

# Estimating Surrounding Vehicles' Pose using Computer Vision

Jesus Nuevo, Ignacio Parra, Jonas Sjöberg, Luis M. Bergasa

**Abstract**—This paper presents a computer vision-based approach to tracking surrounding vehicles and estimating their trajectories, in order to detect potentially dangerous situations. Images are acquired using a camera mounted in the egovehicle. Estimations of the distance, velocity and orientation of other vehicles on the road are obtained by detecting their lights and shadow. Because 3D information is not readily available in a mono-camera system, several sets of constraints and assumptions on the geometry of both road and vehicles are proposed and tested in this paper. Kalman filters are used to track the detected vehicles. We also study the advantages of tracking the vehicles in *road space* (world coordinates), or tracking the position of the lights and shadows on the image. The performance of the approaches is evaluated on video recorded in urban environment.

## I. INTRODUCTION

Vehicle detection and tracking has been the focus of a extensive number of works in recent years, either from cameras that are part of the road infrastructure or from cameras mounted in vehicles. Information about the state of vehicles can be used for a wide variety of purposes ranging from Advanced Driver Assistance Systems (ADAS) to automatic video analysis and detection of potentially dangerous situations.

Detecting and modeling vehicles is difficult because their characteristics vary greatly from one vehicle to another. More importantly, vehicles are driven in uncontrolled environments, where lighting and background can change quickly and unpredictably. The appearance of shadows and occlusions is also frequent.

Several works have been published on vehicle detection and tracking in the last decades [1], [2], [3]. In many cases, the systems fuse image data with other sources, like LIDAR or RADAR, which provide more precise range data [4]. Barth and Franke [5] presented a stereo-camera system to detect and track the vehicles. The stereo vision allowed for 3D data to be readily used. The vehicle was tracked with an Extended Kalman Filter. The system was demonstrated to work during daytime. In [6], a monocular system using an Unscented Kalman Filter to track vehicles was presented. This work assumed a planar road surface. The same authors developed a system [7] that used the distance between the lights of a vehicle in the image to estimate the distance to the vehicle.

Some researchers have used vehicle lights to detect and track vehicles. Most of these works target the problem of

vehicle detection at night, when textures are much more difficult to obtain than during the day. In [8], Alcantarilla *et al.* presented a system that estimated the position of vehicles in front of the egovehicle to lower the light beam automatically. More recently, Fossati *et al.* [9] developed a system to detect the position of the vehicles based on the estimated distance between the rear lights. This system demonstrated good results in environments with many light sources. It worked for vehicles closer than 50 meters, and used color data to filter and pair the lights.

In this paper, we investigate the feasibility of using the car lights and the shadow of the vehicles as measurements. Daytime Running Lamps (DRL) have been mandated in Sweden since the late 1970s. Many other countries in the EU are encouraging its use. In EU, from 2011, ECE R48 will require DRLs conforming to ECE R87 (or full-time low-beam headlamps) on all new motor vehicles. USA and Canada have been working towards the implementation of similar systems since the 90's [10].

Based on this new extended use of DRL, image processing systems based on front/rear lights detection during daytime will be applicable to most road traffic around the world in less than 5 years. Detection and tracking of car lights has some remarkable advantages compared to other techniques: it is more independent from weather conditions, and lights have a well known geometry. Additionally, the appearance of lights is almost the same for all vehicles on the road.

The objective of the system described in this paper is to estimate the surrounding cars trajectory using monocular vision in order to detect potentially dangerous situations that could lead to a crash. The images are obtained from a single front-mounted camera in the egovehicle. Two different cues are searched for in the images: front and rear lights, and the wheels and shadow under the car, as shown in Fig. 1. This process is described in § II. Estimated yaw rate and distance to the car rely on different geometrical models of the road and the vehicles, and are presented in § III. Section IV describes how the lights, wheels and shadows under the car are tracked using a Kalman filter in either image or *road space* (world coordinates). Experimental results are described in § V. The paper closes with conclusions and future work.

## II. IMAGE PROCESSING

### A. Light detection system

Car lights have a very distinctable and stable appearance in video sequences. They have a very well known geometry and show a higher intensity value than their neighbouring pixels. On a first step bright regions in the image are extracted using a binary adaptive threshold set to the 90% of the image mean

J. Nuevo, I. Parra and L.M. Bergasa are with the Department of Electronics, University of Alcalá, Alcalá de Henares, Madrid, Spain. J. Sjöberg is with Signals and Systems at Chalmers University of Technology, Göteborg, Sweden. [jnuevo,parra,bergasa@depeca.uah.es](mailto:jnuevo,parra,bergasa@depeca.uah.es), [jonas.sjoberg@chalmers.se](mailto:jonas.sjoberg@chalmers.se).

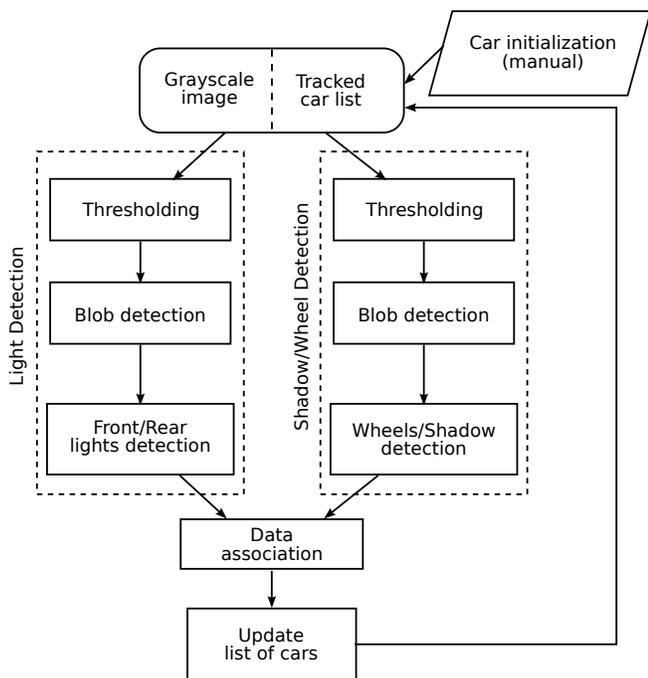


Fig. 1. System flow diagram.

intensity value. Blobs of the brighter regions of the image are then detected and their contour extracted. For every contour, circularity, perimeter and area are stored in a list along with the wheels/shadow information.

### B. Wheels/shadow detection system

A common feature for all the cars in the video sequences is that they show a dark region under the car lights corresponding to the wheels or the shadow under the car. A similar approach as in the previous section was followed, but the adaptive threshold is set to the 5% of the average intensity level for the image in order to detect the darker areas surrounding the car lights.

Using geometry clues based on the previously estimated position of the lights in the image, a new Region of Interest (ROI) is defined for the dark contours detected. Every dark region inside this ROI is labeled as wheel/shadow contour and stored. This contours are merged using the prior knowledge of the car geometry to get the main shadow under the lights, which should show a similar size as both lights contour as shown in Fig. 2. Once we have detected the main shadow, the front and rear wheels are searched for as dark contours in known positions with respect to the lights and main shadow.

## III. DISTANCE AND ORIENTATION ESTIMATION USING GEOMETRICAL INFORMATION

Estimating distances from images obtained with one camera is difficult. Depth information is not readily available as is the case of stereo camera pairs, and must then be estimated based on a set of assumptions on the scene and the sizes of the vehicles. This section describes several sets of assumptions, and discusses their strengths and weaknesses.

### A. Flat-Earth method

This methods assumes that the ground is locally flat. We follow a similar approach to that in [8], where the distance between the egovehicle and the detected vehicles is computed using monocular vision. The perspective camera model [11] used can be seen in Fig. 3. The origin of the vehicle coordinate system is located at the central point of the camera lens. The  $x$  and  $y$  coordinates of the vehicle coordinate system are parallel to the image plane and the  $Z$  axis is perpendicular to the plane formed by the  $X$  and  $Y$  axis. A vehicle at a look-ahead distance  $z$  from the camera will be projected into the image plane at vertical and horizontal coordinates  $(u, v)$  respectively. Vertical and horizontal mapping models will be carried out. The vertical model considers flat road and uses the following parameters:

- $z$ : Look-ahead distance for planar ground (mm)
- $h_{cam}$ : Elevation of the camera above the ground (mm)
- $h_{light}$ : Elevation of the vehicle rear lights (mm)
- $h'_{light}$ : Elevation of the vehicle front lights (mm)
- $\theta_{cam}$ : Camera pitch angle relative (rad)
- $\theta_z$ : Incident angle of the precedent vehicle light in the camera relative to the pitch axis (rad)
- $(u, v)$ : Horizontal and vertical image coordinates (pixels)
- $(u_0, v_0)$ : Optical center vertical coordinate (pixels)
- $f$ : Focal length (pixels)

Several of these parameters are assumed to be known and fixed. Elevation of the vehicle lights is supposed to be constant and equal for all car models, as is the elevation and pitch angle of the camera. The longitudinal axis of the detected vehicle and egovehicle are assumed to be always tangential to the road, which also implies the absence of speed bumps or other irregularities on the road. An estimation of the distance is calculated as follows: to each scan line at  $v$ , there corresponds a pitch angle relative to the local



Fig. 2. Detail of the merging of the dark regions under the car. Detected lights are enclosed by the white rectangle.

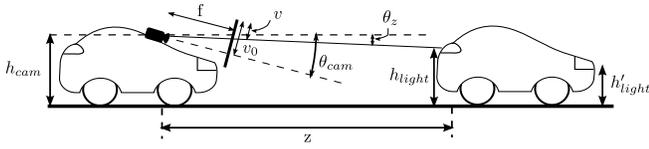


Fig. 3. Car distance estimation using the car position of car lights in the image, assuming flat Earth.

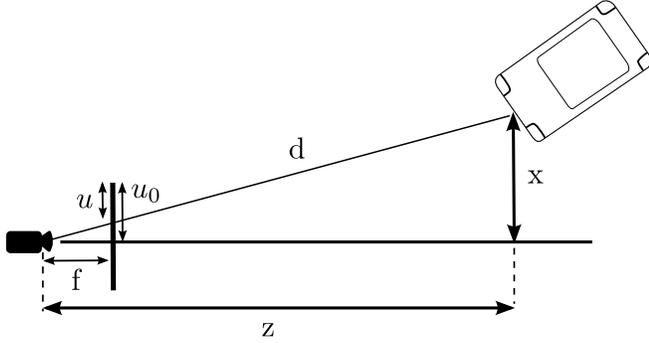


Fig. 4. Geometry of the car distance correction considering horizontal position of the vehicle.

tangential plane of  $\theta_z$ . The angle  $\beta$  is given by

$$\beta = \theta_{cam} - \theta_z, \quad \beta = \arctan\left(\frac{v_0 - v}{f}\right) \quad (1)$$

with

$$\tan(\theta_z) = \frac{h_{cam} - h_{light}}{z}. \quad (2)$$

From this, the planar look-ahead distance corresponding to  $v$  is obtained as

$$z = \frac{h_{cam} - h_{light}}{\tan(\theta_{cam} - \arctan\left(\frac{v_0 - v}{f}\right))}. \quad (3)$$

A distance estimate  $d$  is obtained by introducing the horizontal coordinate of the image ( $u$ ), as shown in Fig. 4. In this formulation,

$$x = \frac{z \cdot (u - u_0)}{f} \quad (4)$$

where  $u$  is the horizontal image coordinate,  $u_0$  is the optical center horizontal coordinate and  $f$  is the optical length. Finally the distance  $d$  to the car is computed as an Euclidean distance  $d = \sqrt{x^2 + z^2}$ .

The flat-Earth assumption normally holds for cars that are closer than 10-15 meters to the egovehicle, but fails at longer distances or when the road is steep. The estimation error grows linearly with errors in the values of  $h_{light}$  and  $h_{cam}$ , but it is non-linear with the non-flatness of the ground.

### B. Shadow-lights method

The flat-Earth assumption can be avoided if a more adequate reference to measure the lights elevation is chosen. The points where the wheels contact the ground could be used. The wheels are visible in most situations (see Fig. 2), but depending on the car model they are partially occluded

or can not be distinguished from the shadow under the car. In the latter case the shadow can also be used as reference. As can be seen in Fig. 5, only the distance  $\Delta v$  is needed, and the distance  $z$  can be obtained from it.

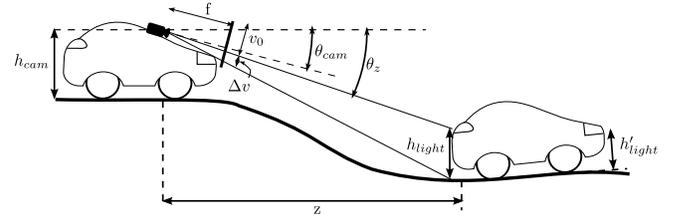


Fig. 5. Car orientation estimation using the position of lights and shadow.

This method only requires a correct camera calibration and proper detection of the lights and shadow/wheels of the vehicle, and its only assumption is that the elevation  $h_{lights}$  (or  $h'_{lights}$ ) of the lights is constant and equal for all vehicles. The shadow is dependent on the incident light and if it is noticeably oblique, the detection of it could be imprecise.

### C. Frontal-facing car method

Shadow and wheels are more difficult to locate properly than car lights. Assuming that the car is facing the camera, and that the distance between the lights is fixed and the same for all vehicles, we can obtain a rough estimate of the distance  $d$  just from the distance  $\Delta u = u_r - u_l$  between the position of the lights in the image.

When the vehicle is not facing frontal to the camera, which will be the case in curves, the uncertainty of the orientation and distance can be solved using the method in III-E or obtained from the dynamic vehicle model in section IV.

### D. Orientation computation using lights position

The estimation of the car orientation is carried out using the same perspective camera model as above in section III-A. The projection of the car lights into the image plane  $u_{right}$  and  $u_{left}$  will determine the car orientation given that we know their distance to the car ( $z_{right}$  and  $z_{left}$ ) and that the distance between the lights is similar for most vehicles and known. The geometry of the problem can be seen in Fig. 6.

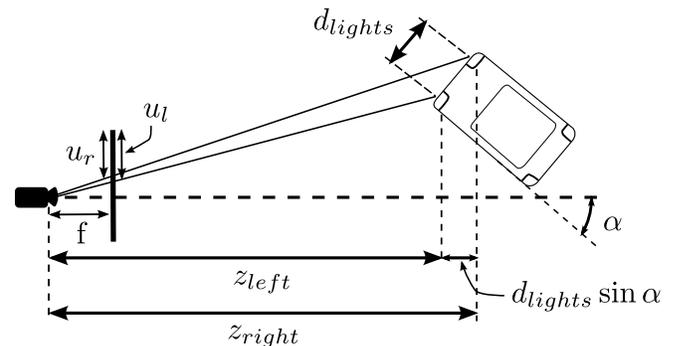


Fig. 6. Car orientation estimation using the lights position.

The estimated angle  $\alpha$  for the car can be obtained from

$$\begin{cases} z_{left} + d_{lights} \cdot \sin \alpha = z_{right} \\ x_{left} + d_{lights} \cdot \cos \alpha = x_{right} \end{cases} \quad (5)$$

and solving for  $\alpha$

$$\begin{cases} \alpha = \arcsin\left(\frac{z_{right} - z_{left}}{d_{lights}}\right) \\ \alpha = \arccos\left(\frac{x_{right} - x_{left}}{d_{lights}}\right) \end{cases} \quad (6)$$

Due to the limited resolution of the image when the vehicle is far away the distance between  $z_{right}$  and  $z_{left}$  is very small, leading to underestimated  $\alpha$  angles.

### E. Orientation computation using car length

The problem of the method above is that the actual distance between the lights is small compared to the distance to the car. In Eq. 6, the denominator  $d_{lights}$  can be very small. Thus, the estimation of the angle  $\alpha$  is very sensitive to measurement errors of the values of  $d_{lights}$ . The length  $L$  of the car is usually more than twice as the width, and thus is a measure providing better resolution. The geometry of the problem can be seen in Fig. 7. The length of the car can be estimated by detecting the wheels or the shadow under the car.

The estimated angle  $\alpha$  for the car can be obtained solving

$$\frac{u_0 - u_s}{f} = \frac{x_{left} - L \sin \alpha}{z_{left} + L \cos \alpha} \quad (7)$$

where  $u_0$  is the central point of the focal plane and  $u_s$  is the detected most distant point of the shadow. Solving for  $\sin \alpha$  the following equation can be obtained

$$\sin^2 \alpha - 2 \cdot \frac{K_1 \cdot f}{K_2 \cdot u_s} \cdot \sin \alpha + \frac{K_1^2 - 1}{K_2} = 0 \quad (8)$$

with

$$K_1 = \frac{f \cdot x_{left} - u \cdot z_{left}}{L \cdot u_s}, \quad K_2 = \frac{f^2 + u_s^2}{u_s^2}. \quad (9)$$

As above, the shape of the shadow is subject to the position of the sun and weather conditions, and the detection error can be considerable. When the vehicle is far away, poor resolution may lead to the detected  $L \cos \alpha = z_{shadow} - z_{left}$  being bigger than the value of  $L$  itself, which is an impossible situation.

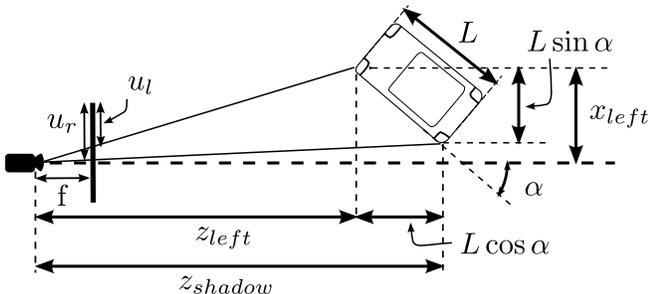


Fig. 7. Car orientation estimation using car length.

## IV. VEHICLE MODEL AND TRACKING

A simple constant velocity model is used to characterize the movement of the vehicles. Position and speed are relative to the egovehicle. For a first approximation to the problem of vehicle tracking, a Kalman Filter (KF) is used. Considering our simplified model and an urban scenario as the one in the test sequences, with vehicles driving at low speeds, a KF is enough to track most vehicles.

Fig. 8(a) shows several measurements of lights obtained after processing the image. A rectangle is drawn around detected lights. In order to remove spurious or unrelated measurements, a gate is placed around the actual or expected position of the lights. Measurements outside the gate are unlikely to be produced by the car being tracked, and are not considered. The same process is applied for shadows. Fig. 8(b) shows an example of a gate drawn in an image. Measurement gate for lights is drawn with dashed line, and gate for shadow is drawn with solid line.



(a) Detected lights in a frame (b) Gating of measurements

Fig. 8. Detection of lights and gating of measurements.

Measurements in the gate are associated with the corresponding lights using nearest neighbor (NN). If more than one light is found within the gate, an estimation of distance is computed for all possible pairs of lights. The pair that yields the closest estimation to the expected value is chosen. Using pairs instead of single lights solves part of the association problem, and reduces the probability of choosing an unlikely pair of lights. Global nearest neighbor (GNN) or better techniques would have to be used otherwise. If no pair is found to be valid, estimation from single lights are considered. As a last resort, when no detected light is found to be valid, an estimate of the distance is obtained from the position of the shadow of the car, if available, using the *flat-Earth* assumption. Most frequently only one measurement is available for the shadow, and thus NN obtains good results.

Two variations of vehicle tracking, depending on the variables used in the state vector of the KF, have been tested.

### A. Tracking in road space

This type of tracking uses the actual position of the vehicle in road space, computed from the pixel values using one of the techniques above, with state vector  $\mathbf{s} = (z, x, \alpha, \dot{z}, \dot{x}, \dot{\alpha})$ . The measurement gate is placed around the expected position of the vehicle in  $(z, x)$ , and the selected values used as inputs to the KF.

These computations carry the errors described in the previous section. If they are too different from those of previous frames, it is possible that the gate will be placed

far from the current measurements, and tracking will be lost for that frame.

We have tested two different gate sizes: one with a fixed size in meters around the expected position of the vehicle, and an ellipsoidal gate, whose sizes depend on the distance to the car and the variance of the error, as given by the KF.

### B. Tracking in image space

This type of tracking uses the position of the vehicles in the images. The measurement gate is placed around the expected position of the lights and shadow in the image. If both lights have been detected in the previous frame, the size of the gate is a function of the distance in pixels between the lights. Otherwise, the gate takes a fixed size.

This kind of *indirect tracking* is simpler because it only performs the road space to image space transformation once, which in addition makes it more robust if the required assumptions do not hold in a frame. On the other hand, road space tracking provides actual filtered estimations of the pose  $(z, d, \alpha)$  of the tracked vehicle, while image space tracking does not.

## V. TESTS AND RESULTS

Data used to test our system was collected by FOT-researchers at the Vehicle and Traffic Safety Centre (SAFER) at Chalmers, for evaluation purposes. The videos were recorded in an urban environment, at a frame rate of  $\sim 10$  frames per second. Images are  $640 \times 480$  pixels.

The data collected does not include readings from LIDAR scanners or any other means of obtaining ground-truth values of the distance to the cars and their speed, which makes performance evaluation subjective. Five vehicles appear clearly in front of the egovehicle. Seven cars drive in the other direction on the same street, plus another four that are only visible for a few frames. Several other vehicles appear in the images, moving or parked on the streets, but are far from the egovehicle or are occluded.

The car that stays the longest within view does so for about 400 frames. A few of these frames are shown in Fig. 9. The vehicle is first detected at around 40 meters, and it gradually gets as close as 10 meters before disappearing from the images. Fig. 9(i) shows the distance estimation for the sets of assumptions using an ellipsoidal gate, and an indirect gate for flat-Earth and frontal-facing. In all cases the distance estimation is quite noisy, due to both the low frame rate of the video, and errors inherent to assumptions made.

Flat-Earth assumption works well for most of the sequence, with the exception of around frame 300. Fig. 9(c) and Fig. 9(d) show a change in the slope of the street. Indirect tracking estimation then results in an over estimation of the distance to the vehicle, while the direct estimation misses a few frames, as the measurements do not fall in the gate.

Frontal-facing estimation, on the other hand, works better when indirect tracking is used. The problem of this assumption is its dependence of the direction of the vehicle, and a quick, slight turn of the vehicle being tracked would result in a measurement that will fall outside the gate. The sensitivity

of the estimation increases with the distance, due to the resolution of the camera. When the vehicle is frontal to the egovehicle, the estimation is slightly shorter than the other methods.

Finally, shadow-light estimation produces similar results to flat-Earth, with slightly larger values when the shadow extends from the actual size of the vehicle. This situation does not take place in our test data, but could turn problematic when the sun is low on the horizon, producing long shadows.

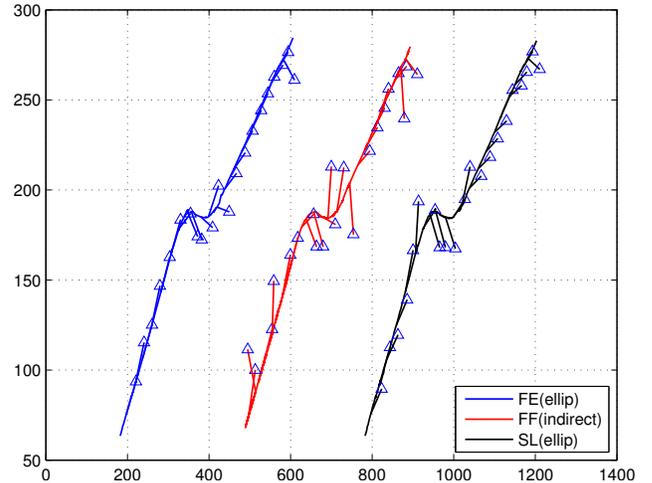


Fig. 10. Trajectory in meters and angle estimation. Lines indicated with a triangle indicate the estimated angle. (Estimations for frontal-facing and shadow-lights are offset for clarity.)

Fig. 10 shows the trajectory of the vehicle for 3 sets of assumptions described in § III, and the estimation of the angle at some points of the path. As expected, flat-Earth provides the smoothest estimation, while frontal-facing estimates are the noisiest.

From the results, it is clear that all methods using tracking in *road space* obtain similar results. At longer distances (first frames in Fig. 9(i)), the frontal-facing assumption has a higher error due to the poor estimation of the angle of the vehicle, which is used to correct the estimation. Flat-Earth and shadow-lights also obtain a more consistent estimate of the value of the angle, although the estimation of the trajectory is virtually the same for all methods, as shown in Fig. 10. Tracking in *road space* provides a smoother estimation of the distances, and it is more robust to outlier data than tracking in *image space*, as it is the case around frame #300. Measurements in *image space* need additional checks to ensure they correspond to physically valid positions of the vehicles.

## VI. CONCLUSIONS AND FUTURE WORK

Estimation of distances with a single camera presents a series of problems because no 3D data is available. This implies that some assumptions about the elements in the scene have to be made. In our case, these include the dimensions of the vehicles to be tracked, and the geometry

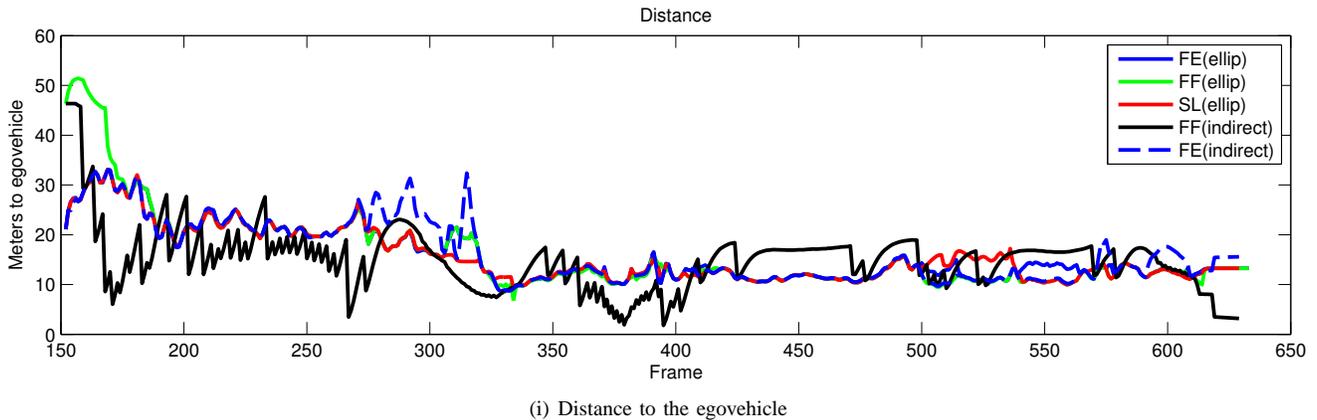
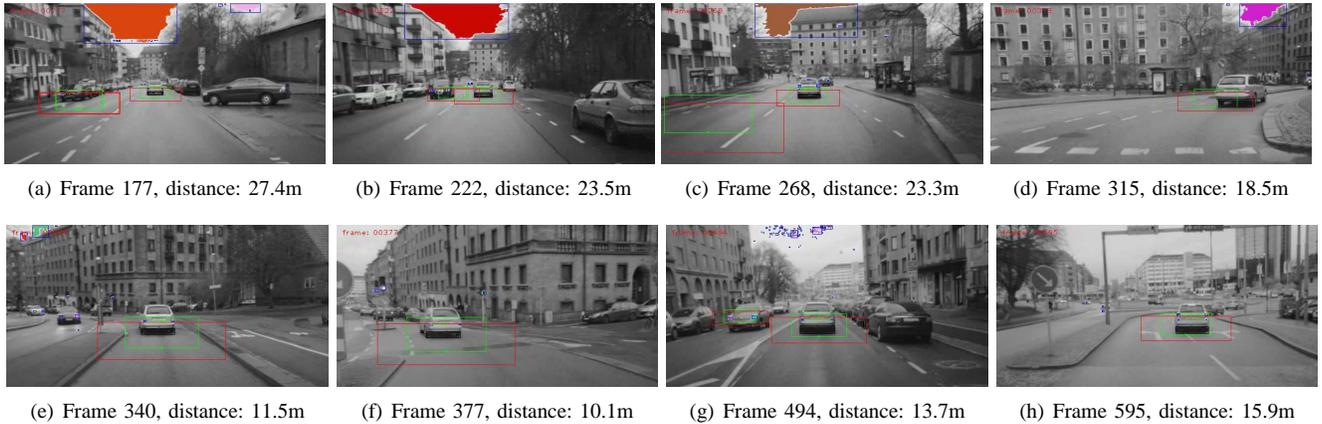


Fig. 9. Tracking of a car and distance to ego-vehicle, for several assumptions (FE = Flat-Earth, FF = Front Facing, SL = Shadow-light). Image samples 9(a)-9(h) correspond to results assuming flat Earth, with tracking in road space.

of the road where they are. We have tested several sets of assumptions on a test video sequence, and found that assuming a flat Earth results in the most stable estimation, although tracking can be lost in presence of pitch and slope changes that make the assumption fail. Performance of the shadow-light assumption is similar. Due to the low frame rate of the video, vehicles can only be tracked properly for speeds lower than 50 km/h (relative to the ego-vehicle).

In the future, we plan to combine the images with other sensor data, like RADAR or LIDAR. Automatic detection of the vehicles will be performed from RADAR data. We will implement pitch-correction techniques to reduce the sensitivity of the flat-Earth assumption to changes in pitch. We will study alternative initialization techniques that rely only on computer vision, and evaluate the convenience of night-only initialization methods, such as [9] for daytime operation. Further tests will be carried out to assess the performance of the system.

## VII. ACKNOWLEDGMENTS

This work has been supported by the Spanish Ministry of Education and Science by means of Research Grant DPI2005-07980-C03-02 and the Regional Government of Madrid by means of Research Grant CCG06-UAH/DPI-0411. Financial support through SAFER, Vehicle and Traffic Safety Centre at Chalmers, is gratefully acknowledged.

## REFERENCES

- [1] M. Betke, E. Haritaoglu, and L. Davis, "Real-time multiple vehicle detection and tracking from a moving vehicle," *Machine Vision and Applications*, vol. 12, no. 2, pp. 69–83, 2000.
- [2] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1064–1072, 2004.
- [3] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, 2006.
- [4] C. Premevida, G. Monteiro, U. Nunes, and P. Peixoto, "A Lidar and Vision-based Approach for Pedestrian and Vehicle Detection and Tracking," in *IEEE ITSC*, 2007, pp. 1044–1049.
- [5] A. Barth and U. Franke, "Estimating the driving state of oncoming vehicles from a moving platform using stereo vision," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 560–571, Dec 2009.
- [6] D. Ponsa, A. Lopez, J. Serrat, F. Lumberras, and T. Graf, "Multiple vehicle 3d tracking using an unscented kalman," in *IEEE Intell. Transp. Sys. Conf., ITSC*, 13-15 2005, pp. 1108 – 1113.
- [7] D. Ponsa, A. Lopez, F. Lumberras, J. Serrat, and T. Graf, "3d vehicle sensor based on monocular vision," in *IEEE Intell. Transp. Sys. Conf., ITSC*, 13-15 2005, pp. 1096 – 1101.
- [8] P. Alcantarilla, L. Bergasa, P. Jimenez, M. Sotelo, I. Parra, D. Fernandez, and S. Mayoral, "Night time vehicle detection for driving assistance lightbeam controller," in *2008 IEEE Intelligent Vehicles Symposium*, 2008, pp. 291–296.
- [9] A. Fossati, P. Schönmann, and P. Fua, "Real-time vehicle tracking for driving assistance," *Machine Vision and Applications*, pp. 1–10, 2010.
- [10] Canada Ministry of Justice, "Motor vehicle safety regulations," <http://laws-lois.justice.gc.ca>, C.R.C., c. 1038.
- [11] E. Dickmanns and B. Myśliwetz, "Recursive 3-d road and relative ego-state recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 199–213, Feb 1992.