Stopped Vehicle Detection System for Outdoor Traffic Surveillance

Gonçalo Monteiro, João Marcos, and Jorge Batista *
Institute for System and Robotics
DEEC – FCT – University of Coimbra – Portugal

Abstract

In this paper a stopped vehicle detection system for outdoor traffic surveillance is presented. This system is based on the pixel history cache called Codebook. A temporal validation was also used to discard some false positives. This methodology has proved to be quite robust in terms of different weather conditions, lighting and image quality. Some experiments carried out on some highway scenarios demonstrate the robustness of the proposed solution.

1 Introduction

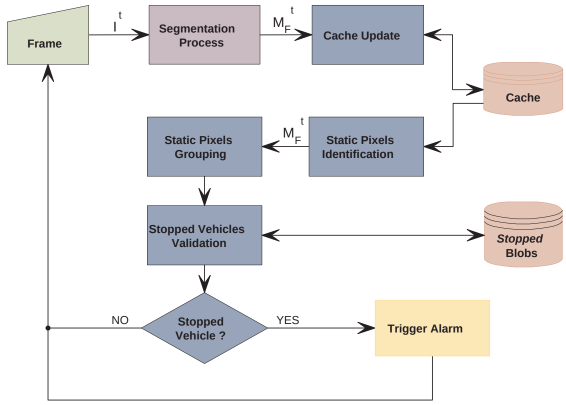
In order to ensure a safe and efficient driving, it is important to classify vehicles' behaviors and to understand their interactions in typical traffic scenarios. Not long ago, this burdensome task was performed by human operators at traffic control centers. However, increasing number of available cameras dictated the need for automatic traffic surveillance systems.

The work presented in this paper is part of an automatic traffic surveillance system [1, 4, 3]. A vehicle stopped on the road or on the hard shoulder can represent a serious threat. An immediate detection of a stopped vehicle could help prevent serious accidents by warning the oncoming vehicles and highway assistance services or, ultimately, by warning the police. This paper focus on the detection of vehicles stopped on highways. For that purpose a segmentation process was used to identify vehicles in the image. The result of the segmentation process was used as input in a stopped vehicles detection system. This stopped vehicles detection system has three main phases. Firstly, it is verified if there are any static pixels segmented during a certain period of time. Secondly, those static pixels are then grouped into blobs. The static pixels identification is based on a pixel history cache analysis. Finally, a blob temporal validation is applied to discard false positives in traffic jam situations. If the validation succeeds, an alarm is triggered in the traffic telematic system.

2 Robust Vehicle Segmentation

The core of an incident detection system has to be an accurate and robust segmentation process. In this paper, a robust segmentation process for outdoor scenarios was used [3]. This system is based on background subtraction, and uses three background to model the background variations. Two distinct thresholds are used: a per-pixel threshold for a robust adaptation to the scene variations, and a global threshold, that is the minimum value, to eliminate camera noise scenario features. It is also used a shadow/highlight detection algorithm based on cross-correlation to discard the blobs generated by lighting variation. This segmentation process proved to be a good basis for an incident detection system.

*This work was supported by BRISA, Auto-estradas de Portugal, S.A.



(a)

Figure 1. Overview of the stopped vehicles detection process here proposed.

3 Stopped Vehicles Detection

The main assumption for the stopped vehicles detection method is that some foreground pixels with the same color that appears during a large period, are probably part of a stopped vehicle. After a blob has been formed by these pixels grouping, it will only be considered as stopped if it holds approximately the same position and dimensions for a predefined validation time (see Fig. 1).

3.1 Stopped Pixels Recognition

The stopped pixels are identified by the analysis of the segmented pixels. A pixel is classified as *static* if it is labeled as foreground with the same color with a certain frequency. A pixel history cache [2] is used to analyze the pixel color frequency. For each pixel it is analyzed a set of colors that appears at least one time in the last T_h frames. Cache is an array of the same size of the image. Each entry corresponds to a pixel and it has a list of Codewords (called Codebook). A Codeword saves the RGB color components and a validation buffer that keeps occurrence history of a color in the pixel in the last α frames. Occlusions, vibrations or lighting changes that can temporarily hide or change a vehicle's pixel color are taken into account with the pixel history cache.

For each foreground pixel in the present frame it is checked if the color matches any Codeword present in the Codebook. A match event exists if the euclidean distance between two colors, in RGB space, is below $\epsilon = 30$. If there is a match, with a Codeword in Cache, RGB components are updated by a weighted average, and the validation buffer is updated with '1'. For all the other Codewords present in the Codebook the validation buffer is updated with '0'. If there is no Codeword matching, a new Codeword is created in the pixel Codebook. The validation buffer of the Codewords of non segmented pixels are also updated with '0'. Finally, all the Cache Codewords that not appear at least one time in the last T_h frames are deleted ($T_h = 25$). A pixel is validated as static if exists a Codeword on its Codebook with β occurrences in the last α frames, where α is the validation buffer size (namely $\beta = 40$ and $\alpha = 64$). This strategy handles the occlusion problem mentioned above. Fig. 2 b) shows chromatically the number of occurrences of the most frequent Codeword.

3.2 Stopped Vehicle Validation

Once identified the static pixels, they are grouped into a blob. The validation process of a possible stopped vehicle starts if the blob has a significant area. Not all the blobs that result from static pixels grouping are really stopped vehicles. They can be vehicles moving slowly in the image or wrong segmented regions. The main idea of the validation process is to track the blob in validation and verify if it maintains the same position with the same dimensions during a certain validation period. In this situation, the blob is considered as a stopped vehicle, and an alarm is triggered.

4 Experimental Results

The system was tested with a real set of image sequences from highways traffic surveillance cameras with different weather conditions, lighting, image quality and fields of view. The system was also tested in real time in some Portuguese high traffic density highway scenarios and in a car-park at Coimbra University. Table 4 shows the experimental results obtained in some different scenarios. In Fig. 2, it is illustrated the stopped vehicles detection. In the experiments performed to the system it was verified that the mis-detection of stopped vehicles was due to the lack of image contrast. Most of the false positives are related to a wrong segmentation result. The system is able to detect stopped vehicles over a 320×240 pixel image at 11 fps on a 3.2 GHz *P4 Intel Processor* under Linux OS, most of the processing time is used by the segmentation process.

(a) (b)

Figure 2. Stopped vehicles detection system experiments. a) stopped vehicle detection (red box); b) chromatic representation of the number of occurrences of the most frequent Codeword.

Table 1. Experiments conducted on the stopped vehicle detection system.

	Period	Stopped	Hits	FP
Tunnel	8 minutes	4	4	1
Car-Park	24 hours	85	82	18
Highway 1	24 hours	0	0	7
Highway 2	24 hours	2	2	4

5 Conclusions

In this paper, a stopped vehicles detection system based on a pixel history cache was presented. The experiments conducted on a large number of scenes demonstrate that this system is able to robustly detect stopped vehicles under different weather conditions, lighting, image quality and image compression variation.

References

[1] J. Batista, P. Peixoto, C. Fernandes, and M. Ribeiro. A dual-stage robust vehicle detection and tracking for real-time traffic monitoring. In *IEEE Int. Conference on Intelligent Transportation Systems*, 2006.

[2] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis. Real-time foreground-background segmentation using codebook model. In *Real-time Imaging*, pages 167–256, 2005.

[3] G. Monteiro, J. Marcos, M. Ribeiro, and J. Batista. Robust segmentation for outdoor traffic surveillance. In *IEEE ICIP 08*, 2008.

[4] G. Monteiro, M. Ribeiro, J. Marcos, and J. Batista. Wrong way drivers detection based on optical flow. In *IEEE ICIP 07*, 2007.