Automated Camera Stabilization and Calibration for Intelligent Transportation Systems

Marcel Bruckner
Technische Universität München
Boltzmannstraße 15, 85748 Garching
marcel.bruckner@tum.de

0.1. Static calibration

Intelligent Transportation System are inherently dependent on the calibration of the different sensors. To track and predict traffic the system has to know the poses of the different sensors relative to some reference coordinate system. This enables the ITS to accurately measure the position of vehicles within the single sensor ranges and at the overlapping boundaries.

Previous experiments have shown that a calibration process based on an IMU is not feasible in our case. Instead we focus on a calibration procedure based on visual landmarks in the video feed. The landmarks are mapped to their partially known world positions from high definition road maps.

Retrive objects from the HD maps (??) In this work we focus on the permanent delineator objects that are easily visible in the video feeds.

The world position of the objects can be retrieved using the mathematical operations defined in the OpenDRIVE standard.

This gives us the the base origin point $o = (x,y,z)^T$ of the object in the transverse mercator projection [2]. The base origin point is the world position of the lower end of the object where it ends in the ground or another object.

Additionally we retrieve a directional heading axis $d = (x,y,z)^T$ and the height h of the object.

These values enable us to approximate the real-world objects by sampling points $s \in S$ in world position along the center line of the object, where

$$S = \{o + \lambda * d : \quad \lambda \in [0,h]\} \tag{1}$$

Mapping objects to pixels To calibrate the camera we need to establish a mapping

$$s_c \mapsto p_c \Leftrightarrow o_c, d_c, \lambda_c \mapsto p_c \tag{2}$$

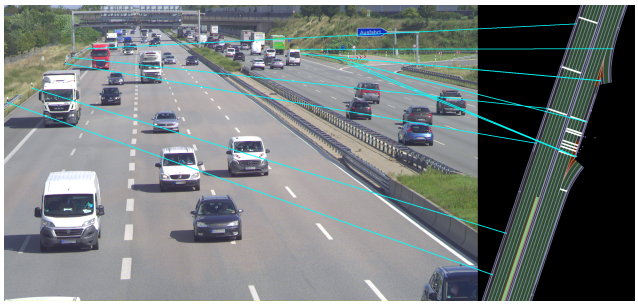

Figure 1: Left: The current camera frame. Right: A part of the HD maps (**??**). Cyan: The mapping $s_c \mapsto p_c$ from objects (right) to their corresponding pixels (left).

from pixels $p_c = (u,v)^T \in P$ from the video stream to the corresponding sampled points s_c from the object they belong to.

This leaves us with a set C of correspondences $\{p_c, s_c\}$.

This mapping is currently done by human interaction and not fully automated. To minimize the work an aiding system to mark pixels was implemented that outputs a list of pixels that can easily be mapped to the list of objects.

Relaxation of problem by line approximation Approximating the objects by lines removes the need for exactly known 3D correspondences as usually needed in calibration problems. Nonetheless it does not require to jointly estimate the full world position of the objects and camera pose jointly as in Bundle-Adjustment problems. The assumptions we made are:

- Objects are symmetric around their directional heading axis.
- Projected pixels of the objects are also symmetric around the projected directional axis.

We then take the mean over each row of pixels as the approximate pixel positions of the center line.

Calibration procedure Our camera is modelled using the pinhole camera model. The pinhole projection from samples of the world objects to pixels is formulated as

$$p_c = \pi\left(R * T * (o_c + \lambda_c * h_c)\right) \tag{3}$$

$$z * \pi(x) = \begin{bmatrix} f_x, & 0, & c_x, & 0 \\ 0, & f_y, & c_y, & 0 \\ 0, & s, & 1, & 0 \end{bmatrix} * x \tag{4}$$

where R is the cameras world rotation in Euler angles, T is the cameras world translation and π is the camera projection to image space based on the camera intrinsic parameters.

The optimal value for π, R, T is found, if it holds for all correspondences:

$$0 = p_c - \pi\left(R * T * (o_c + \lambda_c * h_c)\right) = p_c - \hat{p}_c \tag{5}$$

This places constraints on the values π, R, T can take and enables us to recover the camera pose and intrinsics only from the correspondences.

We estimate the camera pose by minimizing a modified version of the least-squares reprojection error

$$\min_{T,R,\Lambda,W} E(P,S,\pi,T,R,\Lambda,W) \tag{6}$$

formulated as

$$\begin{aligned} E(P,S,\pi,T,R,\Lambda,W) = & \sum_{c\in C} \rho(\|w_c * [p_c - \hat{p}_c]\|^2) \\ & + \sum_{c\in C} \alpha * \rho(\|(1-w_c)\|^2) \\ & + \sum_{c\in C} \beta * \rho(\|\Delta(\lambda_c, 0, h_c)\|^2) \\ & + \sum_{\pi_i \in \pi} \gamma * \rho(\|\Delta(\pi_i, \pi_i * 0.9, \pi_i * 1.1)\|^2) \\ & + \delta * \rho(\|\Delta(R_x, 60, 110)\|^2) \\ & + \delta * \rho(\left\|\Delta(R_y, -10, 10)\right\|^2 \end{aligned} \tag{7}$$

where P is the set of mapped pixels in the image, S is the set of mapped corresponding sampled points from the objects and Λ is the set of λ values associated with the sampled points. This formulation allows the optimization over the line approximations of the objects and jointly optimizes for the camera parameters T, R and the $\lambda \in \Lambda$ parameters of the line objects.

The calculation of the exact position of $s_c \sim \lambda$ allows the optimizer to search the whole space of real numbers for λ. Nonetheless we penalize values for λ that exceed the physical height of the object by

$$\Delta(x,l,u) = \begin{cases} x-u, & \text{if } x > u \\ x-l, & \text{if } x < l \\ 0, & \text{else} \end{cases} \tag{8}$$

This regularization enables a robust estimation procedure that can flexibly adjust to the missing exact world positions.

Initialization In contrast to most pose estimation problems our approach drops the need for good initialization. By regularization of the λ values enough flexibility is given to optimize over an infinite space of values, but enforces the solution of the λ to lie within the interval of $\lambda \in [0,h]$.

$$\bar{s} = \frac{1}{|C|} \sum_{c\in C} o_c \tag{9}$$

$$T_0 = \begin{pmatrix} 1,0,0, & \bar{s}_x \\ 0,1,0, & \bar{s}_y \\ 0,0,1, & \bar{s}_z + 1000 \\ 0,0,0, & 1 \end{pmatrix} \tag{10}$$

$$R_0 = \mathbb{I}^{4\times 4} \tag{11}$$

It is sufficient to initialize with $\lambda_c = 0$ for all correspondences. The camera rotation is defined to be zero with the camera facing in negative world z axis. By placing the camera at some distance over the mean of the known object positions with zero rotation the optimization always converges to the desired minimum.

0.2. Static calibration

In the following we evaluate the implemented static calibration algorithm and assess the algorithmic and systematic errors.

0.2.1 Ensure no Systematic Error

The proposed pose estimation algorithm from Equation 6 converges to a pair of optimal translation T and rotation R values for the camera pose. These T, R values approximate the projection from the world objects to the pixels as described in Equation 3.

Using a maps provider we assured that the resulting values are within reasonable ranges and that there are no systematic errors in the optimization. The results are displayed in Figure 3

0.2.2 Points needed for convergence

The correspondences build up a system of linear equations. This system of equations is solvable if there exist a more or equal number of constraints on the parameters than there are degrees of freedoms in the system:

$$p_c = \pi\left(R * T * (o_c + \lambda_c * h_c)\right), \forall c \in C \tag{12}$$

This is the case if:

$$\begin{aligned} 2 * |C| &\geq 5+3+3+1*|C| \\ |C| &\geq 11 \end{aligned} \tag{13}$$

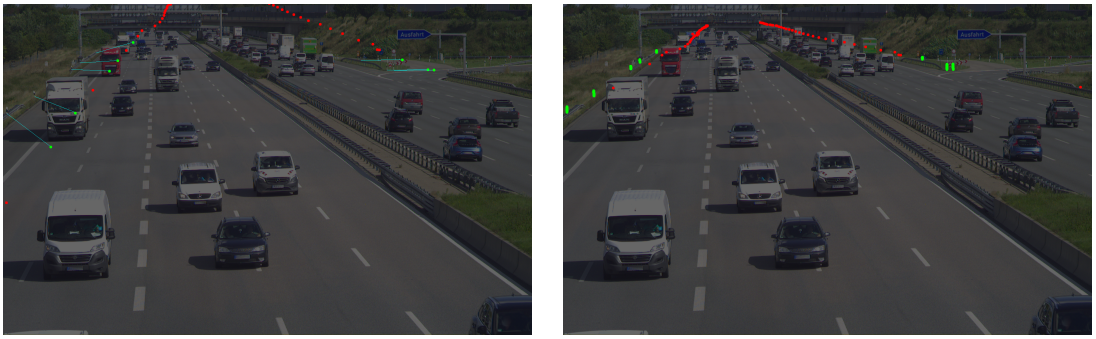

Figure 2: Left: Sampled points of objects that are mapped to pixel locations (green) and sampled points without known corresponding pixels (red) rendered by a poorly calibrated camera model. The mapping from the points to their expected pixels is drawn in cyan. Right: The same sampled points after the calibration procedure. The rendered positions of the sampled points align with the pixels of the objects they are mapped to and the drawn mapping disappears as the distance approaches 0.

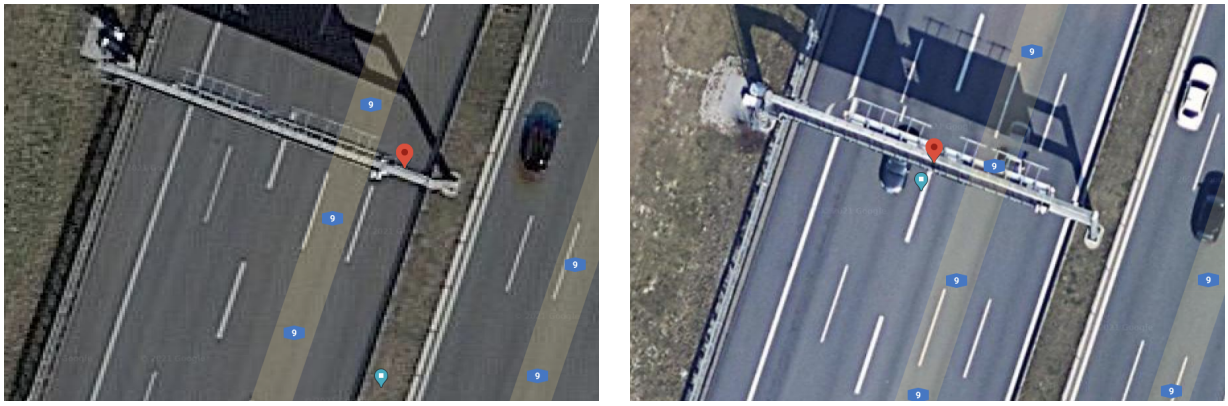

Figure 3: Left: The positions of the cameras S40 Near (red) and S40 Far (green) and their respective looking directions (yellow). Right: The positions of the cameras S50 Near (red) and S50 Far (green) and their respective looking directions (yellow). It displays that the rotation of the cameras is in a reasonable range so that the cameras look along the highway as expected. Also the cameras are within reasonable translational bounds around their real world location as subsubsection 0.2.4 shows.

As each of the pixel per correspondence gives us two constraints and we optimize over the 5 intrinsic parameters (Equation 4), the 3 extrinsic translation, 3 extrinsic rotation parameters and one λ parameter per correspondence. We see that 11 points are enough to recover the pose.

0.2.3 Structure of points

The best result are shown when there are at least two correspondences per object. These correspondences should be the top and bottom most visible pixel of the object. If there exists only 1 or a low number of near pixels, the algorithm cannot precisely recover the camera pose, as it is free to move the correspondence along the center line of the object. The algorithm therefore cannot distinguish the solutions where the camera is placed low, thus projecting a high point of an object to a low pixel correspondence, or if it should place the camera higher and lower the correspondences world position along the center line.

We thus conclude that the best solution is recovered the more the pixels fill the whole object, and the more spread to the top and bottom of the object the correspondences are.

0.2.4 Expectable Error Bounds

0.2.5 Expectable Deviations among Estimations

The proposed pose estimation algorithm is based on the minimization of the reprojection-error Equation 3. As with all optimization problems convergence is reached when the values that are optimized don't change anymore.

The optimization jointly optimizes for the 6 camera pa-

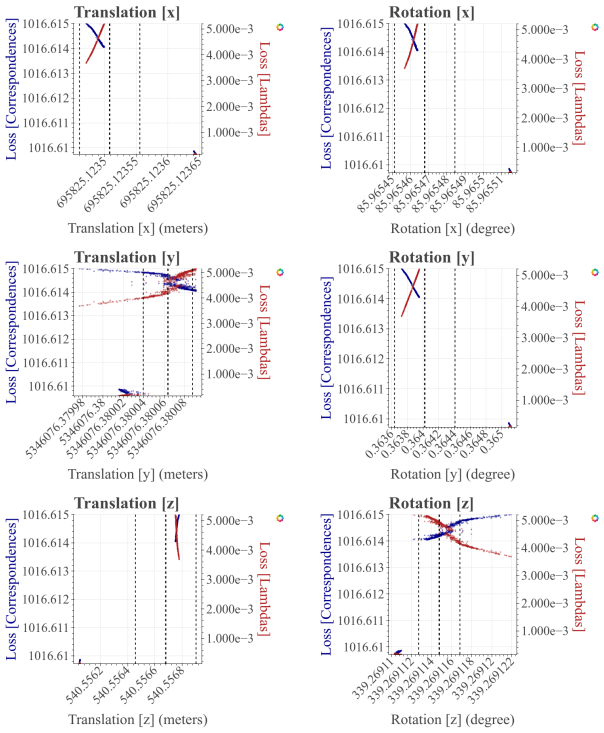

Figure 4: Left: The resulting translational parameters plotted against the remaining losses. Right: The resulting rotational parameters plotted against the remaining losses. The mean of the distributions is displayed as thick dashed line. The smaller dashed lines display the standard deviation σ. The σ of the translations does not exceed $2 * 10^{-4} m = 0.2mm$. For the rotation σ is at most $4 * 10^{-4} deg$.

rameters, 3 for the translation, 3 for the Euler angle rotation and one λ parameter per correspondence. The resulting high-dimensional problem exceeds multiple minima, whereas each represents a configuration for the camera pose that well explains the dependency between pixels, world objects and the camera.

As stated previously the loss landscape does exhibit a multitude of local minima, thus the optimization procedure converges to different sets of parameters.

Figure 4 displays the resulting parameters for the camera S40 Far. We plot the loss of the correspondence residuals and the λ residual blocks against each of the parameters. The translation parameters are in meters and relative to the transverse mercator projection [1]. The rotation parameters are in degree of Euler angles.

The plots show that the standard deviation σ of the translations does not exceed $2 * 10^{-4} m = 0.2mm$. For the rotation σ is at most $4 * 10^{-4} deg$.

We have shown the expectable error for the parameters resulting from the sensor and map inaccuracies in subsub-section 0.2.4. We conclude that in relation to these the algorithmic error can be neglected.

In the appendix **??** we additionally evaluate the other cameras.

References

[1] Johann Heinrich Lambert. *Anmerkungen und zusätze zur entwerfung der land-und himmelscharten*. Number 54. W. Engelmann, 1894. 4

[2] PROJ contributors. *PROJ coordinate transformation software library*. Open Source Geospatial Foundation, 2021. 1