3D Object Trajectory Reconstruction using Instance-Aware Multibody Structure from Motion and Stereo Sequence Constraints

Sebastian Bullinger*, Christoph Bodensteiner*, Michael Arens* and Rainer Stiefelhagen†

***Abstract*— Three-dimensional environment perception is a key element of autonomous driving and driver assistance systems. A common image based approach to determine three-dimensional scene information is stereo matching, which is limited by the stereo camera baseline. In contrast to stereo matching based methods, we present an approach to reconstruct three-dimensional object trajectories combining temporal adjacent views for object point triangulation. We track two-dimensional object shapes on pixel level exploiting instance-aware semantic segmentation techniques and optical flow cues. We apply Structure from Motion (SfM) to object and background images to determine initial camera poses relative to object instances as well as background structures and refine the initial SfM results by integrating stereo camera constraints using factor graphs. We compute object trajectories using stereo sequence constraints of object and background reconstructions. We show qualitative results using publicly available video data of driving sequences. Due to the lack of suitable ground truth, we create a synthetic benchmark dataset of stereo sequences with vehicles in urban environments. Our algorithm achieves an average trajectory error of 0.09 meter using the dataset. The dataset is on our website[1] publicly available.**

I. Introduction

A. Trajectory Reconstruction

Three-dimensional object motion trajectories are crucial for autonomous driving and driver assistance systems to avoid collisions and to perform path planing. There is a variety of sensor types to capture three-dimensional information corresponding to moving objects. In comparison to active sensors like Lidar or Radar, cameras reduce weight and size of the system and lower production costs. We propose an approach to reconstruct three-dimensional object motion trajectories using two cameras as sensors. Previous methods [1], [2] use stereo matching to determine 3D objects points. However, 3D stereo measurement precision deteriorates quickly with camera distance [3], [4] due to limited stereo camera baselines. To tackle this problem we combine temporal adjacent views using Structure from Motion, which allows us to exploit virtual camera baselines that are not restricted by the stereo camera setup. For example, even small object rotations may result in big virtual camera baseline differences. In many scenes, objects cover only a minority of pixels. This increases the difficulty of reconstructing object motion trajectories using image data. In such cases, current state-of-the-art Structure from Motion (SfM) approaches [5], [6] consider moving object observations most likely as outliers and reconstruct background structures instead. Previous works, e.g. [7], [8], detect moving objects by applying motion segmentation or keypoint tracking. Recent progress in instance-aware semantic segmentation [9], [10] and optical flow [11], [12] techniques allow for object tracking on pixel level [13] and handle stationary objects naturally. We extend the approach in [13] to track objects on pixel level in stereo video data. Stereo object tracking allows us to compute object and background reconstructions using SfM [5], [6]. We refine the SfM reconstruction results by exploiting stereo projection constraints using factor graphs [14]. Known stereo camera baselines resolve the scale ambiguity between object and background reconstruction and allow us to compute metric object motion trajectories. In contrast to stereo matching based methods, our approach allows to build an holistic models for each moving object.

B. Related Work

Our pipeline uses semantic segmentation and structure from motion to reconstruct object trajectories. [15] presents Fully Convolutional Networks for semantic segmentation, which are trained end-to-end. [9], [10] extended this concept and proposed instance-aware semantic segmentation approaches. We considered [9], [16] and [10] to detect objects on pixel level as well as [12] and [11] to compute the optical flow of adjacent images. We observe that [10] and [12] achieve the best segmentation and optical flow results. [12] computes consistent optical flow vectors also for large object displacements.

The field of Structure from Motion (SfM) consists of iterative [17], [5], [6], [18] and global approaches [5], [18]. In our experiments [6] and [5] created the most reliable object and background reconstructions.

A factor graph is a suitable graphical model to represent SfM and Simultaneous Localization and Mapping (SLAM) problems. We use factor graphs to model stereo camera constraints. We apply state-of-the-art SfM libraries [5], [6] to perform data association and initialization. We use the GTSAM library [19] to define factor graphs corresponding to SfM reconstructions. GTSAM [19] does not provide functionality to perform data association or initialization.

Previous works [20], [7], [21], [22], [23] proposed methods to reconstruct object and vehicle trajectories in monocular video data. In contrast to our approach, these methods

*Sebastian Bullinger, Christoph Bodensteiner and Michael Arens are with the Department of Object Recognition, Fraunhofer IOSB, Ettlingen, Germany. E-Mail: <firstname>.<lastname>@iosb.fraunhofer.de

†Rainer Stiefelhagen is with the Department of Computer Science, Karlsruhe Institute of Technology, Karslruhe, Germany. E-Mail: rainer.stiefelhagen@kit.edu

[1]Project page: http://s.fhg.de/trajectory

need to define object motion constraints to tackle the scale ambiguity inherent to monocular image based reconstructions. [22] presents a synthetic dataset to evaluate the reconstruction of vehicle trajectories in monocular video data quantitatively. The authors in [1] reconstruct vehicle shapes and trajectories in stereo video data using off-the-shelf ego-motion and stereo matching based reconstruction algorithms. [2] presents a combination of object proposals, stereo matching, visual odometry and scene flow to compute three-dimensional vehicle tracks in traffic scenes. We use the stereo matching based object trajectory reconstruction method proposed in [24] as baseline. We considered different off-the-shelf stereo matching methods [25], [26], [27] to compute the corresponding disparity values, since usage of and fair comparisons to ConvNet based stereo matching approaches like [28], [29] are limited due to the lack of pre-trained models and required fine-tuning in the target domain. We observe that [26] computes more stable object specific disparities than [25], [27].

C. Contribution

The core contributions of this work are as follows. (1) We present a new framework to reconstruct the three-dimensional trajectory of moving instances of known object categories in stereo video data leveraging state-of-the-art semantic segmentation and structure from motion approaches. Our method allows to track two-dimensional object shapes on pixel level in stereo image sequences. In contrast to stereo matching methods, our approach leverages views from different time steps for object point triangulation. (2) We show how stereo constraints modeled with factor graphs improve inital SfM reconstructions. This SfM refinement step allows us to determine metric object motion trajectories. (3) We create a benchmark dataset of synthetic stereo sequences capturing driving vehicles in urban environments suitable to evaluate image based vehicle trajectory reconstruction methods quantitatively. (4) We demonstrate the usefulness of our method by showing qualitative results of reconstructed object motion trajectories using publicly available driving sequences.

II. Object Motion Trajectory Reconstruction

The pipeline of our approach is shown in Fig. 1. The input is an ordered stereo image sequence. We track two-dimensional object shapes on pixel level across video sequences exploiting instance-aware semantic segmentation [16] to identify object shapes and optical flow [11] to associate extracted object shapes in corresponding stereo images and subsequent frames. Without loss of generality, we describe the motion trajectory reconstruction of single objects. We apply SfM [5], [6] to object and background images as shown in Fig. 1. Object images denote pictures containing only color information of single object instances. Similarly, background images show only environment structures. We combine information of object and background SfM reconstructions to determine consistent object motion trajectories. We use factor graphs [14] to refine object and

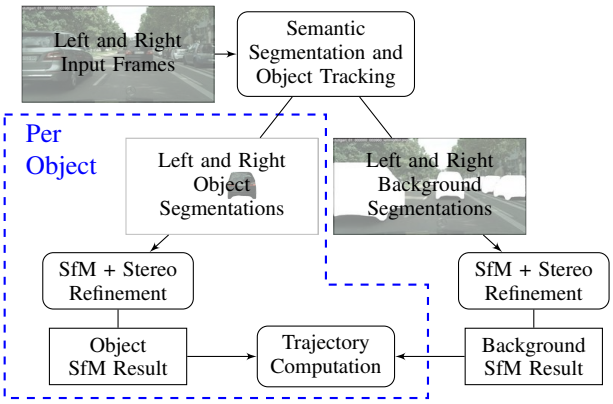
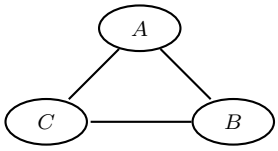
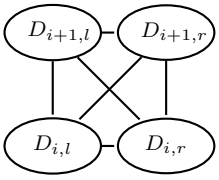


Fig. 1: Overview of the trajectory reconstruction pipeline. Boxes with corners denote computation results and boxes with rounded corners denote computation steps.



(a) Visualization of the general three-dimensional assignment problem as 3-partite graph.



(b) Visualization of the stereo MOT assignment problem as 4-partite graph.

Fig. 2: Comparison of the three-dimensional and the stereo MOT assignment problem. The circles visualize the partition of the nodes of the corresponding multipartite graph. There exist no edges between nodes of the same subset. The lines denote that elements of a subset share edges with the elements of a different subset. $D_{i,l}$, $D_{i,r}$, $D_{i+1,l}$ and $D_{i+1,r}$ denote the objects detections visible in the left and right image at time i and $i+1$.

background reconstructions, which allows us to use stereo camera baseline to resolve the scale ambiguity of the SfM results.

Point triangulation using stereo correspondences is limited by the baseline of the corresponding stereo camera [4]. Our method circumvents this problem by exploiting information of subsequent frames to triangulate 3D points. Already small object rotations may result in big virtual camera baseline changes. In contrast to stereo matching methods, the proposed approach builds object models reflecting the information of each frame. To build an holistic object model with stereo matching requires additional steps to fuse triangulated points of subsequent frames.

A. Stereo Online Multiple Object Tracking

The proposed stereo Multiple Object Tracking (MOT) approach extends the monocular tracking algorithm presented in [13] and is depicted in Fig. 3. [13] use optical flow matches to associate instance-aware semantic segmentations between subsequent frames to track the two-dimensional shape of objects of known categories on pixel level across monocular video sequences. This approach allows to naturally associate

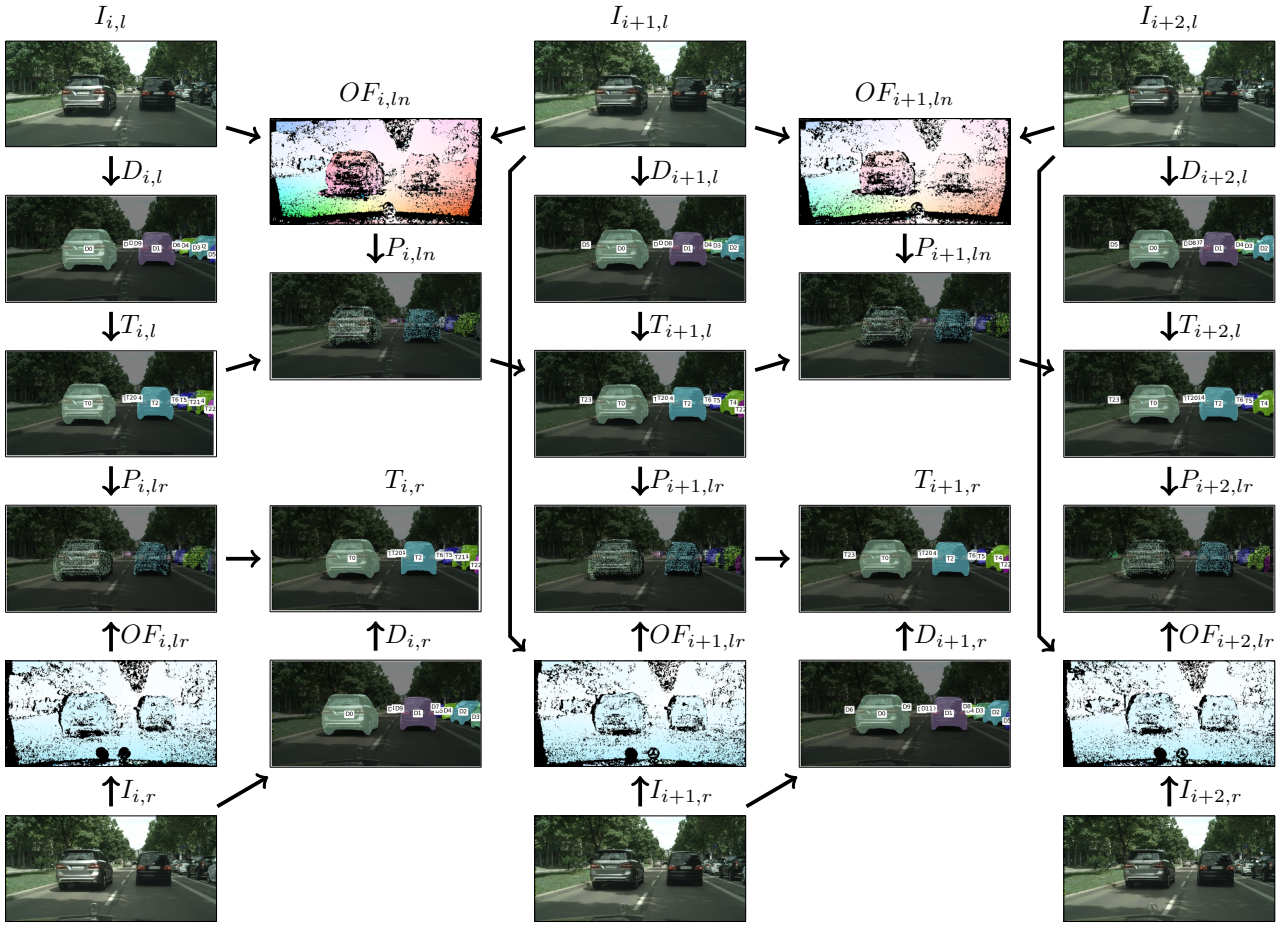


Fig. 3: Scheme of the stereo object tracking algorithm. The variables have the following meaning. I: image, OF: optical flow, D: detection, P: prediction, T: tracker state, i: image index, l: left, r: right, ln: left-next, lr: left-right. Arrows show the relation of computation steps. A computation step depends on the results connected with incoming arrows. The tracked objects $T_{i,l}$ at time i in the left image are predicted to the next image using the optical flow $OF_{i,ln}$ of image $I_{i,l}$ and $I_{i+1,l}$. The predictions $P_{i,ln}$ are associated with the detections $D_{i+1,l}$ to update the left tracker state. Simultaneously, the tracked object objects $T_{i,l}$ are predicted to the corresponding right image of the same time step using the optical flow $OF_{i,lr}$. The predictions $P_{i,lr}$ are associated with the detections $D_{i,r}$ to compute the tracker state $T_{i,r}$ of the objects in right image at time step i. Corresponding objects in the left and the right tracker state $T_{i,l}$ and $T_{i,r}$ share the same identifier, which is not necessarily the case for detections in $D_{i,l}$ and $D_{i,r}$. The used optical flow color coding is defined in [30].

objects in subsequent frames as well as objects in left and right images of the stereo camera. [13] uses the Kuhn-Munkres algorithm [31] to solve the two-dimensional assignment problem (AP), i.e. to determine object associations of objects between image pairs. The two-dimensional AP consists of finding a maximum weight matching in a weighted bipartite (or 2-partite) graph. An improved version of the Kuhn-Munkres algorithm [32] solves the two-dimensional AP in $\mathcal{O}(n^3)$, where n is the number of elements to be assigned. There exist different three-dimensional extensions of the two-dimensional AP and a few special cases can be solved in polynomial time [33]. However, the general three-dimensional AP is NP-hard [34]. In the stereo MOT case object instances in the left image $I_{i,l}$ and the right image $I_{i,r}$ at time i as well as the object instances in the left image $I_{i+1,l}$ and the right image $I_{i+1,r}$ at time $i+1$ must be associated. Therefore, the stereo MOT AP corresponds to the general four-dimensional assignment problem (see Fig. 2(b)). Fig. 2(a) and 2(b) show that the stereo MOT AP comprises the general three-dimensional AP and is therefore NP-hard. We do not solve the associations of $I_{i,l}$, $I_{i+1,l}$, $I_{i,r}$ and $I_{i+1,r}$ simultaneously, since (a) the stereo MOT AP is NP-hard and (b) the simultaneous determination of two subsequent stereo image pairs requires the computation of three optical flow fields in addition to $OF_{i,lr}$ and $OF_{i,ln}$. Here, $OF_{i,lr}$ and $OF_{i,ln}$ denote the optical flow between image $I_{i,l}$ and $I_{i,r}$ as well as $I_{i,l}$ and $I_{i+1,l}$. Instead, we apply the following greedy approximation of the stereo MOT AP by solving two two-dimensional assignment problems. This allows us to determine object correspondences in $I_{i,l}$ and $I_{i,r}$ as well as $I_{i,l}$ and $I_{i+1,l}$ in $\mathcal{O}(n^3)$.

We associate object instances in the left images $I_{i,l}$ and $I_{i+1,l}$

using the object affinity matrix presented in [13] as input for the Kuhn-Munkres algorithm to compute the tracker state $T_{i+1,l}$. In this case the affinity matrix is defined according to (1). Here, $O_{p,d}$ denotes the overlap of the prediction with index p in $P_{i,ln}$ and the detection with index d in $D_{i+1,l}$. Let n_p and n_d denote the number of predictions in $P_{i,ln}$ and the number of detections in $D_{i+1,l}$.

$$\mathbf{A}_t = \begin{bmatrix} O_{1,1} & \cdots & O_{1,d} & \cdots & O_{1,n_d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ O_{p,1} & \cdots & O_{p,d} & \cdots & O_{p,n_d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ O_{n_p,1} & \cdots & O_{n_p,d} & \cdots & O_{n_p,n_d} \end{bmatrix} \quad (1)$$

Fig. 3 shows an example for $P_{i,ln}$ and $D_{i+1,l}$. The tracker state $T_{i+1,l}$ contains only tracks of object instances in images corresponding to the left camera. We use the optical flow between left and right images $OF_{i+1,lr}$ to associate the tracker state of left images $T_{i+1,l}$ with objects visible in the corresponding right image. The association between predictions $P_{i+1,lr}$ and detections $D_{i+1,r}$ in the right images are also computed using an affinity matrix and the Kuhn-Munkres algorithm. In this case $O_{p,d}$ denotes the overlap of prediction p in $P_{i+1,lr}$ and detection d in $D_{i+1,r}$. n_p denotes the number of predictions in $P_{i+1,lr}$ and n_d denotes the number of detections in $D_{i+1,r}$. The overlap $O_{p,d}$ is an affinity measure that reflects locality and visual similarity.

B. Object Motion Trajectory Computation

We follow the pipeline outlined in Fig. 1 and apply SfM simultaneously to object and background images. We denote corresponding reconstruction results with $sfm^{(o)}$ and $sfm^{(b)}$. Each object image has a corresponding background image, i.e. the background image extracted from the same input frame. We consider only object-background-cameras-pairs that are part of $sfm^{(o)}$ as well as $sfm^{(b)}$, i.e. we remove cameras without corresponding object or background camera from the reconstructions. Let $\mathbf{o}_j^{(o)}$ denote the 3D points contained in $sfm^{(o)}$. The superscript (o) in $\mathbf{o}_j^{(o)}$ describes the corresponding coordinate frame. The variable j denotes the index of the points in the object point cloud. We combine information of object-background-image-pairs to define object motion trajectories parameterized by a single parameter. The object reconstruction $sfm^{(o)}$ contains object point positions $\mathbf{o}_j^{(o)}$ as well as corresponding camera centers $\mathbf{c}_i^{(o)}$ and rotations $\mathbf{R}_i^{(o)} \in SO(3)$. We convert the object points $\mathbf{o}_j^{(o)}$ defined in the coordinate frame system (CFS) of the object reconstruction to points $\mathbf{o}_j^{(i)}$ in the camera CFS of camera i using $\mathbf{o}_j^{(i)} = \mathbf{R}_i^{(o)} \cdot (\mathbf{o}_j^{(o)} - \mathbf{c}_i^{(o)})$. We use the camera center $\mathbf{c}_i^{(b)}$ and the corresponding rotation $\mathbf{R}_i^{(b)} \in SO(3)$ contained in the background reconstruction $sfm^{(b)}$ to transform object points in camera coordinates to the background CFS using $\mathbf{o}_{j,i}^{(b)} = \mathbf{c}_i^{(b)} + {\mathbf{R}_i^{(b)}}^T \cdot \mathbf{o}_j^{(i)}$. The transformation between points in the object CFS and points in the background CFS is defined in (2).

$$\mathbf{o}_{j,i}^{(b)} = \mathbf{c}_i^{(b)} + {\mathbf{R}_i^{(b)}}^T \cdot \mathbf{R}_i^{(o)} \cdot \left(\mathbf{o}_j^{(o)} - \mathbf{c}_i^{(o)}\right) \quad (2)$$

The naive combination of object and background reconstruction results in inconsistent object motion trajectories due to the scale ambiguity of SfM [35]. We adjust the scale between object and background reconstruction using the baseline of the stereo cameras in object and background reconstructions as reference. We can recover the full object motion trajectory computing (2) for each object-background-image-pair. We use $\mathbf{o}_{j,i}^{(b)}$ of all cameras and object points as representation of the object motion trajectory.

C. SfM Stereo Refinement and Outlier Removal

Reconstructions of dynamic objects using state-of-the-art SfM tools occasionally contain incorrectly registered cameras as well as incorrectly triangulated object points due to small object sizes, changing illumination and reflecting surfaces. Fig. 7 shows a few examples. Incorrect camera baselines hamper the correct estimation of the scale ratio between object and background reconstruction. We model stereo projection constraints to refine the previously computed SfM reconstructions by leveraging factor graphs [14]. In the following, we describe the factor graph based refinement for the object reconstruction. The refinement of the background reconstruction is performed analogously.

A factor graph is a bipartite graph $G = (\mathcal{F}, \Theta, \mathcal{E})$ with two node types: factor nodes $f_k \in \mathcal{F}$ and variable nodes $\theta_l \in \Theta$. The edges $e_{k,l} \in \mathcal{E}$ connect factor and variable nodes. A factor graph is a graphical model used to represent the factorization of a function f according to equation (3),

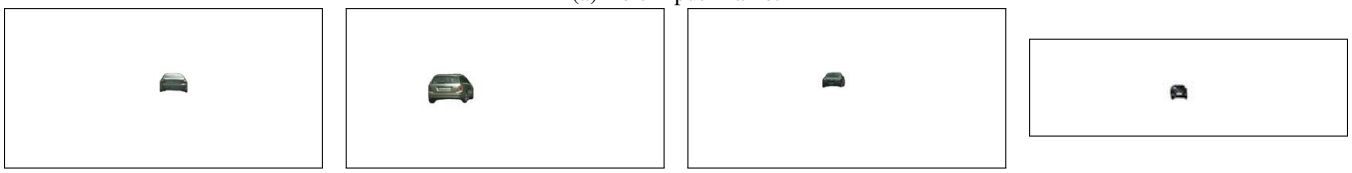
$$f(\Theta) = \prod_k f_k(\Theta_k) \quad (3)$$

where Θ_k is the set of variables θ_l adjacent to f_k, i.e. each $\theta_l \in \Theta_k$ is connected with an edge to f_k.

In our case the variable nodes Θ represent stereo camera poses θ_s and triangulated object points θ_p. We use a set of stereo factors f_k to reflect the relation of triangulated object points projected into specific stereo cameras and their corresponding observations. In order to map the observation constraints in the SfM result onto the stereo factors, we determine for each triangulated object point in the SfM reconstruction all pairs of corresponding feature observations m_l and m_r of the left and the right image of the same time step. We add stereo projection factors of the form $f_k(\theta_p, \theta_s; m_{l,h}, m_{r,h}, m_v, \mathbf{K}, b)$, where $m_{l,h}$ and $m_{r,h}$ denote the horizontal pixel positions of the measurements m_l and m_r. m_v denotes the averaged vertical pixel position of corresponding left and right observations. $\mathbf{K}$ and b represent the calibration matrix and the stereo camera baseline. Note that in $f_k(\theta_p, \theta_s; m_{l,h}, m_{r,h}, m_v, \mathbf{K}, b)$ the parameters θ_p and θ_s are variable nodes, whereas $m_{l,h}, m_{r,h}, m_v, \mathbf{K}$ and b are fixed (measured) values.

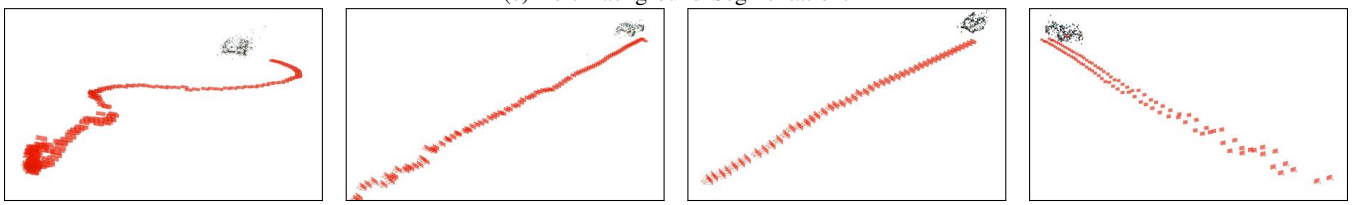
Using a Gaussian measurement model translates the determination of the optimal values for the variable nodes θ_l into

(a) Left Input Frame.



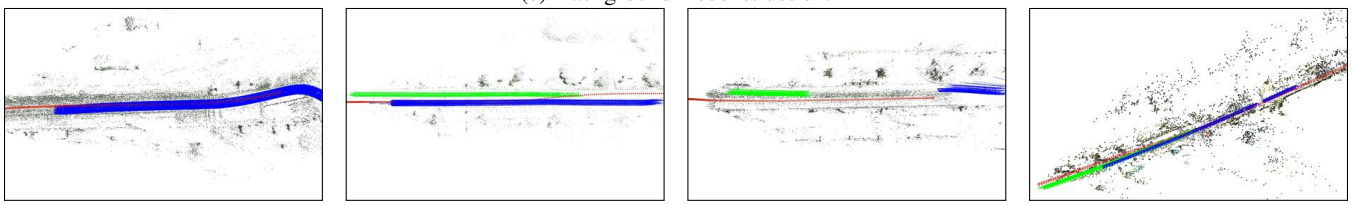
(b) Left Object Segmentation.

(c) Left Background Segmentation.

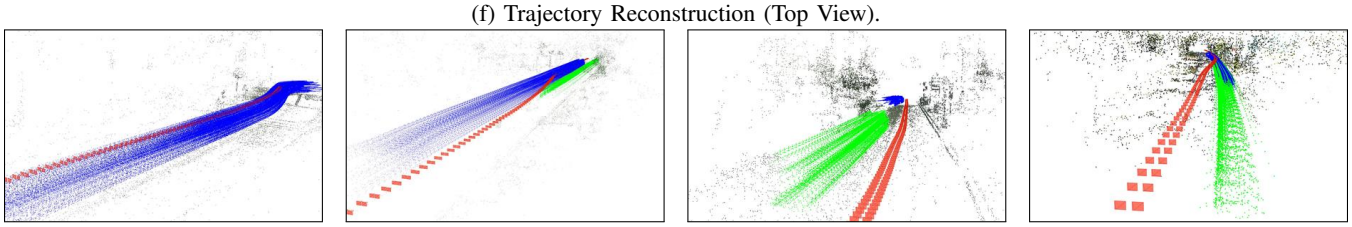


(d) Object Reconstruction.

(e) Background Reconstruction.



(f) Trajectory Reconstruction (Top View).



(g) Trajectory Reconstruction (Side View).

Fig. 4: Vehicle trajectory reconstruction using three sequences (stuttgart01-stuttgart03) contained in the Cityscape dataset [36] and one sequence (2011_09_26_drive_0013) of the KITTI dataset [37]. Object segmentations and object reconstructions are shown for one of the vehicles visible in the scene. The reconstructed cameras are shown in red. The vehicle trajectories are colored in green and blue. The figure is best viewed in color.

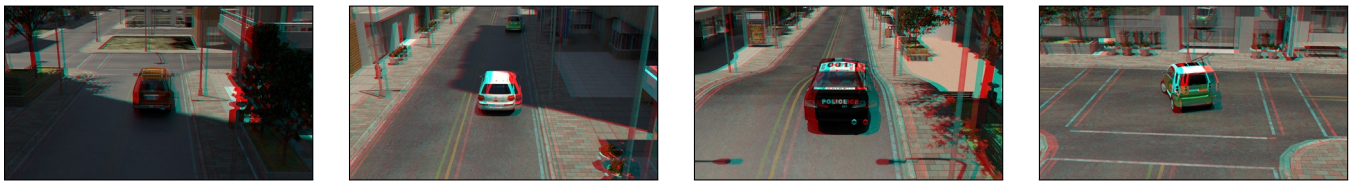


Fig. 5: Anaglyph images representing stereo information of the sequences contained in the presented virtual vehicle trajectory dataset. Information of left and right images are highlighted with green and red, respectively.

the nonlinear least-squares problem shown in (4).

$$\underset{\Theta}{\operatorname{argmin}}(-\log f(\Theta)) = \underset{\Theta}{\operatorname{argmin}} \sum_k \|h_k(\Theta_k) - z_k\|^2_{\sum_k} \quad (4)$$

Here, $\|\cdot\|^2_{\sum_k}$ denotes the squared Mahalanobis distance with covariance matrix $\sum_k$. $h_k(\Theta_k)$ and z_k denote the measurement function and measurement corresponding to the stereo factor f_k. For more details see [38]. To determine the maximum a posteriori estimate, we apply the Levenberg-Marquardt algorithm to (4), which solves the nonlinear least-squares problem iteratively. We initialize the stereo camera variable nodes θ_s with the pose of the left cameras $[\mathbf{R}_i^{(o)}|\mathbf{t}_i^{(o)}]$ with $\mathbf{t}_i = -\mathbf{R}_i^{(o)}\mathbf{c}_i$ and the landmark variables nodes θ_p with the triangulated points $\mathbf{o}_j^{(o)}$. The resulting reconstructions show consistent camera stereo baselines. Fig. 7 shows a comparison of initial and refined reconstructions.

We determine for all 3D object points in the stereo-refined reconstruction result an objectness score by projecting each point onto the tracked object segmentation for all cameras. This allows us to remove outliers using the semantic outlier filtering presented in [23].

Monocular projection factors in combination with odometry factors between left and right cameras provide an alternative to stereo projection factors. However, this increases the amount of variable and factor nodes, which results in a higher computation time.

D. Stereo Matching Baselines

We use the object trajectory reconstruction method proposed in [24] as baseline for our experiments. This approach combines stereo matching based object point triangulations and camera poses obtained by the environment SfM reconstruction. Stereo matching [25], [26] exploits knowledge of the stereo camera setup to determine correspondences, i.e. matches are determined along scan lines. This allows to compute pixelwise disparity functions $d_i(\cdot)$ for each time step i. With the object tracking approach described in section II-A we determine object specific pixel disparities $d_i(u, v)$, where (u, v) denotes a pixel corresponding to the object in the left image at time i. For each time step i we back-project the pixel disparity triplets $(u, v, d_i(u, v))$ according to equation (5) to determine the corresponding homogeneous points $(x_u, v_v, z, w_{u,v,i})$.

$$\begin{bmatrix} x_u \\ y_v \\ z \\ w_{u,v,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -c_u \\ 0 & 1 & 0 & -c_v \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{-1}{b} & \frac{c_u - c'_u}{b} \end{bmatrix} \cdot \begin{bmatrix} u \\ v \\ d_i(u,v) \\ 1 \end{bmatrix} \quad (5)$$

Here, f and (c_u, c_v) denote the focal length and principal point in pixels. b is the baseline of the stereo camera in the background SfM coordinate frame system. Normalizing $(x_u, y_v, z, w_{u,v,i})^T$ yields the actual three-dimensional object point $\mathbf{p}_j^{(i)}$ in camera coordinates. To compute the full object trajectory we transform the object point cloud for each time step i into world coordinates with $\mathbf{p}_{j,i}^{(b)} = \mathbf{c}_i^{(b)} + \mathbf{R}_i^{(b)^T} \cdot \mathbf{p}_j^{(i)}$. In contrast to the trajectory reconstruction method proposed in section II-B, this approach does not leverage information of subsequent frames to triangulate object points.

III. Dataset

In order to perform a quantitative evaluation of reconstructed object motion trajectories we require accurate object shape models as well as synchronized object and camera poses at each time step. The registration and synchronization of object and camera poses is a complex process and the corresponding results contain noise and artifacts like drift. [22] circumvents these problems using a virtual dataset for monocular vehicle trajectory evaluation. By extending the presented Blender framework we render stereo sequences and create corresponding stereo camera ground truth poses. Fig. 5 shows a few anaglyph images representing rendered images corresponding to the same stereo camera. We set the baseline to 0.3 meter, which lies between the stereo baselines used in common real word datasets [37], [36]. Overall the dataset contains 35 sequences of five different vehicles on seven motion trajectories.

IV. Experiments and Evaluation

Our object trajectory reconstruction pipeline uses [10] and [12] for object segmentation and optical flow computations. We leverage [6] for object and [5] for background reconstructions. For all SfM reconstructions, we applied a camera model with fixed focal length, principal point and no radial distortion. We use the stereo matching approach proposed by [26] with the object trajectory reconstruction method presented in [24] as baseline. The current implementation of our pipeline does not meet real time requirements.

A. Qualitative Evaluation

Due to the lack of suitable ground truth data we show qualitative trajectory reconstruction results using publicly available datasets of driving sequences presented in [36] and [37]. Fig. 4 shows intermediate and final results produced by the proposed method. Fig. 7 shows a comparison of object SfMs result before and after refinement using factor graphs.

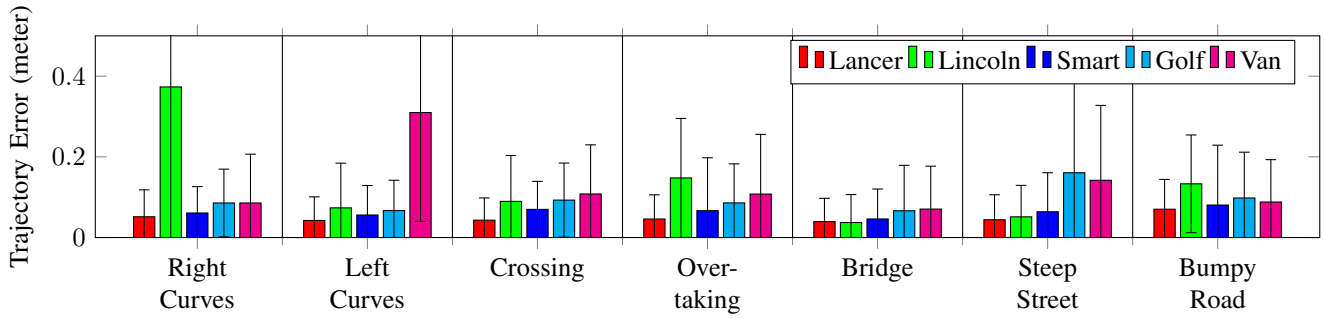
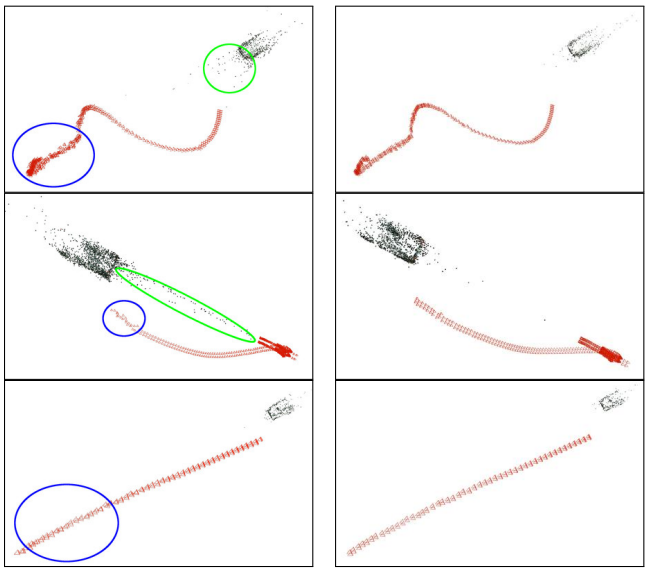


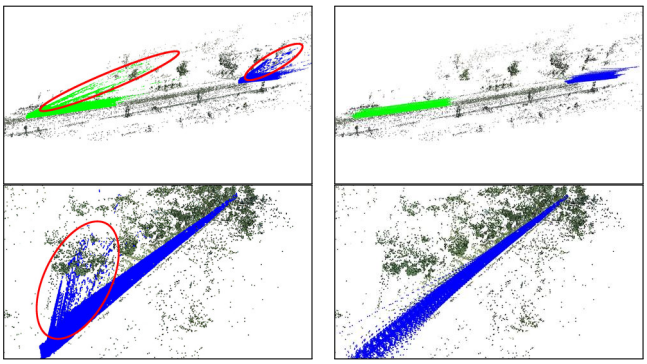
Fig. 6: Quantitative evaluation of the proposed method using the dataset presented in [22]. The dataset contains seven different vehicle trajectories (*Right Curves*, *Left Curves*, *Crossing* ...) and five different vehicle models (*Lancer*, *Lincoln Navigator*, ...). The bars shows the trajectory error in meter and the intervals the corresponding standard deviations.



(a) Initial object SfM result.

(b) Refined object SfM result.

Fig. 7: Comparison of initial SfM object reconstructions and corresponding refinements using stereo constraints. The cameras are shown in red. The blue and green circle emphasizes incorrectly registered cameras and triangulated points.



(a) Stereo Matching Baseline.

(b) Proposed Method.

Fig. 8: Trajectory reconstruction examples using sequences of the CityScapes dataset. The red circles emphasize incorrectly triangulated trajectory points.

Method	Average Trajectory Error (meter)					
	Lancer	Lincoln	Smart	Van	Golf	Overall
Ours	**0.05**	0.13	**0.06**	**0.09**	**0.13**	**0.09**
Baseline	0.06	**0.06**	0.07	0.10	0.27	0.11
[23]	0.11	0.09	0.14	0.21	0.30	0.17
[22]	0.20	0.23	0.33	0.33	0.47	0.31

TABLE I: Trajectory error per vehicle of the presented benchmark dataset. Our approach achieves an average trajectory error of 0.09 m considering all sequences and outperforms the monocular methods presented in [22] and [23].

The refinement using factors graphs improves the consistency of camera poses and triangulated object points. According to the Fresnel equations [39] the reflection intensity depends on the angle between camera and object surface. Since our method uses temporal adjacent views to triangulate object points it is less prone to reflection based point correspondences. Fig. 8 shows a comparison of the proposed method and the baseline.

B. Quantitative Evaluation

Due to the lack of suitable real world benchmark datasets for vehicle trajectory reconstruction using stereo data, we extended the monocular rendering framework presented in [22]. We apply the monocular trajectory error defined in [22] for binocular sequences by back-projecting the reconstructed object point cloud using only left cameras. The trajectory error is the average trajectory-point-mesh distance, i.e. the shortest distance of each object point to the vehicle mesh at the corresponding time step. The trajectory error is affected by background camera poses registration errors and incorrect vehicle point triangulations. Table I compares our method with two monocular reconstruction methods and the baseline described in section II-D.

V. Conclusions

This paper presents a novel approach to reconstruct three-dimensional object motion trajectories using stereo image sequences. Our method tracks objects on pixel level across the input videos. This allows us to use SfM to reconstruct different objects simultaneously. In contrast to previously published stereo 3D object trajectory reconstruction methods, our approach leverages temporal adjacent frames for object and background reconstruction. Thus, the presented method circumvents stereo camera baseline limitations. The usage of temporal adjacent views for point triangulation reduces for example the number of outliers caused by reflections. We observe that a refinement of SfM reconstructions using stereo camera projection constraints improves the accuracy of camera poses and reduces the number of incorrectly triangulated object points. Our approach builds an object model and registers camera poses w.r.t. this object model, which is not possible with previously presented stereo matching based reconstruction pipelines. We showed qualitative results on the Cityscape and the KITTI dataset due to the lack of real world stereo 3D object motion trajectory benchmark datasets with suitable ground truth data. We created a set of virtual stereo sequences and corresponding ground truth data to evaluate our method quantitatively. The dataset is publicly available on our website. In future work we will analyze robustness of the presented approach w.r.t decreasing object sizes.

References

[1] F. Engelmann, J. Stückler, and B. Leibe, "SAMP: shape and motion priors for 4d vehicle reconstruction," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017.

[2] A. Ošep, W. Mehner, M. Mathias, and B. Leibe, "Combined image- and world-space tracking in traffic scenes," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[3] C. Chang and S. Chatterjee, "Quantization error analysis in stereo vision," in *Conference Record of the Twenty-Sixth Asilomar Conference on Signals, Systems Computers*, 1992.

[4] P. Pinggera, D. Pfeiffer, U. Franke, and R. Mester, "Know your limits: Accuracy of long range stereoscopic object measurements in practice," in *European Conference on Computer Vision (ECCV)*, 2014.

[5] P. Moulon, P. Monasse, R. Marlet, and Others, "Openmvg. an open multiple view geometry library.," 2013.

[6] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[7] A. Kundu, K. M. Krishna, and C. V. Jawahar, "Realtime multibody visual slam with a smoothly moving monocular camera," in *International Conference on Computer Vision (ICCV)*, 2011.

[8] K. Lebeda, S. Hadfield, and R. Bowden, "2D or not 2D: Bridging the gap between tracking and structure from motion," in *Asian Conference on Computer Vision (ACCV)*, 2014.

[9] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[11] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[12] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine patchmatch for large displacement optical flow," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] S. Bullinger, C. Bodensteiner, and M. Arens, "Instance flow based online multiple object tracking," in *IEEE International Conference on Image Processing (ICIP)*, 2017.

[14] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, 2006.

[15] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[16] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[17] C. Wu, "Visualsfm: A visual structure from motion system," 2011.

[18] C. Sweeney, *Theia Multiview Geometry Library: Tutorial & Reference*. University of California Santa Barbara., 2014.

[19] F. Daellert, "Factor graphs and gtsam: A hands-on introduction," tech. rep., GT-RIM-CP&R-2012-002, 2012.

[20] K. E. Ozden, K. Cornelis, L. V. Eycken, and L. J. V. Gool, "Reconstructing 3d trajectories of independently moving objects using generic constraints," *Computer Vision and Image Understanding*, 2004.

[21] S. Song, M. Chandraker, and C. C. Guest, "High accuracy monocular SFM and scale correction for autonomous driving," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[22] S. Bullinger, C. Bodensteiner, M. Arens, and R. Stiefelhagen, "3d vehicle trajectory reconstruction in monocular video data using environment structure constraints," in *IEEE European Conference on Computer Vision (ECCV)*, 2018.

[23] S. Bullinger, C. Bodensteiner, and M. Arens, "Monocular 3d vehicle trajectory reconstruction using terrain shape constraints," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

[24] S. Bullinger, C. Bodensteiner, and M. Arens, "3d object trajectory reconstruction using stereo matching and instance flow based multiple object tracking," in *International Conference on Machine Vision Applications (MVA)*, 2019.

[25] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[26] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision (ACCV)*, 2010.

[27] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *European Conference on Computer Vision (ECCV)*, 2014.

[28] Z. Liang, Y. Feng, Y. Guo, H. Liu, W. Chen, L. Qiao, L. Zhou, and J. Zhang, "Learning for disparity estimation through feature constancy," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[29] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[30] S. Baker, S. Roth, D. Scharstein, M. J. Black, J. P. Lewis, and R. Szeliski, "A database and evaluation methodology for optical flow," in *IEEE International Conference on Computer Vision (ICCV)*, 2007.

[31] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

[32] J. K. Wong, "A new implementation of an algorithm for the optimal assignment problem: An improved version of munkres' algorithm," *BIT Numerical Mathematics*, vol. 19, no. 3, pp. 418–424, 1979.

[33] K. C. Gilbert and R. B. Hofstra, "Multidimensional assignment problems," *Decision Sciences*, vol. 19, no. 2, pp. 306–321, 1988.

[34] A. Frieze, "Complexity of a 3-dimensional assignment problem," *European Journal of Operational Research*, vol. 13, no. 2, pp. 161 – 164, 1983.

[35] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second ed., 2004.

[36] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[37] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *Int. J. Rob. Res.*, vol. 32, no. 11, 2013.

[38] F. Dellaert and M. Kaess, "Factor graphs for robot perception," *Foundations and Trends in Robotics*, vol. 6, no. 1-2, p. 1139, 2017.

[39] M. Born, E. Wolf, A. B. Bhatia, P. C. Clemmow, D. Gabor, A. R. Stokes, A. M. Taylor, P. A. Wayman, and W. L. Wilcock, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. Cambridge University Press, 7 ed., 1999.