Integrating Recognition and Reconstruction for Cognitive Traffic Scene Analysis from a Moving Vehicle

Bastian Leibe[1], Nico Cornelis[2], Kurt Cornelis[2], and Luc Van Gool[1,2]

[1] ETH Zurich, Switzerland
{leibe,vangool}@vision.ee.ethz.ch
[2] KU Leuven, Belgium
{firstname.lastname}@esat.kuleuven.be

Abstract. This paper presents a practical system for vision-based traffic scene analysis from a moving vehicle based on a cognitive feedback loop which integrates real-time geometry estimation with appearance-based object detection. We demonstrate how those two components can benefit from each other's continuous input and how the transferred knowledge can be used to improve scene analysis. Thus, scene interpretation is not left as a matter of logical reasoning, but is instead addressed by the repeated interaction and consistency checks between different levels and modes of visual processing. As our results show, the proposed tight integration significantly increases recognition performance, as well as overall system robustness. In addition, it enables the construction of novel capabilities such as the accurate 3D estimation of object locations and orientations and their temporal integration in a world coordinate frame. The system is evaluated on a challenging real-world car detection task in an urban scenario.

1 Introduction

Our target application is the analysis of traffic scenes, especially the detection of parked and moving cars in crowded urban areas. Such an analysis has straightforward applications in automatic driver assistance systems for identifying potentially dangerous traffic situations and as a basis for higher-level assistance functions. For example, the accurate localization of parked cars may be used to direct a focus of attention to image locations at which an inadvertent child might suddenly enter the street. As most of the child's body will be occluded by other vehicles, detection is particularly difficult in those situations, and contextual priming may buy precious reaction time.

However, detection from a moving vehicle is notoriously difficult because of the combined effects of egomotion, blur, unknown scene content, significant partial occlusion, and rapidly changing lighting conditions between shadowed and brightly lit areas. In addition, geometric scene context, which has been routinely used for surveillance and tracking applications from static cameras (e.g. [7, 12]), is far harder to obtain in a moving vehicle, where continuous recalibration is needed due to the changing environment and vehicle pitch during acceleration and deceleration. While considerable progress has been made in relatively clean highway situations (e.g. [2, 1]), the reliable detection of vehicles and pedestrians in crowded urban areas is still an important challenge [5].

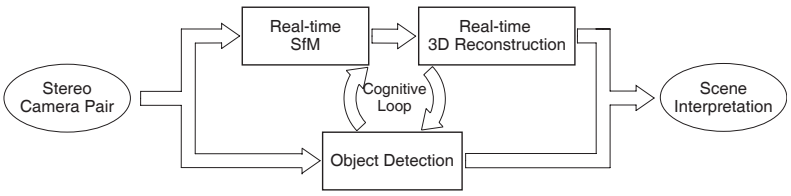


Fig. 1. Overview of our system integrating recognition and geometry estimation.

In this paper, we focus purely on vision as the most informative sensor. However, we integrate different cues and processing modalities: structure-from-motion (SfM), stereo reconstruction, and object detection. Our system is based on the idea of cognitive loops. While each of the component modules in isolation is limited, their interaction and exchange of information can compensate for the individual weaknesses and contribute to a reliable system response. Thus, the SfM and reconstruction modules collect knowledge about the scene geometry and the camera's relative pose in it. However, relying on the assumption that a dominant part of the scene change is caused by egomotion, the estimation breaks down in crowded traffic situations. By detecting other moving objects and factoring out their influence on the scene change, the recognition module helps to obtain more reliable estimates. The recognition system, on the other hand, can profit immensely from knowledge about the scene geometry by applying ground plane constraints that the SfM and reconstruction modules can deliver.

The paper is structured as follows. The following section gives an overview of the proposed system. Sections 2 and 3 then describe the two main components and their interaction in detail. Section 4 finally presents experimental results.

System Overview. Figure 1 shows a visualization of our system setup. Our input data are two video streams recorded by a calibrated stereo rig mounted on top of the test vehicle, which are annotated with GPS/INS measurements. From this data, a Structure-from-Motion (SfM) algorithm first computes a camera pose for each image. Subsequently, these poses are used to generate a compact reconstruction of the surrounding road surface and facades using a fast dense-stereo algorithm [3]. Both of those stages are highly optimized and run at about 25 fps. In parallel, an object detection module is applied to both camera images in order to detect cars in the scene. The three modules are integrated in a tight cognitive loop. For each image, the object detection module receives scene geometry information, extracted from the previous frame, from the other two modules and feeds back information about detected objects to them, which is then used for processing the next frame. Thus, the modules exchange information that helps compensate for their individual failure modes and improves overall system performance. The next sections explain the different modules in detail.

2 Real-time Geometry Estimation

In the first pathway, our system computes and permanently updates an estimate of the surrounding scene geometry. As space does not permit an in-depth discussion of well-known algorithms for Structure-from-Motion pipelines [6] and dense stereo [14], we

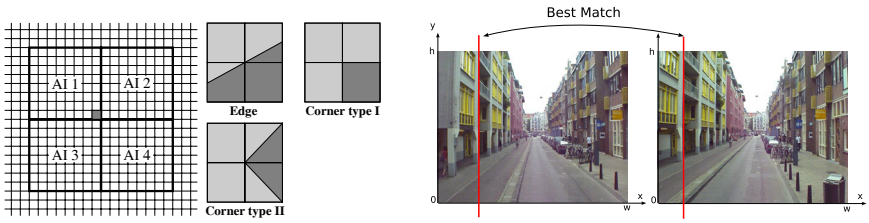


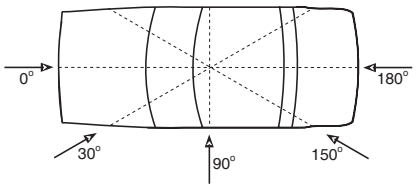
Fig. 2. (left) The fast feature measure used for SfM. (right) Rectified stereo pair.

limit the description to those changes that were made to allow for high-speed processing. Details of the described algorithms can be found in [3].

Real-time Structure-from-Motion Computation. A real-time feature matcher extracts image feature points by finding local maxima of a very simple feature measure, based on the average intensity (AI) of four sub-regions (Fig. 2(left)): $d = abs((\mathrm{AI1} + \mathrm{AI4}) - (\mathrm{AI2} + \mathrm{AI3}))$. The extracted features are matched between consecutive images based on a fast sum of absolute intensity differences and then fed into a classic SfM pipeline, which reconstructs feature tracks and refines 3D point locations by performing triangulation. Sufficient baseline between images is guaranteed by only accepting a new image when the GPS or odometry signals sufficient movement. For efficiency reasons, only the green channel of one of the cameras is processed during SfM. A bundle adjustment routine is running in parallel with the main SfM algorithm to refine camera poses and 3D feature locations for previous frames and thus reduce drift. Additional GPS and odometry information can be used to guide feature matching during fast turns, to compensate for remaining drift, and to transfer the cameras into a global world coordinate system. The drift-compensated and globally aligned cameras are then rectified so that their up-vector is parallel to the world gravity vector. This ensures that 3D lines parallel to the gravity vector are displayed as vertical lines in each stereo pair.

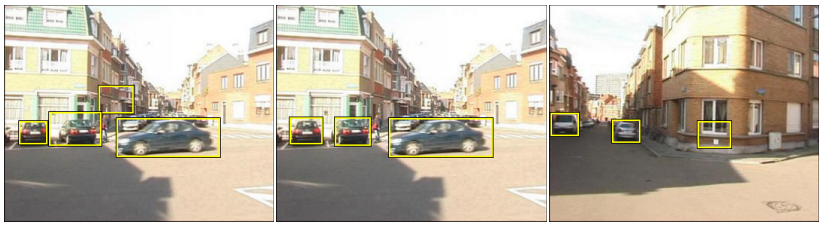
Real-time 3D Reconstruction. Next, a real-time geometry module reconstructs building facades using the (realistic) assumption that those can be modeled by ruled surfaces (i.e. surfaces made up of non-intersecting line segments) which are parallel to the gravity vector. For each rectified stereo pair, disparity values are computed for every vertical line using a single dynamic programming pass which is based on the ordering constraint and a robust line-based similarity measure (c.f. Fig.2(right)). Besides the tremendous gain in speed compared with algorithms which run dynamic programming on each horizontal scan line, the reconstruction becomes more accurate, as information over each vertical scan line can be integrated. The reconstructed volumes from all stereo pairs are then integrated over time into a topological map by a voting based carving algorithm. Finally, the road itself is reconstructed by fitting lines through the known contact points of the test vehicle's wheels with the road. This way of road reconstruction is not only faster than using dense stereo algorithms, but also more accurate since roads are often not textured enough for dense stereo.

Derivation of Geometric Constraints. For each image, the geometry module computes an estimate of the current ground plane by fitting a plane through the reconstructed road surface around the wheel contact points and extrapolating it along the current view-



	#images	mirrored
0°	117	
30°	138	X
90°	50	
150°	119	X
180°	119	

Fig. 3. (left) Visualization of the viewpoints the single-view detectors were trained on. (right) Number of training images used for each view.



(a) (b) (c)

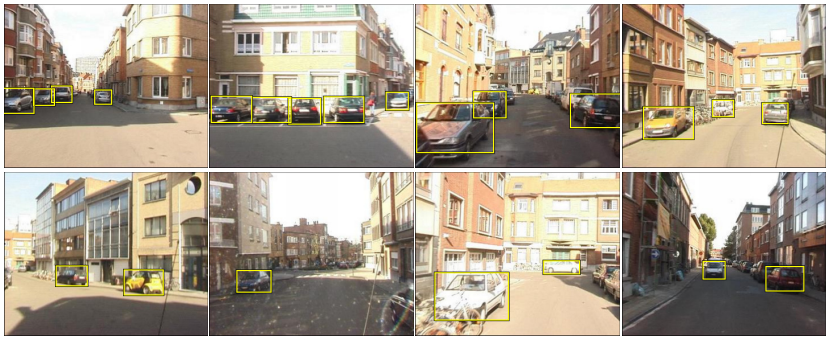
Fig. 4. Effect of scene geometry constraints: (a) object hypotheses before and (b) after ground plane constraints are enforced; (c) False positive that is filtered out by facade constraints.

ing direction. By intersecting this plane with already reconstructed building facades, we can restrict the possible space in which objects may occur. This information is then passed to the recognition module to guide and improve object detection performance.

3 Object Detection

The recognition system is based on the ISM approach [8]. A bank of 5 single-view ISM detectors is run in parallel to capture different aspects of cars (see Fig. 3 for a visualization of their distribution over viewpoints). For efficiency reasons, we make use of symmetries and run mirrored versions of the same detectors for the other semi-profile views. All detectors share the same set of initial features: *Shape Context* descriptors [11], computed at *Harris-Laplace*, *Hessian-Laplace*, and *DoG* interest regions [11, 10]. During training, extracted features are clustered into appearance codebooks, and each detector learns a dedicated spatial distribution for the codebook entries that occur in its target aspect. During recognition, features are again matched to the codebooks, and activated codebook entries cast probabilistic votes for possible object locations and scales according to their learned spatial distributions. The votes are collected in 3-dimensional Hough voting spaces, one for each detector, and maxima are found using MSME [8].

Integration of Ground Surface Constraints. Geometric scene constraints, such as the knowledge about the ground surface on which objects can move, can help detection in several important respects. First, they can restrict the search space for object hypotheses to a corridor in the $(x, y, scale)$ volume, thus allowing significant speedups and filtering out false positives. Second, they make it possible to evaluate object hypotheses under a



(a) (b) (c) (d)

Fig. 5. (top) Car detections on typical images from the city scenario. (bottom) Examples for the difficulties in this scenario: (a) motion blur, (b) lens flaring, (c) bright lighting (d) strong shadows.

size prior and "pull" them towards more likely locations. Last but not least, they allow to place object hypotheses at 3D locations, so that they can be corroborated by temporal integration. In the following, we use all three of those ideas to improve detection quality.

Given the camera calibration from SfM and a ground plane estimate from the 3D reconstruction module, we can estimate the 3D location for each object hypothesis by projecting a ray through the base point of its bounding box and intersecting it with the ground plane. If the ray passes above the horizon, we can trivially reject the hypothesis. In the other case, we can estimate its real-world size by projecting a second ray through the bounding box top point and intersecting it with a vertical plane through its 3D base. Using this information, we can formally express the likelihood for the real-world object H given image I by the following marginalization over the image-plane hypotheses h:

$$p(H|I) = \sum_h p(H|h,I)p(h|I) \sim \sum_h p(h|H)p(H)p(h|I) \tag{1}$$

where $p(H)$ expresses a prior for object sizes and distances, and $p(h|H)$ reflects the accuracy of our 3D estimation. In our case, we enforce a uniform distance prior up to a maximum depth of 70m and model the size prior by a Gaussian. The hypothesis scores are thus modulated by the degree to which they comply with scene geometry, before they are passed to the next stage (Fig. 4(a,b)).

Multi-view Integration. In order to fuse the single-view hypotheses into a consistent system response, we next apply the following multi-view integration stage. We first compute a top-down segmentation for each hypothesis h according to the method described in [8]. This yields two per-pixel probability maps $p(\mathit{figure}|h)$ and $p(\mathit{ground}|h)$ per hypothesis. With their help, we can express the hypothesis likelihood $p(h|I)$ in terms of the pixels it occupies:

$$p(h|I) = \sum_{\mathbf{p}\in I} p(h|\mathbf{p}) = \sum_{\mathbf{p}\in Seg(h)} p(\mathbf{p} = \mathit{figure}|h)p(h). \tag{2}$$

where $Seg(h)$ denotes the segmentation area of h, i.e. the pixels for which $p(\mathbf{p} = \mathit{figure}|h) > p(\mathbf{p} = \mathit{ground}|h)$. We then search for the optimal combination of hy-

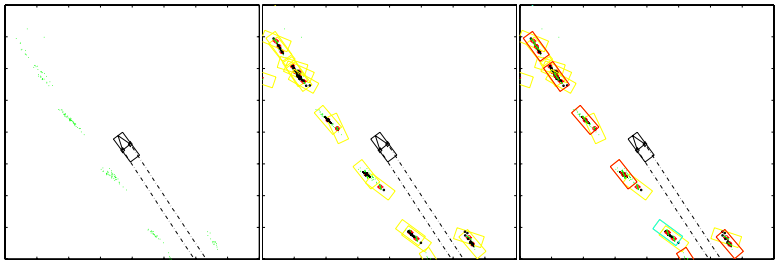


Fig. 6. Visualization of the temporal integration stage: (a) estimated 3D object locations (in green); (b) real-world object hypotheses obtained by mean-shift clustering (in yellow); (c) final hypotheses selected by the QBOP (in red).

potheses that best explains the image content under the constraint that each pixel can be assigned to at most one hypothesis. This is achieved by solving the following Quadratic Boolean Optimization Problem (QBOP):

$$\max_m m^T Q m = m^T \begin{bmatrix} q_{11} & \cdots & q_{1M} \\ \vdots & \ddots & \vdots \\ q_{M1} & \cdots & q_{MM} \end{bmatrix} m \tag{3}$$

where $m = (m_1, m_2, \ldots, m_M)$ is a vector of indicator variables, such that $m_i = 1$ if hypothesis h_i is accepted and 0 otherwise. Q is an interaction matrix whose diagonal elements q_{ii} express the merits of each individual hypothesis, while the off-diagonal elements q_{ij} express the cost of their overlap. In theory, we could directly use the hypothesis likelihood to define the merit. However, since we are dealing with incomplete information from sparsely sampled interest regions, we have to add a regularization term incorporating the number of pixels N in the figure-ground segmentation, as well as a normalization factor $A_{\sigma,v}(h)$, expressing the *expected area* of a hypothesis at its detected scale and aspect. The merit terms thus becomes

$$q_{ii} = -\kappa_1 + \frac{p(h_i|H_i)p(H_i)}{A_{\sigma,v}(h_i)} \left((1-\kappa_2)N + \kappa_2 \sum_{\mathbf{p} \in Seg(h)} p(\mathbf{p} = fig.|h_i) \right). \tag{4}$$

For the interaction terms, we measure the hypothesis overlap in the image and subtract the contribution of the overlapping area from the hypothesis $h^* \in \{h_i, h_j\}$ that is farther away from the camera.

$$q_{ij} = -\frac{1}{2} \frac{p(h^*|H^*)p(H^*)}{A_{\sigma,v}(h^*)} \sum_{\mathbf{p} \in Seg(h_i) \cap Seg(h_j)} ((1-\kappa_2) + \kappa_2 p(\mathbf{p} = fig.|h^*)) \tag{5}$$

This formulation allows to select the best global interpretation for each image from the output of the different single-view detectors. Since typically only a subset of hypotheses produces overlaps, it is generally sufficient to compute a fast greedy approximation to the optimal solution. Examples for the resulting detections are shown in Figure 5.

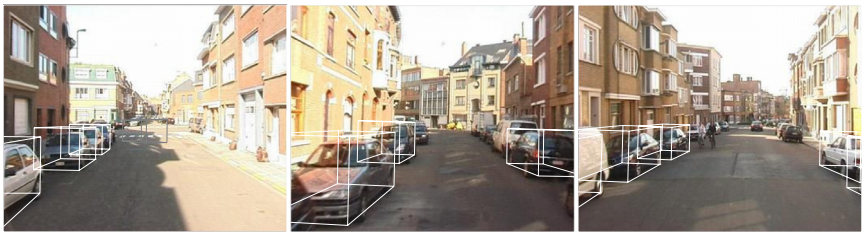


Fig. 7. Online 3D car location estimates (using only information from previous frames).

Integration of Facade Constraints. Using the information from 3D reconstruction, we add another step to check if hypothesized 3D object locations lie behind reconstructed facades (c.f. Fig. 4(c)). As this information will typically only be available after a certain time delay (i.e. when our system has collected sufficient information about the facade), this filter is applied as part of the following temporal integration stage.

Temporal Integration. The above stages are applied to both camera images simultaneously. The result is a set of 3D object hypotheses for each frame, registered in a world coordinate system. Each hypothesis comes with its 3D location, a 3D orientation vector inferred from the selected viewpoint, and an associated confidence score. Since each individual measurement may be subject to error, we improve the accuracy of the estimation process by integrating the detections over time.

Figure 6 shows a visualization of the integration procedure. We first cluster consistent hypotheses by starting a mean-shift search with adaptive covariance matrix from each new data point H and keeping all distinct convergence points $\mathcal{H}$ (Fig. 6(b)). We then select the set of hypothesis clusters that best explains our observations by again solving a QBOP, only this time in the 3D world space:

$$\tilde{q}_{ii} = -\tilde{\kappa}_1 + \sum_{H \in \mathcal{H}_i} e^{-(t-t_i)/\tau}((1-\tilde{\kappa}_2) + \tilde{\kappa}_2\, p(H|\mathcal{H}_i)p(H|I))\,. \tag{6}$$

$$\tilde{q}_{ij} = -\frac{1}{2} \sum_{H \in \mathcal{H}_i \cap \mathcal{H}_j} e^{-(t-t^*)/\tau}((1-\tilde{\kappa}_2) + \tilde{\kappa}_2\, p(H|\mathcal{H}^*)p(H|I) + \tilde{\kappa}_3\, O(\mathcal{H}_i, \mathcal{H}_j)) \tag{7}$$

where $p(H|\mathcal{H}_i)$ is obtained by evaluating the location of H under the covariance of $\mathcal{H}_i$; $\mathcal{H}^*$ denotes the weaker of the two hypothesis clusters; and $O(\mathcal{H}_i, \mathcal{H}_j)$ measures the overlap between their real-world bounding boxes, assuming average car dimensions. This last term is the main conceptual difference to the previous formulation in eqs. (4) and (5). It introduces a strong penalty term for hypothesis pairs that overlap physically. In order to compensate for false positives and moving objects, each measurement is additionally subjected to a small temporal decay with time constant τ. The results of this procedure are displayed in Fig. 6(c).

Estimating Car Orientations. Finally, we refine our orientation estimates for the verified car hypotheses using the following two observations. First, the main estimation errors are made both along a car's main axis and along our viewing direction. Since the latter moves when passing a parked car, the cluster's main axis is slightly tilted towards our egomotion vector (c.f. Fig.6(a)). Second, the semi-profile detectors, despite being

trained only for 30° views, respond to a relatively large range of viewpoints. As a result, the orientation estimates from those detectors are usually tilted slightly away from our direction of movement. In practice, the two effects compensate for each other, so that a reasonably accurate estimate of a car's main axis can be obtained by averaging the two directions. Some typical examples of the resulting 3D estimates are shown in Fig. 7.

Feedback into SfM and Reconstruction Modules. The results of the previous stages have demonstrated that object detection can benefit considerably from knowledge about the scene geometry, delivered by the SfM and 3D reconstruction modules. However, those modules can also benefit from the results of object detection.

As discussed above, the SfM module relies on the assumption that a dominant part of the scene change is caused by egomotion. As a result, moving and/or shiny cars degrade the accuracy of the estimated camera positions. Although RANSAC outlier rejection [4] can to a certain degree compensate for this, there are many natural car motions that can be misinterpreted as static because of ambiguities in their image projection. E.g. following a car in the same lane at more or less the same speed on a straight sketch makes it clearly indistinguishable from a static object at infinity. Also, a car approaching on the other lane with a speed correlated to ours is indistinguishable from a static car parked somewhere in the middle of both lanes. Similarly, the fast 3D reconstruction relies on the assumption that the scene can be represented by ruled surfaces. Obviously, this is no longer the case when cars are parked in front of the facades. As a result, the cars introduce erroneous measurements into the dense stereo calculations which may influence the accuracy of the resulting scene geometry estimate (and thus of the ground plane estimate that will be provided to the detection module for the next frame).

The object detection module therefore completes the cognitive loop by feeding back information about its detections into the SfM and Reconstruction modules. By informing the SfM algorithm where cars can be expected, features will not be instantiated or tracked in those areas, thereby avoiding erroneous measurements which would result from tracking non-stationary points on moving and shiny cars. Similarly, object detection helps the reconstruction module by segmenting out all detected cars, so that the dense stereo reconstruction can focus on image areas that fulfill the ruled surface assumption. This continuous feedback of information is a crucial point for guaranteeing system reliability in complex real-world scenarios.

4 Experimental Results

In order to evaluate our method, we applied it to a test sequence, recorded by a camera vehicle over a distance of approximately 500m. The stereo input streams were captured at the relatively low resolution of 380×288 pixels due to restrictions of the recording setup. Altogether, the data set comprises 1175 image pairs, which are processed at their original resolution by the SfM and reconstruction modules and bilinearly interpolated to twice that size for object detection (similar to [10]). The 5 single-view detectors were trained on images taken from the LabelMe database [13], for which viewpoint annotations and rough polygon outlines were already available (c.f. Fig.3). In all experiments, we set $\kappa_2 = 0.95$, $\tilde{\kappa}_2 = 0.5$, and plot performance curves over the value of κ_1.

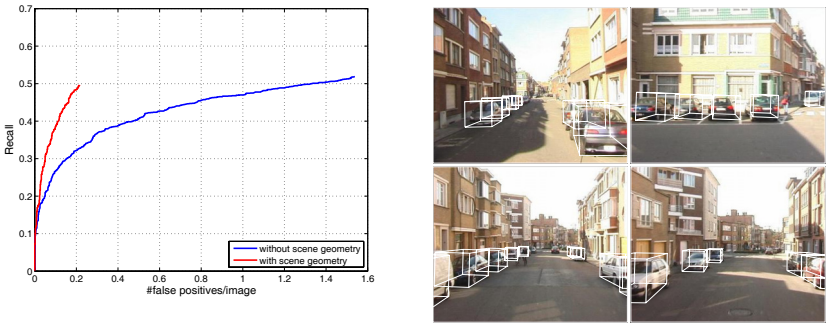


Fig. 8. (left) Comparison of the detection performance with and without scene geometry constraints. (right) 3D car location estimates using also information from future frames.

For a quantitative estimate of the performance improvement brought about by the inclusion of geometry constraints, we annotated the first 600 frames of the video sequence by marking all cars that were within a distance of 50m and visible by at least 40-50%. It is important to note that this includes many difficult cases with partial visibility, so it is unrealistic to expect perfect detection in every frame. We then evaluated the detection performance with and without ground plane constraints using the evaluation criterion from [9]. The results of this experiment are shown in Figure 8. As can be seen from the plot, detection reaches a level of about 50% recall in both cases. While the original recognition system yields 1.3 false positives per image at this level of recall, the inclusion of ground plane constraints significantly reduces the false positive rate to one every five images at 50% recall, or even one every ten images at 40% recall.

Counted over its full length, the sequence contains 77 (sufficiently visible) static and 4 moving cars, all but 6 of which are correctly detected in at least one frame. The online estimation of their 3D locations and orientation usually converges at a distance between 15 and 30m and leads to a correct estimate for 68 of the static cars; for 5 more, the obtained estimate would also have been correct, but does not reach a sufficiently high confidence level to be accepted. The estimates can further be improved by backpropagating also information from future frames (Fig. 8(right)). The SfM and reconstruction modules also profit from the feedback from object detection in terms of increased robustness. However, the exact benefit is hard to quantify, since no ground truth was available for the 3D measurements.

5 Discussion and Conclusion

In this paper, we have presented a system for cognitive traffic scene analysis that closely integrates structure-from-motion, 3D reconstruction, and object detection into a cognitive loop. At first view, it might seem unintuitive to incur the overhead of executing all three of those components in parallel, just to improve recognition performance. However, rather the opposite is the case: each individual task becomes considerably easier by its integration in the cognitive loop and the continuous feedback from the other

modules. As we have shown in this paper, the close interaction between the different modules increases both the recognition and 3D estimation performance, as well as the robustness of the entire system. In addition, our highly efficient implementation of the SfM and reconstruction modules allows them to run at video frame rate, so that their inclusion entails no additional delay. Although our current implementation of the object detector is not optimized for real-time processing yet, its individual stages are sufficiently simple, so that a time-efficient implementation is well possible.

In future work, we will aim to improve the representation of moving cars by adding a dedicated motion model. Secondly, we plan to extend recognition to other traffic participants, such as pedestrians and bicyclists, which was hitherto hindered by the poor resolution of our input video streams. Inferring a selective focus of attention from the detected car locations will help overcome this problem. Last but not least, we will optimize the implementation of our object detector for inclusion into a real-time application.

Acknowledgments. This work is funded, in part, by the EU project DIRAC (IST-2005-27787). We also wish to acknowledge the support of the K.U.Leuven Research Fund's GOA project MARVEL, Wim Moreau for the construction of the stereo rig, and TeleAtlas for providing additional video data to test on.

References

1. L. Andreone, P.C. Antonello, M. Bertozzi, A. Broggi, A. Fascioli, and D. Ranzato. Vehicle detection and localization in infra-red images. In *Intel. Vehicles Symp.'02*, 2002.
2. M. Betke, E. Haritaoglu, and L.S. Davis. Real-time multiple vehicle tracking from a moving vehicle. *MVA*, 12(2):69–83, 2000.
3. N. Cornelis, K. Cornelis, and L. Van Gool. Fast compact city modeling for navigation previsualization. In *CVPR'06*, 2006.
4. M. Fischler and R. Bolles. Random sampling consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Comm. ACM*, 24:381–395, 1981.
5. D. Gavrila and V. Philomin. Real-time object detection for smart vehicles. In *ICCV'99*, pages 87–93, 1999.
6. R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2000.
7. D. Koller, K. Daniilidis, and H.-H. Nagel. Model-based object tracking in monocular image sequences of road traffic scenes. *IJCV*, 10(3):257–281, 1993.
8. B. Leibe and B. Schiele. Scale invariant object categorization using a scale-adaptive mean-shift search. In *DAGM'04*, Springer LNCS, Vol. 3175, pages 145–153, 2004.
9. B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR'05*, 2005.
10. D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2), 2004.
11. K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 27(10), 2005.
12. A. Mittal and L.S. Davis. M2 tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 51(3):183–203, 2003.
13. B.C. Russell, A. Torralba, K.P. Murphy, and W.T. Freeman. Labelme: a database and web-based tool for image annotation. Technical Report AIM-2005-025, MIT AI Lab, 2005.
14. D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002.