or cost effective. While these detectors do provide useful information and data about traffic flows at particular points, they generally do not provide useful data for traffic flows over space. It is not possible to move detectors; further, they cannot provide useful information such as vehicle trajectories, routing information, and paths through the network. The number of road accident increases day by day, on average, at least one person died at every minute due to some road accident. Auto accidents injure at least 10 million people every year, two or 3 million are seriously. It leads to the hospital bill, damaged property, and other costs will add up to 1-3 percent of the world’s gross domestic product [1-2]. With the aim of reducing injury and accident severity, precrash sensing is becoming an area of active research among automotive manufacturers, suppliers and universities. Several national and international projects have been launched over the past several years to investigate new technologies for improving safety and accident prevention.

Vehicle accident statistics disclose that the main threats a driver is facing are from other vehicles. Consequently, developing on-board automotive driver assistance systems aiming to alert a driver about driving environments and possible collision with other vehicles has attracted a lot of attention. In these systems, robust and reliable vehicle detection is the first step. Vehicle detection—and tracking —has many applications including platooning (i.e., vehicles traveling in high speed and close distance in highways), stop and go (vehicles traveling in low speeds and close distance in cities), and autonomous driving.

This paper presents a review of traffic surveillance which includes all current concepts on vision-based system. Traffic surveillance is done on a road vehicle detection system. Here the camera is mounted either on a dynamic location like a moving vehicle or on a fixed location like a traffic/driveway monitoring systems.

Detection and tracking of moving objects is an important task in the analysis of video data for applications such as video surveillance, traffic
monitoring and analysis, human detection and tracking, and gesture recognition. A common approach to detect moving objects is background subtraction, where each frame in the video is compared to a background or reference frame, and pixels that deviate significantly from the background are considered to be moving or foreground objects [3]. Vehicle detection using optical sensors is very challenging due to huge within class variabilities in vehicle appearance. Vehicles may vary in shape (Fig. 1a and 1c), size, and color. The appearance of a vehicle depends on its pose (Fig. 1b) and is affected by nearby objects. Complex outdoor environments (e.g., illumination conditions (Fig. 1d), unpredictable interaction between traffic participants, cluttered background (Fig. 1e) are difficult to control. On-road vehicle detection also requires faster processing than other applications since the vehicle speed is bounded by the processing rate. Another key issue is robustness to vehicle’s movements and drifts. More general overviews on various aspects of intelligent transportation systems (e.g., infrastructure-based approaches such as sensors detecting the field emitted by permanent magnetic markers or electric wires buried in the road) as well as vision-based intelligent transportation systems (e.g., driver monitoring, pedestrian detection, sign recognition, etc.) can be found in [2], [4-7]. Several special issues have also focused on computer vision applications in intelligent transportation systems.

Figure 1. The variety of vehicle appearances poses a big challenge for vehicle detection

2. Sensors

The most common approach to vehicle detection is using active sensors [8] such as radar-based (i.e., millimeter-wave) [9], laser-based (i.e., LIDAR) [10-11], and acoustic-based [12]. In radar, radio waves are transmitted into the atmosphere, which scatters some of the power back to the radar’s receiver. A Lidar (i.e., “Light Detection and Ranging”) also transmits and receives electromagnetic radiation, but at a higher frequency; it operates in the ultraviolet, visible, and infrared region of the electromagnetic spectrum. The reason that these sensors are called active is because they detect the distance of objects by measuring the travel time of a signal emitted by the sensors and reflected by the objects. Their main advantage is that they can measure certain quantities (e.g., distance) directly without requiring powerful computing resources. Radar-based systems can “see” at least 150 meters ahead in fog or rain, where average drivers can see through only 10 meters or less. Lidar is less expensive to produce and easier to package than radar; however, with the exception of more recent systems, Lidar does not perform as well as radar in rain and snow. Laser-based systems are more accurate than radars; however, their applications are limited by their relatively higher costs. Prototype vehicles employing active sensors have shown promising results. However, when a large number of vehicles move simultaneously in the same direction, interference among sensors of the same type poses a big problem. Moreover, active sensors have, in general, several drawbacks, such as low spatial resolution and slow scanning speed. This is not the case with more recent laser scanners, such as SICK [9], which can gather high spatial resolution data at high scanning speeds.

Optical sensors, such as normal cameras, are usually referred to as passive sensors [8] because they acquire data in a non intrusive way. One advantage of passive sensors over active sensors is cost. With the introduction of inexpensive cameras, we could have both forward and rearward facing cameras on a vehicle, enabling a nearly 360 degree field of view. Optical sensors can be used to track more effectively cars entering a curve or moving from one side of the road to another they can also works in low visible condition [13-14]. Also, visual information can be very important in a number of related applications, such as lane detection, traffic sign recognition, or object identification (e.g., pedestrians and obstacles), without requiring any modifications to road infrastructures. Several systems presented in [5] demonstrate the principal feasibility of vision-based driver assistance systems. Optical cameras are also mounted on moving platform like helicopter with gyrosopic mount to capture the traffic [15]. These cameras are mounted on traffic/driveway monitoring system. The author proposed a multi-modal approach based on photogrammetry combined with positioning...
technologies is used to obtain 3-D coordinates of chosen geographic objects, providing a search area for conventional feature trackers.

3. Vision-based intelligent vehicle research and transportation system

The Vision-based vehicle detection for driver assistance has received considerable attention over the last 18 years. There are at least three reasons for the blooming research in this field: 1) the startling losses both in human lives and finance caused by vehicle accidents, 2) the availability of feasible technologies accumulated within the last 30 years of computer vision research, and 3) the exponential growth in processor speeds have paved the way for running computation-intensive video-processing algorithms even on a low-end PC in real-time.

With the ultimate goal of building autonomous vehicles, many government institutions, automotive manufacturers and suppliers, and R&D companies have launched various projects worldwide, involving a large number of research units working cooperatively. These efforts have produced several prototypes and solutions, based on rather different approaches [5-6]. Looking at research on intelligent vehicles worldwide, Europe pioneers the research, followed by Japan and United States. In Europe, the PROMETHEUS project (Program for European Traffic with Highest Efficiency and Unprecedented Safety) started this exploration in 1986. More than 13 vehicle manufactures and research institutes from 19 European countries were involved. Several prototype vehicles and systems were designed and demonstrated as a result of PROMETHEUS. In 1987, the UBM (Universitaet der Bundeswehr Munich) test vehicle VaMoRs demonstrated the capability of fully autonomous longitudinal and lateral vehicle guidance by computer vision on a 20 km free section of highway at speed up to 96 km/h. Vision was used to provide input for both lateral and longitudinal control. That was considered as the first milestone. Further development of this work has been in collaboration with von Seelen’s group [16] and Daimler-Benz VITA project (Vision Technology Application) [17]. Long range autonomous driving has been demonstrated by the VaMP of UBM in 1995. The trip was from Munich to Odense, Denmark, more than 1,600 km. About 95 percent of the distance was driven without intervention of the safety driver [4]. Another experimental vehicle, mobile laboratory (MOB-LAB), was also part of the PROMETHEUS project [18]. It was equipped with four cameras, several computers, monitors, and a control-panel to give a visual feedback and warnings to the driver.

One of the most promising subsystems in the MOBLAB was the Generic Obstacle and Lane Detection (GOLD) system. The GOLD system, utilizing a stereo rig in the MOBLAB, addressed both lane and obstacle detection at the same time. The lane detection was based on a pattern matching technique, while the obstacle detection was reduced to the determination of the free-space in front of the vehicle using the stereo image pairs without 3D reconstruction. The GOLD system has been ported on ARGO, a Lancia Thema passenger car with automatic steering capabilities [19]. Although the first research efforts on developing intelligent vehicles were seen in Japan in the 1970s, significant research activities have been triggered by prototype vehicles built in Europe in the late-1980s and early-1990s. MITI, Nissan, and Fujitsu pioneered the research in this area by joining forces in the project “Personal Vehicle System” [20], a project with deep influence on Japan. In 1996, the Advanced Cruise-Assist Highway System Research Association (AHSRA) was established among automobile industries and a large number of research centers in Japan [39]. The Japanese Smart Way concept car will implement some driver aid features, such as lane keeping, intersection collision avoidance, and pedestrian detection. A model deployment project was planned to be operational by 2003 and national deployment in 2015 [5]. In the United States, a number of initiatives have been launched to address this problem. In 1995, the US government established the National Automated Highway System Consortium (NAHSC), and launched the Intelligent Vehicle Initiative (IVI) in 1997. Several promising prototype vehicles/systems have been investigated and demonstrated within the last 15 years [21]. The Navlab group at Carnegie Mellon University has a long history of development of automated vehicles and intelligent systems for driver assistance. The group has produced a series of 11 vehicles, Navlab 1 through Navlab 11. Their applications have included off-road scouting, automated highways, run-off-road collision prevention, and driver assistance for maneuvering in crowded city environments. In 1995, NavLab5 demonstrated long range partially autonomous driving (i.e., automatic lateral control) on highways from the east coast to the west. With a more than 5,000 km trip, 98 percent of the distance
was driven without intervention of the human safety driver [22]. The latest model in the Navlab family is the Navlab 11, a robot Jeep Wrangler equipped with a wide variety of sensors for short-range and midrange obstacle detection [23-24]. Major motor companies including Ford and GM have poured great effort into this research and already demonstrated several promising concept vehicles. US government agencies are very supportive of intelligent vehicle research. Recently, the US Department of Transportation (USDOT) has launched a five year, 35 million dollar project with GM to develop and test preproduction rear-end collision avoidance system. In March 2004, the whole world was stimulated by the “grand challenge” organized by The US Defense Advanced Research Projects Agency (DARPA). In this competition, 15 fully autonomous vehicles attempted to independently navigate a 250-mile (400 km) desert course within a fixed time period, all with no human intervention whatsoever—no driver, no remote-control, just pure computer-processing and navigation horsepower, competing for a 1 million cash prize. Although, even the best vehicle (i.e., “Red Team” from Carnegie Mellon) made only seven miles, it was a very big step towards building autonomous vehicles in the future.

The Automated Traffic Surveillance and Control (ATSAC) system developed by the City of Los Angeles Department of Transportation (DOT) was implemented in 1992 at 800 intersections across L.A., the ATSAC has proven to be an important and innovative traffic management tool. The ATSAC program has several key features that distinguish it from conventional or other traffic management systems. It has traffic adaptive signal timing to adjust signals based on road traffic and help smooth the flow of city traffic. It is able to detect road incidents or tie-ups and alert emergency or repair vehicles for dispatch. While according signal priority to light rail transit vehicles, it is capable of detecting system malfunctions continually so that repairs can be made instantly. With closed-circuit video monitoring and real-time relay of traffic information, the system seeks to continuously identify traffic congestion levels and incidents. Compared with other computer-controlled signal systems, ATSAC has the most extensive use of roadway detectors and computation of traffic flow measures to evaluate its own performance.

The most extensive TCSS (Traffic Control and Surveillance System) network, Tsing Ma Traffic Control Area, was introduced in Hong Kong in May 1997. The system had two different types of ATC technologies – the Split, Cycle Offset Optimization Technique (SCOOT) (an IT system for the control of traffic signals in urban areas currently in version MC3 (managing, congestion communications and control) and developed by Siemens, PEEK and TRL) and the Sydney Coordinated Adaptive Traffic System (SCATS) (Tyco Integrated Systems), which links multiple traffic signal control systems [25]. Video image detection systems (VIDS) employ machine vision technology to automatically analyze traffic data collected with Closed Circuit Television (CCTV) systems. Their applications have used to monitor freeway conditions, arterials and intersections, detect incidents and classify vehicles. There are over 5,000 VIDS in existence today throughout the world. Simple VIDS, for signal actuation, speed estimation or traffic counts, are implemented throughout Europe, the United States and Australia. These are present on a more limited basis in Asia (principally in China and Japan). More complex VIDS that automatically detect incidents, estimate queue lengths and actuate variable message signs are less widespread. Examples include: 1) the Migrazur system on the Escota Network of 430km of freeways in the southern Provence-Alpes-Cote d'Azur regions of France; 2) DIVA system in the Les Halles tunnel in central Paris; 3) the A4-A31/Brescia-Padova freeway in Northern Italy; 4) The City Link project in Sydney, one of the biggest and most sophisticated projects covering 22km of high-quality roads.

Active Road Management Assisted by Satellite (ARMAS) is a system for monitoring vehicles via satellite based on the European Geostationary Navigation Overlay Service (EGNOS). EGNOS works by enhancing the data provided by the US GPS system, offering greater precision and signal continuity. It was started in April 2003. The main applications of ARMAS are going to be in the Road domain (Urban and Highways), with particular interest in tolling based on Satellite Positioning. The traffic surveillance project Traficon (B) at imagelab of Universita di Medona e Reggio Emila, Italy. In Mumbai and Bangalore, B-TRAC is going on for traffic surveillance which will be completed in 2010.
4. Vehicle detection

On-board vehicle detection systems have high computational requirements as they need to process the acquired images at real-time or close to real-time to save time for driver reaction. Searching the whole image to locate potential vehicle locations is prohibitive for real-time applications. The majority of methods reported in the literature follow two basic steps: 1) the locations of possible vehicles in an image and 2) where tests are performed to verify the presence of vehicles in an image (see Fig. 3 and Fig. 4). Although there is some overlap in the methods employed for each step, this taxonomy provides a good framework for discussion throughout this survey.

4.1 Possible location

Various possible location based approaches have been proposed in the literature, which can be classified into one of the following three categories: 1) knowledge-based, 2) stereo-based, and 3) motion based. The objective of the possible location step is to find candidate vehicle locations in an image quickly for further exploration. Knowledge-based methods employ a priori knowledge to hypothesize vehicle locations in an image. Stereo-based approaches take advantage of the Inverse Perspective Mapping (IPM) [26] to estimate the locations of vehicles and obstacles in images. Motion-based methods detect vehicles and obstacles using optical flow. The hypothesized locations from the possible location step form the input to the prospective location step, where tests are performed to verify the correctness of the possibility.

4.1.1 Knowledge base method

Knowledge-based methods employ a priori knowledge to hypothesize vehicle locations in an image. We review below some representative approaches using information about symmetry, color, shadow, geometrical features (e.g., corners, horizontal/vertical edges), texture, and vehicle lights. An example-based algorithm for moving vehicle detection is proposed. First, a novel scheme for adaptive background estimation is introduced. Then, the image is divided into many small non overlapped blocks. The candidates of the vehicle part can be found from the blocks if there is some change in gray level between the current image and the background. A low dimensional feature is produced by applying principal component analysis to two histograms of each candidate, and a classifier based on a support vector machine is designed to classify it as a part of a real vehicle or not. Finally, all classified results are combined, and a parallelogram is built to represent the shape of each vehicle.

4.1.2 Symmetry

As one of the main signatures of man-made objects, symmetry has been used often for object detection and recognition in computer vision [27]. Images of vehicles observed from rear or frontal views are in general symmetrical in the horizontal and vertical directions. This observation has been used as a cue for vehicle detection in several studies [28-29]. An important issue that arises when computing symmetry from intensity, however, is the presence of homogeneous areas. In these areas, symmetry estimations are sensitive to noise. Information about edges was included in the symmetry estimation to filter out homogeneous areas (see Fig. 5). In a different study, [30] formulated symmetry detection as an optimization problem which was solved using Neural Networks (NNs).
4.1.3 Colour

Although few existing systems use color information to its full extent for HG, it is a very useful cue for obstacle detection, lane/road following, etc. Several prototype systems have investigated the use of color information as a cue to follow lanes/roads or segment vehicles from background [31, 32]. [32] used two closely positioned cameras to extend the dynamic range of a single camera. One camera was set to capture the shadowed area by opening its iris and the other the sunny area by using a closed iris. Combining color information (i.e., red, green, and blue) from the two images, he formed a six-dimensional color space. A Gaussian distribution was fit to this color space and each pixel was classified as either road or non road pixel. [31] used a nonparametric learning based approach for object segmentation and recognition. A multivariate decision tree was utilized to model the object in the RGB color space from a number of training examples. Among various color spaces, the RGB color space ensures that there is no distortion in the initial color information, however, color features are highly correlated—it is difficult to evaluate the difference of two colors from their distance in RGB color space. In [33] chose the L*a*b color space instead. The L*a*b color space has the property that it maps equally distinct color differences into equal Euclidean distances. An incremental region fitting method was investigated in the L*a*b color space for road segmentation.

4.1.4 Shadow

Using shadow information as a sign pattern for vehicle detection was initially discussed in [34]. By investigating image intensity, it was found that the area underneath a vehicle is distinctly darker than any other areas on an asphalt paved road. A first attempt to deploy this observation can be found in [35], although there was no systematic way to choose appropriate threshold values. The intensity of the shadow depends on the illumination of the image, which in turn depends on weather conditions. Therefore, the thresholds cannot be, by any means, fixed. To segment the shadow area, a low and a high threshold are required. However, it is obvious that it is hard to find a low threshold for a shadow area. The high threshold can be estimated by analyzing the gray level of the “free driving space”—the road right in front of the prototype vehicle. The authors [36] followed the same idea and proposed a method to determine the threshold values. Specifically, a normal distribution was assumed for the intensity of the free driving space. The mean and variance of the distribution were estimated using Maximum Likelihood (ML) estimation. The high threshold of the shadow area was defined as the limit where the distribution of the road gray values declined to zero on the left of the mean, which was approximated by \[ m - 3\sigma \], where \( m \) is the mean and \( \sigma \) is the standard deviation. This algorithm is depicted in Fig. 6. It should be noted that the assumption about the distribution of the road pixels might not always hold true.

4.1.5 Corners

Exploiting the fact that vehicles in general have a rectangular shape with four corners (upper-left, upper-right, lower-left, and lower-right), [37-39] proposed a corner-based method to hypothesize vehicle locations. Four templates, each of them corresponding to one of the four corners, were used to detect all the corners in an image, followed by a search method to find the matching corners (i.e., a
valid upper-left corner should have a matched lower-right corner). The hybrid method based on bounding box and feature tracking is used for tracking the vehicle.

4.1.6 Vertical/horizontal edges

Different views of a vehicle, especially rear/frontal views, contain many horizontal and vertical structures, such as rear window, bumper, etc. Using constellations of vertical and horizontal edges has shown to be a strong cue for hypothesizing vehicle presence. In an effort to find pronounced vertical structures in an image, [40] used edge detection to find strong vertical edges. To localize left and right position of a vehicle, they computed the vertical profile of the edge image (i.e., by summing the pixels in each column) followed by smoothing using a triangular filter. By finding the local maximum peaks of the vertical profile, they claimed that they could find the left and right position of a vehicle. A shadow method, similar to that in [41], was used to find the bottom of the vehicle. Because there were no consistent cues associated with the top of a vehicle, they detected it by assuming that the aspect ratio of any vehicle was one. [42] Proposed a method called Local Orientation Coding (LOC) to extract edge information. An image obtained by this method consists of strings of binary code representing the directional gray-level variation in the pixel's neighborhood. These codes carry essentially edge information. [30] Also used LOC, together with shadow information, for vehicle detection. [43] proposed to extract the general structure of a traffic scene by first segmenting an image into four regions: pavement, sky, and two lateral regions using edge grouping. Groups of horizontal edges on the detected pavement were then considered for hypothesizing the presence of vehicles. [13], utilized edge information to detect distant cars.

They proposed a coarse-to-fine search method looking for rectangular objects. In [44], vertical and horizontal edges were extracted separately using the Sobel operator. Then, two edge-based constraint filters (i.e., rank filter and attached line edge filter) were applied on those edges to segment vehicles from background. The edge-based constraint filters were derived from prior knowledge about vehicles. Assuming that lanes have been successfully detected, hypothesized vehicle presence by scanning each lane starting from the bottom to a certain vertical position, corresponding to a predefined maximum distance in the real world. Potential candidates were obtained if a strong horizontal segment delimited by the lane borders had been found. A multiscale approach which combines sub sampling with smoothing to hypothesize possible vehicle locations more robustly was proposed in [45-46] to address the above problems.

4.1.7 Texture

The presence of vehicles in an image causes local intensity changes. Due to general similarities among all vehicles, the intensity changes follow a certain texture pattern [47]. This texture information can be used as a cue to narrow down the search area for vehicle detection. Entropy was first used as a measure for texture detection. For each image pixel, a small window was chosen around it, and the entropy of that window was considered as the entropy of the pixel. Only regions with high entropy were considered for further processing. Another texture-based segmentation method suggested in uses co-occurrence matrices introduced in [48]. The co-occurrence matrix contains estimates of the probabilities of co-occurrences of pixel pairs under predefined geometrical and intensity constraints. Fourteen statistical features were computed from the co-occurrence matrices. For typical textures of geometrical structures, like trucks and cars, four measurements out of the 14 were found to be critical for object detection (i.e., energy, contrast, entropy, and correlation). Using co-occurrence matrices for texture detection is more accurate in general than using the entropy method mentioned earlier since co-occurrence matrices employ second order statistics as opposed to histogram information employed by the entropy method. However, computing the co-occurrence matrices is expensive. A normal flow computation and a 2D multiframe motion analysis for locating [49].

4.1.8 Vehicle lights

Most of the cues discussed above are not helpful for nighttime vehicle detection—it would be difficult or impossible to detect shadows, horizontal/vertical edges, or corners in images obtained at night conditions. A salient visual feature during night time is the vehicle lights. A morphological analysis [50] was used to detect vehicle light pairs in a narrow inspection area. The morphological operator also considered the shape, size, and minimal distance between vehicles to provide hypotheses.
4.2 Stereo-Vision-Based Methods

There are two types of methods that use the stereo information for vehicle detection. One uses disparity map, while the other uses an anti perspective transformation—Inverse Perspective Mapping (IPM). We assume that camera parameters have already been computed through calibration [81].

4.2.1 Disparity Map

The difference in the left and right images between corresponding pixels is called disparity. The disparities of all the image points form the disparity map. If the parameters of the stereo rig are known, the disparity map can be converted into a 3D map of the viewed scene. Computing the disparity map is very time consuming due to the requirement of solving the correspondence problem for every pixel; however, it is possible to do it in real-time using a Pentium class processor or embedded hardware [51]. Once the disparity map is available, all the pixels within a depth of interest according to a disparity interval are determined and accumulated in a disparity histogram. If an obstacle is present within the depth of interest, then a peak will occur at the corresponding histogram bin (i.e., similar idea to the Hough transform). It was argued that, to solve the correspondence problem, area-based approaches were too computationally expensive, and disparity maps from feature-based methods were not dense enough. A local feature extractor (i.e., “structure classification”) was proposed to solve the correspondence problem faster. According to this approach, each pixel was classified into various categories (e.g., vertical edge pixels, horizontal edge pixels, corner edge pixels, etc.) based on the intensity differences between the pixel and its four direct neighbors. To simplify finding pixel correspondences, the optical axes of the stereo-rig were aligned in parallel (i.e., corresponding points were on the same row in each image). Accordingly, their search for corresponding pixels was reduced to a simple test (i.e., whether two pixels belong to the same category or not). Obviously, there are cases where this approach does not yield unique correspondences. To address this problem, they further classified the pixels by their associated disparities into several bins by constructing a disparity histogram. The number of significant peaks in the histogram indicated how many possible objects were present in the images.

4.2.2 Inverse Perspective Mapping

The term “Inverse Perspective Mapping” does not correspond to an actual inversion of perspective mapping, which is mathematically impossible. Rather, it denotes an inversion under the additional constraint that inversely mapped points lay on the horizontal plane. If we consider a point p in the 3D space, perspective mapping implies a line passing through this point and the center of projection N, see Fig. 7. To find the image of the point, we intersect the line with the image plane. IPM is defined by the following procedure: For a point pi’ in the image, we trace the associated ray through N towards the horizontal plane. The intersection of the ray with the horizontal plane is the result of the inverse perspective mapping applied to the image point pi’. If we compose both perspective and inverse perspective, the horizontal plane is mapped onto itself, while elevated parts of the scene appear distorted. Assuming a flat road, [52] used stereo vision to predict the image seen from the right camera, given the left image, using IPM. Specifically, they used IPM to transform every point in the left image to world coordinates, and reprojected them back onto the right image, which were then compared against the actual right image. In this way, they were able to find contours of objects above the ground plane. Instead of warping the right image onto the left image, [18-19] computed the IPM of both the right and left images. Then, they took the difference between the two remapped left and right images. Due to the flat-road assumption, anything elevating out from the road was detected by looking for large clusters of nonzero pixels in the difference image. In the ideal case, the difference image contains two triangles for each obstacle that correspond to the left and right boundaries of the obstacle. This is because, except for those pixels on the left and right boundaries of the obstacle, all other pixels are the same in the left and right remapped images. Locating those triangles, however, was very difficult due to texture, irregular shape, and non homogeneous brightness of obstacles. To deal with these issues, they used a polar histogram to detect the triangles. Given a point on the road plane, the polar histogram was computed by scanning the difference image and counting the number of over threshold pixels for every straight line originating from that point. [53] They clustered the elevated 3D points based on their distance from the ground plane to generate hypotheses. Each hypothesis was tracked over time.
and further verified using Kalman filters [54]. This system assumed that the dynamic behavior of the host vehicle was known, and the path information was stored in a dynamic map. The system was able to detect vehicles up to 150 m under normal daytime weather conditions. Although only two cameras are required to find the range and elevated pixels in an image, there are several advantages to use more than two cameras [79-80]: 1) repeating texture can confuse a two cameras system by causing matching ambiguities, which can be eliminated when additional cameras are present and 2) shorter baseline systems are less prone to matching errors while longer baseline systems are more accurate. The combination is better than either one alone. Williamson and Thorpe investigated a trinocular system. The trinocular rig was mounted on top of a vehicle with the longest baseline being 1.2 meters. The third camera was displaced 50 cm horizontally and 30 cm vertically to provide a short baseline. The system reported a capacity of detecting objects as small as 14 cm at range in excess of 100 m. Due to the additional computational costs; however, binocular system is more preferred in the driver assistance system.

Figure 7. Geometry of perspective mapping

4.2.3 Motion-Based Methods

All the cues discussed so far use spatial features to distinguish between vehicles and background. Another cue that can be employed is relative motion obtained via the calculation of optical flow [76]. It estimated optical flow from spatiotemporal derivatives of the gray value images using a local approach. They further clustered the estimated optical flow to eliminate outliers. Assuming a calibrated camera and known ego-motion, they detected both moving and stationary objects. Generating a displacement vector for each pixel (i.e., dense optical flow) is time consuming and also impractical for a real-time system. In contrast to dense optical flow, “sparse optical flow” is less time consuming by utilizing image features, such as corners [56-57] local minima and maxima [58], or Color Blobs [59]. Although it can only produce a sparse flow, feature based methods can provide sufficient information for HG. Moreover, in contrast to pixel-based optical flow estimation methods where pixels are processed independently, feature-based methods utilize high-level information. Consequently, they are less sensitive to noise.

Figure 8. Obstacle detection: (a) left and (b) right stereo images, (c) and (d) the remapped images, (e) the difference image, and (f) corresponding polar histogram

4.3 Possible location verification

The input to the HV step is the set of hypothesized locations from the HG step. During HV, tests are performed to verify the correctness of a hypothesis. Approaches to HV can be classified mainly into two categories: 1) template-based and 2) appearance-based. Template-based methods use predefined patterns from the vehicle class and perform correlation.

Appearance-based methods, on the other hand, learn the characteristics of the vehicle class from a set of training images which should capture the variability in vehicle appearance. Usually, the variability of the non vehicle class is also modeled to improve the performance. Each training image is represented by a set of local or global features. Then, the decision boundary between the vehicle and non vehicle classes is learned either by training a classifier (e.g., NNs, Support Vector Machines (SVMs)) or by modeling the probability distribution of the features in each class (e.g., using the Bayes rule assuming a Gaussian distribution). A method for the detection of moving objects in airborne thermal videos is presented. For the frame to frame registration, points of interest are extracted and tracked through the sequence. Then, the motion of the sensor is estimated using projective
planar homographies as the transformation model [60].

Figure 9. The results: One frame of the a) small road, b) autobahn, c) forest, d) people scene.

4.3.1 Template based method

Template-based methods use predefined patterns of the vehicle class and perform correlation between the image and the template. Some of the templates reported in the literature represent the vehicle class “loosely,” while others are more detailed. It should be mentioned that, due to the nature of the template matching methods, most papers in the literature do not report quantitative results and demonstrate performance through examples. A hypothesis verification scheme based on the presence of license plates and rear windows was proposed by [43]. This can be considered as a loose template of the vehicle class. No quantitative performance was including in the paper. A template [65] based on the observation that the rear/frontal view of a vehicle has a “U” shape (i.e., one horizontal edge, two vertical edges, and two corners connecting the horizontal and vertical edges). During verification, they considered a vehicle to be present in the image if they could find the “U” shape. Ito et al. used a very loose template to recognize vehicles. Using active sensors for HG, they checked whether or not pronounced vertical/horizontal edges and symmetry existed. Due to the simplicity of the template, they did not expect very accurate results, which was the main reason for employing active sensors for HG. They argued that the visual appearance of an object depends on its distance from the camera. Consequently, they used two slightly different generic object (vehicle) models, one for nearby objects and another for distant objects. This method, however, raises the question of what model to use in a specific location. Instead of working with different generic models, distance-dependent sub sampling was performed before the verification step in [61]. A template, called “moving edge closure,” was used in [50] which was predefined range, they claimed vehicle detected. Nighttime vehicle detection was also addressed in this work. Basically, pairs of headlights were considered as templates for vehicle detection. A rather loose template was also used, where hypotheses were generated on the basis of road position and perspective constraints. The template contained a priori knowledge about vehicles: “A vehicle is generally symmetric, characterized by a rectangular bounding box which satisfies specific aspect ratio constraints.” The model matching worked as follows: Initially, the hypothesized region was checked for the presence of two corners representing the bottom of the bounding box, similar to the “U” shape idea. The presence of corners was validated using perspective and size constraints. Then they detected the top part of the bounding box in a specific region determined, once again, by perspective and size constraints. Once the bounding box was detected successfully, they claimed vehicle presence in that region. This template could be very fast, however, it introduces some uncertainties, given that there might be other objects on the road satisfying those constraints (e.g., distant buildings).fit to groups of moving points. To get the moving edge closure, they performed edge detection on the area covered by the detected moving points, followed by the external edge connection.

4.3.2 Appearance method

HV using appearance models is treated as a two-class pattern classification problem: vehicle versus non vehicle. Building a robust pattern classification system involves searching for an optimum decision boundary between the classes to be categorized. Given the huge within-class variability’s of the vehicle class, we can imagine that this is not an easy task. One feasible approach is to learn the decision boundary based on training a classifier using the feature sets extracted from a training set. Appearance-based methods learn the characteristics of vehicle appearance from a set of training images which capture the variability in the vehicle class. Usually, the variability of the non vehicle class is also modeled to improve performance. First, a large number of training images is collected and each training image is represented by a set of local or global features. Then, the decision boundary between the vehicle and non vehicle classes is learned either by training a classifier or by modeling the probability distribution of the features in each class. Various feature extraction methods have been investigated in the context of vehicle detection. Based on the method used, the features extracted can be classified as either
local or global. Global features are obtained by considering all the pixels in an image. [62] Used standard Principal Components Analysis (PCA) for feature extraction, together with a nearest-neighbor classifier, reporting 89% accuracy. However, their training database was quite small (93 vehicle images and 134 non vehicle images), which makes it difficult to draw any useful conclusions. An inherent problem with global feature extraction approaches is that they are sensitive to local or global image variations (e.g., pose changes, illumination changes, and partial occlusion). Local feature extraction methods on the other hand are less sensitive to these effects. Moreover, geometric information and constraints in the configuration of different local features can be utilized either explicitly or implicitly. Different from [38], in [40], PCA was used for feature extraction and Neural Networks (NNs) for classification. First, each sub image containing vehicle candidates was scaled to 20 x 20, then it was subdivided into 25 4 x 4 subwindows. PCA was applied on every subwindow (i.e., local PCA) and the output was provided to a NN to verify the hypothesis. [42], [63] used the LOC method to extract edge information. The histogram of LOC within the area of interest was then provided to a NN classifier, a Baye’s classifier and combination of both for classification. For NN, the number of nodes in the first layer was ranges from 350-450 while the number of hidden nodes was 10-40. They used 2,000 examples for training and the whole system ran in real-time. The performance of the neural net classifier was 94.7%, which is slightly better than their Baye’s classifier 94.4%, also very close to the combined classifier (95.7%). Two models for vehicle detection proposed [64]: one for sedans and the other for trucks. Two different model generation methods were used. The first one was designed manually, while the second one was based on a statistical algorithm using about 50 typical trucks and sedans. Classification was performed using NNs. The input to the NNs was the Hausdorff distances between the hypothesized vehicles and the models, both represented in terms of the LOC. The NN classified every input into three classes: sedans, trucks, or background. The histogram of LOC, together with a NN, for vehicle detection is used. The Hausdorff distance was used for the classification of trucks and cars. No quantitative performance was reported in [65] or [64]. A statistical model of vehicle appearance was investigated by [66]. A view-based approach employing multiple detectors was used to cope with viewpoint variations. The statistics of both object and “non-object” appearance were represented using the product of two histograms with each histogram representing the joint statistics of a subset of Haar wavelet features and their position on the object. A three-level wavelet transform was used to capture the space, frequency, and orientation information. This three level decomposition produced 10 sub bands and 17 subsets of quantized wavelet coefficients were used.

Bootstrapping was used to gather the statistics of the non vehicle class. The best performance reported in a paper was found as 92%. A different statistical model was investigated by Weber et al. They represented each vehicle image as a constellation of local features and used the Expectation-Maximization (EM) algorithm to learn the parameters of the probability distribution of the constellations. They used 200 images for training and reported 87% accuracy. An over complete dictionary of Haar wavelet features was utilized in [67] for vehicle detection.

They argued that this representation provided a richer model and spatial resolution and that it was more suitable for capturing complex patterns. The over complete Haar wavelet features were derived from a set of redundant functions, where the wavelets at level n was 1/4 x 2n instead of 2n. They referred it to as quadruple density dictionary. A total of 1,032 positive training patterns and 5,166 negative training patterns were used for training and the ROC showed that the false positive rate was close to 1 percent when the detection rate approached to 100%. The actual values of the wavelet coefficients are not very important for vehicle detection [68]. In fact, coefficient magnitudes indicate local oriented intensity differences, information that could be very different even for the same vehicle under different lighting conditions. Following this observation, they proposed using quantized coefficients to improve detection performance. The quantized wavelet features yielded a detection rate of 93.94 % compared to 91.49 % using the original wavelet features. Using Gabor filters for vehicle feature extraction was investigated. Gabor filters provide a mechanism for obtaining orientation and scale tunable edge and line detectors. Vehicles contain strong edges and lines at different orientation and scales; thus, these types of features are very effective for vehicle detection. The hypothesized vehicle sub images were subdivided into nine overlapping subwindows. Gabor filters were then applied on each subwindow separately. The magnitudes of the responses of the Gabor filters were collected from each subwindow and represented by
three moments: the mean $\mu$, the standard deviation $\sigma$, and the skewness $\kappa$. Classification was performed using Support Vector Machines (SVMs) yielding an accuracy of 94.8%. A “vocabulary” of information-rich vehicle parts was constructed automatically by applying the Forstner interest operator onto a set of representative images, together with a clustering method [69]. Each image was represented in terms of parts from this vocabulary to form a feature vector, which was used to train a classifier to verify hypotheses. Some successful detection was reported under high degree of clutter and occlusion, and an overall 90.5% accuracy was achieved. Following the same idea (i.e., detection using components), Leung investigated a different vehicle detection method. Instead of using the Forstner interest operator, differences of Gaussians were applied onto images in scale space, and maxima and minima were selected as the key-points. At each of the key points, the Scale Invariant Feature Transform (SIFT) was utilized to form a feature vector, which was used to train a SVM Classifier. Leung tested his algorithm on the UIUC data showing slightly better performance.

Figure 10. Vehicle detection using the MBR algorithm

5. Tracking of vehicle

In the last decades, one of the most important efforts in ITS research has been the development of visual surveillance systems that could help reduce the number of traffic incidents and traffic jams in urban and highway scenarios. Although the large number of systems based on different types of sensors and their relative performance, vision based systems are very useful to collect very rich information about road traffic. Tracking of vehicle some becomes difficult due to bad weather condition. The tracking of vehicle based on dynamic template derived from their temporal coherence proposed by [70]. The proposed solution is based on a dual stage approach, using a pixel-level stage to extract foreground object from background scenes and a block-level stage to detect and track vehicles. The pixel-level stage combines a multi-background modeling with a dynamic thresholding, using low-scale quasi-connected-components as a first stage for image object grouping/cleaning.

Figure 11. Trajectory of the car and the locations where it stopped (in blue).

The block level performs an 8x8 block-region analysis defining a block energy function that is used to label the blocks belonging to different vehicles and track them over a stack of images, Kalman or particle filter [71-72], proposed incorporation of temporal differencing and shape detection in an appearance based object tracking algorithm. A rule-based method is developed to track the objects between frames, based on the values of the variables. The proposed system uses a forward moving camera in moving car to estimate the distance of the car to other vehicle on the road.

The author proposed an explicit contour model, which not only provides a good approximation to the contours of all classes of vehicles but also embeds the contour dynamics in its parameterized template. The optimization procedure is kept efficient through incremental computation and conservative hypothesis pruning [73]. The vehicles were identified among many stabilization artifacts and tracked, with a simple tracker based on spatiotemporal connected components analysis. Dynamic objects are identified using both background elimination and background registration techniques [74]. A dual-stage approach [75], using a pixel level stage to extract foreground object from background scenes and a block-level stage to detect and track. An adaptive background learning and subtraction method is applied to a real life traffic video sequence to obtain more accurate Spatio-temporal information of the vehicle objects.

6. Conclusion

Moving object tracking is a key task in video monitoring applications. The common problem is occlusion detection. We have presented a survey of vision-based on-road vehicle detection systems—one of the most important components of any driver assistance system. On-road vehicle detection using
optical sensors is very challenging and many practical issues must be considered. Depending on the range of interest, different methods seem to be more appropriate. An integrated system of highway traffic surveillance and control is likely to form an increasingly important role as part of the Intelligent Transport Systems of the near future. The technology exists today and, as this successful project demonstrates, works well in real life, and can be adapted to any type of limited-access highway. Although the benefits may be hard to measure, they are clearly worthwhile for society as a whole. Video data is one of the most promising traffic surveillance techniques nowadays. The experimental results on real traffic video data show that our vehicle detector has strong abilities to deal with different weather and illuminating conditions. More than one stage processing will increase the robustness of the system (Batista et al.). In spite of the technical challenges that lie ahead, we believe that some degree of optimism is justifiable based on the progress that this domain has seen over the last few years. Judging from the research activities in this field worldwide, it is certain that it will continue to be among the hottest research areas in the future. Major motor companies, government agencies, and universities, are all expected to work together to make significant progress in this area over the next few years. Rapidly falling costs for the sensors and processors combined with increasing image resolution provides the basis for a continuous growth of this field.

References

[17] Ulmer B.: ‘Vita—an Autonomous Road Vehicle (ARV) for Collision Avoidance in


[85] B. Leibe and B. Schiele. Scale Invariant Object Categorization Using a Scale-Adaptive


