

Mid-level Segmentation and Segment Tracking for Long-Range Stereo Analysis

Simon Hermann^{1,*}, Anko Börner², and Reinhard Klette¹

¹ The *.enpeda.* Project, Department of Computer Science
The University of Auckland, New Zealand

² DLR (German Aerospace Center), Berlin-Adlershof, Germany

Abstract. This paper presents a novel way of combining dense stereo and motion analysis for the purpose of mid-level scene segmentation and object tracking. The input is video data that addresses long-range stereo analysis, as typical when recording traffic scenes from a mobile platform. The task is to identify shapes of traffic-relevant objects without aiming at object classification at the considered stage. We analyse disparity dynamics in recorded scenes for solving this task. Statistical shape models are generated over subsequent frames. Shape correspondences are established by using a similarity measure based on set theory. The motion of detected shapes (frame to frame) is compensated by using a dense motion field as produced by a real-time optical flow algorithm. Experimental results show the quality of the proposed method which is fairly simple to implement.

1 Introduction

The classification of traffic-relevant objects (e.g. vehicles, bicyclists, pedestrians, or traffic signs) is a common goal in vision-based *driver assistance systems* (DAS). If an object is potentially dynamic, then it is important to understand its current state (e.g., currently static, or moving with a particular trajectory). This task requires solutions for scene segmentation, object detection and tracking, and eventually also for object classification.

Monocular or stereo vision, LIDAR, or infrared cameras are sensors considered for solving such tasks. We apply one pair of grey-level cameras for stereo vision that records the space in front of the *ego-vehicle* (i.e. the vehicle the cameras operate in). Given a 3D world coordinate system, stereo analysis provides depth information about the traffic scene in form of a cloud of 3D points. After ground manifold estimation and possibly its removal [14,22], subsequent tasks are free-space estimation [2], obstacle detection using occupancy grids [16], or object segmentation by point clustering [17]. A tracking process involves motion analysis [4,15], e.g. using particle filters [6,17].

A recent approach [20] uses disparity information for the purpose of real-time 3D scene flow computation. Object motion in the 3D world is calculated using ego-motion estimation [1]. A motion likelihood is assigned to each pixel and a binary graph-cut segmentation algorithm identifies independently moving objects [21].

* The first author thanks the German Academic Exchange Service (DAAD) for financial support.

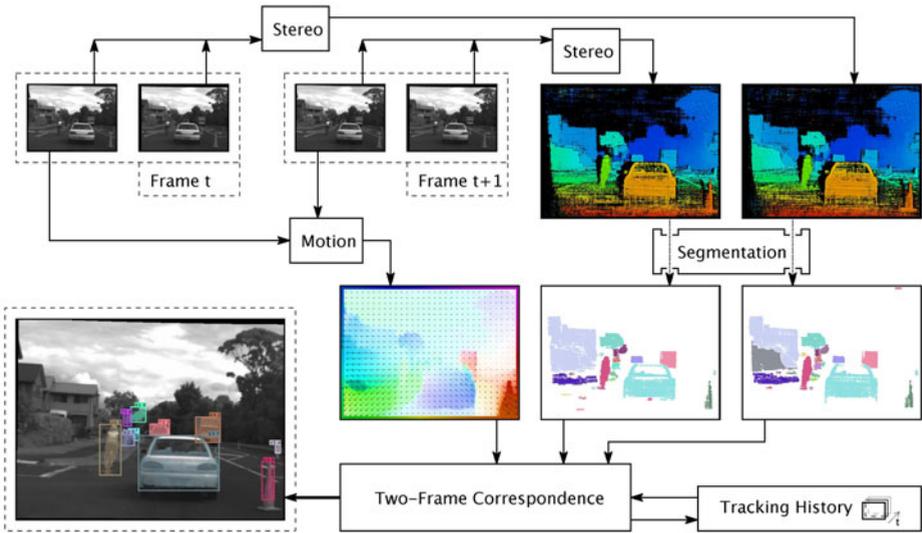


Fig. 1. Upper left: input data at t and $t + 1$. Upper right: disparity maps. Middle and lower right: optical flow map and segmentation results. Lower left: a final result showing enclosing rectangles (the *bounding boxes*) for tracked segments.

This paper proposes a new technique of mid-level scene segmentation and segment tracking. The key feature of our approach is that both, segmentation and tracking are entirely performed within the disparity data generated by dense stereo algorithms. Our approach has the advantage that DAS tasks are solved before reprojecting disparity values into the 3D world. This eliminates one source of errors, and supports the design of 2D algorithms for segmentation and tracking on disparity maps. The resulting implementation, as outlined below, is of great simplicity.

Disparity-based segmentation is already extensively used for pedestrian detection [7,27] by identifying back- and foreground areas in an image [27], before applying intensity-based segmentation.

Ground-manifold approximation [14] and scene-flow computation [20] can be solved in disparity space.

In [11] the problems that occur by tracking 3D data is highlighted and the authors argue on the benefits of performing these task in disparity space. However, they evaluate their proposed method only on synthetic data and do not provide any result image. In [24] segmentation and tracking is performed in disparity space, but with a different approach.

To the best knowledge of the authors, the proposed method constitutes a novel approach for scene segmentation and tracking in disparity space. Our implementation was evaluated on three real-world traffic sequences of 400 or 250 stereo frames. Segmentation results are also compared with available ground truth.

At this stage we purposely do not consider intensity information from the input images for segmentation. The reason is that we want to highlight the segmentation and tracking quality that is possible by exploiting disparity information only. Of course,

future work should incorporate intensity information into the proposed segmentation and tracking process.

The Proposed Method. Figure 1 sketches the workflow of our approach. The input are two stereo pairs at times t and $t + 1$. A stereo matcher computes for each stereo pair a dense disparity map. (The left image is our reference frame). Each disparity map is segmented according to a three-step segmentation process:

First, we post-process the disparity map with a mode-based filter that removes ‘noisy disparities’ (e.g. in occluded areas, or in irrelevant areas such as the sky). *Second*, a road or ground manifold is estimated and subtracted from the stereo map. This prevents objects from being connected by similar disparities at road level. *Third*, the resulting disparity map is segmented by employing a simple region-growing algorithm.

The segmentation process is described in detail in Section 3. Results during the segmentation process are shown in Fig. 1. Final patches are very similar in shape and location at t and $t+1$. A human would easily identify corresponding patches in segmentation maps f_t and f_{t+1} . A set-theoretical metric is applied for quantifying correspondences between 2D patches.

The metric encodes the ratio between overlap and total area of both patches, and thus also the similarity in 2D shape. The latter can be assumed because it is reasonable to neglect roll or tilt of objects in vision-based DAS.

We expect invariance of projected 2D object shape between subsequent frames (at least for rigid objects) recorded at 25 Hz or more, but aim at handling (minor) changes in object size due to varying distances to the ego-vehicle, and translational changes in positions due to recording highly dynamic scenes. To compensate for translational changes we calculate a dense motion field using a real-time state-of-the-art optical flow algorithm. Calculated motion vectors are used to shift pixels (of an object) from segmentation map f_t into new positions in f_{t+1} . So far we do not rescale a patch before we apply the metric because different objects cannot occupy the same image region, and a change in size did not appear to be very crucial for identifying corresponding patches.

After correspondence analysis, a temporal filter calculates the size of the current patch. This size is used for rescaling the bounding box. Correspondences between patches define the tracking history.

Outline of the Paper. Section 2 presents the used stereo and motion analysis algorithms with comments about their parametrization. Section 3 explains ground manifold estimation, the mode filter used for stereo post-processing, and the proposed segmentation for one stereo pair into objects (or patches). Section 4 establishes correspondences between patches in subsequent frames using a known shape metric, and describes the proposed tracking mechanism that works on image sequences and uses shape priors. Section 5 summarizes experiments about segmentation and tracking. Section 6 concludes.

2 Stereo and Motion Analysis

Semi-global Stereo Matching. For generating a dense disparity map D , we follow the original semi-global stereo matching algorithm [9] that minimizes the energy

$$E(D) = C(D) + S(D) \quad (1)$$

where $C(D)$ refers to the dissimilarity or data cost and $S(D)$ to the smoothness cost which incorporates a first-order data prior.

For the data term we apply the census cost function which calculates the Hamming distance of two binary signature vectors which are assigned to corresponding pixels; see the census transform in [25]. It has been shown [10] that this function is very descriptive and robust, even under strong illumination variations, which is crucial for real-world applications. In our implementation we use a 9×3 window as we work on a 32-bit machine and favour a stronger data contribution along the epipolar line.

The semi-global smoothness constraint integrates multiple optimal 1D energies along different accumulation paths using a dynamic programming approach. Since the number of paths is very limited (usually not more than eight paths) it is referred to as semi-global matching (SGM).

Our implementation uses four accumulation paths (up, down, left, right). Subpixel accuracy is obtained using equiangular interpolation [18]. To enforce uniqueness, two disparity maps are calculated and a left-right consistency check is performed. A disparity passes this test if corresponding disparities do not deviate by more than 0.7 disparity levels. The consistency check is performed to invalidate occluded areas and to remove ill-defined disparity values. Figure 2 shows a calculated disparity map. The colour code runs from “hot” (large disparities) to “cold” (small disparities).

Dense Motion Estimation. We considered methods published in [4,26] for dense motion field calculation. Both methods are based on the total variation approach by Horn-Schunck, but instead of minimizing a global energy based on the L_2 norm (as in the original work), they minimize an energy that uses the L_1 norm as data term. Although performance between both methods is rather similar, the numerical schemes of both methods are quite different. While [4] uses a fixed point iteration procedure to solve Lagrange equations, [26] employs a duality-based approach. Both algorithms are suitable for parallel implementation and can achieve real-time performance. The implementation reported in this paper follows [4]. Figure 3 shows a result for this algorithm. The rectangular frame also shows the used colour key.

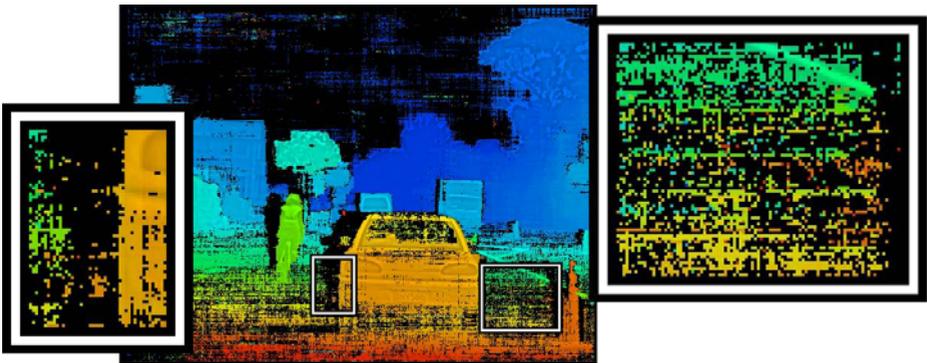


Fig. 2. Result of the SGM algorithm with the proposed settings. Enlarged windows show cases to be processed by our mode filter.

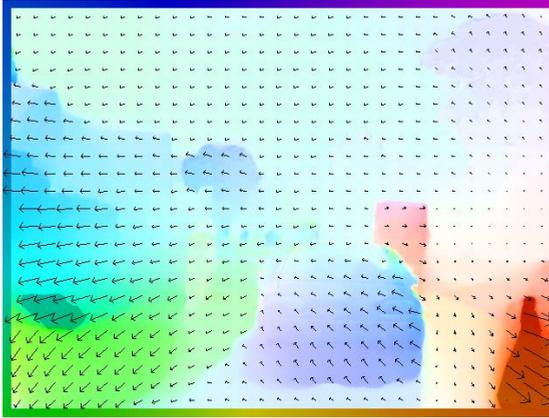


Fig. 3. Result of the used pyramidal optical flow implementation following [4]

3 Segmentation in Disparity Space

Mode Filtering. [9] proposed to apply a median filter to disparity maps before performing the consistency check. The filter removes disparity outliers and performs edge-preserving local smoothing of disparity values. As a result, more pixels pass the consistency check, thus increasing the denseness of the final disparity map.

However, for the purpose of segmentation, denseness is not the primary goal. On the contrary, enhancing occluded areas or invalidating disparities close to object boundaries helps in the stereo segmentation process. Since occluded areas have a low disparity denseness, a filter that supports the segmentation process should identify this characteristic and invalidate remaining pixels in that area. Additionally, the filter should invalidate image regions where disparity information is rather undefined. Such areas are quite often affected by noise. The enlarged window in Fig. 2, left, shows disparity values which correspond to the disparity level of the car; they are propagated into the occluded area. The enlarged window on the right shows a noisy disparity region on the road.

Intuitively, we are looking for a filter which generates both, a disparity value that is statistically dominant within a local neighbourhood, and an index that indicates the support or likelihood for this dominant value. If the corresponding disparity value of a neighbourhood is close enough to the identified dominant disparity, and the support is sufficiently high, then the disparity value should remain unchanged. Otherwise it should be discarded.

Computing the mode of a neighbourhood fulfils both needs. The mode identifies one dominant value that corresponds to the highest occurrence of a value in a domain with a fixed number of elements. The ratio of occurrences to the number of elements in the domain serves as index for the support. Since we work on sub-pixel disparity levels we need to count occurrences of disparities within intervals. The centre of the interval that contains the most disparities represents the dominant disparity value η .

Let \mathcal{N}_p be the pixel neighbourhood of a pixel p . The ordered sequence \mathcal{S} of disparity values within this neighbourhood is $\mathcal{S} = \langle s_0, s_1, \dots, s_n \rangle$. We define for $\xi \geq 0$ and each disparity s_i in \mathcal{S} a subset

$$\mathcal{I}_i = \{s : s \text{ in } \mathcal{S} \wedge s_i \leq s < s_i + \xi\} \tag{2}$$

of all disparity values in \mathcal{N}_p . Assume that the maximum

$$c_m = \max_{i=0,1,\dots,n} \text{card}(\mathcal{I}_i) \tag{3}$$

is for set \mathcal{I}_m . The centre

$$\eta = s_m + \frac{\xi}{2} \tag{4}$$

of the corresponding interval defines the mode of the neighbourhood \mathcal{N}_p . The support ν is the ratio of the number of elements in \mathcal{I}_m to the number of elements within the neighbourhood \mathcal{N}_p :

$$\nu = \frac{c_m}{\text{card}(\mathcal{N}_p)} \tag{5}$$

Let D_p be the disparity value at pixel p in disparity map D . We define our *mode filter* by the following rule:

$$D_p = \begin{cases} D_p & \text{if } |D_p - \eta| < \xi \wedge \nu > \psi \\ \text{invalid} & \text{otherwise} \end{cases} \tag{6}$$

This filter has two input parameters. ξ defines the disparity range of considered intervals and ψ defines the percentage how many disparities need to be in the mode interval. Choosing $\psi > 0.5$ avoids ambiguous cases when a maximum cardinality is taken by more than just one set \mathcal{I}_m . Figure 4 shows the result of this filter.

Ground Manifold Removal. It is common practice to approximate the ground manifold by a planar surface. A simple and fast, yet very robust method [14] uses a *v-disparity map* defined as follows:

Let $\{0, 1, \dots, d_{max}\}$ be the disparity range of an $M \times N$ disparity map D in the ij -plane. This 2D array is projected into a 2D v -disparity array V in the dj -plane:

$$V(d, j) = \sum_{i=1}^M \Psi(D_{ij} = d) \tag{7}$$

for $0 \leq d \leq d_{max}$ and $1 \leq j \leq N$, where $\Psi(\text{true}) = 1$ and equals zero otherwise. In [14], a dominant straight line is detected by Hough transform after binarization of V .

We approximate a dominant straight line by linear regression, where those points are iteratively discarded whose residues lie above the convergence threshold. For the initial point cloud, one representative is chosen for each disparity d , namely the second largest¹ j -value to where a value is projected. This line has the property of ‘rising from below’ towards the scattered points in the v -disparity map and it is ‘resting’ at a stable position. Disparities close to this ground manifold are removed prior segmentation.

¹ The origin of map V is in the top-left corner.

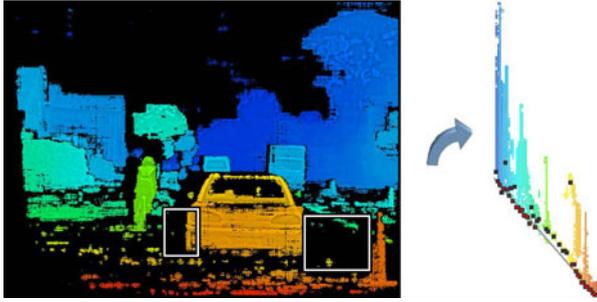


Fig. 4. Left: Disparity map after applying the mode filter. Right: The corresponding v-disparity map with calculated ‘lower straight envelope’.

Figure 4 shows a result for our regression method. For better visibility, we just project the disparity values and not their frequencies. The initial point cloud is marked by black squares; points contributing to the final regression line have a red dot inside of their square.

The Segmentation Algorithm. After stereo post-processing and ground manifold removal we segment the disparity map into consistent regions called *patches*: a patch is a 4-connected component such that any two pixels in this region can be connected by a 4-path inside the region such that the disparity difference between two consecutive disparity values is always less than a defined threshold. This segmentation rule is entirely defined in disparity space.

In other words, we decompose the disparity image into smooth stereo patches which can be adjacent but they are separated due to different disparity levels. Advantages of this segmentation method are that it is easy to implement, fast in execution, and results in a unique decomposition.

4 Correspondence Analysis and Tracking

Let $\mathcal{A}^t = \{P_0, P_1, \dots, P_n\}$, $\mathcal{A}^{t+1} = \{Q_0, Q_1, \dots, Q_m\}$ be the sets of patches obtained from two consecutive disparity images at times t and $t + 1$. In the following, symbol P refers always to a patch extracted from frame f_t , and Q to a patch from frame f_{t+1} . We search for corresponding patches such that each P corresponds to one Q at most, and each Q to at most one P . Some P 's or Q 's may not have corresponding patches.

Dissimilarity Measure. For measuring the dissimilarity of two patches P and Q , both given as sets of pixels in the same ij -plane, we apply a metric [13] defined by the ratio of the cardinality of the symmetric difference of P and Q to the cardinality of their union:

$$\Gamma(P, Q) = \frac{\text{card}(P \cup Q) - \text{card}(P \cap Q)}{\text{card}(P \cup Q)} \quad (8)$$

This metric equals zero if and only if both sets are equal, and equals one if both sets have no pixel (i, j) in common.

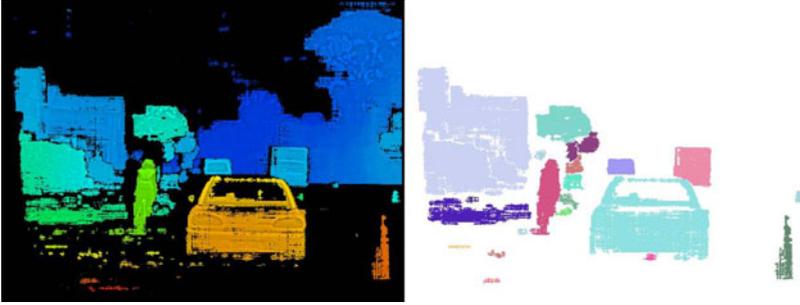


Fig. 5. Left: Result after mode filtering and ground manifold removal. Right: Result of the segmentation algorithm.

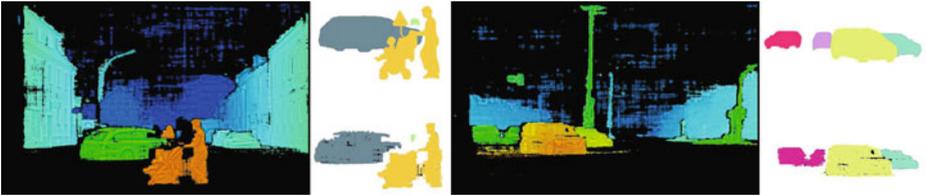


Fig. 6. Results of our segmentation experiment. Top: Silhouettes available as ground truth. Bottom: our segmentation results.

Two-Frame Correspondences. Because the position of patch P can change from frame f_t to frame f_{t+1} we use the relative motion as obtained by a dense optical flow algorithm (as specified above) to compensate for a translational component. Let $(u_p, v_p)^T$ be the flow at pixel p . We calculate the mean flow in the region occupied by P in f_t as

$$(u_P, v_P)^T = \frac{1}{\text{card}(P)} \sum_{p \in P} (u_p, v_p)^T \quad (9)$$

We shift the pixels in set P by $(u_P, v_P)^T$ into a new set \bar{P} .

For each $P_i \in \mathcal{A}^t$ we identify now the index j such that $\Gamma(\bar{P}_i, Q_j)$ is minimal for all $j = 0, \dots, m$.

Temporal Tracking. In order to track corresponding stereo patches over multiple frames we apply the two-frame correspondence procedure repeatedly. As for all tracking algorithms, incorporating a-priori knowledge improves the robustness of the tracking results. Therefore, a very simple statistical filter is incorporated into our framework.

The filter stores for each tracked patch its history of τ corresponding patches at previous time slots. For experiments reported below we used $\tau = 6$, thus less than a quarter of a second.

For keeping the history of patches, we shift the positions of pixels in all patches of the history by the mean flow calculated from the current motion map. This means that the τ -th patch is shifted τ times when keeping track of the history.

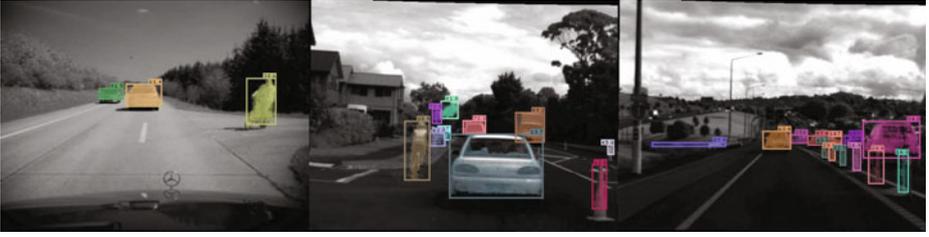


Fig. 7. Left to right: Bounding boxes of tracked patches for the sequences `intern on bike`, `cyclist` and `motorway`

To incorporate the prior knowledge of the history, a weight ω_t is assigned to each pixel of all patches based on the time instance t when it was added. The weights are then accumulated at each pixel location of the image domain over all patches of the current history. Thus we generate a new patch that represents the accumulated ‘knowledge’ of its history, and we use actually this generated patch for the two-frame correspondence.

Two-frame correspondence is now based on all pixels inside the $M \times N$ image domain where we have that $\omega_\Sigma > 0.5$ for the accumulated weight. Naturally, weights should decrease as t decreases, such that pixels from more recently added patches yield a higher contribution. For the six pixel sets in a history we use the weights (0.1, 0.1, 0.15, 0.15, 0.2, 0.3).

If there are patches $Q \in \mathcal{A}^{t+1}$ after the two-frame correspondence procedure that were not matched, then they are added as potentially new patches for the next time instance. If these patches are matched in the next time instance then they are confirmed as being a new patch, otherwise they are discarded. A confirmed patch is considered to be identified if it is tracked at least over three frames. Its occupied area is defined by the accumulated weights from the history. If a patch is lost for more than τ frames then it is removed from the tracked object list.

5 Experiments and Results

We show that segmentation and tracking in the disparity domain is of adequate quality to support common DAS tasks. A comparison with other segmentation and tracking methods needs to be left to a future paper.

Used Data for Evaluation. We have chosen three image sequences from a public benchmark site [5] that provides experimental data for traffic-related applications. The first two sequences (from Set 4, see [12]) are called `cyclist` and `motorway` and consists of 400 frames each, the name of the third sequence (from Set 1, see [19]) is `intern on bike` and it has 250 frames. We run the proposed approach on those sequences and perform a visual evaluation. We are especially interested in stability of the segmentation and over how many frames an object is tracked. We focus our brief discussion on objects that are relevant for a traffic situation.

Segmentation Experiment. In [3] a method for segmentation and tracking of independently moving objects (IMO) is presented. The segmentation is based on evaluating probabilities at pixels whether they are in motion in real world coordinates or static, based on using scene flow and ego-motion information. Although this approach is not comparable to the approach presented here, we use the ground truth image provided by [3] (publicly available in Set 7 of [5]) as indication for the quality of the segmentation part of our algorithm. After incorporating ego-motion analysis into our framework (at a later stage), segmented objects can be labelled ‘static’ or ‘in motion’, as in a scene flow approach.

Results and Discussion. Figure 6 shows resulting stereo maps of two traffic scenes provided by the study [3] as generated by our SGM implementation with subsequent mode filtering and ground manifold removal. On the right of both maps, contours are shown that correspond to moving objects within the scene. Contours on the top are colour-coded ground truth segmentations. Below are results obtained with our segmentation procedure. Black pixels are caused by invalid disparities. The results indicate in general that prominent objects are segmented within reasonable DAS quality limits.

In the scene shown on the left of Fig. 6, the umbrella of the baby buggy is segmented as an individual object in our method. The reason is that fine structures like these are rarely reconstructed by stereo algorithms which results in disconnected smaller segments. A motion based segmentation is here of benefit, because umbrella and baby buggy should show a similar motion pattern.

In the scene shown on the right of Fig. 6, the two cars which are close to the ego-vehicle are clearly segmented. The two cars further away, however, are connected due to disparity noise. Again, a motion based approach should in theory cope with this situation by identifying different motion patterns. However, the immediately relevant objects closer to the ego-vehicle are segmented with very high accuracy.

Figure 7 shows Frame 141 from the `intern on bike` sequence, Frames 60 from the `cyclist` sequence, and Frame 20 from the `motorway` sequence. Bounding boxes enclose all pixels belonging to the tracked patches. The number in the top right corner indicates a mean estimate of the distance between ego-vehicle and object (in metres in the 3D world). Each object obtains an ID and a unique colour for its bounding box. If a tracked object is discarded and later picked up again, the colour will differ. For better visibility, we removed a few tracked objects in the visualization of results, but not in the tracking algorithm (e.g. all objects higher than 6 m and in cases of the `cyclist` sequence, everything left of the cyclist).

Reliable detection is possible up to 100 m when objects are in clear sight, meaning objects are lost only after a distance of 100 m or more, and are picked up by the tracking module before coming close to the 100 m mark. This range depends on camera parameters (e.g. baseline, focal length).

In the `cyclist` sequence we track cyclist, car, the van in the background, and the approaching truck robustly during the whole sequence. The truck is picked up at a distance of 137 m, and the van is lost at a distance of 130 m. The car is lost after 80 m, but this is mainly due because it vanishes at a corner behind a tree.

In the `intern on bike` sequence the program picked up all relevant objects until they disappear from the video. The bike is picked up at a distance of 89 m, the first

approaching car at 121 m and the second approaching car at 95 m. In the motorway sequence all the numerous objects are tracked with a good robustness.

6 Conclusions

The main contribution of our work is to propose a concept and an implementation for solely disparity-based segmentation and tracking. Further contributions are the proposal of a mode filter for stereo post-processing which we found is crucial for an efficient flood flow segmentation algorithm. Or the application of a metric for set correspondences in the DAS context.

Methods and filters designed for 3D data analysis (see upper row in Fig. 1) could be applied to our method. Subsequent processing steps after disparity re-projection, such as 3D shape models or particle filters, will help to increase reliability. Another way of interpretation is, of course, that our presented approach can be considered as additional source of information for standard DAS methods.

Our results indicate the possibility of reliable segmentation and tracking in the disparity domain. It is difficult to compare our results with those provided by other methods, and more ground-truth data such as Set 7 on [5] would help.

Issues with our proposed segmentation approach: First, segmented objects may, depending on the stereo map, be connected in one frame and disconnected in the next; there is a need to generate a filter that resolves this behaviour based on prior knowledge. Second, small objects close to larger objects may not be segmented into independent objects because of insufficient disparity dynamics. Note that this applies in 3D data processing in general: Only if objects are moving then motion-based segmentation is able to identify different objects.

For the next stage we consider to include standard temporal filters to improve the stability and robustness such that more challenging traffic scenarios can also be processed with the same quality as for the sequences used in this paper.

Acknowledgments. The authors thank James Milburn for his implementation of the used dense optical flow algorithm.

References

1. Badino, H.: A Robust Approach for Ego-Motion Estimation Using a Mobile Stereo Platform. In: Jähne, B., Mester, R., Barth, E., Scharr, H. (eds.) IWCM 2004. LNCS, vol. 3417, pp. 198–208. Springer, Heidelberg (2007)
2. Badino, H., Franke, U., Pfeiffer, D.: The Stixel World - A Compact Medium Level Representation of the 3D-World. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 51–60. Springer, Heidelberg (2009)
3. Barth, A., Siegemund, J., Meißner, A., Franke, U., Förstner, W.: Probabilistic Multi-Class Scene Flow Segmentation for Traffic Scenes. In: Goesele, M., Roth, S., Kuijper, A., Schiele, B., Schindler, K. (eds.) DAGM 2010. LNCS, vol. 6376, pp. 503–512. Springer, Heidelberg (2010)
4. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High Accuracy Optical Flow Estimation Based on a Theory for Warping. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)
5. enpeda. image sequences analysis test site, <http://www.mi.auckland.ac.nz/EISATS>

6. Franke, U., Rabe, C., Badino, H., Gehrig, S.: 6D-Vision: Fusion of Stereo and Motion for Robust Environment Perception. In: Kropatsch, W.G., Sablatnig, R., Hanbury, A. (eds.) DAGM 2005. LNCS, vol. 3663, pp. 216–223. Springer, Heidelberg (2005)
7. Gómez, G.D.: A global approach to vision-based pedestrian detection for advanced driver assistance systems. PhD thesis, Univ. Autónoma de Barcelona (2010)
8. Haller, I., Pantillie, C., Oniga, F., Nedeveschi, S.: Real-time semi-global dense stereo solution with improved sub-pixel accuracy. In: IVS, pp. 369–376 (2010)
9. Hirschmüller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. CVPR 2, 807–814 (2005)
10. Hirschmüller, H., Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. IEEE Trans. Pattern Analysis Machine Int. 31, 1582–1599 (2009)
11. Ivekovic, S., Clark, D.: Multi-Object Stereo Filtering in Disparity Space. In: COGIS (2009)
12. Klette, R., Krüger, N., Vaudrey, T., Pauwels, K., van Hulle, M., Morales, S., Kandil, F., Haeusler, R., Pugeault, N., Rabe, C., Markus, L.: Performance of correspondence algorithms in vision-based driver assistance using an online image sequence database. IEEE Trans. Vehicular Technology (2011)
13. Klette, R., Rosenfeld, A.: Digital Geometry - Geometric Algorithms for Digital Picture Analysis. Morgan Kaufmann, San Francisco (2004)
14. Labayrade, R., Aubert, D., Tarel, J.-P.: Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation. In: IVS, pp. 646–651 (2002)
15. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IUW, pp. 121–130 (1981)
16. Oniga, F., Nedeveschi, S., Meinecke, M.M.: Occupancy grids detected from dense stereo using an elevation map representation. In: WIT, pp. 133–138 (2009)
17. Petersson, L., Fletcher, L., Zelinsky, A., Barnes, N., Arnell, F.: Towards safer roads by integration of road scene monitoring and vehicle control. Int. J. Robotic Res. 25, 53–72 (2006)
18. Shimizu, M., Okutomi, M.: An analysis of subpixel estimation error on area-based image matching. In: Proc. Digital Signal Processing, vol. 2, pp. 1239–1242 (2002)
19. Vaudrey, T., Rabe, C., Klette, R., Milburn, J.: Differences between stereo and motion behaviour on synthetic and real-world stereo sequences. In: IVCNZ, pp. 1–6 (2008)
20. Wedel, A., Badino, H., Rabe, C., Loose, H., Franke, U., Cremers, D.: B-spline modeling of road surfaces with an application to free space estimation. In: IVS, pp. 828–833 (2008)
21. Wedel, A., Meißner, A., Rabe, C., Franke, U., Cremers, D.: Detection and Segmentation of Independently Moving Objects from Dense Scene Flow. In: Cremers, D., Boykov, Y., Blake, A., Schmidt, F.R. (eds.) EMMCVPR 2009. LNCS, vol. 5681, pp. 14–27. Springer, Heidelberg (2009)
22. Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., Cremers, D.: Efficient dense scene flow from sparse or dense stereo data. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 739–751. Springer, Heidelberg (2008)
23. Wegener, P.: A technique for counting ones in a binary computer. Comm. ACM 3, 322 (1960)
24. Yu, Q., Araujo, H., Wang, H.: A Stereovision Method for Obstacle Detection and Tracking in Non-Flat Urban Environments. Autonomous Robots 19, 141–157 (2005)
25. Zabih, R., Woodfill, J.: Non-Parametric Local Transform for Computing Visual Correspondence. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 801, pp. 151–158. Springer, Heidelberg (1994)
26. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV- L^1 optical flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 214–223. Springer, Heidelberg (2007)
27. Zhao, L., Thorpe, C.: Stereo and neural network-based pedestrian detection. IEEE Trans. Int. Transportation Systems 1, 148–154 (2000)