Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Science Letters:

Blind spot detection using vision for automotive applications

Miguel Ángel SOTELO[†], José BARRIGA
(*Department of Electronics, University of Alcalá, Alcalá de Henares 28871, Spain*)
[†]E-mail: sotelo@depeca.uah.es
Received Feb. 12, 2008; revision accepted July 29, 2008

Abstract: This paper describes a vision-based system for blind spot detection (BSD) in intelligent vehicle applications. A camera is mounted in the lateral mirror of a car with the intention of visually detecting cars that are located in the so-called blind spot and cannot be perceived by the vehicle driver. The detection of cars in the blind spot is carried out using computer vision techniques, based on optical flow and a double-stage data clustering technique for robust vehicle detection.

Key words: Computer vision, Optical flow, Blind spot detection (BSD), Intelligent vehicles
doi:10.1631/jzus.A0820111 **Document code:** A **CLC number:** TP391.41

INTRODUCTION

Given that the main sensor for providing car detection is vision, some car suppliers have considered treating this problem using infrared LEDs or 24-GHz short-range radar. In contrast to the use of vision, infrared and radar-based systems make use of active systems that can cause interference with other vehicles equipped with similar devices. On the contrary, cameras are cheap passive sensors that do not emit any beams or waves. Vision-based systems are thus more suitable for mass production in the automotive industry (Mobileye, 2007; Volvo, 2007) and massive deployment on roads and highways.

In this letter, a vision-based blind spot detection (BSD) system has been developed for intelligent vehicle applications. In our system, images are analyzed using optical flow techniques (Giachetti *et al*., 1998) to detect pixels that move in the same direction as the ego-vehicle. Pixels producing movement as described are grouped following the clustering techniques described in (Kailunailen, 2002). The resulting clusters are considered as potential vehicles overtaking the ego-vehicle. A major novelty of the proposed BSD is the use of a double-stage detection mechanism for providing robust vehicle detection. In the first stage, a pre-detector system computes the mass centre of the resulting clusters and determines whether the detected cluster is a potential vehicle according to the size of detected pixels. In the second stage, another detector looks for the appearance of vehicles' frontal parts. Any object looking like the frontal part of a vehicle is considered as a potential vehicle, whenever the mass centre pre-detector triggers the pre-detection signal. Thus, a sufficiently big object in the image plane, producing optical flow in the same direction as the ego-vehicle and exhibiting a part similar to the frontal part of a car, is validated as a car entering the blind spot. The position of the vehicle in the image plane is computed and tracked using a Kalman filter. Tracking continues until the vehicle disappears from the scene, and an alarm signal is triggered to indicate the driver that a vehicle has entered the blind spot zone.

SYSTEM DESCRIPTION

The description of the algorithm is provided in Fig.1 in the form of flow diagram. As can be observed, there are several computation steps based on optical flow computing at the image level, pixel-wise clustering, analysis of clusters and final vehicle detection.

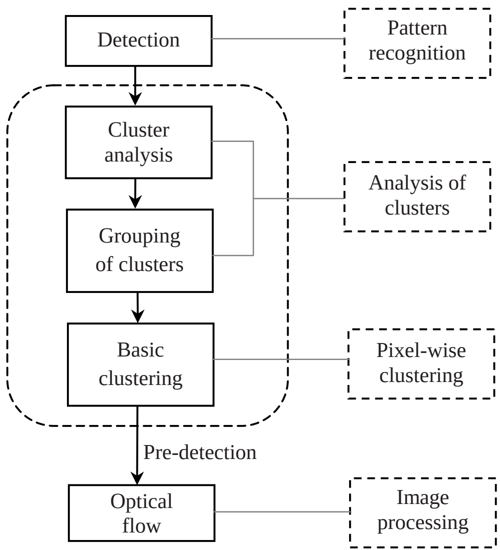

Fig.1 Flow diagram of the blind spot detection algorithm

According to the preceding statements, the system relies on the computation of optical flow using vision as the main sensor to provide information about the road scene. In order to reduce computation time, optical flow is computed only on relevant points in the image. These points are characterized to exhibit certain features that discriminate them from other points in their environment. Normally, these salient features have prominent values of energy, entropy, or similar statistics. In this work, a salient feature point has been considered as the one that exhibits a relevant differential value. Accordingly, a Canny edge extractor is applied to the original incoming image. Pixels providing a positive value after the Canny filter are considered for the calculation of optical flow. This is because relevant points are needed for optical flow computation and matching of the points has to be done between two consecutive frames.

Canny edge pixels are consequently matched and grouped together to detect clusters of pixels that can be considered as candidate vehicles in the image. Classical clustering techniques are used to determine groups of pixels, as well as their likelihood to form a single object. Even after pixel clustering, some clusters can still be clearly regarded as belonging to the same real object. A second grouping stage is then carried out among different clusters to determine which of them can be further merged into a single blob. For this purpose, simple distance criteria are considered. Two objects that are very close to each other are finally grouped together in the same cluster. The reason for computing a two-stage clustering process relies on the fact that by selecting a small distance parameter in the first stage, interesting information about clusters in the scene can be obtained. Otherwise, using a large distance parameter in the single clustering process, highly gross clusters would have been achieved, losing all information about the granular content of the points that provide optical flow in the image.

The selected clusters constitute the starting point for locating candidate vehicles in the image. For that purpose, the detected positions of clusters are used as a seed point to search for a collection of horizontal edges that could potentially represent the lower part of a car. The candidate is located on the detected horizontal edges that meet certain conditions of entropy and vertical symmetry. Some of the most critical aspects in BSD are listed below:

(1) Shadows on the asphalt due to lampposts, other artefacts or a large vehicle overtaking the ego-vehicle on the right lane;

(2) Self-shadow reflected on the asphalt (especially problematic in sharp turns like in round-about points), or on road protection fences;

(3) Robust performance in tunnels;

(4) Avoiding false alarms due to vehicles on the third lane.

The flow diagram of the double-stage detection algorithm is depicted in Fig.2. As can be observed, there is a pre-detector that discriminates whether the detected object is behaving like a vehicle or not. If so, the frontal part of the vehicle is located in the region of interest and a pre-warning message is issued. In addition, the vehicle mass centre is computed. In case the frontal part of the vehicle is properly detected and its mass centre can also be computed, a final warning message is issued; and at that point vehicle tracking starts. Tracking is stopped when the vehicle gets out of the image. Sometimes, the shadow of the vehicle remains in the image for a while after the vehicle disappears from the scene, provoking the warning alarm to hold on for 1~2 s. This is not a problem, however, since the overtaking car is running in parallel with the ego-vehicle during that time although it is out of the image scene. Thus, maintaining the alarm in such cases turns out to be a desirable side effect. After being located, vehicle candidates are classified by using an SVM classifier previously trained with the samples obtained from real road images.

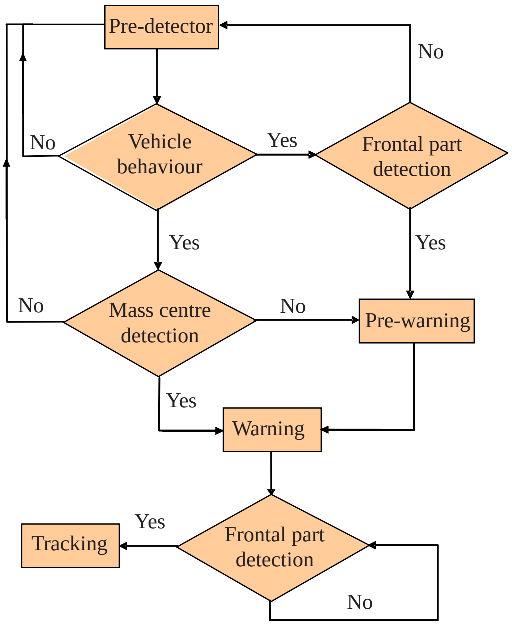

Fig.2 Flow diagram of the double-stage detection algorithm

IMPLEMENTATION AND RESULTS

A digital Fire-i camera (IEEE 1394a) providing 640×480 gray scale images at 30 frames/s was mounted in the lateral mirror of a real car equipped with a Pentium IV 2.8 GHz PC running Linux Knoppix 3.7 and OpenCV libraries 0.9.6. The installation of the camera was carried out using a supplementary element attached to the left-hand side lateral mirror of the car. Thus, the driver maintains full visibility of the scene by means of the lateral mirror, while incorporating a new element that provides additional safety by using the vision-based processing. The car was manually driven for several hours in real highways and roads. After the experiments, the system achieved a detection rate of 99% (1 missing vehicle), producing 5 false positive detections. Fig.3 shows an example of BSD in a sequence of images. The indicator depicted in the upper-right part of the figure toggles from green to blue when a vehicle enters the blind spot area (indicated by a green polygon). A blue bounding box depicts the position of the detected vehicle.

Fig.3 Example of blind spot detection in a sequence of images. The indicator in the upper-right part of the figure toggles from green to blue when a car is detected in the blind spot

Video results in real traffic conditions using our BSD system can be retrieved from ftp://www.depeca.uah.es/pub/vision/BSD.

CONCLUSION AND FUTURE WORK

We have developed a vision-based blind spot detection (BSD) system for active safety in automotive applications. The anticipation of vehicles in the blind spot of the driver is provided in a range of some 20 m. The BSD can warn the driver about the presence of other vehicles in the blind spot area, thus providing assistance in lane change and overtaking manoeuvres. Our results are encouraging yielding a detection rate of 99% while keeping low false positive rates. The current research focuses on the development of SVM-based vehicle recognition, as proposed in (Blanc and Steux, 2007) for increasing the detection rate and decreasing the false alarm rate. It has also been demonstrated in (Sotelo *et al.*, 2005), where SVM was used for vehicle detection in an ACC (adaptive cruise control) application. In addition, we look forward to combining the BSD camera with a rear-looking camera to provide what can be denoted as ‘panoramic BSD’.

References

Blanc, N., Steux, B., 2007. LaRASideCam: A Fast and Robust Vision-based Blind Spot Detection System. Proc. IEEE Intelligent Vehicles Symp., Istanbul, Turkey, p.480-485.

Giachetti, A., Campani, M., Torre, V., 1998. The use of optical flow for road navigation. *IEEE Trans. on Rob. Autom.*, **14**(1):34-48. [doi:10.1109/70.660838]

Kailunailen, J., 2002. Clustering Algorithms: Basics and Visualization. T61.195, Special Assignment 1. Laboratory of Computer and Information Science, Helsinki University of Technology.

Mobileye, 2007. Blind Spot / Lane Change by Mobileye. Http://www.mobileye-vision.com

Sotelo, M.A., Nuevo, J., Ocaña, M., Bergasa, L.M., 2005. A monocular solution to vision-based ACC in road vehicles. *LNCS*, **3643**:507-512. [doi:10.1007/11556985_65]

Volvo, 2007. Blind Spot Information System (BLIS) by Volvo. Http://www.volvo.com