Visual Odometry based on Stereo Image Sequences with RANSAC-based Outlier Rejection Scheme

Bernd Kitt, Andreas Geiger and Henning Lategahn
Institute of Measurement and Control Systems
Karlsruhe Institute of Technology
{bernd.kitt, geiger, henning.lategahn}@kit.edu

***Abstract*— A common prerequisite for many vision-based driver assistance systems is the knowledge of the vehicle's own movement. In this paper we propose a novel approach for estimating the egomotion of the vehicle from a sequence of stereo images. Our method is directly based on the trifocal geometry between image triples, thus no time expensive recovery of the 3-dimensional scene structure is needed. The only assumption we make is a known camera geometry, where the calibration may also vary over time. We employ an Iterated Sigma Point Kalman Filter in combination with a RANSAC-based outlier rejection scheme which yields robust frame-to-frame motion estimation even in dynamic environments. A high-accuracy inertial navigation system is used to evaluate our results on challenging real-world video sequences. Experiments show that our approach is clearly superior compared to other filtering techniques in terms of both, accuracy and run-time.**

I. Introduction

The estimation of the movement of a camera, especially a stereo-camera rig, is an important task in robotics and advanced driver assistance systems. It is also a prerequisite for many applications like obstacle detection, autonomous driving, *simultaneous localization and mapping (SLAM)* and many other tasks. For all of these applications, the relative orientation of the current camera frame with respect to the previous camera frame or a static reference frame is needed. Often, this localization task is performed using imprecise wheel speed sensors and *inertial measurement units (IMUs)* [13] or expensive high-accuracy IMUs. In recent years, camera systems became cheaper, more compact and the computational power even on standard PC hardware increased dramatically. This is why high resolution images can be provided at high frame rates and processed in real-time. The information given by such images suffices for precise motion estimation based on visual information [1], called *visual odometry* (e.g., Nistér et. al. [18]).
Compared to other sensors, visual odometry promises several advantages: One main advantage of visual odomentry is the high accuracy compared to wheel speed sensors. Especially in slippery terrain where wheel speed sensors often yield wrong motion estimates, visual odometry is more precise [12]. Other approaches use GPS sensors or IMUs to mitigate this effect. Drawbacks of GPS- or IMU-based approaches are the low accuracy and the high sensor costs respectively. The local drift rates given by visual odometry are mostly smaller than the drift rates given by IMUs except for expensive high-accuracy hardware which fuses GPS-measurements with inertia sensor information [13].
In this work we estimate the relative displacement between two consecutive camera positions using stereo sequences captured in urban environments. Such data is especially challenging due to the presence of independently moving objects, which violate the static world assumption. To deal with outliers a rejection step based on random sampling is proposed and evaluated. The 6 degrees of freedom (6DoF) egomotion is estimated merely from image measurements. No additional information such as odometry data or GPS information is used as in [1] or [7]. Furthermore we do not restrict the degrees of freedom by using a special (nonholonomic) motion model, making our approach widely applicable.

A. Related Work

In recent years many algorithms for visual odometry have been developed, which can roughly be devised into two categories, namely methods using monoscopic cameras (e.g., [25]) or methods using stereo rigs. These approaches can be further separated into methods which either use feature matching (e.g., [13], [23], [24]) between consecutive images or feature tracking over a sequence of images (e.g., [7], [2], [14]). If a calibrated multi-ocular camera setup is available, the 3-dimensional scene can be reconstructed via triangulation. Based on the point clouds of the static scene in two consecutive images, the iterated closest point (ICP) algorithm is often used for egomotion estimation as described in [17]. Monocular cameras mainly require tracking image features (e.g. corners) over a certain number of images. Using these feature tracks, also the scene structure can be computed using structure from motion [18]. In most cases, the multi-ocular algorithms yield better performances than monocular approaches [4]. Additionaly, if multi-camera approaches are used, the scale ambiguity present in the monocular case is eliminated [1]. Further approaches combine visual odometry with other sensors to increase the accuracy of the results and reduce drift, a problem inherent to all incremental positioning methods. While Dornhege et. al. [7] additionally make use of an IMU, Agrawal et. al. (e.g., [1], [3], [2]) use GPS and wheel encoders, thus fusing a wide variety of sensor types for optimal performance. Clearly, the use of GPS information limits drift due to the system's global nature. Furthermore, approaches making assumptions about the observer's motion have been developed. For example, Scaramuzza et. al. [19]

use nonholonomic constraints of wheeled vehicles in order to reduce the motion model's parameter space.
Compared to the method proposed by [2], where a visual odometry algorithm based on bundle adjustment [8] is combined with IMU and GPS data, the focus of our approach lies on estimating the motion solely based on visual inputs. Because of the higher computational complexity of bundle adjustment compared to frame-to-frame motion estimation we employ the latter method. Our visual odometry algorithm is briefly summarized in the next section.

B. System Overview

We propose an algorithm for egomotion estimation in all six degrees of freedom using a fully calibrated stereo-camera rig, i.e. the intrinsic as well as the extrinsic calibration parameters are given. It is noteworthy, that the calibration is not assumed to be fixed over the sequence, such that the proposed approach can also be applied to active stereo-camera rigs.
In a first step, we extract and match corner-like image features between two consecutive stereo image pairs. Based on these feature correspondences, the egomotion of the vehicle is estimated using the trifocal tensor which relates features between three images of the same static scene. A similar approach is introduced by Yu et. al. [26] using a monocular camera. We extend this approach to stereo camera rigs to gain robustness and avoid scale ambiguity. Furthermore we use an *Iterated Sigma Point Kalman Filter (ISPKF)* to cope with the non-linearities in the measurement equation. Outliers are detected via a *random sample consensus (RANSAC)* based outlier rejection scheme [19]. This procedure guarantees, that outliers which stem from false matches or features located on independently moving objects are rejected prior to the final motion estimation step. Thus our algorithm can also be deployed in dynamic environments. We do not require tracked image features over multiple frames. Instead feature matches between consecutive stereo image frames are sufficient, hence not requiring any reinitialization procedure like most tracking approaches [26].

The remainder of this paper is organized as follows: Section II describes our camera model and the relations between point correspondences in image triples. In Section III the proposed approach is introduced. Experimental results of the proposed method using image sequences captured in urban environments are given in Section IV. We close the paper with a short conclusion and an outlook on future work.

II. Geometry of Image Triples

A. Camera Model

This section describes the camera model used in the proposed approach.
Let $\boldsymbol{K}$ be the 3×3 calibration matrix which encapsulates the intrinsic parameters of the camera. The mapping between the camera coordinates $\boldsymbol{X}_C$ and the homogeneous image coordinates $\widetilde{\boldsymbol{x}}$ can be described as follows [11]:

$$\widetilde{\boldsymbol{x}} = (u, v, w)^\mathsf{T} = \boldsymbol{K} \cdot \boldsymbol{X}_C \tag{1}$$

Here $\widetilde{(.)}$ denotes homogeneous notation. In general, the camera coordinate frame and the world coordinate frame are not aligned, but the two coordinate frames are related via a translation vector $\boldsymbol{t}$ and a rotation matrix $\boldsymbol{R}$, the extrinsic calibration of the camera. Given a 3-dimensional point $\boldsymbol{X}_W = (X_W, Y_W, Z_W)^\mathsf{T}$ in the world reference frame, the corresponding point $\boldsymbol{X}_C = (X_C, Y_C, Z_C)^\mathsf{T}$ in the camera coordinate frame is computed via:

$$\boldsymbol{X}_C = \boldsymbol{R} \cdot \boldsymbol{X}_W + \boldsymbol{t} \tag{2}$$

Combining equations (1) and (2) the mapping of a 3d object point onto the image plane is described as

$$\widetilde{\boldsymbol{x}} = \boldsymbol{P} \cdot \widetilde{\boldsymbol{X}}_W \tag{3}$$

where $\boldsymbol{P} = \boldsymbol{K} \cdot \left[\boldsymbol{R}|\boldsymbol{t}\right]$ is a 3×4 projection matrix [11].

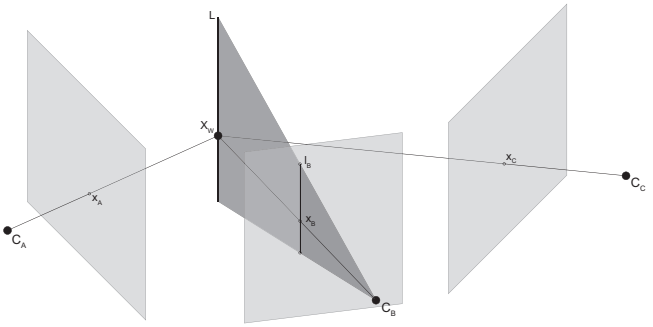


Fig. 1: Relationship between corresponding points in three images. This figure depicts the point-line-point transfer which maps a given point correspondence $\boldsymbol{x}_A \leftrightarrow \boldsymbol{x}_B$ into the third image, assuming that the trifocal tensor $\mathcal{T}$ between the three images is known.

B. Relationship between three Images

The $3 \times 3 \times 3$ *trifocal tensor* $\mathcal{T}$ describes the relationship between three images of the same static scene. It encapsulates the projective geometry between the different viewpoints and is independent from the structure of the scene.
Knowing the projection matrices of the three cameras, i.e. $\boldsymbol{P}_A = \boldsymbol{K}_A \cdot \left[\boldsymbol{R}_A|\boldsymbol{t}_A\right]$, $\boldsymbol{P}_B = \boldsymbol{K}_B \cdot \left[\boldsymbol{R}_B|\boldsymbol{t}_B\right]$ and $\boldsymbol{P}_C = \boldsymbol{K}_C \cdot \left[\boldsymbol{R}_C|\boldsymbol{t}_C\right]$, the entries of the trifocal tensor are given by

$$\mathcal{T}_i^{qr} = (-1)^{i+1} \cdot \det \begin{pmatrix} \sim \boldsymbol{a}^i \\ \boldsymbol{b}^q \\ \boldsymbol{c}^r \end{pmatrix}, \tag{4}$$

where $\sim \boldsymbol{a}^i$ denotes matrix $\boldsymbol{P}_A$ without row i and $\boldsymbol{b}^q$ and $\boldsymbol{c}^r$ represent the q-th row of $\boldsymbol{P}_B$ and the r-th row of $\boldsymbol{P}_C$ respectively [11].
Here, we make use of the trifocal tensor's ability to map two corresponding feature points $\boldsymbol{x}_A \leftrightarrow \boldsymbol{x}_B$ in images A and B into image C. Figure 1 illustrates this procedure graphically: An arbitrary image line $\boldsymbol{l}_B$ through point $\boldsymbol{x}_B$ is projected into

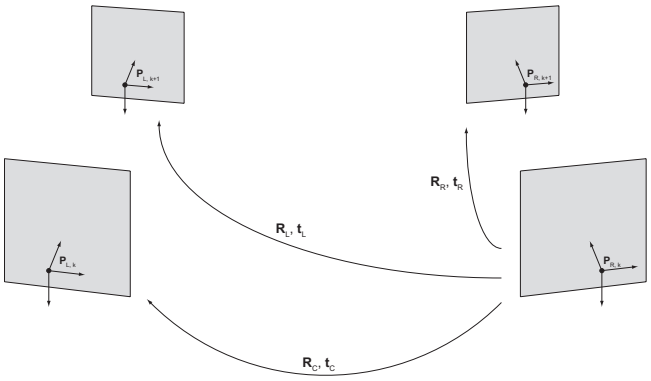


Fig. 2: This figure depicts the configuration of the cameras of a stereo at two consecutive time steps, including the geometric relations between the images.

3d space.
Given both, the line $\boldsymbol{l}_B$ and the trifocal tensor $\mathcal{T}$, the point $\boldsymbol{x}_C$ in image C which corresponds to the point correspondence $\boldsymbol{x}_A \leftrightarrow \boldsymbol{x}_B$ in the first two images is given by

$$\boldsymbol{x}_C^k = \boldsymbol{x}_A^k \cdot \boldsymbol{l}_{B,j} \cdot \mathcal{T}_i^{jk}. \tag{5}$$

The following section details the application of this relationship to egomotion estimation.

III. Kalman Filter based Visual Odometry

Estimating the camera motion at each time step is performed using two consecutive stereo image pairs. The motion parameters are integrated temporally by means of an Iterated Sigma Point Kalman Filter.
Figure 2 shows the configuration of a stereo rig at two consecutive steps in time. Depicted are the image planes, the camera coordinate frames and the orientations of the cameras with respect to the previous right camera. While the pose of the previous left camera is given by the known extrinsic calibration $\{\boldsymbol{R}_C, \boldsymbol{t}_C\}$ of the stereo rig, the parameters $\{\boldsymbol{R}_R, \boldsymbol{t}_R\}$ and $\{\boldsymbol{R}_L, \boldsymbol{t}_L\}$ are defined by the egomotion and by a combination of extrinsic camera calibration and egomotion respectively.

A. Motion Parameterization

To parameterize motion, i.e. the spatial orientation of the camera coordinate frame related to the world reference frame, we use the translation vector $\boldsymbol{t} = (t_X, t_Y, t_Z)^\mathsf{T}$ and the rotation matrix $\boldsymbol{R}(\Theta, \Phi, \Psi)$. The rotation of the camera is parameterized in Euler angles, as a concatenation of rotations around the three axis of the world reference frame[1]. In this work we define the rotation as follows:

$$\boldsymbol{R}(\Theta, \Phi, \Psi) = \boldsymbol{R}_Z(\Theta) \cdot \boldsymbol{R}_X(\Phi) \cdot \boldsymbol{R}_Y(\Psi) \tag{6}$$

The spatial motion, represented by $\boldsymbol{t}$ and $\boldsymbol{R}$, can be computed for every time step if the egomotion $(V_X, V_Y, V_Z, \omega_X, \omega_Y, \omega_Z)$ of the stereo rig and the time difference ΔT between two consecutive frames is known.

[1]The world reference frame is shifted in every time step. Hence it always aligns with the camera coordinate frame of the previous right image.

Here V_i and ω_i denote translational and rotational velocities, respectively. Given the egomotion and the time difference the translation and rotation are thus given by:

$$\boldsymbol{t} = (V_X \cdot \Delta T, V_Y \cdot \Delta T, V_Z \cdot \Delta T)^\mathsf{T} \tag{7}$$

$$\boldsymbol{R}(\omega_Z \cdot \Delta T, \omega_X \cdot \Delta T, \omega_Y \cdot \Delta T) \tag{8}$$

B. Trifocal Constraints for Visual Odometry

Figure 2 shows that the projection matrices of the four cameras can be computed if the intrinsic and extrinsic calibration of the cameras and the egomotion is known. Without loss of generality, the camera coordinate frame of the previous right camera is aligned with the world reference frame $\boldsymbol{P}_{R,k} = \boldsymbol{K}_R \cdot \big[\boldsymbol{I}|\boldsymbol{0}\big]$. The remaining projection matrices are defined as follows:

$$\boldsymbol{P}_{L,k} = \boldsymbol{K}_L \cdot \big[\boldsymbol{R}_C|\boldsymbol{t}_C\big] \tag{9}$$

$$\boldsymbol{P}_{R,k+1} = \boldsymbol{K}_R \cdot \big[\boldsymbol{R}_R|\boldsymbol{t}_R\big] \tag{10}$$

$$\boldsymbol{P}_{L,k+1} = \boldsymbol{K}_L \cdot \big[\boldsymbol{R}_L|\boldsymbol{t}_L\big] \tag{11}$$

Here k describes the discrete time step at which the images were captured. Using the projection matrices parameterized as above, two trifocal tensors can be determined. One which relates the previous image pair to the current right frame and one which relates the previous image pair to the current left frame. By equation (4) we have:

$$\mathcal{T}_R = \mathcal{T}(\boldsymbol{K}_R, \boldsymbol{K}_L, \boldsymbol{R}_C, \boldsymbol{t}_C, \boldsymbol{R}_R, \boldsymbol{t}_R, \Delta T) \tag{12}$$

$$\mathcal{T}_L = \mathcal{T}(\boldsymbol{K}_R, \boldsymbol{K}_L, \boldsymbol{R}_C, \boldsymbol{t}_C, \boldsymbol{R}_L, \boldsymbol{t}_L, \Delta T) \tag{13}$$

These two trifocal tensors depend on the motion of the stereo rig and the camera calibration. Using the trifocal tensors, a non-linear mapping of the point correspondence $\boldsymbol{x}_{R,k} \leftrightarrow \boldsymbol{x}_{L,k}$ into the current images via $\boldsymbol{x}_{R,k+1} = h_R(\mathcal{T}_R, \boldsymbol{x}_{R,k}, \boldsymbol{x}_{L,k})$ and $\boldsymbol{x}_{L,k+1} = h_L(\mathcal{T}_L, \boldsymbol{x}_{R,k}, \boldsymbol{x}_{L,k})$ is defined.
Different kinds of feature detectors and descriptors are possible: Popular choices include Harris et. al. [10], Shi et. al. [20] or local image descriptors like the SIFT descriptor proposed by Lowe et. al. [16] or the SURF descriptor proposed by Bay et. al. [5]. Those descriptors are highly distinctive and thus allow robust matchings.

C. Bucketing

In a first step, we detect and match image features in both stereo pairs. Afterwards a subset is chosen by means of bucketing [27]: The image is divided into several non-overlapping rectangles (see figure 3). In every bucket we keep a maximal number of feature points. This benefits in several ways. First, the smaller number of features reduces the computational complexity of the algorithm which is an important prerequisite for real time applications. Second, this technique guarantees that the used image features are well distributed along the z-axis, i.e. the roll-axis of the vehicle. This turns out to be important for a good estimation of the linear and angluar velocities. The distribution of image

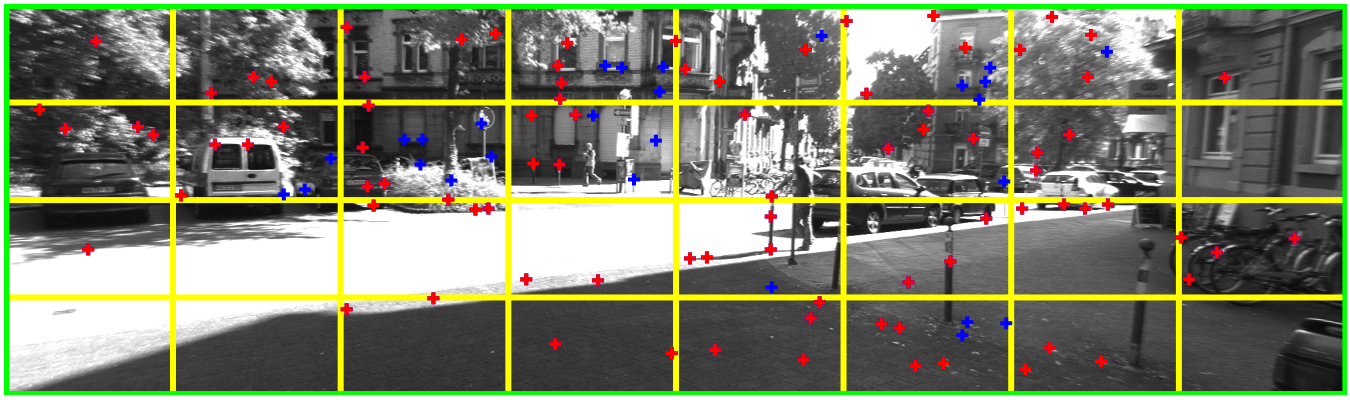


Fig. 3: This figure depicts the results of our bucketing mechanism: The green rectangle defines the region of interest in which features are selected, the yellow lines depict individual buckets. All crosses represent matches found in both stereo image pairs, red crosses denote the selected, blue crosses denote the rejected features.

features along the z-axis ensures that far as well as near features are used for the estimation process. This results in a precise estimation of the overall egomotion of the vehicle. Third, the used image features are uniformly distributed over the whole image. This benefits twice: In dynamic scenes where most of the detected features lie on independently moving objects, our technique guarantees that not all image features fall on independently moving objects but also on the static background. Second, the bucketing reduces the drift rates of the approach. In our experiments with simulated data we observed, that high drift rates follow from biased scene points. This effect is mitigated by the use of bucketing.

D. RANSAC based outlier rejection

The remaining feature points located on independently moving objects are rejected using RANSAC based outlier rejection: We randomly choose subsets of feature correspondences and estimate the egomotion based on this subsets, whereas the number of used subsets is given by

$$n = \frac{log\left(1-p\right)}{log\left(1-\left(1-\epsilon\right)^{s}\right)}. \tag{14}$$

Here s is the minimum number of data points needed for estimation, p is the probability that at least one sample contains inliers solely and ϵ defines the assumed percentage of outliers in the data set [6]. Because of the low number of data points ($s = 3$) necessary for motion estimation, the number of samples is low even with a serious number of outliers. After the Kalman Filter converges, we compute all inliers using the Euclidean reprojection error. A feature is considered as an inlier, if the Euclidean reprojection error is lower than a certain threshold. A final estimation step with all inliers of the best sample is performed to give the final egomotion estimate. The proposed bucketing technique combined with the RANSAC based outlier rejection scheme yields a robust egomotion estimation even in the presence of independently moving objects.

To integrate information about the dynamic behaviour of the ego-vehicle, a Kalman Filter is used for filtering, as outlined in the following section.

E. Kalman Filtering

The Kalman Filter is a two-step estimator making use of a prediction step and an update step. It is used to estimate the current state of a dynamic system, which is assumed to be disturbed by zero-mean white noise. To estimate the instantaneous state, disturbed measurements are used. It is assumed, that the measurements and the state are related via a linear transform. It is also assumed that the given measurements are disturbed by zero-mean white noise [9].

In our case, the relations between the instantaneous state $\boldsymbol{y} = \left(V_X, V_Y, V_Z, \omega_X, \omega_Y, \omega_Z\right)^{\mathsf{T}}$ and the measurements, i.e. the relations between the egomotion and the feature positions in the current frames, given by $\boldsymbol{x}_{R,k+1} = h_R\left(\mathcal{T}_R, \boldsymbol{x}_{R,k}, \boldsymbol{x}_{L,k}\right)$ and $\boldsymbol{x}_{L,k+1} = h_L\left(\mathcal{T}_L, \boldsymbol{x}_{R,k}, \boldsymbol{x}_{L,k}\right)$ respectively, are non-linear. The discrete-time space filter equations are given by

$$\boldsymbol{y}_{k+1} = f\left(\boldsymbol{y}_k\right) + \boldsymbol{w}_k \tag{15}$$

$$\boldsymbol{z}_{k+1} = h\left(\boldsymbol{y}_{k+1}\right) + \boldsymbol{v}_{k+1} \tag{16}$$

where $\boldsymbol{y}_k$ is the state of the system at time step k, $f\left(.\right)$ is the non-linear system equation, $h\left(.\right)$ is the non-linear measurement equation described above. $\boldsymbol{z}_{k+1} = \left[u_{R,k+1,1}, \ldots, v_{L,k+1,N}\right]^{\mathsf{T}}$ denotes the $4N$-dimensional measurement vector and $\boldsymbol{w}_k \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{Q}_k\right)$ and $\boldsymbol{v}_{k+1} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{R}_{k+1}\right)$ are the system noise and the measurement noise respectively, which are assumed to be uncorrelated. Here the 6×6 matrix $\boldsymbol{Q}_k$ and the $4N \times 4N$ diagonal matrix $\boldsymbol{R}_{k+1}$ denote the state and measurement error covariance matrices respectively [21], and N denotes the number of feature correspondences used for filtering.

To use Kalman Filters for non-linear problems, linearization around the current state is often performed using a first order Taylor-approximation. This yields the *Extended Kalman Filter (EKF)*. To reduce the approximation error caused by Taylor approximation, the update step is often iterated. In such cases $h\left(.\right)$ is linearized around the estimated state of the current iteration. Repeating this step yields

the well-known *Iterated Extended Kalman Filter (IEKF)*. In general, the iteration process is abandoned if any predefined termination criteria is fulfilled. In our case of highly non-linear equations the results of *Extended Kalman Filters* are mostly poor. The reason for this is that the used Taylor-approximation is only a first order approximation. A better choice in such cases is the usage of Kalman Filters based on the *Unscented Transform (UT)* [22]. Such filters propagate mean and covariance based on sigma points. Their estimates are mostly better than estimates of *Extended Kalman Filters* because the unscented transform incorporates information about higher order moments in the estimation process. Examples for filters propagating mean and covariance based on sigma points are the *Unscented Kalman Filter (UKF)* [15] or the *Iterated Sigma Point Kalman Filter (ISPKF)* [21]. See [22], [21] for more details on Kalman Filtering techniques.

In the prediction step of the proposed algorithm we assume constant velocity between consecutive time steps, so the system equation simplifies to $\boldsymbol{y}_{k+1} = \boldsymbol{y}_k + \boldsymbol{w}_k$. This assumption is nearly fulfilled if the camera provides images with a fairly high frame-rate. Even if this assumption is violated (e.g. in the case of acceleration, deceleration or turns), the update step guarantees reliable motion estimation. In our case, the measurements are the features in the current images. For every feature correspondence in the previous image pair the expected coordinates in the current images are predicted. Given the measured point correspondences, the system equation and the measurement equation, Kalman Filtering can be performed.

Besides the reduction in linearization error, the ISPKF has another benefit compared to EKF based filtering. In our experiments, the convergence of the ISPKF is approximately 60 times faster than the convergence of the IEKF, without the need for analytical derivatives. In average, the ISPKF converges in three iterations, whereas the IEKF needs about 200 iteration for convergence to the same solution. A detailed analysis of the convergence between those filtering techniques for different termination thresholds is given in section IV-B.

IV. Experimental Results

For our experiments we used simulated as well as real data sets. The real data sets were captured from our experimental vehicle, equipped with a stereo camera rig and a high accuracy inertial navigation system which combines inertial measurements with a GPS-receiver and wheel speed sensors for measuring motion, pose and orientation of the vehicle. Therefore, the INS yields a good reference for the linear motion along the roll-axis and the yaw-rate of the car. In the following, the INS trajectories are used as ground truth for our experiments. As features we used Harris corners in combination with block matching on the image derivatives, for efficiency reasons. However, also other features can be equally employed: With similar results, we also tried SURF features [5]. Because of the average linear speed of 7m/s and a maximum speed of 17m/s in our real world experiments, scale invariant features benefit especially in those situations. Compared to the average linear movements of about 1m/s reported by Agrawal et. al. (e.g., [1]) the speed in our experiments is significantly higher.

	ISPKF	**IEKF**	**UKF**	**EKF**
positioning error	33.5	34.3	33.5	105.9
standard deviation	15.8	15.6	15.9	31.8

TABLE I: Average positioning error and standard deviation (in meters) at the end of the sequences occuring from drift using different simulated sequences each over a length of 2000m.

A. Comparison with other Filtering Techniques

Because of the non-linearities in the measurement equation, we compared a variety of other filtering techniques in our approach: We evaluated the *Unscented Kalman Filter (UKF)* proposed by [15], the *Extended Kalman Filter (EKF)* and the *Iterated Extended Kalman Filter (IEKF)*. The evaluation was perfomed on different simulated data sets. Each of them consisting of 2000 frames and 40 scene points without outliers, which are investigated in the next section. The average linear motion used for this experiments was 10m/s. The measurements were disturbed by zero-mean Gaussion noise with a standard deviation of 0.7 pixels. The results of *ISPKF*, *IEKF* and *UKF* are similar to the ground truth. However, the result of the *EKF* is considerably worse, because this type of filter cannot cope well with the non-linear measurement equation. A detailed analysis between the different filtering techniques is shown in table I. While the ISPKF, IEKF and UKF perform similar with respect to drift errors, we prefer using the ISPKF due to considerably lower run-times, which are further analyzed in section IV-B.

B. Convergence Analysis

For performance analysis, we compared the number of iterations for the ISPKF and the IEKF using different termination thresholds. Therefore, we used different simulated data sets, each consisting of 1000 frames. In each frame 40 scene points were used for egomotion estimation. The measurements were disturbed by Gaussian noise with no outliers. The number of iterations until convergence is nearly independent for the ISPKF (threshold[2]: $10^{-1} \rightarrow 3$ iterations, threshold: $10^{-5} \rightarrow 4$ iterations), for the IEKF the number of iterations increases dramatically (threshold: $10^{-1} \rightarrow 4$ iterations, threshold: $10^{-5} \rightarrow 464$ iterations) when reducing the threshold.

C. Analysis of the Outlier Rejection Scheme

To analyze the benefits of our outlier rejection scheme, we created different simulated data sets, each with 20% outliers. Using these data sets we performed ISPKF based

[2]The threshold means, that every parameter in the estimation vector change less than this threshold between two iterations. The unit is m/s for the linear velocities and °/s for the angular velocities, respectively.

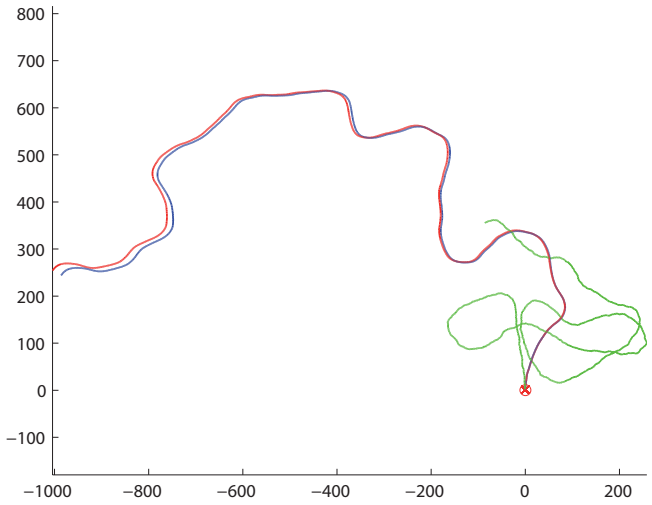


Fig. 4: This figure depicts the trajectories of the approach with and without the proposed outlier rejection. The trajectory with activated outlier rejection (blue) is very similar to the ground truth (red). Without outlier rejection, the trajectory (green) differs significantly from the ground truth.

egomotion estimation with and without outlier rejection. The remaining parameters are the same for both filters, the termination threshold for the iteration was set to 10^{-3} (in m/s and °/s respectively). The results for one of the data sets with an average linear motion of $10m/s$ consisting of 2000 frames is shown in figure 4. The positioning error at the end of the trajectory with activated outlier rejection is $24.29m$, corresponding to approximately 1.3% of the travelled distance. In contrast, the trajectory without outlier rejection differs significantly from the ground truth.

D. Runtime Evaluation

Since we focus on real-time applications, we evaluated the possible frame rates for the egomotion estimation. Therefore we used a simulated dataset consisting of 1000 frames. The average number of frames per second (fps) of the egomotion algorithm depending on the number of used image features is given in table II. As threshold for the termination criterion we used 0.001 for all parameters of the motion estimate.

features	10	20	30	40	50	60	70	80
fps	27	20	15	10	8	6	5	4

TABLE II: Average number of frames per second which can be processed by the proposed algorithm depending on the number of used image features.

E. Real-World Experiments

For our real-world experiments we captured different image sequences in urban environments with high traffic. An example image (at a resolution of 1344×391 pixels) is depicted in figure 3. The stereo camera rig was mounted on top of the vehicle with a base line of $0.7m$. The results for three challenging data sets with different length and speed can be seen in figure 5. Especially the parking sequence shown in figure 5a is challenging because of the $360°$ turn during the parking maneuver. As depicted, the trajectory before the parking procedure is closely aligned with the trajectory after the parking procedure. The estimated trajectories are similar to the trajectories given by the INS. The occuring drift which is an inherent drawback of all local approaches is comparatively small.

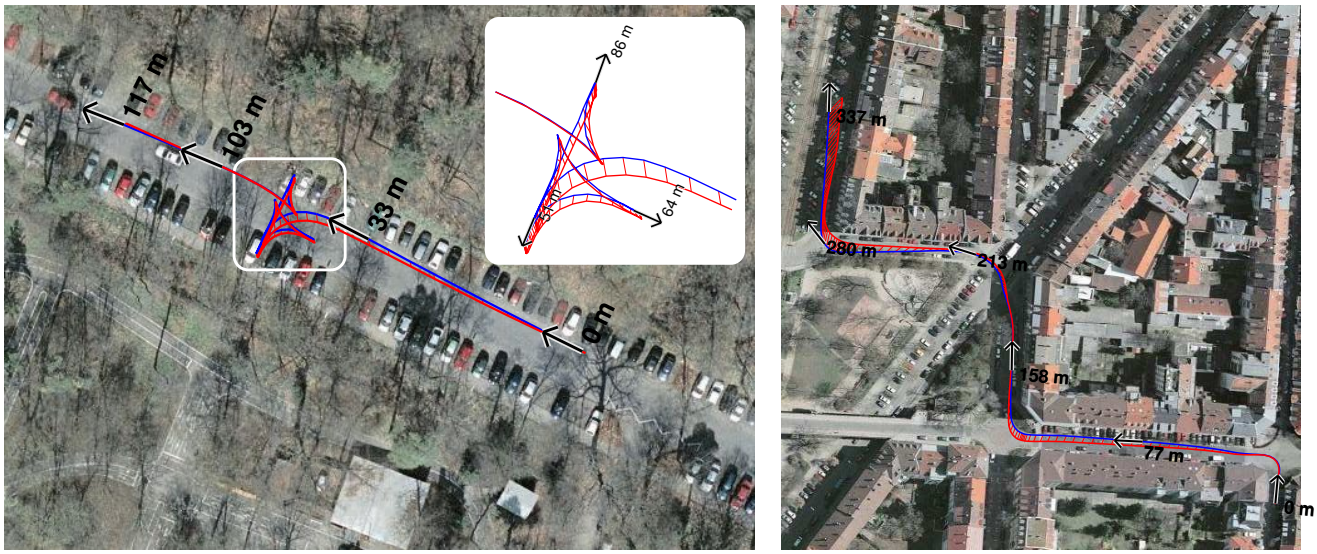
V. Conclusions and Future Works

In this paper we presented an approach for estimating the 6DoF egomotion of a stereo camera rig based on corresponding image features. The proposed approach is based on the trifocal geometry between image triples. Therefore no reconstruction of the 3d object points is required. The algorithm neither needs a rectified stereo-camera rig nor a time consuming preprocessing rectification of the captured images. Merely, the intrinsic and extrinsic calibration of the cameras need to be known.
The experimental results show, that the proposed algorithm yields a good estimate of the egomotion in urban environments compared to the high accuracy INS. Because of the iteration in the update step of the Kalman Filter, effects of non-linearity are dealt with in a principal way during the estimation process.
The main novelty of the proposed approach is the usage of the trifocal tensor between image triplets in combination with a RANSAC based outlier rejection scheme. This allows motion estimation based on measurements in the images without recovering the 3d scene structure. Recovering of the scene structure based on the disparity is – especially for far scene points – unreliable because depth accuracy decreases with distance. The bucketing technique yields a good distribution of feature points over the image, guaranteeing that the majority of the features lie on the static background of the scene and not on independently moving objects on the one hand. On the other hand a uniform distribution of the features along the roll-axis is present. This results in a precise estimation of the linear and angular velocities. The RANSAC based outlier rejection scheme sorts out remaining features on independently moving objects prior to the final estimation process. Taken together, this yields an accurate egomotion estimation, even in dynamic environments.

To improve the proposed approach we are working on a better model of the system, which accounts for the dynamic behaviour of the vehicle more precisely than the constant velocity assumption in the presented approach. We also try to estimate the mechanical parameters involved in the dynamic behaviour of the vehicle jointly with the motion parameters in our future research.

Acknowledgements

The authors gratefully acknowledge the contribution of the German collaborative research center on Cognitive Automobiles (SFB/Tr28), granted by Deutsche Forschungsgemeinschaft.



(a) Driven path for a sequence with 1200 frames.

(b) Driven path for a sequence with 1200 frames.

Fig. 5: This figure depicts the results of the proposed egomotion estimation (blue), compared to the trajectory given by the inertial-measurement-unit (red) for different challenging sequences in urban environments (image source: GoogleEarth).

REFERENCES

[1] M. Agrawal and K. Konolige, "Real-time localization in outdoor environments using stereo vision and inexpensive gps," in *Proceedings of the 18th International Conference on Pattern Recognition*, 2006, pp. 1063 – 10 068.

[2] ——, "Rough terrain visual odometry," in *Proceedings of the International Conference on Advanced Robotics*, August 2007.

[3] M. Agrawal, K. Konolige, and R. C. Bolles, "Localization and mapping for autonomous navigation in outdoor terrains: A stereo vision approach," in *Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision*, 2007.

[4] H. Badino, "A robust approach for ego-motion estimation using a mobile stereo platform," in *First International Workshop on Complex Motion*, 2004, pp. 198 – 208.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346 – 359, June 2008.

[6] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel, "1-point ransac for ekf-based structure from motion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2009.

[7] C. Dornhege and A. Kleiner, "Visual odometry for tracked vehicles," in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*, 2006.

[8] C. Engels, H. Stewénius, and D. Nistér, "Bundle adjustment rules," in *Photogrammetric Computer Vision*, September 2006.

[9] M. S. Grewal and A. P. Andrews, *Kalman Filtering Theory and Practice Using MATLAB*, third edition ed. Wiley, 2008.

[10] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Converence*, 1988, pp. 147 – 151.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in computer vision*, second edition ed. Cambridge University Press, 2008.

[12] D. M. Helmick, Y. Cheng, D. S. Clouse, L. H. Matthies, and S. I. Roumeliotis, "Path following using visual odometry for a mars rover in high-slip environments," in *Proceedings of the IEEE Aerospace Conference*, vol. 2, March 2004, pp. 772 – 789.

[13] A. Howard, "Real-time stereo visual odometry for autonomous ground vehicles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, September 2008, pp. 3946 – 3952.

[14] A. E. Johnson, S. B. Goldberg, Y. Cheng, and L. H. Matthies, "Robust and efficient stereo feature tracking for visual odometry," in *IEEE International Conference on Robotics and Automation*, May 2008, pp. 39 – 46.

[15] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," in *Proceedings of the IEEE*, vol. 92, no. 3, March 2004, pp. 401 – 422.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91 – 110, 2004.

[17] A. Milella and R. Siegwart, "Stereo-based ego-motion estimation using pixel-tracking and iterative closest point," in *Proceedings of the Fourth IEEE International Conference on Computer Vision Systems*, 2006.

[18] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," in *IEEE Computer Society Conference Computer Vision and Pattern Recognition*, vol. 1, 2004, pp. 652 – 659.

[19] D. Scaramuzza, F. Fraundorfer, and R. Siegwart, "Real-time monocular visual odometry for on-road vehicles with 1-point ransac," in *Proceedings of the IEEE Conference on Robotics and Automation*, May 2009.

[20] J. Shi and C. Tomasi, "Good features to track," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 1994, pp. 593 – 600.

[21] G. Sibley, G. Sukhatme, and L. Matthies, "The iterated sigma point kalman filter with applications to long range stereo," in *Proceedings of Robotics: Science and Systems*, August 2006.

[22] D. Simon, *Optimal State Estimation*, first edition ed. Wiley, 2006.

[23] A. Talukder, S. Goldberg, L. Matthies, and A. Ansar, "Real-time detection of moving objects in a dynamic scene from moving robotic vehicles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, October 2003, pp. 1308 – 1313.

[24] A. Talukder and L. Matthies, "Real-time detection of moving objects from moving vehicles using dense stereo and optical flow," in *IEEE International Conference on Intelligent Robots and Systems*, vol. 4, September 2004, pp. 3718 – 3725.

[25] K. Yamaguchi, T. Kato, and Y. Ninomiya, "Vehicle ego-motion estimation and moving object detection using a monocular camera," in *Proceedings of the 18th International Conference on Pattern Recognition*, 2006, pp. 610 – 613.

[26] Y. K. Yu, K. H. Wong, M. M. Y. Chang, and S. H. Or, "Recursive camera-motion estimation with the trifocal tensor," *IEEE Transactions on Systems, Man and Cybernetics – Part B*, vol. 36, no. 5, pp. 1081 – 1090, October 2006.

[27] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Artifical Intelligence*, vol. 78, no. 1 – 2, pp. 87 – 119, 1995.