COOPERATIVE IMAGE ORIENTATION CONSIDERING DYNAMIC OBJECTS

P. Trusheim*, M. Mehltretter, F. Rottensteiner, C. Heipke

Institute of Photogrammetry and GeoInformation, Leibniz University Hannover, Germany
(trusheim, mehltretter, rottensteiner, heipke)@ipi.uni-hannover.de

Commission I, WG I/9

KEY WORDS: Image Orientation, Dynamic Scene, Bundle Adjustment, Cooperative Localisation

ABSTRACT:

In the context of image orientation, it is commonly assumed that the environment is completely static. This is why dynamic elements are typically filtered out using robust estimation procedures. Especially in urban areas, however, many such dynamic elements are present in the environment, which leads to a noticeable amount of errors that have to be detected via robust adjustment. This problem is even more evident in the case of cooperative image orientation using dynamic objects as ground control points (GCPs), because such dynamic objects carry the relevant information. One way to deal with this challenge is to detect these dynamic objects prior to the adjustment and to process the related image points separately. To do so, a novel methodology to distinguish dynamic and static image points in stereoscopic image sequences is introduced in this paper, using a neural network for the detection of potentially dynamic objects and additional checks via forward intersection. To investigate the effects of the consideration of dynamic points in the adjustment, an image sequence of an inner-city traffic scenario is used; image orientation, as well as the 3D coordinates of tie points, are calculated via a robust bundle adjustment. It is shown that compared to a solution without considering dynamic points, errors in the tie points are significantly reduced, while the median of the precision of all 3D coordinates of the tie points is improved.

1. INTRODUCTION

Precise and reliable positioning is one of the main pre-requisites for automated driving. Densely built-up areas, in particular, still present challenges for classic positioning methods such as those offered by global navigation satellite systems (GNSSs). For this reason, additional sensors are commonly adopted to improve positioning and to detect possible errors (Garcia-Fernandez and Schön, 2019). Besides laser scanners, passive optical sensors, such as RGB cameras, are increasingly used for this purpose, which have the advantage of relatively low cost. In general, positioning using cameras is not only relevant for automated driving-related applications in urban areas (Cavegn et al., 2016; Cavegn, 2020), but also in drone navigation (Stoven-Dubois et al., 2018) and robotics (Zou et al., 2019).

Apart from the support provided by additional sensors, the cooperation of several vehicles can also be employed to improve positioning. Cameras allow to recognise moving cooperating participants as dynamic ground control points (GCPs)[1]. Dynamic GCPs are moving traffic participants that communicate their position information to other traffic participants, which can then use this information to improve their own positioning (Trusheim et al., 2021). This is especially beneficial if the own positioning ability is weak or does not exist at all, for example, in the case of poor or no GNSS signal (Molina et al., 2017; Stoven-Dubois et al., 2018; Trusheim and Heipke, 2020).

In most cases, a bundle block adjustment or a graph-based SLAM procedure is used to determine the 6 degrees of freedom (DoF) orientation of each image. However, as these algorithms involve the assumption of a static environment, complications arise, especially in the case of traffic scenarios in which several dynamic objects are involved. In addition, in cooperative sensing dynamic objects can cover large parts of the image, especially in a convoy formation (Trusheim et al., 2021). While a robust adjustment can help to find such errors, this solution may fail if the majority of tie points are dynamic. Therefore, more and more approaches use selection functions that categorise the points into static and dynamic ones, using geometric criteria as well as neural networks, before performing an adjustment (Bescos et al., 2018, 2021; Zhao et al., 2021).

In this paper, an approach is presented to identify dynamic parts of a scene and thus dynamic points in synchronised stereoscopic image sequences showing urban traffic. First, feature extraction is performed, and in parallel a neural network is employed to extract dynamic objects; the two results are then superimposed to distinguish between points associated with potentially dynamic objects and the static environment. Based on the tie points in the static environment, initial image orientations are calculated for every epoch, i.e., every frame of the image sequences. The 3D coordinates of potentially dynamic points are then calculated per epoch via forward intersection and are subsequently analysed for movement. Points that are found to be static (e.g. points on parked cars) are added to the point cloud of the static environment for a final bundle adjustment.

The main contribution of this paper is a novel method to separate dynamic and static image points of a stereoscopic image sequence showing ego-motion. The separation is done using a Convolutional Neural Network (CNN) to detect potentially dynamic image regions, which are further checked for position stability. Furthermore, we investigate the effects of dynamic points on the image orientation results computed via bundle adjustment with dynamic GCPs. For this purpose, three variants have been evaluated: (a) all feature points without any preselection are used as tie points, (b) only the feature points in the static environment are used as tie points, (c) feature points

* Corresponding author

[1] We use the term "dynamic" instead of "kinematic", which is used in (Molina et al., 2017), because of the possibility to also include measurements of inertial measurements units (IMUs) in our model.

located on potentially dynamic objects but found to be static are added to the points of variant (b) and are also used as tie points.

The remainder of this work is structured as follows: In Section 2 an overview of related literature is given. This is followed by a detailed description of the methodology in Section 3. The data set and the experimental setup are presented in Section 4. The results of the experiments are shown and discussed in Section 5. Finally, the work is concluded and an outlook on promising directions for future work is given in Section 6.

2. RELATED WORK

Optical sensors can provide important information for the task of positioning in areas that are challenging for GNSS sensors. For instance, it is shown in (Cavegn et al., 2016) and (Cavegn, 2020) that the errors of checkpoints of approximately $40\ cm$ achieved by a GNSS/IMU sensor combination could be improved by a factor of 10 using image orientation. These results show the potential of combining traditional localisation sensors with visual observations. However, GCPs were used to achieve these results, which are not always available.

One possibility to improve the accuracy and reliability of image orientation is the integration of several cameras. This approach is frequently employed in robotics, as shown in the survey of Zou et al. (2019). One example is CoSLAM (Zou and Tan, 2013) in which images from several cameras are used in a common bundle adjustment to calculate the 3D coordinates of tie points located in the static environment and to build a common map; the authors find that image coordinates of tie points of individual moving objects can be removed by outlier detection. There are also applications for road traffic mapping. In (Stoven-Dubois et al., 2020), for example, it is shown that by jointly recording a map by several vehicles equipped with GNSS receivers and monoscopic cameras, the residuals of checkpoints, which in this case consist of road signs, can be reduced from up to $10\ m$ to approximately $2\ m$.

The cooperation with other traffic participants can open up further opportunities. Stoven-Dubois et al. (2018) introduce an unmanned aerial vehicle (UAV) tandem system for surveying objects in GNSS denied areas. A so-called surveying UAV flies next to the object to be surveyed and takes images. Because of the proximity to the object, the GNSS signal of this drone may be weak or totally absent, therefore this UAV is tracked by another UAV that flies at a higher altitude with a good GNSS signal. MapKITE (Molina et al., 2017; Nahon et al., 2019) also uses a tandem system. The authors combine a terrestrial mobile mapping van with a UAV. The van has a much higher payload and, thus, can carry heavier and also more accurate equipment. In that approach, the UAV uses the vehicle as a kinematic GCP. For accurate automated localisation, a circular target is placed on the vehicle roof. As the drone can profit from the highly accurate localisation of the van, better geo-referencing of the image data of the drone is achieved as a result. In (Trusheim and Heipke, 2020) and (Trusheim et al., 2021) we discuss the usability of similar dynamic GCPs in the context of road traffic scenarios. We compare the precision of different cooperation methods: (a) sharing all image data among the cooperating vehicles and employing a centralised bundle adjustment, and (b) using one vehicle as dynamic GCP for the observations of the other vehicle. It is shown that both methods offer a precision improvement of more than $20\ \%$ compared to a baseline with a non-cooperating approach. While the improvement due to the centralised bundle adjustment was slightly higher, such an approach needs more data exchange.

Moreover, it has been shown in the literature that image coordinates of points located on dynamic objects cannot always be eliminated before image orientation, which decreases the quality of the results. This problem is addressed in (Zhao et al., 2021). The authors detect dynamic image regions by combining object detection using Mask R-CNN (He et al., 2017) and optical flow in indoor scenarios. Mask R-CNN is used to detect movable objects such as people. The remaining parts of the image are checked for potential movements via optical flow. The authors state that they are able to improve the accuracy and robustness of the image orientations in a dynamic indoor scenario compared to ORB-SLAM2 (Mur-Artal and Tardós, 2017), a well-known visual SLAM method. Bescos et al. (2018) use a similar approach for image sequences showing traffic scenarios. Instead of using optical flow, they employ RGB-D images and an approach based on multi-view geometry to detect and eliminate potentially moving objects (e.g., cars). The authors report that they are able to improve the image orientation compared to ORB-SLAM2 in scenarios in which nearly all detected instances are actually moving. Bescos et al. (2021) detect cars using a CNN and subsequently represent them by sets of points identified by some point extraction method. Each set is then assigned a motion model, which is integrated into a bundle adjustment. The authors state that the image orientation is slightly degraded compared to ORB-SLAM in cases in which many cars are not moving, due to the higher number of parameters to be estimated. In contrast, our method subdivides the image regions detected as being potentially dynamic by the CNN into dynamic and static regions, and subsequently only uses points located in the static regions or in the static environment as tie points in the bundle adjustment. In this way, significantly fewer parameters are needed.

3. IDENTIFICATION AND PROCESSING OF DYNAMIC POINTS

This section describes the procedure used to detect the dynamic points in the total set of conjugate points. Figure 1 shows the pipeline developed for our method. Note that our method uses synchronised image sequences recorded by a stereo camera as input data. Each of the frames in such a sequence is considered as an epoch and associated with a timestamp. In the first step, the images of the whole sequence of both cameras of the stereo setup are classified into areas showing potentially dynamic objects and those showing static environment using an object detection approach. All objects which are traffic participants (e.g., cars, bicycles or pedestrians) are assumed to be potentially dynamic, while everything else is assumed to be part of the static environment. In parallel, feature points are detected in all of these images. These feature points are superimposed with the detected objects, resulting in a division of feature points belonging to the static environment and to potentially dynamic objects. The points assigned to the static environment are then matched to derive conjugate points and are employed to calculate initial exterior orientations in a common coordinate system. Using these orientation parameters, the potentially dynamic points are transformed into the common object coordinate system by forward intersection using the two images of the stereo camera for each epoch separately. Point tracking is used to assign points observed over multiple epochs to tracks. The transformed coordinates of all point observations belonging to one track are

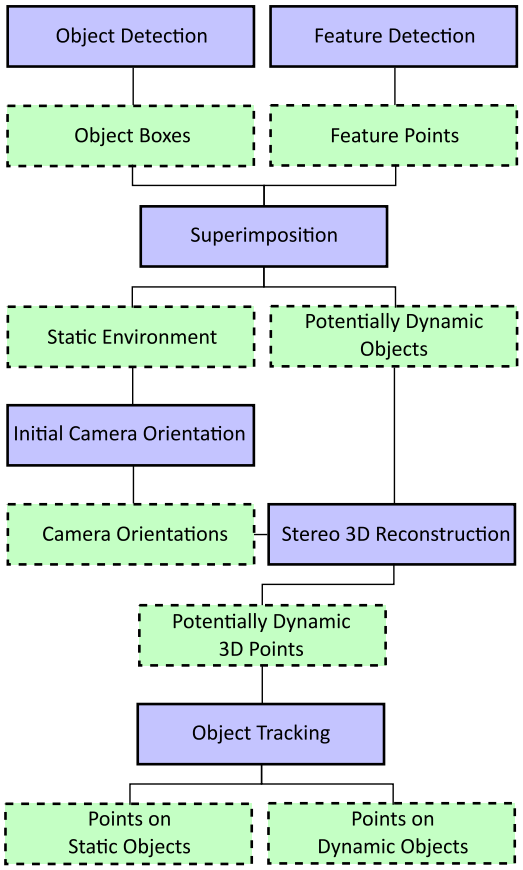


Figure 1. This flowchart shows the procedure used in this paper to identify points on dynamic objects. Blue solid boxes refer to processing steps and green dashed boxes to data.

then analysed to separate static from moving points. These individual steps are described in more detail in the subsequent sections.

3.1 Object Detection

Object detection is used to find instances of potentially moving objects. Neural networks are suitable for this purpose and many models exist in the literature that have been trained on large-scale data sets such as Common Objects in Context (COCO) (Lin et al., 2014). This data set contains the object types that are of interest for our work, such as cars, pedestrians and cyclists. In this work, the You Only Look Once (YOLOv4) approach (Bochkovskiy et al., 2020), pre-trained on COCO, is used. YOLOv4 detects objects and represents them by rectangular axis-parallel boxes. The algorithm produces state-of-the-art results and provides real-time capability. Object detection is applied independently for both images of the stereo camera in every epoch. Regions inside these boxes are marked as being potentially dynamic, and regions outside the boxes are declared to correspond to the static environment. This is a relatively coarse simplification, as the boxes also contain static background.

3.2 Feature Detection

We use the well-known SIFT operator (Lowe, 2004) for feature detection. To ensure that the feature points are evenly distributed across the image, which leads to better stability of the adjustment results, the image is first subdivided into sub-regions of uniform size. Then each sub-region in the image is normalised to detect possible feature points also in areas with low contrast. Normalised grey values gn_i are computed using the maximum g_{max} and the minimum g_{min} grey value of each sub-region with the pixel index i:

$$gn_i = \frac{g_i - g_{min}}{g_{max} - g_{min}} \cdot 255. \tag{1}$$

Subsequently, SIFT feature points are detected and described in these sub-regions independently. The detected feature points $\boldsymbol{x}$ are then sorted based on the contrast value $D(\boldsymbol{x})$. The n best feature points in every sub-region of the image are chosen for further processing. The selected points are divided into potentially dynamic and static image points based on the type of region they are located in (see Sec. 3.1).

3.3 Image Orientation and 3D Coordinates of Potentially Dynamic Points

Using bundle adjustment, the points of the static environment are used to calculate initial image orientations for all epochs in a common (global) coordinate system[2]. Subsequently, in every epoch, the potentially dynamic image points of the stereoscopic image pair are matched and 3D coordinates are computed via forward intersection.

3.4 Separation of Static and Dynamic Points

To check whether the potentially dynamic points are indeed dynamic, they are tracked in image space over time. For this purpose, the detected feature points, of which 3D coordinates were calculated via forward intersection (see Sec. 3.3), are assigned to tracks.

Beginning from a starting frame t_0 of the image sequence of the left camera of the stereo setup, the points in the current frame are compared to the points in a second frame of the same camera, having a temporal distance τ. The comparison is based on the SIFT descriptor. This is repeated for multiple frames, until τ reaches a pre-defiend maximum value τ_{max}. A large valued τ_{max} ensures that a feature point can be tracked over long distances even if it is occluded in between, but also increases the required computation time. The starting frame is then moved one frame forward in time. The point identification numbers (IDs) of points that have already been assigned to tracks in the previous step are kept, so that tracks that have already been started are continued.

To determine if a point is static or dynamic, the RMSE p_{rmse} between all point observations p_i assigned to a track and the centre of gravity $\overline{p}$ of these observations in the global coordinate system is calculated:

$$p_{rmse} = \sqrt{\frac{\sum_{i=1}^{n} \left(|\overline{p} - p_i|\right)^2}{n-1}}, \tag{2}$$

where n is the number of point observations assigned to a track, p is a vector of 3D coordinates in one epoch and i is the index of the point in the track.

Besides the point IDs also the object IDs are kept consistent over time to allow to correctly recognise dynamic objects (see

[2] Matching and bundle adjustment were carried out using COLMAP, (Schönberger et al., 2016; Schönberger and Frahm, 2016)

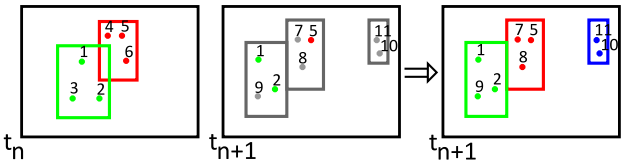


Figure 2. Propagation of object IDs from one frame to the next. Point IDs are shown by a number and object IDs by colour, *grey* indicates that no object ID is assigned.

Fig. 2). For this purpose, points that are located in image regions showing a potentially dynamic object are assigned the ID of this object, in addition to their point ID. Based on the concept of feature tracking, the point ID is used to propagate the object ID assigned to a point to its correspondence in the next frame. As far as the determination of an object ID is concerned, there are two cases: If one or more points associated with an object were already assigned an object ID in an earlier frame, the most frequent object ID is adopted and assigned to all points associated with this object in the current frame (see Fig. 2 *red* and *green* objects). If, on the other hand, none of the points that are located within the image region showing an object were assigned an object ID in an earlier frame, a new object ID is assigned (see Fig. 2 *blue* object).

Figure 2 shows an example of such a propagation of the object IDs. The objects are represented by boxes. The process starts with frame t_n which shows two objects (*green* and *red*), each with three points assigned to it. The next frame t_{n+1} shows three objects with unknown object IDs and five points (*grey*). Some of the points in frame $t_n n + 1$ were already associated an object having an object ID from frame t_n (*green*: 1 and 2; *red*: 5). Through these points the object IDs are propagated to the objects of frame t_{n+1}. Also, points situated on these objects which where not yet connected to an object, are now associated to this object (*green*: 9; *red*: 7 and 8). One image region shows an object which has no points assigned that were associated with any object in the earlier frame (*grey*: 11 and 10). Therefore, this object is assigned a new object ID (*blue*).

The mean of the RMSE values ($p_{rmse,j}$, with index j) of all k tracks associated to an object, weighted by the track length (i.e. the number of points n_j) o_{rmse}, is finally used to determine whether the whole object is dynamic or static:

$$o_{rmse} = \frac{\sum_{j=1}^{k} p_{rmse,j} \cdot n_j}{\sum_{j=1}^{k} n_j}. \quad (3)$$

If this value exceeds a threshold λ, all points of this object are classified as dynamic, otherwise as static.

3.5 Cooperative Image Orientation using Dynamic GCPs

For the calculation of the exterior orientations of the stereo camera as well as the 3D coordinates of the tie points, a bundle adjustment based on the methodology we introduced in our previous work (Trusheim et al., 2021) is used. In this method, the six elements of exterior orientation are modelled as functions of time with equally spaced anchor points as support and linear interpolation in between. As the stereo camera is part of a multi-sensor platform mounted on a vehicle (see also below), the transformation from the image to the global coordinate system is split into two 6 DoF transformations: a constant transformation between the camera coordinate system and the platform coordinate system, the parameters of which are determined in a pre-calibration, and a second transformation between the platform coordinate system and a global coordinate system, which is time-dependent. For the remaining parts of the paper, we refer to the first transformation as mounting calibration and to the second transformation as exterior orientation.

The unknowns of this method are the 3D object space coordinates of the static tie points as well as the six orientation parameters for each anchor point in the global coordinate system. Three types of observations are introduced to compute these unknowns:

1. The image coordinates of the static conjugate points are used as tie points.

2. The image coordinates of so-called marker points. Marker points are points on a cooperating vehicle with known 3D coordinates in the vehicle's coordinate system. The image coordinates and the marker IDs are observed with an algorithm based on a blob detector (Mallick, 2015).

3. The 3D coordinates of the positions of a GNSS-antenna on the platform, given in the global system. These coordinates are introduced to define the geodetic datum and to prevent a block drift. Due to the small number of observations of GNSS coordinates compared to the number tie points, they have only a minor influence on the overall block stability.

4. EXPERIMENTS

We performed three experiments corresponding to three scenarios that differ in the way in which points associated with potentially dynamic objects are treated in the bundle adjustment:

1. In the first scenario all detected conjugate feature points are used as potential tie points in the adjustment (see Sec. 3.2). This scenario is similar to the approach of our previous work (Trusheim et al., 2021), as no pre-selection takes place; it is used as a baseline.

2. In the second scenario, only feature points associated with the static environment are used as potential tie points in the adjustment (see Sec. 3.3). Thus, all points that are associated with potentially dynamic objects are eliminated.

3. In the third scenario, the point set that is associated with potentially dynamic objects is split into a set of dynamic points (e.g., points on driving cars) (see Sec. 3.4) and a set of static points (e.g., points on parked cars), and the observations of these static points, as well as the points of the static environment, are used as tie points in the adjustment.

To calculate the results, a robust adjustment is employed, which eliminates observations with too large residuals during the adjustment process (Klein and Förstner, 1984). In the stochastic model, different accuracies were assumed for the different types of observations. The quantitative evaluation of the results of the different scenarios is based on the precision, derived by error propagation provided by the bundle adjustment.

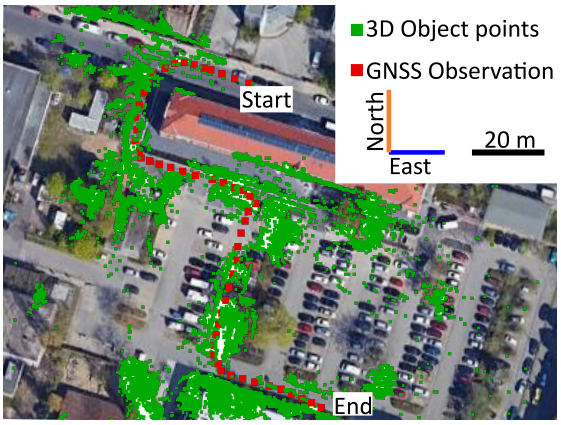


Figure 3. The trajectory driven by the two vehicles, represented by the GNSS observations and the 3D tie points. A satellite image of the area is used as background.

4.1 Data Acquisition

The data has been recorded during a measurement campaign with two vehicles equipped with multi-sensor platforms consisting of a stereoscopic camera system and a GNSS antenna. For the experiments, a road section was chosen in which the two vehicles moved in a tandem formation, one behind the other, in an inner-city area including a car park. In this configuration, the front vehicle is visible in the image of the rear vehicle for most of the track and can be used as a dynamic GCP. Therefore, the rear vehicle will be referred to as *vehicle* and the front vehicle as *dynamic GCP* in the following. For the experiments only images observed by the stereoscopic camera system of the *vehicle* are used.

The total length of the trajectory is $170\,m$. It consists of three 90^o left turns and one 90^o right turn, see also Figure 3; the *vehicle* needed approximately $50\,s$ for the whole trajectory and reached a maximum velocity of $4.5\,\frac{m}{s}$. The frequency of the image acquisition was $5\,Hz$. Therefore, the maximum distance between two images is $0.9\,m$ and the maximum displacement of a 3D tie point in a distance of $20\,m$ is $43\,px$ in horizontal and $28\,px$ in vertical image direction; such displacements can typically be handled by matching algorithms.

The stereoscopic system consists of two Grasshopper 3 USB cameras. They acquired images of 1920×1200 pixels and have a pixel size of $5.85\,\mu m \times 5.85\,\mu m$. The focal length is $11.3\,mm$, equivalent to 1930 pixels. Image acquisition was initiated by an external trigger signal provided to both cameras and to a raspberry pi equipped with a GNSS antenna which saved the GNSS time. Thus, all sensor data is given in the same time system. The GNSS positions were captured using geodetic receivers Septentrio PolaRx5e with a Javad GrAnt G5T antenna at a frequency of $1\,Hz$. The campaign was carried out on Aug. 25, 2020, at 5 pm, thus relatively late in the day. The sky was overcast, which led to somewhat challenging lighting conditions. In Figure 4, an example frame is shown, whereby the *dynamic GCP* is visible; the markers can be seen mounted on the back of that vehicle.

4.2 Experimental Setup

The complete trajectory of the *vehicle* is modelled by anchor points distributed every $0.25\,s$. This leads to a total amount of 197 anchor points.

Figure 4. Image taken by one of the cameras of the *vehicle* at the beginning of the trajectory; the vehicle corresponding to the *dynamic GCP* can be seen in the centre of the image.

For the trajectory of the *dynamic GCP*, anchor points were set every $1\,s$, i.e., corresponding to the frequency of the GNSS observations. Only the images observed by the *vehicle* closest in time to the GNSS observation of the *dynamic GCP* are used to detect the marker points. As the *dynamic GCP* is not continuously visible in the observing camera due to the curves along the trajectory, some regularisation constraints are needed to estimate the rotation angles of the exterior orientation of the *dynamic GCP*. These constraints model the fact that the *dynamic GCP* is oriented in the direction of travel during the whole experiment; the direction of travel is defined by two GNSS observations adjacent in time. The constraints are introduced as soft-constraints.

The image coordinates of the feature points were generated by dividing the image into 12×8 sub-regions. For each region up to 100 of the best feature points ranked by the contrast value $D(\boldsymbol{x})$ were selected limited by a threshold value of $D(\boldsymbol{x}) = 0.04$ according to Section 3.2. As mentioned above, feature point matching and the computation of the initial orientation parameters of the cameras was done using the COLMAP software (Schönberger et al., 2016; Schönberger and Frahm, 2016).

For the final bundle adjustment the following observations and stochastic model are used:

GNSS observations of the *vehicle* and the *dynamic GCP* $(\sigma_{N,E_{GNSS}} = \pm 0.5\,m,\ \sigma_{H_{GNSS}} = \pm 1.0\,m,)$.

Image coordinates of tie points from the stereo camera of the *vehicle* $\left(\sigma_{x,y_{tp}} = \pm 1.5\,px\right)$.

Image coordinates of marker points from the stereo camera of the *vehicle* $\left(\sigma_{x,y_{mp}} = \pm 0.5\,px\right)$.

Soft-constrains introduced for the rotations of the *dynamic GCP* $(\sigma_{R,P_{sc}} = \pm 0.5\,rad,\ \sigma_{Y_{sc}} = \pm 1.0\,rad)$.

For the image coordinates of the marker points, a smaller standard deviation was used compared to those of the tie points, as the markers are specifically designed to be accurately measured in the images. Due to the large standard deviations, the soft constraints do not significantly influence the numerical results, but they do prevent the normal equation matrix from becoming singular.

As mentioned above, in the final bundle adjustment the following unknowns are calculated:

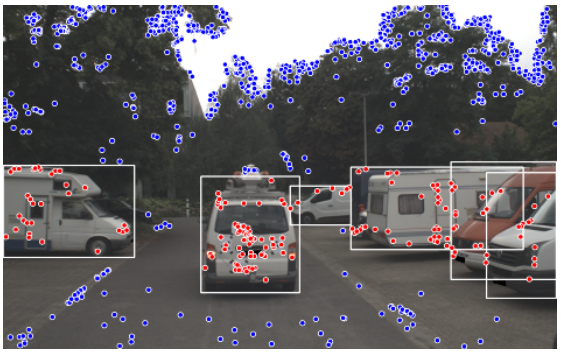


Figure 5. Result of the extracted potentially dynamic objects superimposed with the detected feature points. *Blue*: Feature points in the static environment. *Red*: Feature points situated on potentially dynamic objects.

Anchor points of the trajectories in the global coordinate system of the *vehicle* and the *dynamic GCP*.

3D Tie point coordinates in the global coordinate system for all observed tie points.

5. RESULTS

5.1 Detection of Potentially Dynamic Objects

An example of the bounding boxes of potentially dynamic objects with the detected feature points is displayed in Figure 5. The feature points located in the static environment are coloured in blue and the points located on potentially dynamic objects are shown in red. This example shows, that while object extraction working on individual images only is of course not able to distinguish parked from moving cars, the bounding boxes capture the cars rather well. Also, feature points are distributed across the whole image as required, except for areas with extremely low textures.

5.2 Number of Tie Points and Track Length

Obviously, the number of 3D tie points is largest in Scenario 1 (namely 37.812), while the static environment, being the focus of Scenario 2, only contains 34.659 points, which corresponds to a reduction of 3153 points or 8.3% compared to Scenario 1. For Scenario 3, 647 points are added again, an increase of 2.0% yielding a total of 35.306 points. This is a relatively small number of additional points only. A reason for this small increase is the fact that, in contrast to points in the static environment, which can also be reconstructed via temporal matching, all points added in Scenario 3 must be visible in both images of the stereoscopic camera in one and the same epoch, to be reconstructed in 3D, which limits their number.

For Scenario 3 a window size τ_{max} of 50 frames is employed for tracking. The track length of the potentially dynamic 3D points and their RMSE value are depicted in Figure 6. It can be seen that points can have a track length larger than τ_{max} (one such point is visible in the figure). The reason is that point IDs are preserved beyond the window size (see Sec. 3.4). Also, a certain correlation between track length and RMSE is visible, the Pearson correlation coefficient being 0.69. This is due to the fact, that most dynamic points (having a high RMSE) are located on the *dynamic GCP* and are thus visible in many consecutive frames. Objects with a weighted mean of the RMSE

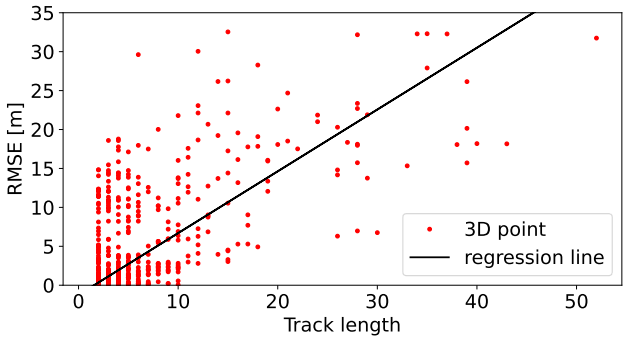


Figure 6. Relation between the track length given by the number of observations of a 3D point, and the RMSE of the position of potentially dynamic 3D points.

values of all associated points larger than $\lambda = 0.75\,m$ are considered to be dynamic, all associated points are eliminated.

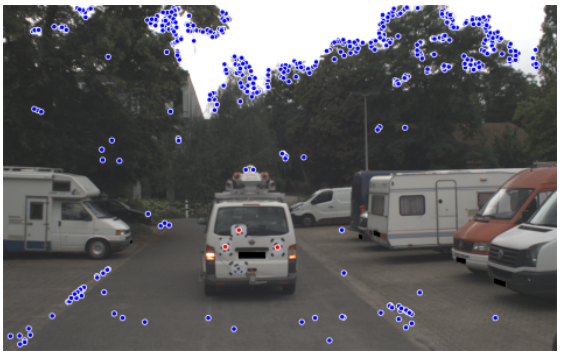
Figure 7 shows the tie point distributions resulting from Scenarios 2 and 3 in the same image frame. It can be seen that our method was able to correctly distinguish between static points (those on parked cars) and dynamic points (on the *dynamic GCP*).

5.3 Precision of the Exterior Orientation

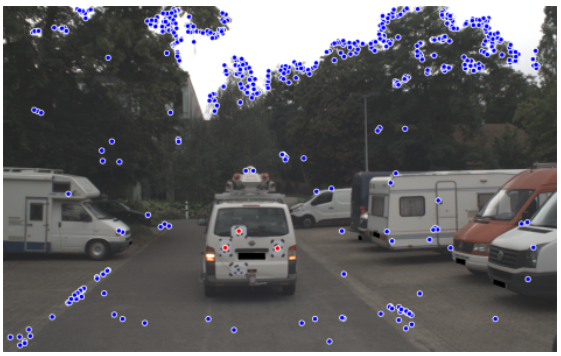
The exterior orientation of the *vehicle* and its precision as a functions of time are depicted in Figure 8. The figures are similar for all three scenarios, therefore only the results of Scenario 1 are presented. Figure 8a shows the six orientation parameters, divided into positions, consisting of an East, a North and a Height component, and the rotations roll, pitch and yaw. Roll is the rotation about the axis in the driving direction of the *vehicle*, pitch is the rotation about the horizontal axis orthogonal to the driving direction and yaw is the rotation about the vertical axis. The figure shows that a large proportion of the trajectory consists of turns, which is particularly evident from the yaw.

Figure 8b shows the precision of the exterior orientation; the precision of the position is expressed in the driving direction, in the horizontal direction orthogonal to the driving direction and in the vertical direction. The precision of the rotation is given in roll, pitch and yaw. It is noticeable that the precision of the projection centre of the *vehicle* in driving direction and the horizontal direction orthogonal to the driving direction, expressed as σ_D and σ_O are clearly better than that in the vertical direction σ_V. This finding can be explained by the distribution of the 3D tie points. In the viewing direction, the 3D body in which 3D tie points lie is bounded by the scene (at some point, there will be an opaque surface), in the two other directions it is bounded by the viewing angle of the cameras. Typically, the extent of this 3D body is larger in the viewing direction, which results in a smaller standard deviation in that direction. Due to the turns, the horizontal size of that 3D body is enlarged over time. On the other hand, the smallest extension of the 3D body is in the vertical direction, as there are only rather small changes of the pitch angle. As a result, the vertical direction is more uncertain than the other two directions, visible in the upper part of Figure 8b, where the green curve lies clearly above the blue and the orange ones. These findings are similar to those we found in (Trusheim et al., 2021).

In addition, Figure 8b shows that the turns affect the precisions, in particular the vertical precision (cf. σ_V, green curve in the

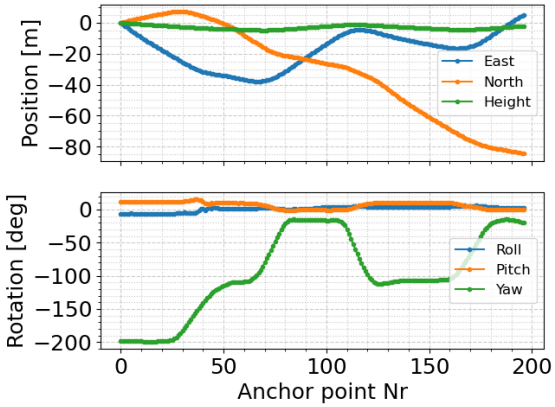


(a) Tie point distribution in Scenario 2.

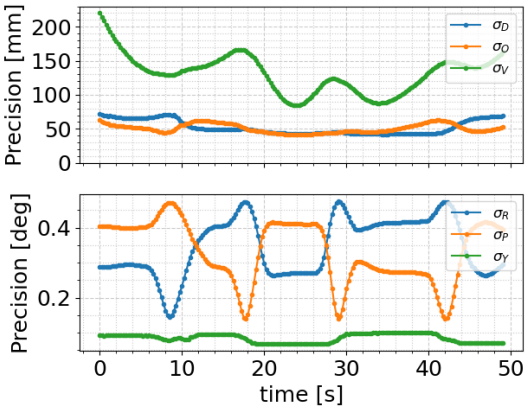


(b) Tie point distribution in Scenario 3.

Figure 7. Example of tie point distribution in an image. Red: observed marker points. Blue: observed tie point.



(a) Exterior orientation of the *vehicle* as a function of time. Top: Position (East, North and Height). Bottom: Rotation angles (roll, pitch, yaw).



(b) Precision of the exterior orientation parameters resulting from Scenario 1. Top: Position (driving σ_D, orthogonal σ_O and vertical σ_V). Bottom: Rotation angles (roll σ_R, pitch σ_P and yaw σ_Y) angle.

Figure 8. Exterior orientation of the *vehicle* and its precision as functions of time

figure). The related standard deviation increases, i.e. the precision becomes worse, during the turns, and decreases during the straight segments. The worst precision of the vertical component is found at the beginning of the trajectory, amounting to $230\,mm$; the best value can be found in the middle of the drive, amounting to $85\,mm$. The reason for the precision becoming worse in the turns is assumed to be the reduced number of tie points in those parts of the trajectory, also observable in shorter track lengths for the related tie points, leading to a weaker connectivity in the block. In contrast, the precision in driving direction and in the horizontal direction orthogonal to the driving direction remain relatively constant and are only slightly affected by the turns: the precision lies between $40\,mm$ and $70\,mm$ over the whole track.

Analysing the precisions of the angles, the main effect of the turns is recognisable in the precisions of roll (σ_R, blue curve) and pitch (σ_P, yellow curve), while the precision of yaw (σ_Y, green curve) is relatively constant at a level of 0.1^o. At each 90^o turn, the directions of the roll and the pitch axis with respect to the global coordinate system are interchanged, which explains also the constant sections of the two curves, which relate to the straight elements of the trajectory. σ_R and σ_P change between two different levels of 0.3^o and 0.4^o, respectively. The different levels can again be explained by the non-symmetric distribution of 3D tie points in the global coordinate system, which leads to a better stability about the initial roll axis.

5.4 Precision of the 3D Coordinates of the Tie Points

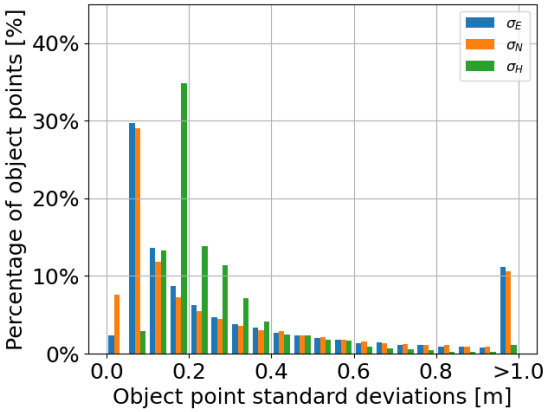
The precisions of the 3D coordinates of the tie points in the global coordinate system are given in Table 1, using the median of all tie points for the three scenarios. It is shown that the median precision of the 3D coordinates of the tie points in Scenarios 2 and 3 differ just slightly, but it is significantly improved in all three components compared to Scenario 1. For the median precision of both planimetric components an improvement of $55\,mm$ could be achieved, as well as an improvement of $5\,mm$ in height. This result confirms our expectation that deleting points on moving objects improves the results. Similar findings are achieved when inspecting the distribution of the precision of the 3D coordinates, see Figure 9 (Scenario 3 is omitted, as results for Scenarios 2 and 3 are visually identical). The figure reveals that by omitting dynamic points the percentage of tie points with smaller standard deviation (higher precision) is increased for all three components. Simultaneously, the percentage of tie points with a lower precision decreases.

Finally, in Figure 10 a section of the resulting point cloud is visualised for Scenarios 1 and 2. It is noticeable that in Scenario 1 (Fig. 10a) a cluster of 3D tie points lies under the street level, which is clearly a mistake. This cluster does not show up in the results of Scenario 2 (Fig. 10b) and Scenario 3 anymore. Consequently, the wrong cluster stems from dynamic points present only in Scenario 1.

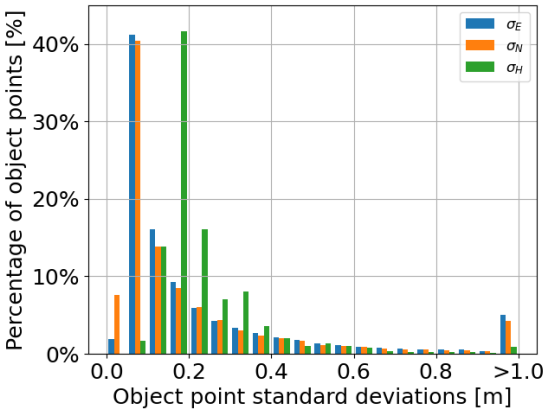
This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-V-1-2022-169-2022 | © Author(s) 2022. CC BY 4.0 License.

	$\tilde{\sigma}_E$ $[mm]$	$\tilde{\sigma}_N$ $[mm]$	$\tilde{\sigma}_H$ $[mm]$
Scenario 1	172	160	196
Scenario 2	117	106	191
Scenario 3	**116**	**105**	**190**

Table 1. Median standard deviations ($\tilde{\sigma}_E$, $\tilde{\sigma}_N$ and $\tilde{\sigma}_H$) of the 3D coordinates of the tie points in East, North and Height, in the different scenarios.

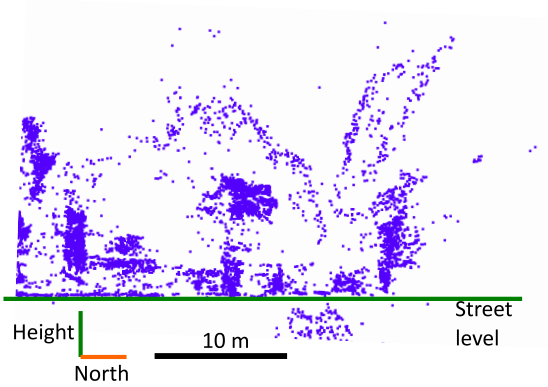


(a) Results for Scenario 1.

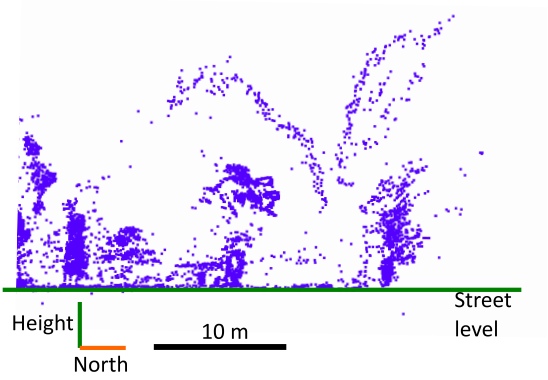


(b) Results for Scenario 2.

Figure 9. Histograms of the standard deviations of the 3D coordinates of the tie points for Scenarios 1 and 2.



(a) Results for Scenario 1.



(b) Results for Scenario 2.

Figure 10. Subset of the 3D point cloud after bundle adjustment.

6. CONCLUSIONS

In this work, a novel method is introduced which can distinguish between static and dynamic points in image sequences acquired with a stereoscopic camera in motion. The method consists of a combination of SIFT feature extraction with detections of potentially dynamic objects using YOLO, followed by bundle adjustment. The method, as it presented in this paper, is not capable of achieving results in real-time because of the use of bundle adjustment. We would at least need to introduce a window-based version (see e.g., (Beder and Steffen, 2008)) to achieve this goal.

In the experiments, the influence of dynamic objects in the tie point cloud in the context of cooperative image orientation is shown. It is found that compared to an adjustment without eliminating dynamic points, the precision in the trajectory does not change very much. However, errors in the 3D tie points are prevented, and consequently, the precision of the 3D coordinates of the tie points are improved. The integration of points located on potentially dynamic, but actually static objects, e.g., parked cars, does not have a significant effect on the precision of the 3D tie points. Most probably, this is due to the relatively low amount of additional points in our example. In general, it can be stated that the elimination of dynamic points offers clear advantages for reducing the errors in the resulting 3D point cloud and simultaneously increases their precision. For the 6 DoF orientations, however, the advantages are found to be less clear in our experiments. In future work, additional experiments taking into account other traffic scenarios, multiple vehicles, more sophisticated interpolation schemes for the trajectories and additional sensors are to be investigated.

While in this work, precision provided by the adjustment is used as quantitative measure of quality, we will also investigate differences in the resulting accuracy in future work, for example, through an improved GNSS solution for both, the *vehicle* and the *dynamic GCP*. Finally, individual dynamic 3D points that are currently eliminated can be introduced into the adjustment as tie points with the help of a motion model. To do so, we strive to cluster them and assign a common motion model to all points of a cluster representing a dynamic object, to reduce the number of additional unknowns. Such clusters can also contain dynamic GCPs.

ACKNOWLEDGEMENTS

This work was supported by the German Research Foundation (DFG) as a part of the Research Training Group i.c.sens [GRK2159].

REFERENCES

Beder, C., Steffen, R., 2008. Incremental Estimation Without Specifying A-Priori Covariance Matrices for the Novel Parameters. *2008 Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 1–6.

Bescos, B., Campos, C., Tardos, J. D., Neira, J., 2021. DynaSLAM II: Tightly-Coupled Multi-Object Tracking and SLAM. *Robotics and Automation Letters*, 6(3), 5191–5198.

Bescos, B., Fácil, J. M., Civera, J., Neira, J., 2018. DynaSLAM: Tracking, Mapping, and Inpainting in Dynamic Scenes. *Robotics and Automation Letters*, 3(4), 4076–4083.

Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y. M., 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv eprints, arXiv-2004.*

Cavegn, S., 2020. Integrated Georeferencing for Precise Depth Map Generation Exploiting Multi-Camera Image Sequences from Mobile Mapping. PhD thesis, Aerospace Engineering and Geodesy, University of Stuttgart.

Cavegn, S., Nebiker, S., Haala, N., 2016. A Systematic Comparison of Direct and Image-Based Georeferencing in Challenging Urban Areas. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B1, 529–536.

Garcia-Fernandez, N., Schön, S., 2019. Optimizing Sensor Combinations and Processing Parameters in Dynamic Sensor Networks. *Proceedings of the 32nd International Technical Meeting of the Satellite Division of the Institute of Navigation*, 2048–2062.

He, K., Gkioxari, G., Dollar, P., Girshick, R., 2017. Mask R-CNN. *Proceedings of the International Conference on Computer Vision*, 2961–2969.

Klein, H., Förstner, W., 1984. Realization of Automatic Error Detection in the Block Adjustment Program PAT-M43 Using Robust Estimators. *International Archives of Photogrammetry and Remote Sensing*, XXV-3a, 234–245.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L., 2014. Microsoft COCO: Common Objects in Context. *European Conference on Computer Vision*, 8693, Springer, Cham., 740–755.

Lowe, D. G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91–110.

Mallick, S., 2015. Blob Detection Using OpenCV (Python, C++). https://www.learnopencv.com/blob-detection-using-opencv-python-c/. Accessed 06 April 2021.

Molina, P., Blázquez, M., Cucci, D., Colomina, I., 2017. First Results of a Tandem Terrestrial-Unmanned Aerial mapKITE System with Kinematic Ground Control Points for Corridor Mapping. *IEE Remote Sensing*, 9(1), 60.

Mur-Artal, R., Tardós, J. D., 2017. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEE Transactions on Robotics*, 33(5), 1255–1262.

Nahon, A., Molina, P., Blázquez, M., Simeon, J., Capo, S., Ferrero, C., 2019. Corridor Mapping of Sandy Coastal Foredunes with UAS Photogrammetry and Mobile Laser Scanning. *Remote Sensing*, 11(11), 1352.

Schönberger, J. L., Frahm, J.-M., 2016. Structure-from-Motion Revisited. *Conference on Computer Vision and Pattern Recognition*, 4104–4113.

Schönberger, J. L., Zheng, E., Pollefeys, M., Frahm, J.-M., 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. *European Conference on Computer Vision*, 501–518.

Stoven-Dubois, A., Dziri, A., Leroy, B., Chapuis, R., 2020. Graph-Based Approach for Crowdsourced Mapping: Evaluation through Field Experiments. *2020 16th International Conference on Control, Automation, Robotics and Vision*, 260–265.

Stoven-Dubois, A., Jospin, L., Cucci, D., 2018. Cooperative Navigation for an UAV Tandem in GNSS Denied Environments. *Proceedings of the 31st International Technical Meeting of the Satellite Division of the Institute of Navigation*, 24–28.

Trusheim, P., Chen, Y., Rottensteiner, F., Heipke, C., 2021. Cooperative Localisation Using Image Sensors in a Dynamic Traffic Scenario. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B1-2021, 117–124.

Trusheim, P., Heipke, C., 2020. Precision of Visual Localization Using Dynamic Ground Control Points. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B1-2020, 363–370.

Zhao, X., Zuo, T., Hu, X., 2021. OFM-SLAM: A Visual Semantic SLAM for Dynamic Indoor Environments. *Mathematical Problems in Engineering*, 2021 Article ID 5538840, 1–16.

Zou, D., Tan, P., 2013. CoSLAM: Collaborative Visual SLAM in Dynamic Environments. *Transactions on Pattern Analysis and Machine Intelligence*, 35(2), 354–366.

Zou, D., Tan, P., Yu, W., 2019. Collaborative Visual SLAM for Multiple Agents: A Brief Survey. *Virtual Reality & Intelligent Hardware*, 1(5), 461–482.