MULTI-FEATURE STEREO VISION SYSTEM FOR ROAD TRAFFIC ANALYSIS

Quentin Houben[1], Juan Carlos Tocino Diaz[1], Nadine Warzée[1], Olivier Debeir[1] and Jacek Czyz[2]
[1]*LISA, Universté Libre de Bruxelles, Avenue Franklin Roosevelt 50 CP165/57, Brussels, Belgium*
[2]*Macq Electronique, Rue de l'Aronef, 2, B-1140 Brussels, Belgium*
qhouben@ulb.ac.be, jtocinod@ulb.ac.be

Keywords: Visual traffic analysis, Multi-camera, Object detection, Vehicle Classification.

Abstract: This paper presents a method for counting and classifying vehicles on motorway. The system is based on a multi-camera system fixed over the road. Different features (maximum phase congruency and edges) are detected on the two images and matched together with local matching algorithm. The resulting 3D points cloud is processed by maximum spanning tree clustering algorithm to group the points into vehicle objects. Bounding boxes are defined for each detected object, giving an approximation of the vehicles 3D sizes. A complementary 2D quadrilateral detector has been developed to enhance the probability of matching features on vehicle exhibiting little texture such as long vehicles. The algorithm presented here was validated manually and gives 90% of good detection accuracy.

1 INTRODUCTION

This work presents an application of multi-camera systems to vehicle detection and classification on motorway. Traffic analysis is an active research domain. The behaviour of the road users and the type of vehicle they use becomes a main issue for motorway administrators. We propose here a multi-camera approach to tackle the difficult problem of vehicle recognition and to determine its main characteristics (dimensions and class of the vehicle). Until a few years ago, the main measurement tool in traffic analysis was the inductive loop (Gibson et al., 1998). This system is expensive, requires a lot of effort to be installed, and is not effective in stop and go situations. The laser-based systems (Lourenco et al., 2002) are accurate but are still quite expensive and have problems with high reflective surfaces, like some car roofs. Video analysis remains a good solution since hardware becomes more and more inexpensive and powerful, allowing real-time results. The installation of cameras is relatively cheap and the maintenance cost is low.

Most of the existing solutions are based on a mono-camera system. Several approaches have been developed (Kastrinaki et al., 2003). Background methods are massively used since they demand small computer effort and are simple to realize. The static background is generally defined by forming a mathematical or exponential average of successive images. The background is then subtracted of the images in order to extract moving vehicles (Tan et al., 2007). Other methods use tracked features that are compared to models (Dickinson and Wan, 1989),(Hogg et al., 1984) or used in a more general pattern recognition (Viola and Jones, 2004). All these methods give limited informations about the dimensions of the vehicle (length, width, height) and perform poorly in vehicle class recognition.

In the approach discussed in this work, a multi-camera grayscale system is considered. Two cameras are disposed over the road (on a bridge for example) with distance of 2 m between them. A multi-camera system allows to obtain 3D informations of the scene. These informations allow to determine the dimensions of the vehicle and thus obtain more accurate informations about the vehicle class. With the height information, a distinction can be made, for example, between a minivan and an estate car. As opposed to the mono-camera systems, vehicles are detected and tracked in the 3D world after a matching step between the two images, based on a multi-feature correspondence system. This system includes phase congruency matching and edges matching. Some heavy vehicles, like semi-trailer trucks, are detected and processed separately with more adapted algorithms, based on an original method of quadrilateral surfaces recognition. The feature-based matching and quadri-

lateral detection are complementary: a vehicle has either texture which yields strong local features that can be matched across different views, or has large flat regions which can be processed by the proposed quadrilateral detector. This method does not need vehicle models and therefore is more robust to variability of the vehicle types.

This paper is organized as follows: Section 2 briefly depicts general stereo vision issues; Section 3 presents the stereo construction algorithm; 3D points processing is discussed in Section 4; Quadrilateral detector is discussed in Section 5; Section 6 presents the results of the proposed method; Section 7 concludes and gives perspectives for this approach.

2 GENERAL STEREO VISION ISSUES

Our approach requires a stereo vision algorithm efficient and fast enough to work in real time. Two major issues must be examined: the cameras system calibration and the identification of matching points between the two images. We will focus here on the second point, assuming that the first one is already treated (Zhang, 1998), (Zhang, 2000). There exists a considerable amount of methods on the stereo correspondences problem. We can classify them into two main categories : dense matching methods, that give a correspondence map for each pixel in the image, and sparse matching methods, that give correspondence only for some points of interest. The dense matching methods, exhaustively listed by Scharstein and Szeliski in (Scharstein and Szeliski, 2002), do not suit the application requirements since the objects we try to detect have uniform texture or have reflective surfaces (roads and cars). Matching pixels of such surfaces is very difficult. Furthermore these algorithms generally demand a lot of computation resources. The sparse matching methods require firstly a features identification step (edge detection, corner detection...), which is done separately in the two images. These features can be matched with local or global algorithms. On one hand, global algorithms search a global matching solution for all features by minimizing cost functions. We can cite dynamic programming methods (Ohta and Kanade, 1985), (Kim et al., 2005), graph cut methods (Boykov et al., 2001) and belief propagation methods (Yang et al., 2006). These last methods are efficient but both belief propagation and graph cut are typically computationally expensive and therefore real-time performance is difficult to achieve (Yang et al., 2006). The dynamic programming method has been tested in the framework of this project but does not improve significately the results. On the other hand, local methods depending only on values within a finite window around the considerate pixel, are definitely faster.

Our approach uses multiple types of features and match them by normalised cross-correlation, which is at the same time simple and robust. The resulting difference of horizontal coordinates between two matching points, called disparity, gives an estimate of the distance of the points in the 3D world.

3 STEREO CONSTRUCTION

Both acquired images are first rectified with the cameras calibration data and corrected if optical distortions appear. The use of rectified images reduces significately the complexity of process, since two corresponding points in the left and right image will have equal vertical coordinates.

Different features are identified separately on each image. The first characteristic points used in our implementation are maximum phase congruency points (where the Fourier components of the image are maximally in phase). These are less sensitive to difference of overall contrast between two images and give more points than more classic features such as Harris corners. The phase congruency is computed with wavelets transforms as described in (Kovesi, 1999). When all maximum phase congruency points are obtained, each point of the left image is compared to the points lying on the same horizontal line in the right image. A maximum disparity is set to reduce the search space and accelerate the process. Several similarity measurement systems between surrounding pixels area have been studied in the literature. Our method uses normalized cross-correlation of the phase congruency values in a square window (W_1, W_2) around the two points, defined by

$$C(W_1,W_2) = \frac{\sum(p_1(i,j)-\overline{p}_1)(p_2(i,j)-\overline{p}_2)}{||(p_1(i,j)-\overline{p}_1)(p_2(i,j)-\overline{p}_2)||} \quad (1)$$

where the sum is taken over (i,j), index of points in the square windows W_1 and W_2, $p_1(i,j)$ and $p_2(i,j)$ are the phase congruency at the pixel (i,j) in the image 1 and image 2 respectively, and $\overline{p}_1, \overline{p}_2$, their mean over the square windows W_1, W_2.

A list of scores in the right image is obtained for each point of the left image and in a similar way for each point of the right image. A "winner-take-all" strategy is then used to match the different points: a match is considerate as valid if the correlation is maximum among all the correlations computed on the

same horizontal line in the left image and in the right image. More formally, if x_l is a point in the left image for which we search the corresponding point among the right points x'_i, and W_l and W'_i are square windows centred respectively on x_l and x'_i, x_l corresponds to x'_k if

$$k = arg_i(max_i\, C(W_l, W'_i)) \qquad i = 1..N' \qquad (2)$$

where N' is the number of points of interest on the horizontal line in the right image. Furthermore we check symetricaly that the determined x'_k gives a maximum correlation in the left image with x_l:

$$l = arg_j(max_j\, C(W_j, W'_k)) \qquad j = 1..N \qquad (3)$$

where N is the number of points of interest on the horizontal line in the left image.

This method is relatively fast and presents few outliers. However, the number of 3D points determined in this way is slightly insufficient to exploit these data in all kinds of lighting conditions and for all kinds of vehicles.

The second type of features points are determined in a different way. Edges are firstly detected in both images using a classic method such as Sobel. The images are then scanned line by line. Each intersection between these lines and the detected edges gives one point of interest. These features are then characterised by three parameters : the strength of the edge, the intensity profile on the edge, and the average intensity on the left and right side of the point. For each hypothetical match, a score is computed taking into account the comparison of all these criteria. The final matching is then done in the same way as the phase congruency features; we consider that a match is valid if the score is maximum among all the scores on the same epipolar line in the left image and in the right image.

These two types of features cover very well all objects of interest (trucks, cars, motorbikes...).

4 3D POINTS PROCESSING

At each features match corresponds a 3D point. The coordinates are obtained with the intrinsic parameters of the cameras, using a minimizing algebraic distance algorithm (Hartley and Zisserman, 2004). The plane equation of the road and its principal axis are supposed to be known. The 3D points above the road level are considerate as belonging to a vehicle. The aggregation of the 3D points into vehicle groups is achieved by the minimum spanning tree clustering algorithm ; all the points classified as possible vehicle points form a minimum spanning tree. The 3D points are connected by weighted edges. The minimum spanning tree is built in such way to minimize the sum of the weights. The weights used here are based on Euclidean distance between points. The edges that have a weight greater than a threshold are cut, forming distinct clusters. This threshold is defined by a constant modulated by the points density around the two considered points. The distance used to weight the edges is anisotropic due to the nature of the tackled problem. The weight of an edge between two vertices (x_{11}, x_{12}, x_{13}) and (x_{21}, x_{22}, x_{23}), is defined as :

$$d = \sqrt{\alpha(x_{11} - x_{21})^2 + \beta(x_{12} - x_{22})^2 + \gamma(x_{13} - x_{23})^2} \qquad (4)$$

where α , β and γ are parameters adjusted to give more importance to distances that are parallel to the road axis, x_{i1} is the horizontal axis perpendicular to the road, x_{i2}, the axis parallel to the road and x_{i3}, the axis normal to the road. The 3D points are thus clustered in vehicles. Eventual errors are extracted by examinating the distribution of the height coordinates of points inside each group. Isolated points of the distribution are eliminated. Bounding boxes of the groups are then defined around the points (figure 1). These boxes give a good approximation of the dimensions of the vehicle (length, width, and height) and are easy to track. Therefore the vehicle speed can be measured and dimensions can be averaged over several frames. A vehicle is counted when it is detected more than 6 times. This method works well for "classic" vehicle like cars. This class of vehicles presents an average number of 10 3D points. Light coloured vehicles have more points than dark ones. We consider that a minimum of 3 points is needed to detect vehicles. With 8 points, we can obtain robust informations about vehicle size. Small vehicles satisfy thus well these criteria. However long trucks often present textureless surfaces (i.e. with few features) which produce insufficient number of 3D points for further analysis. These vehicles need therefore a complementary approach.

5 TRUCK DETECTOR

The roofs of long trucks are characterized by uniform rectangular area. A general quadrilateral detector can therefore be used, identifying the four corners of the roof on the two images.

The roofs of semi-trailer trucks that interest us in this case are always preceded by a tractor unit. This one unit is generally well covered by different features, which allows a good detection of the front of the

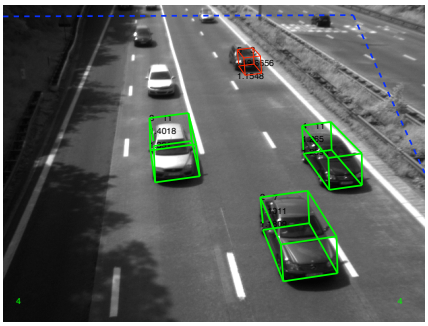

Figure 1: Result of the 3D points clustering : bounding boxes.

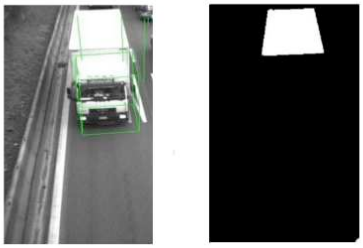

Figure 2: A uniform grayscale zone is detected behind a high box.

vehicle. We can therefore limit the search of quadrilateral surfaces to zones behind 3m high bounding boxes (figure 2). A seed growing method is used to delimit the uniform gray level zone. This detector must be robust enough to recognize the roof of the truck in the segmented zone, which can include segmentation errors, such as side of the semi-trailer, cars that have the same gray level behind the truck, or road details (figure 3).

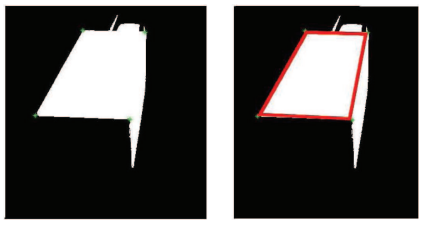

Figure 3: Delimitation of the roof.

The contour of the entire zone is defined and the main straight lines are extracted by examinating the local high curvature points. The intersections between all the extracted lines are considered as potential roof corners. The four final corners will be chosen in such a way that the area of the surface inside these four points is maximum. Several other criteria must be respected :

- the defined quadrilateral must be convex
- corners of the quadrilateral must be close to one of the points of curvature previously detected
- sides of the quadrilateral must be included in the initial segmented zone

This last criterion is adjusted with some tolerance parameter to allow the quadrilateral to include segmentation defaults, like holes, of the initial segmented zone. This method is applied on the two images. If quadrilaterals are detected simultaneously in both images, the four points are matched together and injected in the 3D construction process.

Figure 4 summarizes the all process, from the images capture to the final result.

6 RESULTS

To validate the vehicles detection method, a test was conducted on 3 sequences extracted from a long video of a 4 lanes motorway. These 3 sequences contain a realistic set of vehicle types. A total of 214 vehicles went through the zone covered by the two cameras. A human operator identified the detection errors. These can be classified into 3 categories :

- the vehicle is not detected
- the object detected is not a vehicle
- the vehicle is detected several times

The causes for the miss-detection case are either a bad contrast between a dark car and the shadowed road or a missed image in the camera flow. The first cause could be avoided by using better image properties to permit features detection both in shadowed and lighted zones of the road.

The second category does not appear on the analysed sequence but could be a problem if a mark on the road is permanently miss-matched.

The third category is due either to tracking problem or to over-segmentation of the 3D points, which induces double detections of the same vehicle. This can be avoided using the time parameter, that is not yet used here and will be used in temporal filtering in future development. The results of the 3 sequences (s1, s2, s3) are presented in table 1.

The dimensions of the vehicle are consistent with the vehicle actual characteristics. A test was conducted over 20 vehicles. This test compares the dimensions given by the algorithm of some well identified vehicles (sedans, estate cars, SUV, minivans...) to dimensions furnished by the constructor. The height measurement presents a precision of 92.1%, the length 76.58% and the width 83.35%.

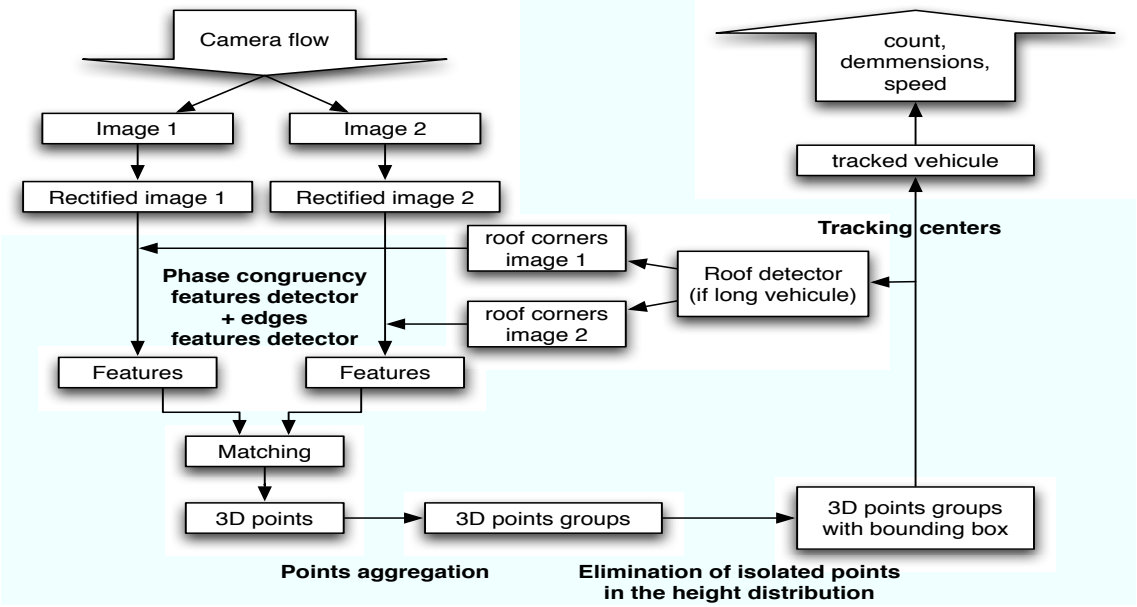

Figure 4: Multi-camera system summary.

Table 1: Table of detection results.

	s 1	s 2	s 3
number of vehicles	107	44	63
number of detected vehicles	110	44	63
total not detected	6	5	7
total not detected in %	5.6	11.4	11.1
total false detections	9	5	7
total false detections in %	9.4	11.4	11.1
- over-segmentation	7	3	2
- tracking error	2	2	5

7 CONCLUSIONS AND FUTURE WORK

In this work an application of multi-camera system for traffic monitoring has been presented. Based on a multi-feature matching and a 3D tracking, the system detects vehicles and determines their dimensions, which is difficult for a classic mono-camera system. Additionally to stereo matching, 2D image processing is used on each camera to detect roofs of long vehicles. This method gives good results even with fast change of lighting condition. Furthermore, 3D reconstructions algorithms are not affected by stop and go situations.

The implementation of the method was realized on Matlab. Implementation on more time-effective language is planed and will allow to measure more precisely computational time requested. A time-filtering method could also be developed to improve the detection results.

An interesting perspective could be a fusion between mono-camera and multi-camera processing. 2D and 3D info could then describe the dynamic scene as a list of 3D objects with position history.

REFERENCES

Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *PAMI*, 23(11).

Dickinson, K. and Wan, C. (1989). Road traffic monitoring using the trip ii system. In *IEE Second International Conference on Road Traffic Monitoring*. The MIT Press TODO check.

Gibson, D., FHWA, Tweedy, C., and Corp., O. (1998). An advanced preformed inductive loop sensor. In *North American Travel Monitoring Exhibition and Conference(NATMEC)*.

Hartley, R. and Zisserman, A. (2004). *Multiple view geometry in computer vision*. Cambridge University Press, second edition.

Hogg, D., Sullivan, G., Baker, K., and Mott, D. (1984). Recognition of vehicles in traffic scenes using geometric models. In *Proceedings of the International Conference on Road Traffic Data Collection, London*. IEE.

Kastrinaki, V., Zervakis, M., and Kalaitzakis, K. (2003). A survey of video processing techniques for traffic applications. In *Image and Vision Computing 21*. Elsevier.

Kim, J., Lee, K., Choi, B., and Lee, S. (2005). A dense stereo matching using two-pass dynamic programming with generalized ground control points. *IEEE CVPR*, 2:1075–1082.

Kovesi, P. (1999). Image features from phase congruency. *Videre: Journal of Computer Vision Research.*

Lourenco, A., Freitas, P., Ribeiroy, M. I., and Marquesy, J. S. (2002). Detection and classification of 3d moving objects.

Ohta, Y. and Kanade, T. (1985). Stereo by intea nad inter-scanline search using dynamic programming. *PAMI.*

Scharstein, D. and Szeliski, R. (2002). a taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *Intl Journal of Computer Vision, vol. 47,no. 1, pp. 742.*

Tan, X.-J., Li, J., and Liu, C. (2007). A video-based real-time vehicle detection method by classified background learning. *World Transactions on Engineering and Technology Education*, 6(1).

Viola, P. and Jones, M. (2004). Robust real-time object detection. *International Journal of Computer Vision.*

Yang, Q., Wang, L., and Yang, R. (2006). Real-time global stereo matching using hierarchical belief propagation. In *BMVC06*, page III:989.

Zhang, Z. (1998). Determining the epipolar geometry and its uncertainty: A review. *IJCV*, 27(2):161195.

Zhang, Z. (2000). Determining the epipolar geometry and its uncertainty. *IEEE TPAMI*, 22:13301334.