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

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

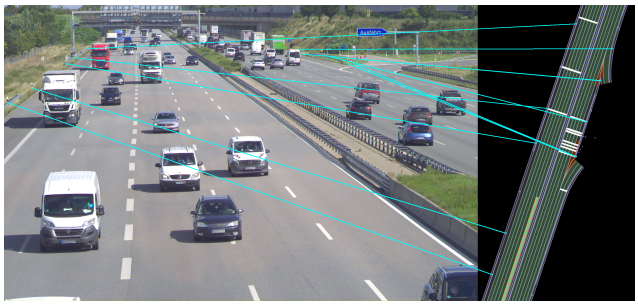

Figure 1: Left: The current camera frame. Right: A part of the HD maps (**??**). Cyan: The mapping $s_c \mapsto p_c$ from objects (right) to their corresponding pixels (left).

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.
- Pixels of the objects are also symmetric around the projected directional axis.
- The pixels in the mapping is equally distributed around

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.

the directional axis as the deviations from tangential offset points cancel out then.

This models that the objects collapse into their center line in 2D and 3D and that tangentially deviating points on the surface in one direction equally cancels out by the opposite point mirrored at 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

$$p_c = \pi\left(\begin{bmatrix}R,T\\0,1\end{bmatrix} * (o_c + \lambda_c * h_c)\right) \tag{3}$$

$$\begin{bmatrix}R,T\\0,1\end{bmatrix} = \begin{bmatrix}R,0\\0,1\end{bmatrix} * \begin{bmatrix}0,T\\0,1\end{bmatrix} = R * T \tag{4}$$

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 T,R 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 R,T can take and enables us to recover the camera pose only from the correspondences.

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

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

formulated as

$$\begin{aligned} E(P,S,T,R,\Lambda,W) = &\sum_{c\in C} \|w_c * [p_c - \hat{p}_c]\|_2^2 \\ &+ \sum_{c\in C} \|penalize(\lambda_i, h_i)\|_2^2 \\ &+ \sum_{c\in C} \|\alpha * (1 - w_i)\|_2^2 \end{aligned} \tag{7}$$

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

$$penalize(\lambda, h) = \begin{cases} \lambda - h, & \text{if } \lambda > h \\ \lambda, & \text{if } \lambda < 0 \\ 0, & \text{else} \end{cases} \tag{8}$$

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 $\lambda \in [0,h]$.

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

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

$$R_0 = \mathbb{I}^{4\times4} \tag{11}$$

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 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 Expectable Error Bounds

0.2.3 Algorithmic Error

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 parameters, 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.

Remaining Correspondences Error We derive a rough scale for the remaining loss over the correspondences and λs, as only a small subset of solution are valid and represent the real camera pose.

$$\begin{aligned} E(P,S,T,R,\Lambda,W) &\simeq \sum_{c\in C} \|p_c - \pi(T * R * s_c)\|_2^2 \\ &= \sum_{c\in C} \|p_c - \hat{p}_c\|_2^2 \end{aligned} \tag{12}$$

where $p_c - \hat{p}_c$ is the remaining distance of actual pixel to the projected pixel.

The projected $\hat{p}_c$ can only be along the center line of the object due to the approximation of objects as lines. The actual pixels p_c will be evenly distributed around the center line, as we expect the objects to be roughly cylindrical (section 0.1).

This implies that for a valid camera pose the remaining distance $p_c - \hat{p}_c$ can be at most half of the pixel width w_s of the object (the maximal pixel distance from the center line).

It thus follows Equation 12

$$\begin{aligned} E(P,S,T,R,\Lambda,W) &\simeq \sum_{c\in C} \|p_c - \hat{p}_c\|_2^2 \\ &= \sum_{o\in O} \sum_{c_o\in C_o} \|p_{c_o} - \hat{p}_{c_o}\|_2^2 \\ &\leq \sum_{o\in O} |C_o| \frac{w_o}{2} \end{aligned} \tag{13}$$

where h_o is the pixel height of the object.

This implies that an upper bound for the remaining error is only dependent on the number of all pixels of the objects and their respective width. We have empirically seen that w_0 in our case does not exceed 10 pixels, thus the remaining error cannot exceed

$$\sum_{o\in O} |C_o| \frac{10}{2} = 5 * \sum_{o\in O} |C_o| = 5 * |C| \tag{14}$$

This shows that the remaining error has to be roughly in the same scale as the number of pixels over all objects.

Expectable Deviations among Estimations 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 subsubsection 0.2.2. We conclude that in relation to these the algorithmic error can be neglected.

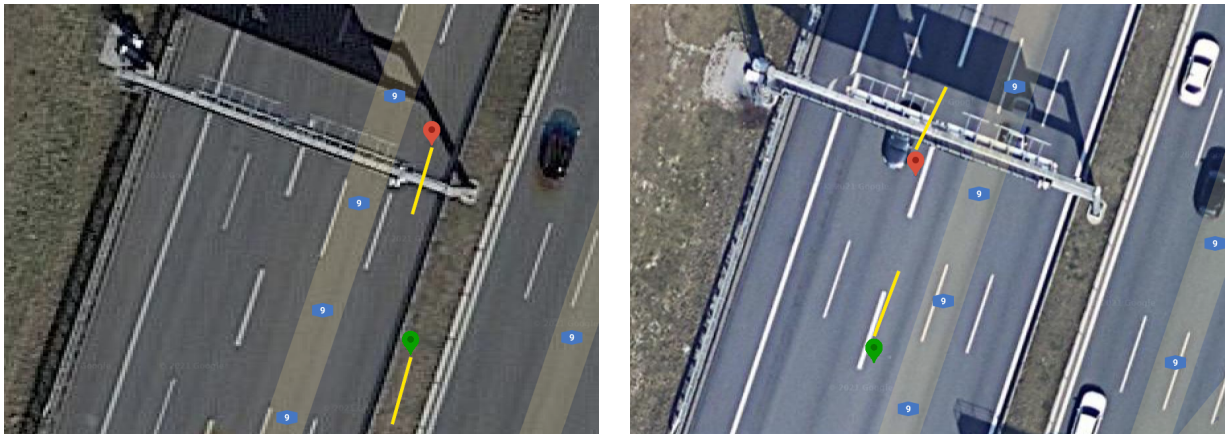

Figure 3: Left: The positions of the cameras S40 Near (red) and S40 Far (green) and their respective looking directions (yellow). Right: The positions of the cameras S50 Near (red) and S50 Far (green) and their respective looking directions (yellow). It displays that the rotation of the cameras is in 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 0.2.2 shows.

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. 3

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

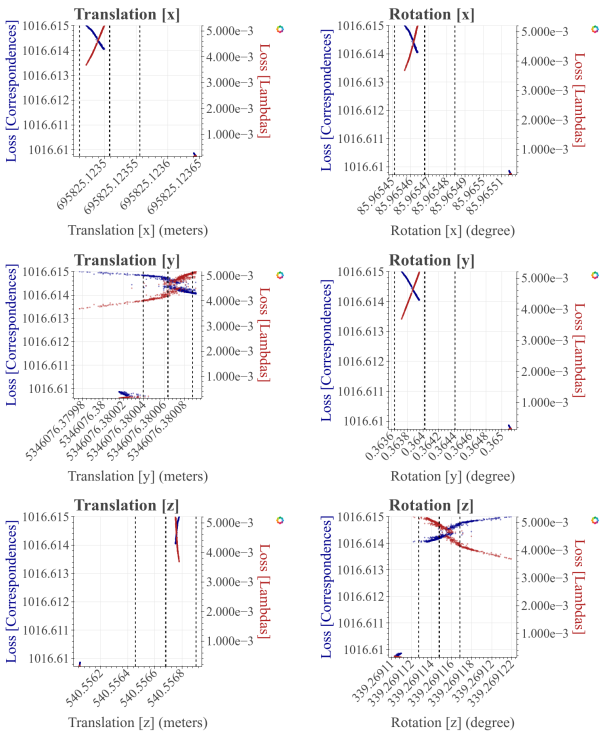

Figure 4: Left: The resulting translational parameters plotted against the remaining losses. Right: The resulting rotational parameters plotted against the remaining losses. The mean of the distributions is displayed as thick dashed line. The smaller dashed lines display the standard deviation σ. The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